



HAL
open science

Exact Sparse Approximation Problems via Mixed-Integer Programming: Formulations and Computational Performance

Sébastien Bourguignon, Jordan Ninin, Hervé Carfantan, Marcel Mongeau

► **To cite this version:**

Sébastien Bourguignon, Jordan Ninin, Hervé Carfantan, Marcel Mongeau. Exact Sparse Approximation Problems via Mixed-Integer Programming: Formulations and Computational Performance. IEEE Transactions on Signal Processing, 2016, 64 (6), pp.1405-1419. 10.1109/TSP.2015.2496367. hal-01254856

HAL Id: hal-01254856

<https://hal.science/hal-01254856v1>

Submitted on 12 Jan 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Exact Sparse Approximation Problems via Mixed-Integer Programming: Formulations and Computational Performance

Sébastien Bourguignon, Jordan Ninin, Hervé Carfantan, and Marcel Mongeau, *Member, IEEE*

Abstract—Sparse approximation addresses the problem of approximately fitting a linear model with a solution having as few non-zero components as possible. While most sparse estimation algorithms rely on suboptimal formulations, this work studies the performance of exact optimization of ℓ_0 -norm-based problems through Mixed-Integer Programs (MIPs). Nine different sparse optimization problems are formulated based on ℓ_1 , ℓ_2 or ℓ_∞ data misfit measures, and involving whether constrained or penalized formulations. For each problem, MIP reformulations allow exact optimization, with optimality proof, for moderate-size yet difficult sparse estimation problems. Algorithmic efficiency of all formulations is evaluated on sparse deconvolution problems. This study promotes error-constrained minimization of the ℓ_0 norm as the most efficient choice when associated with ℓ_1 and ℓ_∞ misfits, while the ℓ_2 misfit is more efficiently optimized with sparsity-constrained and sparsity-penalized problems. Then, exact ℓ_0 -norm optimization is shown to outperform classical methods in terms of solution quality, both for over- and under-determined problems. Finally, numerical simulations emphasize the relevance of the different ℓ_p fitting possibilities as a function of the noise statistical distribution. Such exact approaches are shown to be an efficient alternative, in moderate dimension, to classical (suboptimal) sparse approximation algorithms with ℓ_2 data misfit. They also provide an algorithmic solution to less common sparse optimization problems based on ℓ_1 and ℓ_∞ misfits. For each formulation, simulated test problems are proposed where optima have been successfully computed. Data and optimal solutions are made available as potential benchmarks for evaluating other sparse approximation methods.

Index Terms—sparse approximation, ℓ_0 -norm-based problems, optimization, mixed-integer programming, deconvolution

I. INTRODUCTION

A. Sparse estimation for inverse problems

The problem of *sparse representation* of data $\mathbf{y} \in \mathbb{R}^N$ in a *dictionary* $\mathbf{H} \in \mathbb{R}^N \times \mathbb{R}^Q$ consists in finding a solution $\mathbf{x} \in \mathbb{R}^Q$ to the system $\mathbf{y} = \mathbf{H}\mathbf{x}$ with the fewest non-zero components, *i.e.*, with the lowest *sparsity level*. In *sparse approximation*, in order to account for noise and model errors, the equality constraint is relaxed through the minimization of the data misfit measure $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|$, where $\|\cdot\|$ generally stands for the standard Euclidean norm in \mathbb{R}^N . Such sparsest representation and approximation problems are essentially combinatorial. Finding the best K -sparse solution (the solution with K non-zero components) is usually considered

too difficult in practical large-scale instances. Indeed, the brute-force approach that amounts to exploring all the $\binom{Q}{K}$ possible combinations, is computationally prohibitive. In the abundant literature on sparse approximation, much work has been dedicated to the relaxation approach that replaces the ℓ_0 -“norm” sparsity measure, $\|\mathbf{x}\|_0 := \text{Card}\{i \mid x_i \neq 0\}$, with the ℓ_1 norm $\|\mathbf{x}\|_1 := \sum_i |x_i|$. Many specific convex optimization algorithms have been proposed in the past decade, see for example [1], [2]. In addition, conditions were established for which both the ℓ_0 and the relaxed ℓ_1 problems yield the same *solution support* (the set of non-zero components). These mostly rely on a low sparsity level assumption and on structural hypotheses on the matrix \mathbf{H} , such as low correlation of its columns (see [1] and references therein). Alternatively, greedy algorithms build a sparse solution by iteratively adding non-zero components to an initially empty-support solution [3], [4], [5]. More complex forward-backward methods [6], [7] may show better performance in practice but with higher computation time. Tree-search-based methods also try to improve the classical greedy algorithms using heuristics to reduce the complexity of exhaustive combinatorial exploration (see *e.g.*, [8] and references therein). Other support exploration strategies maintain the desired sparsity level at each iteration, and perform local combinatorial exploration steps [9]. Optimality proofs for all such strategies also rely on very restrictive hypotheses [10], [11]. More “ ℓ_0 -oriented” approaches were proposed, *e.g.*, by successive continuous approximations of the ℓ_0 norm [12], by descent-based Iterative Hard Thresholding (IHT) [13], [14] and by penalty decomposition methods [15]. However, without additional assumptions on \mathbf{H} , one can only prove that the solution found is a local minimum of the optimization problem. Moreover, for IHT, optimality conditions suffer from the same restrictions as the aforementioned greedy methods [13], [14].

In many inverse problems, the model $\mathbf{y} \simeq \mathbf{H}\mathbf{x}$ results from the discretization of an intrinsically continuous physical model. A typical example is sparse deconvolution, where $\mathbf{H}\mathbf{x}$ models the convolution of a spike train (in one-dimensional signals) or of point sources (in imaging problems) by the impulse response of the acquisition device [7], [16], [17]. A similar problem concerns nonlinear parameter identification, where parameters are discretized on arbitrarily thin grids [7], [18], [19] and estimation amounts to finding a sparse solution to a linear problem of high dimension. In such cases, the columns of \mathbf{H} can be highly correlated, so no optimality guaranty can be obtained for greedy and ℓ_1 -norm-based methods.

This work was partially supported by the French Groupement de Recherche ISIS (Information, Signal, Image, viSion) through its “young researchers” program. One of the authors of this work has been supported by French National Research Agency (ANR) through JCJC program (project ATOMIC n° ANR 12-JS02-009-01).

Similar problems also arise for variable selection in machine learning and statistics [6], where the set of features (the columns of \mathbf{H}) is not designed to satisfy any recovery property.

The aforementioned problems essentially focus on the correct estimation of the support of \mathbf{x} . In deconvolution, for example, support identification corresponds to detection and localization of the sources. Since the true sparsity measure is indeed the ℓ_0 norm, a global optimum of ℓ_0 -based formulations is more likely to yield exact support identification than approximate solutions. Consequently, our interest focuses on optimization methods for ℓ_0 -norm-based criteria providing optimality guarantees. Such *exact* approaches are usually discarded, based on the argument that sparse optimization problems are NP hard [20]. It is also commonly considered that exact optimization amounts to combinatorial exploration of all possible supports, which is nothing but a worst-case-scenario argument. Note however that, in order to reduce the number of explored supports, Tropp and Wright [1] mentioned the possible use of *cutting-plane methods*, which are one of the basic elements of resolution of the mixed-integer programs explored in the present paper.

Here, we focus on sparse optimization occurring in certain inverse problems with moderate size yet with a complexity sufficient to make the usual methods fail in estimating the sparsest solutions. Examples include spike-train deconvolution in ultrasonic nondestructive testing (NDT) [16] or Geophysics [17], sparse hyperspectral unmixing [21], and spectral analysis with short data sets [22]. A first objective of this contribution is to show the viability of exact resolution approaches for such problems.

B. Global optimization via Mixed-Integer Programming

We focus on Mixed-Integer Programs (MIP), that is, optimization problems involving both continuous and integer variables. Such problems are well suited to ℓ_0 -norm-based optimization, since the ℓ_0 norm naturally introduces a binary decision variable for each component (zero or non-zero?). In this paper, MIP refers to the minimization of linear or quadratic criteria subject to linear or quadratic inequality constraints. It is commonly claimed that, in the past fifteen years, a factor 10^9 was gained in the required computing time for solving such problems. This gain is due in (roughly) equal parts to (i) hardware improvement, (ii) progress in the resolution of linear programs, and (iii) implementation efficiency of advanced mathematical techniques [23]. Therefore, as it will be shown in this paper, some exact approaches can now be advantageously used to address the moderate-size, yet difficult, applications enumerated at the end of Section I-A.

To our knowledge, the first MIP reformulation of a sparse optimization problem is proposed in [24]. However, the authors argue that the assumption that $|\mathbf{x}|$ is upper bounded, which is required for the MIP reformulation, leads to computational inefficiency. Therefore, they choose to consider only a related problem: the maximum feasible subsystem problem, for which exact solutions can be found only for very small instances ($N = 16$, $Q = 32$) and no result is given concerning the MIP approach. A similar formulation

with binary variables appears in [25], but binary variables are replaced by continuous variables in $[0, 1]$ in order to yield a convex problem, which is obviously not equivalent to the original one. In [26], some exact and approximate reformulations of ℓ_0 -based problems are surveyed. The authors deplore the inherent combinatorial difficulty of such MIP problems but no practical result is provided. Finally, in [27], noise-free sparse *representation* problems are formulated as MIP. Here, we address the noisy case, which opens perspectives to different possible formulations of the optimization problem. Establishing MIP reformulations for such problems, studying their computational efficiency, investigating properties of optimal solutions and comparing them with the results of standard methods are the core of this paper.

C. Objectives of the paper

This paper shows that different possible reformulations of sparse approximation problems can be tackled by MIP solvers. Sparse approximation is intrinsically a bi-objective optimization problem, where both the sparsity measure and the data misfit measure are optimized. In inverse problems, it is usually formulated through the optimization of a weighted sum of the two terms. However, constrained formulations (involving one criterion to be minimized and the other subject to a constraint) may also be well suited to MIP reformulations, which are constrained programs by nature. Therefore, we study the efficiency of MIP solving techniques applied to the following three formulations:

- minimize the ℓ_0 norm under a bounded-data-misfit constraint,
- minimize the data misfit under an ℓ_0 -boundedness constraint,
- minimize a weighted sum of the two criteria.

Additionally, we also consider non-quadratic data misfit measures, which may be appropriate if the error term $\mathbf{y} - \mathbf{H}\mathbf{x}$ is non-Gaussian. Moreover, piecewise-linear alternatives to the ℓ_2 norm $\|\mathbf{x}\|_2 := \sqrt{\sum_i |x_i|^2}$, may also prove to be more attractive computationally, because MIPs essentially rely on the resolution of *linear* subproblems. In particular, the ℓ_1 and ℓ_∞ ($\|\mathbf{x}\|_\infty := \max_i |x_i|$) norms are easily linearized; formulations based on those norms naturally boil down to optimization problems involving linear inequality constraints or linear objective functions. Therefore, we also consider formulations involving ℓ_1 and ℓ_∞ misfits, for which much fewer algorithms have been proposed.

Our work establishes MIP reformulations of nine different sparse approximation problems, which are all evaluated in terms of computational efficiency, depending on the sparsity level and on the noise level. Then, these formulations are compared in their ability to identify the exact support in the presence of noise, depending on the noise statistical distribution. Our experimental results additionally show that the classical methods are far from reaching acceptable results in such cases, whereas the ℓ_0 -norm formulations do yield more satisfactory solutions—but with much higher computing time. Note that experiments in both [24] and [27] involve random

matrices, which are more likely to satisfy the conditions ensuring the optimality of ℓ_1 -norm-based and greedy approaches. In most ill-posed inverse problems, the columns of \mathbf{H} are highly correlated, so that such conditions certainly do not hold.

The remainder of the paper is organized as follows. Section II introduces nine optimization formulations of the sparse approximation problem, and discusses their statistical interpretation and the structure of the solution sets. MIP reformulations are then established in Section III. Then, basic elements concerning the resolution of MIP problems are given in Section IV. Experimental results in Section V are dedicated to the evaluation of computational costs. Section VI compares the solutions obtained through MIP optimization with those of classical sparse approximation algorithms, on both overdetermined and underdetermined sparse deconvolution problems. Simulations in Section VII evaluate the support identification performance of ℓ_p -misfit-constrained formulations as a function of p and of the noise statistical distribution. Finally, a discussion is given in Section VIII.

II. SPARSE OPTIMIZATION PROBLEMS

In sparse approximation, both the sparsity of the solution and the fidelity of its corresponding data approximation are optimized. Therefore, the generic sparse approximation problem, which we are interested in, is the following unconstrained bi-objective optimization problem:

$$\min_{\mathbf{x}} (\|\mathbf{x}\|_0, \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_p), \quad (1)$$

where p is either 1, 2 or ∞ . This section presents nine formulations of this problem and discusses their statistical interpretation and the structure of the sets of solutions.

A. Taxonomy

Various mono-objective optimization problems can be formulated to address the bi-criterion problem (1). For $p \in \{1, 2, \infty\}$, the *bounded-error* problems read

$$\mathcal{P}_{0/p} : \min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_p \leq \alpha_p,$$

and the *sparsity-constrained* problems read

$$\mathcal{P}_{p/0} : \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_p \text{ s.t. } \|\mathbf{x}\|_0 \leq K_p,$$

where α_p and K_p are user-defined threshold parameters. Finally, the *penalized* problems read

$$\mathcal{P}_{0+p} : \min_{\mathbf{x}} \mu_p \|\mathbf{x}\|_0 + \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_p^p, \text{ for } p \in \{1, 2\},$$

$$\text{and } \mathcal{P}_{0+\infty} : \min_{\mathbf{x}} \mu_\infty \|\mathbf{x}\|_0 + \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty,$$

where μ_p and μ_∞ are user-defined penalty parameters.

In this paper, we propose a reformulation of each of these problems as MIPs. To the best of our knowledge, $\mathcal{P}_{0/\infty}$ is the only sparse approximation problem for which a MIP reformulation is mentioned [24]. Remark that the sparse *representation* case (noise-free data), which was recently tackled *via* MIPs in [27] with equality constraint $\mathbf{y} = \mathbf{H}\mathbf{x}$, is the special case of $\mathcal{P}_{0/p}$ with α_p set to 0, for which $\mathcal{P}_{0/1}$, $\mathcal{P}_{0/2}$, and $\mathcal{P}_{0/\infty}$ are obviously equivalent. Recall that the sparsity-based inverse

problems considered here are sparse *approximation* problems: data are always contaminated by measurement noise and the model may be inexact, so that $\alpha_p \neq 0$.

Choosing one of the nine formulations and the values of the parameters (α_p , K_p or μ_p) amounts to selecting some particular solutions among the wide variety of Pareto-optimal solutions of problem (1). Note that, for a given ℓ_p misfit, no equivalence between the three problems $\mathcal{P}_{0/p}$, $\mathcal{P}_{p/0}$ and \mathcal{P}_{0+p} can be obtained because the ℓ_0 norm is not convex. In particular, solutions in the non-convex part of the Pareto frontier cannot be reached by solving the penalized formulation [28].

B. Statistical interpretations and parameter tuning

In practice, one has to choose one among the nine optimization problems and must set a value for the corresponding parameter. Such choices can be based on statistical arguments.

The ℓ_p data-misfit measures, with $p \in \{1, 2, \infty\}$, can be interpreted in terms of likelihood functions. Let p_ϵ be the statistical distribution of the additive noise term $\epsilon = \mathbf{y} - \mathbf{H}\mathbf{x}$. The likelihood function is defined as: $\mathcal{L}(\mathbf{y}; \mathbf{x}) := p_\epsilon(\mathbf{y} - \mathbf{H}\mathbf{x})$. If noise samples ϵ_n , $n = 1, \dots, N$ are independent and identically distributed (i.i.d.) according to a centered Gaussian distribution, then $-\log \mathcal{L}(\mathbf{y}; \mathbf{x})$ is proportional to $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2$ up to an additive constant. Similarly, $-\log \mathcal{L}(\mathbf{y}; \mathbf{x})$ is proportional to $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_1$ (up to an additive constant) if ϵ_n are i.i.d. according to a centered Laplace distribution. Such a heavy-tailed distribution assumption may be appropriate in the presence of impulsive noise [29], [16]. The ℓ_∞ misfit is connected to an i.i.d. uniform noise distribution assumption. Suppose that ϵ is uniformly distributed on $[-a, a]^N$, for some given $a > 0$. Then, the likelihood function is constant for any \mathbf{x} such that $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty \leq a$, otherwise it is zero. Consequently, $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty$ is a maximum-likelihood estimator if noise samples are uniformly distributed on $[-a, a]$, for any $a > \|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\|_\infty$ [30, Ch. 7.1]. In this case, which arises for example when accounting for quantization noise, ℓ_∞ data fitting may be a relevant choice—see [31] for both theoretical and numerical arguments.

Consequently, $\mathcal{P}_{p/0}$ is a sparsity-constrained maximum likelihood estimation problem with the aforementioned corresponding noise distribution assumption. In a Bayesian setting, \mathcal{P}_{0+p} defines a Maximum A Posteriori (MAP) estimate, where the ℓ_0 term results from a Bernoulli-Gaussian prior model with infinite variance Gaussian distribution [7]. Note that, within such a MAP interpretation, solving $\mathcal{P}_{0+\infty}$ reduces to solving $\mathcal{P}_{0/\infty}$. Indeed, since $-\log \mathcal{L}(\mathbf{y}; \mathbf{x})$ is constant if $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty \leq a$ and equals $+\infty$ otherwise, minimizing $\mu_\infty \|\mathbf{x}\|_0 - \log \mathcal{L}(\mathbf{y}; \mathbf{x})$ amounts to minimizing $\|\mathbf{x}\|_0$ subject to $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty \leq a$. Finally, $\mathcal{P}_{0/p}$ considers a maximal tolerance on the approximation error and cannot be interpreted as a maximum likelihood or MAP estimation problem.

The choice of the formulation also depends on available prior information about the considered practical problem. If reasonable bounds on the acceptable approximation error can be inferred, *e.g.*, from the knowledge of the signal-to-noise ratio or from a desired approximation quality, then $\mathcal{P}_{0/p}$ may be preferred. In particular, the parameter α_p can be fixed according to the statistics of $\|\epsilon\|_p$, which can be obtained for any

noise distribution (analytically or numerically). If the sparsity level is fixed or can be upper bounded, *e.g.*, in a compression context, then $\mathcal{P}_{p/0}$ may be appropriate. In \mathcal{P}_{0+p} , the parameter μ_p trades off between the noise level and the sparsity level. With the previous MAP interpretation, for $p \in \{1, 2\}$, it is an explicit function of the noise variance and of the rate of non-zero values in the Bernoulli process. Therefore, tuning μ_p requires more information than tuning the parameters of the two other formulations. When too little prior information is available, a practical solution consists in computing optimal solutions corresponding to different parameter tunings—whatever the considered formulation—and then selecting *a posteriori* the most appropriate one, according to some expert supervision or to model order selection criteria [32].

C. Structure of the solution sets

We investigate hereafter the structure of the solution sets of the different problems, for fixed values of the corresponding parameters α_p , K_p and μ_p . In the following, an *optimal support* refers to a set of indices which supports at least one optimal solution. The ℓ_0 norm is a piecewise constant function, where each value is attained on a finite number of supports. Hence, for any problem \mathcal{P} defined in Section II-A, the set of minimizers can be defined as the finite union of sets of minimizers on each optimal support: if \mathcal{S} denotes the set of optimal supports, then:

$$\text{Arg } \mathcal{P} = \bigcup_{s \in \mathcal{S}} \text{Arg } \mathcal{P}^s,$$

where \mathcal{P}^s denotes the restriction of problem \mathcal{P} to the support s .

Let us characterize the solution set of \mathcal{P}^s . We assume that the sparsity level of all solutions is lower than N and that the matrix \mathbf{H} satisfies the *Unique Representation Property* (URP) [33], that is, any N columns of \mathbf{H} are linearly independent. For any \mathbf{x} supported by s , $\|\mathbf{x}\|_0$ is constant, hence $\mathcal{P}_{p/0}^s$ and \mathcal{P}_{0+p}^s are solved by minimizing $\|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p$, where \mathbf{x}_s (respectively, \mathbf{H}_s) collects the non-zero components in \mathbf{x} (respectively, the corresponding columns of \mathbf{H}). Thanks to the URP, \mathbf{H}_s has full column rank and, for $p = 2$, \mathcal{P}^s admits a unique solution (the least-squares solution). Consequently, the solution sets of $\mathcal{P}_{2/0}$ and \mathcal{P}_{0+2} are both (finite) unions of singletons. The ℓ_p norms for $p = 1$ and $p = \infty$ are not strictly convex, therefore one can only claim that $\min_{\mathbf{x}_s} \|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p$ is attained on a convex set, such that $\mathbf{H}_s \mathbf{x}_s$ lies in an ℓ_p -sphere in dimension $K_s - 1$, centered at \mathbf{y} . Consequently, for $p = 1$ and $p = \infty$, the solution sets of $\mathcal{P}_{p/0}$ and \mathcal{P}_{0+p} are (finite) unions of convex sets of the form $\{\|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p = \text{constant}\}$. Now, consider the solution set of $\mathcal{P}_{0/p}^s$, which is formed by all vectors \mathbf{x}_s such that $\|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p \leq \alpha_p$. In the particular case where $\min_{\mathbf{x}_s} \|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p = \alpha_p$, it comes from the previous arguments that the solution set is a singleton for $p = 2$, and a convex set for $p = 1$ and $p = \infty$. But, in the most frequent case where $\min_{\mathbf{x}_s} \|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p < \alpha_p$, the solution set $\{\mathbf{x}_s \mid \|\mathbf{y} - \mathbf{H}_s \mathbf{x}_s\|_p \leq \alpha_p\}$ is such that $\mathbf{H}_s \mathbf{x}_s$ lies in an ℓ_p -ball of dimension K_s , centered at \mathbf{y} , and the solution set of $\mathcal{P}_{0/p}$ is a finite union of such sets. Consequently, for a given p ,

the solution set of $\mathcal{P}_{0/p}$ is generally “larger” than the solution sets of $\mathcal{P}_{p/0}$ and \mathcal{P}_{0+p} . In particular, the minimizers of these last two problems may be unique. For example, with some additional assumptions on the data \mathbf{y} and on the matrix \mathbf{H} , the solution of \mathcal{P}_{0+2} is unique [34]. On the contrary, the minimizer of $\mathcal{P}_{0/p}$ is certainly not unique, except in very specific cases.

III. MIXED-INTEGER REFORMULATIONS

In this section, we establish the reformulations of optimization problems $\mathcal{P}_{0/p}$, $\mathcal{P}_{p/0}$ and \mathcal{P}_{0+p} for $p \in \{1, 2, \infty\}$, as Mixed-Integer Linear Programs (MILPs), Mixed-Integer Quadratic Programs (MIQPs) or Mixed-Integer Quadratically Constrained (linear) Programs (MIQCPs).

A. Definitions of MILP, MIQP and MIQCP

The general form of an MILP is

$$\min_{\mathbf{v}} \mathbf{c}^\top \mathbf{v}, \text{ subject to (s.t.) } \begin{cases} \mathbf{A}_{\text{in}} \mathbf{v} \leq \mathbf{b}_{\text{in}}, \\ \mathbf{A}_{\text{eq}} \mathbf{v} = \mathbf{b}_{\text{eq}}, \\ \mathbf{l}_b \leq \mathbf{v} \leq \mathbf{u}_b, \\ v_j \in \mathbb{Z}, \forall j \in \mathcal{I}, \end{cases}$$

where $\mathbf{v} \in \mathbb{R}^J$ is the vector of optimization variables; $\mathbf{c} \in \mathbb{R}^J$ defines the linear objective function; $\mathbf{b}_{\text{in}} \in \mathbb{R}^{P_{\text{in}}}$, $\mathbf{b}_{\text{eq}} \in \mathbb{R}^{P_{\text{eq}}}$, $\mathbf{A}_{\text{in}} \in \mathbb{R}^{P_{\text{in}}} \times \mathbb{R}^J$ and $\mathbf{A}_{\text{eq}} \in \mathbb{R}^{P_{\text{eq}}} \times \mathbb{R}^J$ define the inequality and equality constraints; \mathbf{l}_b and $\mathbf{u}_b \in \mathbb{R}^J$ are respectively the vectors of lower and upper bounds of the optimization variables; \mathcal{I} is the index set corresponding to the components of \mathbf{v} that are constrained to be integer-valued.

An MIQP has the general form:

$$\min_{\mathbf{v}} \frac{1}{2} \mathbf{v}^\top \mathbf{F} \mathbf{v} + \mathbf{c}^\top \mathbf{v}, \text{ s.t. } \begin{cases} \mathbf{A}_{\text{in}} \mathbf{v} \leq \mathbf{b}_{\text{in}}, \\ \mathbf{A}_{\text{eq}} \mathbf{v} = \mathbf{b}_{\text{eq}}, \\ \mathbf{l}_b \leq \mathbf{v} \leq \mathbf{u}_b, \\ v_j \in \mathbb{Z}, \forall j \in \mathcal{I}, \end{cases}$$

where \mathbf{F} is a $J \times J$ matrix.

Finally, the form of an MIQCP that is of interest in this paper is:

$$\min_{\mathbf{v}} \mathbf{c}^\top \mathbf{v}, \text{ s.t. } \begin{cases} \frac{1}{2} \mathbf{v}^\top \mathbf{B} \mathbf{v} + \mathbf{d}^\top \mathbf{v} \leq e, \\ \mathbf{A}_{\text{in}} \mathbf{v} \leq \mathbf{b}_{\text{in}}, \\ \mathbf{A}_{\text{eq}} \mathbf{v} = \mathbf{b}_{\text{eq}}, \\ \mathbf{l}_b \leq \mathbf{v} \leq \mathbf{u}_b, \\ v_j \in \mathbb{Z}, \forall j \in \mathcal{I}, \end{cases}$$

where \mathbf{B} is a $J \times J$ matrix, $\mathbf{d} \in \mathbb{R}^J$ and $e \in \mathbb{R}$.

B. Equivalent reformulation techniques

We now present standard reformulation techniques that enables to express each of the nine optimization problems introduced in Section II-A as an MILP, an MIQP or an MIQCP, without any approximation.

1) *Boundedness assumption and “big- M ” reformulation of the ℓ_0 norm:* For each $q = 1, 2, \dots, Q$, let us introduce an additional binary optimization variable b_q , such that

$$b_q = 0 \Leftrightarrow x_q = 0. \quad (2)$$

Then, the non-linear sparsity measure $\|\mathbf{x}\|_0$ is equal to the linear term $\mathbf{1}_Q^T \mathbf{b}$ ($= \sum_q b_q$), where $\mathbf{1}_Q$ is the Q -dimensional all-ones column vector. The logical constraint (2) must however be translated into (in)equality constraints compatible with MIP, ideally through *linear* constraints. One standard way to achieve this (see e.g. [24], [27]) is to assume that a solution \mathbf{x} of the problem under consideration satisfies the following constraints for some sufficiently large pre-defined value $M > 0$:

$$-M\mathbf{1}_Q < \mathbf{x} < M\mathbf{1}_Q. \quad (3)$$

This assumption supposes that the problem admits bounded optimal solutions, which is not restrictive in our practical applications. The parameter M has to be large enough so that $\|\hat{\mathbf{x}}\|_\infty < M$ at any desirable optimal solution $\hat{\mathbf{x}}$. On the other hand, the bound M must be as tight as possible in order to improve computational efficiency; therefore tuning the value of parameter M may be a critical issue [24]. In the problems addressed in this paper, satisfactory results are obtained with a rather simple empirical rule discussed in Section V-B. Note that specific lower and upper bounds for each component x_q could also be advantageously considered [27] if corresponding prior information is available.

The reformulations of ℓ_0 -norm-based constraints and objective functions are obtained through the two following respective lemmas.

Lemma 1: Considering the boundedness assumption (3),

$$\|\mathbf{x}\|_0 \leq K \Leftrightarrow \begin{cases} \exists \mathbf{b} \in \{0; 1\}^Q \text{ such that} \\ \sum_{q=1}^Q b_q \leq K, & (i) \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b}. & (ii) \end{cases}$$

Proof: The \Rightarrow implication is straightforward by considering \mathbf{b} defined by Equation (2). Now, let $\mathbf{b} \in \{0; 1\}^Q$ satisfy (i) and (ii), and suppose $\|\mathbf{x}\|_0 > K$. From (ii), one has $(b_q = 0) \Rightarrow (x_q = 0)$, that is, $(x_q \neq 0) \Rightarrow (b_q = 1)$. Hence $b_q = 1$ for at least $K + 1$ indices q , which contradicts (i). Consequently, $\|\mathbf{x}\|_0 \leq K$. ■

Lemma 2: Considering the boundedness assumption (3),

$$\min_{\mathbf{x} \in \mathcal{F}} \|\mathbf{x}\|_0 \Leftrightarrow \min_{\substack{\mathbf{x} \in \mathcal{F} \\ \mathbf{b} \in \{0,1\}^Q}} \sum_{q=1}^Q b_q \text{ s.t. } -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b},$$

where \mathcal{F} represents the feasible domain of the problem under consideration.

Proof: Similar to that of Lemma 1. ■

Such a reformulation technique is commonly referred to as “big- M ” reformulation. Remark finally that another reformulation of the cumbersome logical constraint (2) consists in introducing the equality constraint $x_q(b_q - 1) = 0$. However, the latter is a bi-linear constraint, typically less interesting in terms of computation time for off-the-self MIP solvers than

linear constraints [35].

2) *Reformulation of the ℓ_1 data misfit measure:* The ℓ_1 misfit term can be written linearly as $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_1 = \sum_n w_n$, with additional constraints $w_n = |y_n - \mathbf{h}_n^T \mathbf{x}|$, $n = 1, \dots, N$, where \mathbf{h}_n^T denotes the n th row of \mathbf{H} . Then, these constraints can be relaxed (exactly) by the linear inequalities: $-\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w}$, with column vector $\mathbf{w} = [w_1, \dots, w_N]^T$, thanks to the two following lemmas.

Lemma 3: $\min_{\mathbf{x} \in \mathcal{F}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_1$

$$\Leftrightarrow \min_{\mathbf{x} \in \mathcal{F}, \mathbf{w} \in \mathbb{R}^N} \sum_n w_n \text{ s.t. } -\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w}.$$

Proof: Let $f_n(\mathbf{x}) = y_n - \mathbf{h}_n^T \mathbf{x}$. The following optimization problems are equivalent:

$$\mathcal{P}_a : \min_{\mathbf{x} \in \mathcal{F}} \sum_n |f_n(\mathbf{x})|$$

$$\mathcal{P}_b : \min_{\mathbf{x} \in \mathcal{F}, \mathbf{w} \in \mathbb{R}^N} \sum_n w_n \text{ s.t. } |f_n(\mathbf{x})| = w_n, \forall n$$

$$\mathcal{P}_c : \min_{\mathbf{x} \in \mathcal{F}, \mathbf{w} \in \mathbb{R}^N} \sum_n w_n \text{ s.t. } |f_n(\mathbf{x})| \leq w_n, \forall n.$$

Indeed, $\mathcal{P}_a \Leftrightarrow \mathcal{P}_b$ is trivial. In order to show that $\mathcal{P}_b \Leftrightarrow \mathcal{P}_c$, one can simply remark that if an optimal solution $(\mathbf{x}^*, \mathbf{w}^*)$ of \mathcal{P}_c is such that $|f_{n_0}(\mathbf{x}^*)| < w_{n_0}^*$ for some index n_0 , then one can straightforwardly construct a better feasible solution for \mathcal{P}_c , which yields a contradiction. ■

Lemma 3 will be used to obtain a MIP reformulation of $\mathcal{P}_{1/0}$ and \mathcal{P}_{0+1} , which involve the ℓ_1 -misfit term in the objective function. For $\mathcal{P}_{0/1}$, which involves the ℓ_1 -misfit term as a *constraint*, we use the following lemma:

Lemma 4: Let $(\mathbf{x}^*, \mathbf{w}^*)$ solve the optimization problem:

$$\mathcal{P}_d : \min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{w} \in \mathbb{R}^N} \|\mathbf{x}\|_0 \text{ s.t. } \begin{cases} \sum_n w_n \leq \alpha_1 \\ -\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w}. \end{cases}$$

Then, \mathbf{x}^* is a solution of $\mathcal{P}_{0/1}$.

Proof: Suppose that $(\mathbf{x}^*, \mathbf{w}^*)$ solves (\mathcal{P}_d) and let $w'_n := |y_n - \mathbf{h}_n^T \mathbf{x}^*|$, $\forall n$. Then, $\sum_n w'_n \leq \sum_n w_n^* \leq \alpha_1$ and $(\mathbf{x}^*, \mathbf{w}')$ is a solution of:

$$\mathcal{P}_e : \min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{w} \in \mathbb{R}^N} \|\mathbf{x}\|_0 \text{ s.t. } \begin{cases} \sum_n w_n \leq \alpha_1 \\ |y_n - \mathbf{h}_n^T \mathbf{x}| = w_n, \forall n. \end{cases}$$

Indeed, since \mathcal{P}_d is a relaxation of \mathcal{P}_e —the feasible set of \mathcal{P}_e is a subset of that of \mathcal{P}_d —, its optimal value, $\|\mathbf{x}^*\|_0$, is a lower bound for the optimal value of \mathcal{P}_e . The solution $(\mathbf{x}^*, \mathbf{w}')$ is clearly feasible for \mathcal{P}_e and it attains the lower bound $\|\mathbf{x}^*\|_0$. Hence, it is optimal for \mathcal{P}_e . Finally, \mathcal{P}_e is clearly equivalent to $\mathcal{P}_{0/1}$. ■

Let us remark finally that problems \mathcal{P}_d and \mathcal{P}_e are not strictly equivalent because they are not minimized by the same couple of vectors (\mathbf{x}, \mathbf{w}) , but they share the same solution set for \mathbf{x} .

3) *Reformulation of the ℓ_∞ data misfit measure:* $\mathcal{P}_{0/\infty}$ naturally brings linear inequality constraints as

$$\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty \leq \alpha_\infty \Leftrightarrow -\alpha_\infty \mathbf{1}_N \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \alpha_\infty \mathbf{1}_N.$$

For both $\mathcal{P}_{\infty/0}$ and $\mathcal{P}_{0+\infty}$, involving the ℓ_∞ norm in the objective function, one can simply introduce an additional scalar variable z such that minimizing $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty$ amounts to minimizing z under the constraints

$$-z\mathbf{1}_N \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq z\mathbf{1}_N.$$

Here again, one can easily show that an optimal solution of the original problem will necessarily satisfy $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|_\infty = z$.

C. Mixed-integer programming reformulations

Given the reformulation techniques of Section III-B, the nine problem reformulations proposed in Table I are straightforward.

IV. MIP RESOLUTION: BASIC ELEMENTS

Mixed-integer programming problems are not easy problems. MILP problems are already NP-hard [36]. As a consequence, reducing sparse approximation problems to MILP, MIQP or MIQCP problems does not *per se* reduce the complexity. Nevertheless, such MIP reformulations not only open up possibilities to prove the optimality (or quantify the sub-optimality) of solutions, but also allows one to benefit from decades of exponential progress in terms of required computing time to solve a given MIP problem. This progress does not simply reflect the doubling of computing power every 18 months, it is also a consequence of fast progress in both the theory and practice of linear programming and discrete optimization (duality, numerical linear algebra, interior-point methods, semi-definite positive relaxations, branch and bound/cut/price methods, decomposition approaches, global optimization, *etc.*) Once the sparse approximation problem is recast as a MIP problem, then state-of-the-art off-the-shelf software can be used, such as BARON, COUENNE, CPLEX, GloMIQO, GUROBI, MOSEK or Xpress-MP—see for example [35] and references therein. We chose to use CPLEX [23] because it is unanimously considered among the best MIP solvers. CPLEX has been developed over the last thirty years and includes the best strategies developed by the MIP community. Moreover, it is freely available for research and teaching purposes.

The main method behind the CPLEX MIP solver is a *branch-and-cut* algorithm. Globally, it implements a *branch-and-bound* strategy (*i.e.*, a tree-structured implicit enumeration algorithm) based on successive continuous relaxations of the integer variables [37]. Each *branch* generates a subproblem by fixing some integer (in our case, binary) variables, and a set of branches is constructed corresponding to the different possible configurations of such variables. Then, the aim is to discard (huge) parts of the remaining combinatorial tree by *lower bounding* the objective function on the corresponding subproblems. To obtain such lower bounds, a continuous relaxation of each subproblem is formed by relaxing the integer variables. Linear constraints, such as Gomory *cutting planes* [38], are added to each generated (continuous relaxation) subproblem, so that the continuous solution converges to an integer solution. Such cutting planes remove part of the feasible domain of the subproblem that does not contain

any integer solution. This approach amounts to attempting to construct the convex hull of the set of integer feasible solutions of each subproblem. CPLEX incorporates several techniques in order to improve performance, such as constraint propagation techniques [39], linear algebra techniques [23] and heuristic techniques to find rapidly a good integer solution. Doing so, parts of the research space are eliminated only if it is proved that they do not contain the global minimum.

The best current integer solution provides an upper bound of the global minimum of the entire problem. The solution of the current relaxed (continuous) subproblem gives a lower bound of the global minimum of the current subproblem with integer constraints under consideration. The worst solution of all relaxed subproblems—the one that achieves the lowest lower bound—gives a certified lower bound of the global minimum of the entire problem. If such a lower bound is attained by the best current integer solution, then a global minimizer is found and optimality is proved. Otherwise, the entire process is iterated by creating new *branches*. The algorithm converges towards such a certified optimum in a finite number of steps. If the solver reaches the time limit, the *duality gap* (the difference between the best solution found and the certified lower bound of the global minimum) provides a measure of sub-optimality of the solution found. Note that tree-search based greedy algorithms such as in [8] also rely on tree-based (local) exploration and on lower-bounding the objective function on sets of solutions, which is used inside a greedy procedure. Therefore, they do not come with any optimality guarantee.

Remark that in most sparsity-inspired signal processing problems, the matrix \mathbf{H} satisfies specific properties that are exploited for efficient computations. In particular, if \mathbf{H} represents a redundant set of transforms based on multi-scale representations such as wavelets [40], matrix-vector products $\mathbf{H}\mathbf{v}$ and $\mathbf{H}^\top\mathbf{v}$ (where \mathbf{v} is some given vector of an appropriate dimension) can be computed by fast algorithms using Fast Fourier Transforms (FFT). In 1D (respectively, 2D) deconvolution problems, \mathbf{H} is a Toeplitz (respectively, Toeplitz-block-Toeplitz) matrix, so that matrix-vector products can also be computed by FFT. No such matrix structure is exploited within the standard mixed-integer optimization algorithms, as they are implemented as general-purpose MIP solvers. The only matrix property that is exploited here is the possible sparsity of the matrix \mathbf{H} to compute fast vector products. This is particularly the case for the finite impulse response (FIR) deconvolution problems that are considered hereafter.

V. EXPERIMENTAL RESULTS: OPTIMIZATION PERFORMANCE

This section presents the test problems and the computational framework. Then, the computational efficiency of the nine MIP reformulations is studied.

A. Definition of test problems

The different MIP reformulations are evaluated on one-dimensional sparse deconvolution problems. Such problems are typically encountered for example in ultrasonic NDT [16]

Problem	Equivalent MIP reformulation		MIP class
Bounded-misfit formulations (parameters: α_2 , α_1 and α_∞)			
$\mathcal{P}_{0/2}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q} \mathbf{1}_Q^\top \mathbf{b}$	s.t. $\begin{cases} \mathbf{x}^\top \mathbf{H}^\top \mathbf{H} \mathbf{x} - 2\mathbf{y}^\top \mathbf{H} \mathbf{x} \leq \alpha_2^2 - \mathbf{y}^\top \mathbf{y} \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MIQCP
$\mathcal{P}_{0/1}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q, \mathbf{w} \in \mathbb{R}^N} \mathbf{1}_Q^\top \mathbf{b}$	s.t. $\begin{cases} \mathbf{1}_N^\top \mathbf{w} \leq \alpha_1 \\ -\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w} \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP
$\mathcal{P}_{0/\infty}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q} \mathbf{1}_Q^\top \mathbf{b}$	s.t. $\begin{cases} -\alpha_\infty \mathbf{1}_N \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \alpha_\infty \mathbf{1}_N \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP
Sparsity-constrained formulations (parameters: K_2 , K_1 and K_∞)			
$\mathcal{P}_{2/0}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q} \mathbf{x}^\top \mathbf{H}^\top \mathbf{H} \mathbf{x} - 2\mathbf{y}^\top \mathbf{H} \mathbf{x}$	s.t. $\begin{cases} \mathbf{1}_Q^\top \mathbf{b} \leq K_2 \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MIQP
$\mathcal{P}_{1/0}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q, \mathbf{w} \in \mathbb{R}^N} \mathbf{1}_N^\top \mathbf{w}$	s.t. $\begin{cases} \mathbf{1}_Q^\top \mathbf{b} \leq K_1 \\ -\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w} \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP
$\mathcal{P}_{\infty/0}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q, z \in \mathbb{R}} z$	s.t. $\begin{cases} \mathbf{1}_Q^\top \mathbf{b} \leq K_\infty \\ -z\mathbf{1}_N \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq z\mathbf{1}_N \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP
Penalized formulations (parameters: μ_2 , μ_1 and μ_∞)			
\mathcal{P}_{0+2}	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q} \mu_2 \mathbf{1}_Q^\top \mathbf{b} + \mathbf{x}^\top \mathbf{H}^\top \mathbf{H} \mathbf{x} - 2\mathbf{y}^\top \mathbf{H} \mathbf{x}$	s.t. $\begin{cases} -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MIQP
\mathcal{P}_{0+1}	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q, \mathbf{w} \in \mathbb{R}^N} \mu_1 \mathbf{1}_Q^\top \mathbf{b} + \mathbf{1}_N^\top \mathbf{w}$	s.t. $\begin{cases} -\mathbf{w} \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq \mathbf{w} \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP
$\mathcal{P}_{0+\infty}$	$\min_{\mathbf{x} \in \mathbb{R}^Q, \mathbf{b} \in \{0,1\}^Q, z \in \mathbb{R}} \mu_\infty \mathbf{1}_Q^\top \mathbf{b} + z$	s.t. $\begin{cases} -z\mathbf{1}_N \leq \mathbf{y} - \mathbf{H}\mathbf{x} \leq z\mathbf{1}_N \\ -M\mathbf{b} \leq \mathbf{x} \leq M\mathbf{b} \end{cases}$	MILP

TABLE I
MIXED-INTEGER PROGRAMMING REFORMULATIONS OF NINE SPARSE APPROXIMATION PROBLEMS.

and in seismic reflection in Geophysics [17]. In the following, \mathbf{x} is a K -sparse sequence in \mathbb{R}^{100} (*i.e.*, $Q = 100$), with uniformly distributed spike locations, where the sparsity level, K , is varying. In order to avoid arbitrary small spike values, each non-zero amplitude is drawn as $\text{sign}(u) + u$, where u is a centered Gaussian sample with unit variance. The matrix \mathbf{H} is the discrete convolution matrix corresponding to the 21-sample impulse response shown in Fig. 1 (left). With the boundary assumption that \mathbf{x} is zero outside its domain, \mathbf{y} is a 120-sample signal (*i.e.*, $N = 120$). White noise is added with variable signal-to-noise ratio (SNR). In this section, noise samples are generated according to a centered normal distribution $\mathcal{N}(0, \sigma^2)$, with σ such that $\text{SNR}_{\text{dB}} = 10 \log_{10} \left(\frac{\|\mathbf{H}\mathbf{x}\|_2^2}{N\sigma^2} \right)$. We name such problems SA_K^{SNR} . Note that they are slightly overdetermined ($Q < N$), whereas typical sparse approximation problems deal with largely underdetermined systems. However, trivial inversion is not satisfac-

tory here, because of the presence of noise and of the ill-conditioned nature of \mathbf{H} ($\text{cond}(\mathbf{H}) \simeq 10^3$).

We consider problems SA_K^{SNR} with K varying between 5 and 11, and SNR varying from $+\infty$ (noise-free data) down to 10 dB. One example of data is given in Fig. 1 (right) for $K = 7$ and SNR = 20 dB. It illustrates the difficulty of sparse deconvolution problems arising for example in ultrasonic NDT [16]. The oscillating impulse response and the proximity of the spikes produce overlapping echoes in the available data. As the echoes cannot be distinguished visually, numerical techniques are required. All data and optimization results of this section are available online as supplementary multimedia material.

B. Machine configuration and implementation details

Optimization is run with IBM ILOG CPLEX V12.6.0 from a Matlab interface on a computer with eight Intel Xeon X5472 processors with Central Processing Units (CPU) clocked at

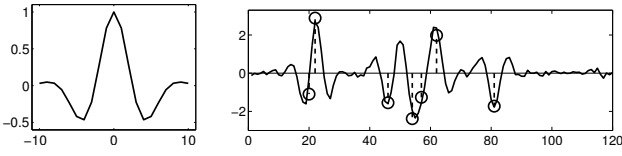


Fig. 1. Example of sparse deconvolution data. Left: impulse response. Right: 7-sparse sequence \mathbf{x} (circles) and noisy convolved data \mathbf{y} , with SNR = 20 dB.

3 GHz. The maximum time allowed for each resolution is set to $T_{\max} = 1000\text{s}$. The other CPLEX parameters are set to their default value. For each problem, the “big- M ” constant is set to $M = 1.1x_{\max}^1$, where $x_{\max}^1 := \|\mathbf{H}^T \mathbf{y}\|_{\infty} / \|\mathbf{h}\|_2^2$ corresponds to the maximum amplitude of 1-sparse solutions estimated by least-squares. If the boundedness assumption (3) is saturated—*i.e.*, one component in the solution $\hat{\mathbf{x}}$ reaches the value $-M$ or M —then the optimization is successively run again with M replaced with $1.1M$, until the obtained solution satisfies $\|\hat{\mathbf{x}}\|_{\infty} < M$. The CPU times given below include such restarts. With this heuristic strategy, in our simulations, only few cases led to such saturation: no restart was necessary in 90 % of the cases, and the average number of restarts in the other cases was approximately 1.6.

C. Evaluation of computational costs

Each of the nine MIP reformulations of Table I is run for fifty random instances (for both spike distributions and noise realizations) of each SA_K^{SNR} problem. In this section, in order to ensure a fair comparison (in terms of computational efficiency), the parameters K_p are set to the true sparsity level K , and parameters α_p and μ_p are set by trial and error, until the sparsity level of each solution equals K . Note that in our case, the matrix \mathbf{H} has full column rank, hence ℓ_0 inequality constraints in $\mathcal{P}_{p/0}$ will yield the same results as if they were equality constraints. That is, when imposing $\|\mathbf{x}\|_0 \leq K$, all the K columns of matrix \mathbf{H} contribute to the reduction of the data misfit. However, formulations with inequality constraints yielded lower computing times. Sparse representation problems (noise-free data, SNR = ∞) are addressed through $\mathcal{P}_{0/p}$ for $p \in \{1, 2, \infty\}$, with threshold $\alpha_p = 10^{-8}$. Remark that in the noise-free case, no sparsity-enhancing algorithm is indeed necessary, since the solution can simply be computed by least squares.

Average CPU times obtained for MIP reformulations are given in Table II. The figures on the left-hand side of each column is the time required to *prove* the global optimality of the solution found. The figures on the right-hand side indicate the time after which the support of the solution was *found*, which is generally much lower. The figures in parentheses indicate the number of instances for which optimality was not proved within 1000s.

All CPU times increase with the sparsity level K , but also with the noise level. In particular, for SNR = 10 dB and $K = 11$, for each formulation, optimality of the solutions was obtained in less than 1000s only on a fraction of the fifty instances. In order to explain such behavior, let us remark that the sparsity level (respectively, the noise level) increases the size of the feasible domain of $\mathcal{P}_{p/0}$ (respectively, of $\mathcal{P}_{0/p}$).

More generally, for all problems, if either the sparsity level or the noise level increases, then the branch-and-bound strategy becomes less efficient in discriminating between concurrent candidates and in eliminating some of them.

With SNR = 30 dB, the lowest CPU times are achieved by solving \mathcal{P}_{0+2} —although optimization did not terminate within 1000s for one instance of SA_{11}^{30} . When the noise level increases, solving $\mathcal{P}_{0/1}$ and $\mathcal{P}_{0/\infty}$ problems becomes more efficient computationally, and their superiority over other problems increases with both the sparsity level and the noise level. In particular, $\mathcal{P}_{0/1}$ problems were always solved exactly in less than 1000s, except for (SNR = 20 dB, $K = 11$). Results are slightly not as good for $\mathcal{P}_{0/\infty}$, where optimization did not terminate within 1000s for three more instances with SNR = 20 dB and $K = 11$. Note that even a small amount of noise severely degrades the resolution performance of problems $\mathcal{P}_{0/p}$. Optimization of $\mathcal{P}_{0/2}$, the only problem with a quadratic constraint, is the most time-consuming among all proposed formulations. Non-linear constraints are known to make the MIP really difficult to solve [35]. An element of explanation can be found by comparing the Lagrangians of formulations involving ℓ_2 misfits. Indeed, the Lagrangian of $\mathcal{P}_{0/2}$ contains trilinear terms. On the contrary, the Lagrangians of $\mathcal{P}_{2/0}$ and of \mathcal{P}_{0+2} are quadratic functions. Therefore, optimizing a linear function under quadratic constraints is more complex than optimizing a quadratic function under linear constraints.

For any $p \in \{1, 2, \infty\}$, solving problems \mathcal{P}_{0+p} generally performs better than solving problems $\mathcal{P}_{p/0}$ at high SNR. On the contrary, as the noise level increases, sparsity-constrained formulations outperform penalized versions. For both formulations, which involve the data misfit in the objective function, using an ℓ_2 misfit measure is the most efficient choice, and both ℓ_1 - and ℓ_{∞} -misfit optimizations behave similarly.

We also note a high dispersion of the required CPU times among the fifty realizations of each problem. For example, the average time for the resolution of \mathcal{P}_{0+1} on SA_9^{20} problems was approximately 11s on forty-eight instances, whereas optimization did not terminate after 1000s on the two other instances. We also remark that, for $\mathcal{P}_{0/2}$, optimality was not proved within 1000s for two instances of the simplest test problem $\text{SA}_5^{+\infty}$. Two other instances of $\text{SA}_5^{+\infty}$ also required a much higher CPU time than the others, which leads to an atypically high average time of 53s, reflecting once again the difficulty of the quadratically-constrained problem $\mathcal{P}_{0/2}$.

Finally, let us evaluate the CPU time of exhaustive combinatorial exploration for $\mathcal{P}_{2/0}$. Using notations introduced in Section II-C, for a given support s with K components, the minimizer of $\mathcal{P}_{2/0}^s$ has the closed-form expression $\hat{\mathbf{x}}_s = (\mathbf{H}_s^T \mathbf{H}_s)^{-1} \mathbf{H}_s^T \mathbf{y}$. Then, the least-squares misfit value is computed by: $\|\mathbf{y} - \mathbf{H}_s \hat{\mathbf{x}}_s\|^2 = \|\mathbf{y}\|^2 - \|\mathbf{H}_s \hat{\mathbf{x}}_s\|^2$. In practice, $\hat{\mathbf{x}}_s$ can be computed by performing Cholesky decomposition of $\mathbf{H}_s^T \mathbf{H}_s$, so that one computation of the objective function mainly amounts to two $K \times K$ triangular inversions. The CPU time, denoted c_K , of one such computation is estimated by averaging over 10^5 inversions. Then, neglecting the cost of Cholesky factorizations, the cost for testing all K -sparse solutions is extrapolated as $\binom{Q}{K} c_K$. It yields approximately

	Error-constrained problems						Sparsity-constrained problems						Penalized problems					
	$\mathcal{P}_{0/2}$		$\mathcal{P}_{0/1}$		$\mathcal{P}_{0/\infty}$		$\mathcal{P}_{2/0}$		$\mathcal{P}_{1/0}$		$\mathcal{P}_{\infty/0}$		\mathcal{P}_{0+2}		\mathcal{P}_{0+1}		$\mathcal{P}_{0+\infty}$	
SNR = ∞																		
K = 5	53	50 ⁽²⁾	0.07	0.03	0.06	0.03												
K = 7	3.8	3.6	0.09	0.03	0.08	0.03												
K = 9	4.4	4.2	0.12	0.03	0.10	0.03												
K = 11	5.2	5.0	0.14	0.03	0.13	0.03												
SNR = 30 dB																		
K = 5	13	7.2	0.3	0.2	0.1	0.1	0.3	0.2	0.5	0.4	0.5	0.4	0.2	0.1	0.4	0.2	0.4	0.2
K = 7	17	6.7	1.4	0.4	0.3	0.1	0.5	0.3	1.4	0.8	1.4	1.0	0.4	0.2	1.0	0.6	0.8	0.4
K = 9	132	37	3.1	1.0	1.7	0.5	3.0	1.0	9.3	3.0	25	7.5	1.9	0.8 ⁽¹⁾	19	2.7	17	2.7
K = 11	399	96 ⁽¹⁸⁾	18	4.1	19	2.1	29	4.0 ⁽¹⁾	72	11 ⁽²⁾	78	17 ⁽⁹⁾	11	2.7 ⁽¹⁾	60	13 ⁽¹⁾	48	14 ⁽⁴⁾
SNR = 20 dB																		
K = 5	13	8.6	0.3	0.2	0.1	0.1	0.3	0.2	0.6	0.3	0.5	0.4	0.3	0.2	0.6	0.3	0.4	0.2
K = 7	48	11 ⁽¹⁾	1.0	0.4	0.5	0.1	1.3	0.6	4.0	1.8	4.2	3.5	6.6	0.9	14	1.5 ⁽¹⁾	29	14
K = 9	285	82 ⁽⁹⁾	6.6	2.7	3.7	0.4	18	3.4 ⁽¹⁾	30	6.4 ⁽¹⁾	72	64 ⁽²⁾	28	13 ⁽²⁾	11	2.8 ⁽²⁾	87	18 ⁽²⁾
K = 11	-	- ⁽⁵⁰⁾	67	16	65	11 ⁽³⁾	166	21 ⁽⁶⁾	244	22 ⁽¹¹⁾	355	260 ⁽²⁰⁾	104	16 ⁽⁵⁾	179	58 ⁽¹³⁾	299	38 ⁽¹⁷⁾
SNR = 10 dB																		
K = 5	20	4.9	0.5	0.2	0.2	0.1	0.4	0.2	1.3	0.6	1.7	1.4	1.9	1.7	2.0	0.5 ⁽²⁾	15	0.8
K = 7	195	27 ⁽²⁾	2.3	0.6	1.4	0.2	3.5	0.7	11	3.0	40	39	42	2.4	73	4.0 ⁽³⁾	66	15 ⁽⁶⁾
K = 9	485	63 ⁽³⁴⁾	24	3.9	26	1.2	54	12 ⁽¹⁾	189	28 ⁽⁴⁾	220	218 ⁽¹¹⁾	96	25 ⁽¹¹⁾	230	11 ⁽¹⁸⁾	215	32 ⁽¹⁶⁾
K = 11	-	- ⁽⁵⁰⁾	287	44 ⁽¹²⁾	211	9.0 ⁽⁹⁾	358	26 ⁽²⁴⁾	670	75 ⁽⁴²⁾	600	598 ⁽³⁹⁾	340	34 ⁽³²⁾	300	13 ⁽⁴³⁾	508	101 ⁽⁴⁴⁾

TABLE II

CPU TIMES (IN SECONDS) OBTAINED BY THE NINE MIP REFORMULATIONS, AS A FUNCTION OF THE SIGNAL-TO-NOISE RATIO (SNR) AND OF THE SPARSITY LEVEL (K), AVERAGED OVER 50 INSTANCES OF EACH PROBLEM. THE FIRST NUMBER INDICATES THE CPU TIME FOR ESTABLISHING OPTIMALITY OF THE SOLUTION. THE SECOND NUMBER GIVES THE CPU TIME AT WHICH THE CORRESPONDING OPTIMUM WAS FOUND. NUMBERS IN PARENTHESES COUNT THE INSTANCES FOR WHICH OPTIMALITY WAS NOT PROVEN IN LESS THAN 1000S. FOR REFERENCE, THE COMPUTATION TIME OF CLASSICAL SPARSE APPROXIMATION ALGORITHMS ON OUR HARDWARE CAN BE FOUND IN FIG. 2.

1500s for $K = 5$, 4 days for $K = 7$, and more than one year for $K = 9$. Problems based on both ℓ_1 and ℓ_∞ misfits require the resolution of a linear program for each support, therefore the corresponding exhaustive search yields still higher computational costs than in the ℓ_2 case. Consequently, exhaustive search cannot be considered as a practical solution, even for such moderate-size problems. In order to emphasize the ability of the MIP solver to remove important parts of the full combinatorial tree search, we give a last indicator. For $\mathcal{P}_{2/0}$ with SNR = 30 dB and $K = 9$, for which all instances were successfully solved, the average number of combinations that were explicitly considered in the branch-and-bound procedure is about $4.5 \cdot 10^5$, to be compared with the total number of $\binom{100}{9} \simeq 1.9 \cdot 10^{12}$ combinations.

VI. EXPERIMENTAL RESULTS: EVALUATION OF SOLUTIONS

We now compare the solutions obtained *via* MIP optimization with those of classical sparse approximation methods: Orthogonal Matching Pursuit (OMP) [5], Single Best Replacement (SBR) [7], Iterative Hard Thresholding (IHT)¹ [13], [14] and the minimization of the ℓ_1 -norm-penalized least-squares criterion (ℓ_1 -relax) using homotopy continuation [41]. All algorithms are tuned so that all solutions have the correct number of spikes. Therefore, all methods are put in a favorable setting for evaluating their ability to retrieve the correct support of the solution.

Recall that for sparse deconvolution, none of the classical methods are theoretically guaranteed to solve the ℓ_0 -norm problem. As seen in the previous section, the MIP

approach can compute an exact solution, but requires a larger computation time. In order to evaluate intermediate solutions found by the MIP solver, we consider several values of the maximum time allowed for each MIP resolution: $T_{\max} = 1s, 10s, 100s$ and $1000s$. If the maximum time is reached, then the current solution is considered—which is the best solution found, without any optimality guarantee. The parameter M is tuned as explained in Section V-B when running optimization with $T_{\max} = 1000s$. Then, this value is also used for lower values of T_{\max} .

In this section, we consider two types of sparse deconvolution problems. The first problems are similar to those of Section V-A, with $N = 120$ and $Q = 100$, and are therefore slightly overdetermined. The second ones are underdetermined problems, the most frequent case in sparse approximation. Such problems may arise in *high resolution* sparse deconvolution. Indeed, the true spike locations are generally continuous-valued (*e.g.*, representing times of flight of reflected waves), and the model $\mathbf{H}\mathbf{x}$ is a discrete approximation of a continuous integral equation. The discretization step is usually chosen equal to the sampling period of the data, thus \mathbf{H} is a discrete convolution matrix. Such a choice may be too rough and can lead to erroneous spike locations [42]. In order to improve the model, one may consider an upsampled convolution model, where both the impulse response and the sparse sequence are upsampled by an integer factor, UF. Then, \mathbf{H} can be viewed as the concatenation of UF discrete convolution matrices. Detail about the corresponding matrix structure can be found in [42]. In the following, we consider UF = 2, so that $N = 120$ and $Q = 200$. Recall however that the intrinsic difficulty of sparse deconvolution is mostly due to the ill-conditioned

¹We used T. Blumensath's implementation of IHT (program `AIHT.m`) available at: <http://www.personal.soton.ac.uk/tb1m08/sparsify/sparsify.html>

nature of matrix \mathbf{H} and to the presence of noise, even in the overdetermined case.

Fifty random instances are run with $\text{UF} = 1$ and $\text{UF} = 2$, $\text{SNR} = \infty$ and $\text{SNR} = 20$ dB, and sparsity levels K varying between 5 and 11. In the noise-free case, for the MIP approach, only $\mathcal{P}_{0/p}$ problems are relevant. Therefore, solutions are computed *via* the optimization of $\mathcal{P}_{0/\infty}$ (with threshold $\alpha_\infty = 10^{-8}$), which was the most efficient computationally (see Section V-C). All data and MIP optimization results of this section are available online as supplementary multimedia material.

In the noisy case, we consider $\mathcal{P}_{2/0}$, since the classical methods rely on ℓ_2 data misfit measures. Two quality indices are computed for each simulated data set:

- the *exact recovery rate*, which is the average number of simulations for which a given algorithm correctly locates the support of the true sequence;
- the *average spike distance* between the estimated and the true sequences. To this aim, a distance similar to that used in neuroscience [43] is defined: both estimated and true spike trains are convolved with a continuous-domain Gaussian kernel (with standard deviation chosen equal to the discretization step of the spikes). Then, the ℓ_2 -norm error between both signals is computed. Such a criterion is less severe than the exact recovery rate if the estimated spikes are slightly shifted with respect to the true ones, and also gives more importance to spikes with high amplitudes.

Note that with $\text{SNR} = \infty$, the true sequence is a minimizer of $\mathcal{P}_{0/p}$, with $p = 1, 2, \infty$. Hence the expected exact recovery rate for the MIP approach is 100%. On the other hand, in the noisy case, the minimizer of $\mathcal{P}_{2/0}$ may not be obtained on the true support. Therefore, successful global optimization does not always produce exact recovery.

Results are summarized in Fig. 2. The top row shows the average exact recovery rate as a function of the number of spikes, and the bottom row plots the average spike distance to the true sequence as a function of the CPU time.

Let us first focus on the noise-free case (left columns). Recall that, in the overdetermined noise-free case, the solution can simply be computed by least squares. Simulations are still of interest, however, in order to compare the algorithms in an ideal and simple context. For all classical algorithms, the exact recovery rate is lower than 40% in the overdetermined case, and decreases as the sparsity level increases (top row). Their performance is still worse in the underdetermined case, where the exact recovery rate is close to zero, except for the simplest problems ($K=5$). Their average spike distance to the true sequence (bottom row) is logically lower for algorithms requiring more computation time. We note in particular the bad results obtained by IHT in the underdetermined case. Such bad performance of IHT was already attested when theoretical optimality conditions do not hold [14]. This is particularly true in the underdetermined case, where the columns of \mathbf{H} are strongly correlated. On the contrary, the MIP strategy correctly retrieves the support in nearly 100% of the noise-free instances, even with the computation time limited to 1s. Actually, only one instance led to erroneous support

identification (for $\text{UF} = 2$ and $K = 5$), meaning that the solution was not found, even within 1000s. The MIP approach also gives an average spike distance close to zero, which means that both the supports *and* the amplitudes of the solutions have been correctly recovered, even in the underdetermined case, but with a larger computation time (from 0.03s to 0.2s for $\text{UF} = 1$, and from 0.1s to 20s for $\text{UF} = 2$). Note however that all classical algorithms are still much faster on such relatively small problems (between 10^{-3} s and 10^{-2} s).

In the more realistic noisy case ($\text{SNR} = 20$ dB), the results of the classical algorithms are very similar to those obtained in the noise-free case, both in terms of exact recovery rate, average spike distance and CPU time. In contrast, the MIP performance deteriorates, and the exact recovery rate quickly decreases as the number of spikes increases. Recall however that, in the presence of noise, the minimizer of $\mathcal{P}_{2/0}$ may *not* retrieve the correct support: in the overdetermined case, for example, the MIP solver returns the optimal solution in less than 1000s in 94% of the instances, whereas the average exact recovery rate is much lower. However, it is still better than that of the classical methods with $T_{\max} = 1$ s, and even much better if T_{\max} is increased to 100s or 1000s. The MIP approach also outperforms classical methods in terms of average spike distance, in particular if T_{\max} is high enough. In the overdetermined case, the average computing time ranges from 0.25s (for $K = 5$) to 350s (for $K = 11$) and, as mentioned earlier, global optimality was obtained in less than 1000s for most simulations. In the underdetermined case, however, an optimum was not proved to be found within 1000s in 51 % of the instances, that mostly correspond to the cases where $K = 9$ and $K = 11$. This analysis corroborates the results in Section V-C: the presence of noise strongly impacts the computing time of the MIP solver, and therefore the quality of the solutions obtained by early stopping.

A typical data set and estimated sparse sequences are shown in Fig. 3. It corresponds to the overdetermined case, the true sequence is 9-sparse and $\text{SNR} = 20$ dB. In this example, the MIP approach is the only algorithm that correctly identifies the support. Note that the resulting ℓ_2 misfit at the MIP solution is lower than the ℓ_2 norm of the noise.

VII. EXPERIMENTAL RESULTS: RELEVANCE OF ℓ_1 -, ℓ_2 - AND ℓ_∞ -NORM DATA MISFITS

In this section, the impact of the data misfit measure (through ℓ_1 , ℓ_2 and ℓ_∞ norms) on the quality of the solution is studied, as a function of the noise distribution—as motivated by the discussion in Section II-B. To this aim, data are simulated in a manner similar to Section V-A and Fig. 1. The 7-sparse spike sequence \mathbf{x}^{true} is fixed, and 200 noise realizations are drawn, where noise samples are i.i.d. according to Gaussian, Laplacian and uniform distributions, successively. The SNR here is set to 15 dB. The three error-constrained problems $\mathcal{P}_{0/p}$ are considered here. We focus on these formulations because, in practical cases, tuning the parameter α_p requires less prior information than tuning the parameter K_p for $\mathcal{P}_{p/0}$ or the parameter μ_p for \mathcal{P}_{0+p} . Indeed, for any given noise distribution p_ϵ , the parameters

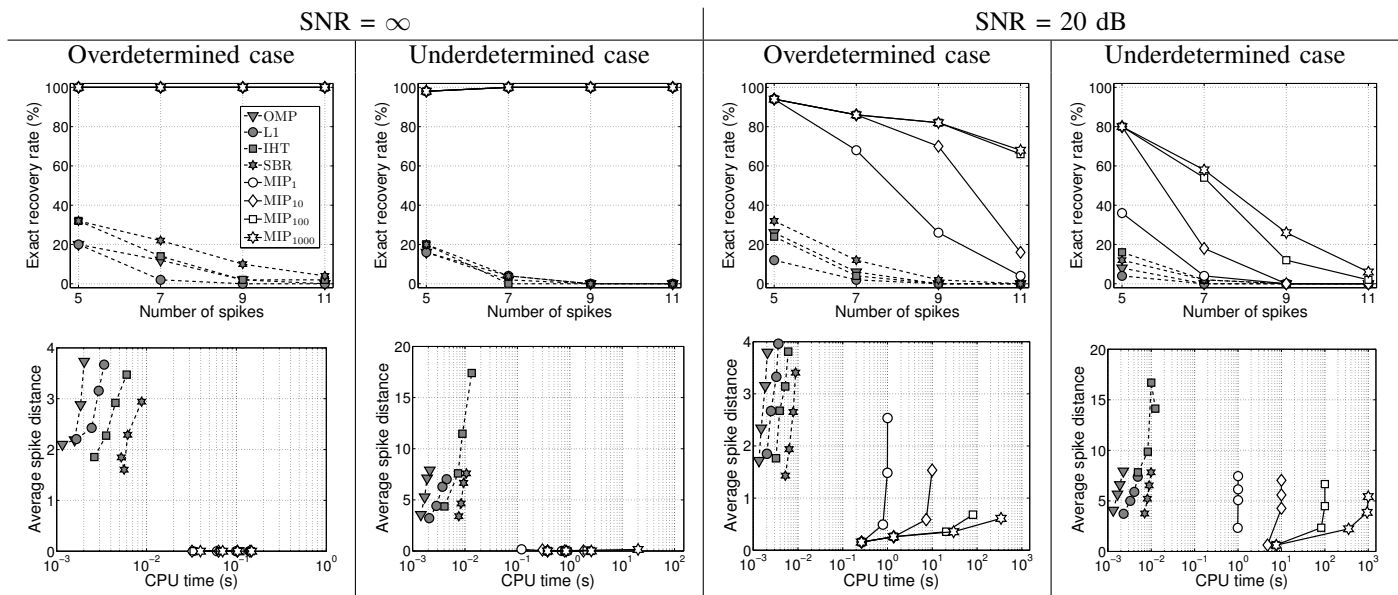


Fig. 2. Performance of classical sparse approximation algorithms and MIP optimization on sparse deconvolution problems, as a function of the signal-to-noise ratio and of the size of the unknown vector. Overdetermined cases correspond to the standard deconvolution problem ($N = 120$, $Q = 100$). Underdetermined cases correspond to an upsampling factor $UF = 2$ such that $N = 120$ and $Q = 200$. Top row: *exact recovery rate* as a function of the sparsity level K . Bottom row: *average spike distance* to the true sequence as a function of the computation time. For each algorithm, the four marks correspond to $K = 5, 7, 9$ and 11 . Note the log-scale and the different scalings of the time axes. The notation $MIP_{T_{\max}}$ corresponds to the MIP optimization strategy running in a maximum of T_{\max} seconds. All results are averaged over 50 random realizations.

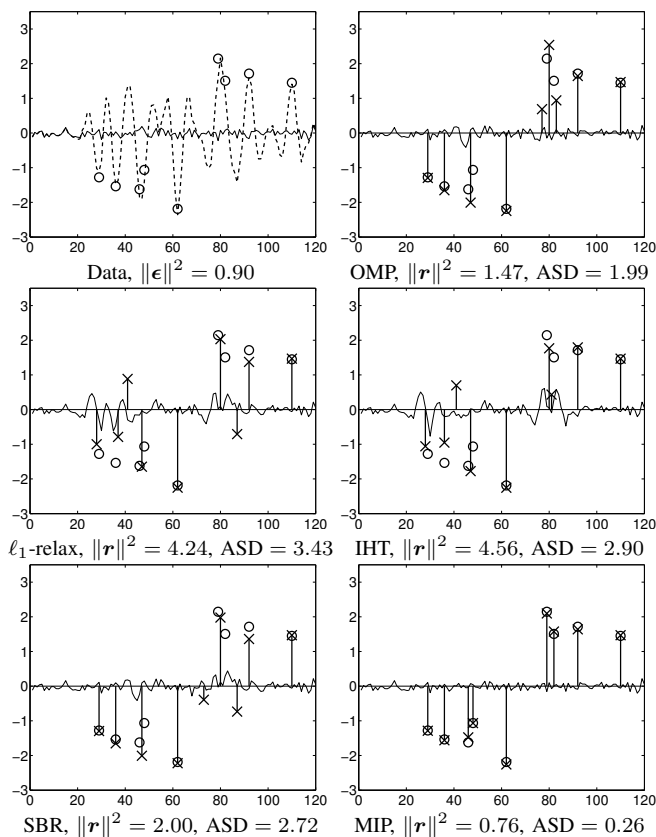


Fig. 3. Data (top-left) and deconvolution results obtained by different sparse approximation algorithms in the overdetermined case ($N = 120$, $Q = 100$), with $SNR = 20$ dB and $K = 9$ spikes. Circles locate the true spikes. On the top-left panel, the dashed (respectively, solid) line represents the data $\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon}$ (respectively, the noise $\boldsymbol{\epsilon}$). On the other panels, crosses show the obtained minimizer $\hat{\mathbf{x}}$ and the solid line represents the residual $\mathbf{r} = \mathbf{y} - \mathbf{H}\hat{\mathbf{x}}$. For each algorithm, the residual norm and the average spike distance to the true sequence (ASD) are given.

α_p , $p \in \{1, 2, \infty\}$, can be naturally estimated from a common statistical rule. More precisely, setting α_p to a value satisfying $\Pr(\|\boldsymbol{\epsilon}\|_p \leq \alpha_p | \boldsymbol{\epsilon} \sim p_{\boldsymbol{\epsilon}}) = 95\%$ amounts to considering that the approximation error cannot be distinguished from noise with probability 95%. Doing so, fair comparisons can be performed between the solutions of the three problems $\mathcal{P}_{0/p}$.

These three problems share the same objective function, hence they can be compared through their minimum value, that is, the estimated number of spikes. In order to evaluate the quality of the solutions, we also consider the *support error* $e^{\text{supp}}(\hat{\mathbf{x}}, \mathbf{x}^{\text{true}}) := \|\hat{\mathbf{b}} - \mathbf{b}^{\text{true}}\|_0$, where \hat{b}_q (respectively, b_q^{true}) equals 1 if $\hat{x}_q \neq 0$ (respectively, $x_q^{\text{true}} \neq 0$), and 0 otherwise. Optimization terminated in less than 1000s for all simulations. Fig. 4 shows the distribution of the ℓ_0 norms (left column) and of the support errors (right column) for the three noise distributions, and for the three ℓ_p data misfits.

We first note that the correct value of the ℓ_0 norm is the most frequently reached in all cases, except if an ℓ_{∞} misfit is used with Laplacian noise. When the estimated ℓ_0 norm is wrong, it is generally lower than the true value. Indeed, in most cases (statistically, in 95% of the cases from the definition of α_p), the noise realization satisfies $\|\boldsymbol{\epsilon}\|_p < \alpha_p$. That is, the true sequence \mathbf{x}^* satisfies the misfit-bound constraint. Thus, the threshold α_p allows a higher approximation error than the noise in the data, which enables the possibility of solutions with fewer spikes. Consequently, in such cases, $\|\hat{\mathbf{x}}\|_0 \leq \|\mathbf{x}^*\|_0$. This is particularly true if the noise distribution is heavy-tailed, where the ℓ_p -norm of the noise samples may be much below the threshold α_p , as show the ℓ_0 norm estimation statistics in the case of Laplacian noise. On the other hand, if $\|\boldsymbol{\epsilon}\|_p \geq \alpha_p$, then the true sequence does not correspond to a solution of the optimization problem, and $\|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\|_p < \|\mathbf{y} - \mathbf{H}\mathbf{x}^*\|_p$. In

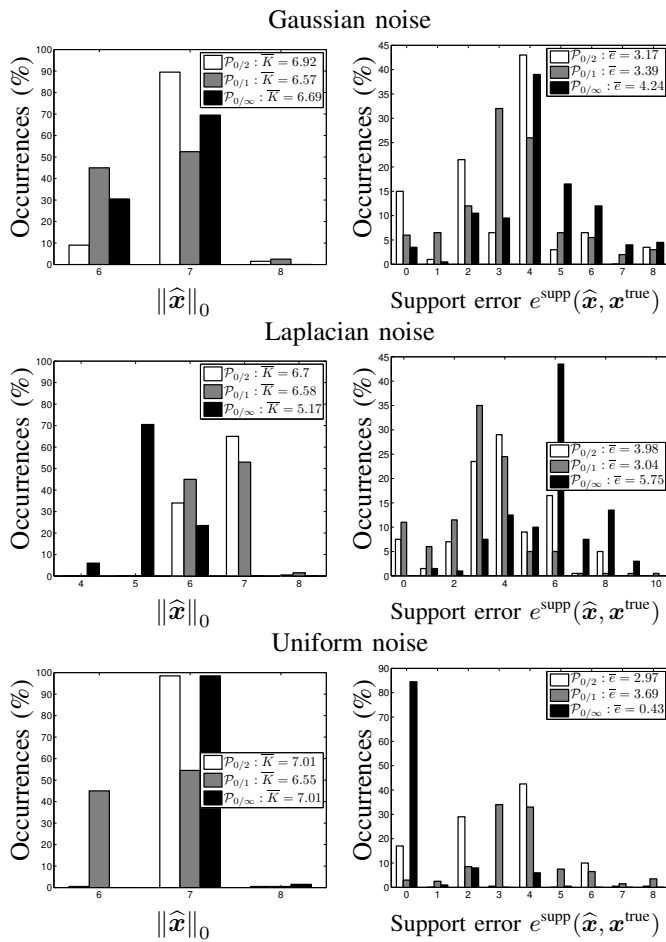


Fig. 4. Estimation results obtained for the deconvolution of a 7-sparse sequence with SNR = 15 dB, averaged over 200 noise realizations, in the case of Gaussian (top), Laplacian (center), and uniform (bottom) noise distributions. Error-constrained problems $\mathcal{P}_{0/p}$ are considered, with $p = 2$ (white), $p = 1$ (gray) and $p = \infty$ (black). Left: distributions of the ℓ_0 norm of the solutions for the three ℓ_p misfits. For each problem, \bar{K} indicates the average estimated ℓ_0 norm value. Right: distributions of the support errors. For each problem, \bar{e} indicates the average support error between the estimated and the true sequences.

such cases, one may have $\|\hat{\mathbf{x}}\|_0 > \|\mathbf{x}^*\|_0$. In our simulations, only very few instances led to such an overestimation of the ℓ_0 norm.

As it could be expected, the lowest support errors are achieved by using the ℓ_2 (respectively, ℓ_1 and ℓ_∞) misfit in the case of Gaussian (respectively, Laplacian and uniform) noise. For each noise distribution, the corresponding misfit yields the smallest average support error, and more frequently achieves correct support identification—even if, for Gaussian or Laplacian noise, it is only obtained in a few cases (respectively, 15% and 11%). We also remark that switching from ℓ_2 to ℓ_1 misfits with Gaussian noise only slightly degrades the support identification performance, whereas optimization is computationally more efficient in the ℓ_1 case—see Section V-C. Much better support identification is achieved with uniform noise combined with the ℓ_∞ misfit, which yields exact identification in 90 % of the cases, whereas ℓ_1 and ℓ_2 data fitting achieve much worse results. Note that with both ℓ_∞ and ℓ_2 misfits, the ℓ_0 norm is correctly estimated in 98% of the cases. However,

support recovery performance is much worse in the ℓ_2 case, as some spikes are misplaced. Such superiority of ℓ_∞ data fitting for uniform noise was already attested in [31] in a non-sparse context.

Fig. 5 displays typical results obtained for one particular realization of the noise process. For Gaussian (respectively, Laplacian and uniform) noise distributions, one example is shown such that ℓ_2 (respectively, ℓ_1 and ℓ_∞) data fitting yields a solution with the most frequent support error obtained among the 200 realizations. Note that, in each case, the solution shown corresponds to *one* solution of the considered optimization problem, that is, with the lowest ℓ_0 norm that satisfies the bounded- ℓ_p -misfit constraint. Recall indeed that, for most values of threshold parameters α_p , problems $\mathcal{P}_{0/p}$ feature an infinite number of solutions—see Section II-C. Consequently, the presented solution is almost certainly not the solution with minimal ℓ_p misfit. With Gaussian noise, the minimizer of $\mathcal{P}_{0/2}$ has the correct ℓ_0 norm, but with two misplaced spikes, that leads to a support error equal to 4. With the minimizer of $\mathcal{P}_{0/1}$, two spikes are slightly misplaced, and a third one is not detected. The minimizer of $\mathcal{P}_{0/\infty}$ also has the correct ℓ_0 norm, but its spikes are very badly located. For Laplacian noise, the most frequent support error for the minimizer of $\mathcal{P}_{0/1}$ is 3, which corresponds to one misplaced spike and the non-detection of one spike. Note that on the presented example, both minimizers of $\mathcal{P}_{0/1}$ and of $\mathcal{P}_{0/2}$ identify the same support, whereas the solution of $\mathcal{P}_{0/\infty}$ features only four spikes (among which one is erroneous). In the case of uniform noise, the solution of $\mathcal{P}_{0/\infty}$ correctly locates all spikes. The solution of $\mathcal{P}_{0/1}$ misplaces one spike and misses another one, and the solution of $\mathcal{P}_{0/2}$ is still worse, with three misplaced spikes—although with the correct sparsity level.

VIII. DISCUSSION

In this paper, nine sparse approximation problems involving the ℓ_0 norm were considered and reformulated as mixed-integer programs (MIP). Bounded-error, sparsity-constrained and penalized formulations were studied, involving ℓ_p -norm data misfit measures, for $p \in \{1, 2, \infty\}$. Thanks to efficient dedicated MIP solvers, we demonstrated that moderate-size sparse approximation problems can be solved exactly, whereas exhaustive search remains computationally prohibitive for such instances. In particular, the use of a branch-and-bound strategy, coupled with efficient cutting planes methods, allows most combinations to be discarded without being evaluated.

Computational costs were evaluated on simulated difficult sparse deconvolution problems. Simulated data and corresponding optimal solutions are made available as potential benchmarks for evaluating other (potentially suboptimal) sparse approximation algorithms². Our experiments show that misfit-constrained minimization of the ℓ_0 norm is the most efficient optimization formulation when associated with ℓ_1 and ℓ_∞ misfit measure. Conversely, the ℓ_2 misfit measure is advantageously used as an objective function, not as a

²Matlab implementations of the nine formulations are available at <http://www.irccyn.ec-nantes.fr/~bourguignon>

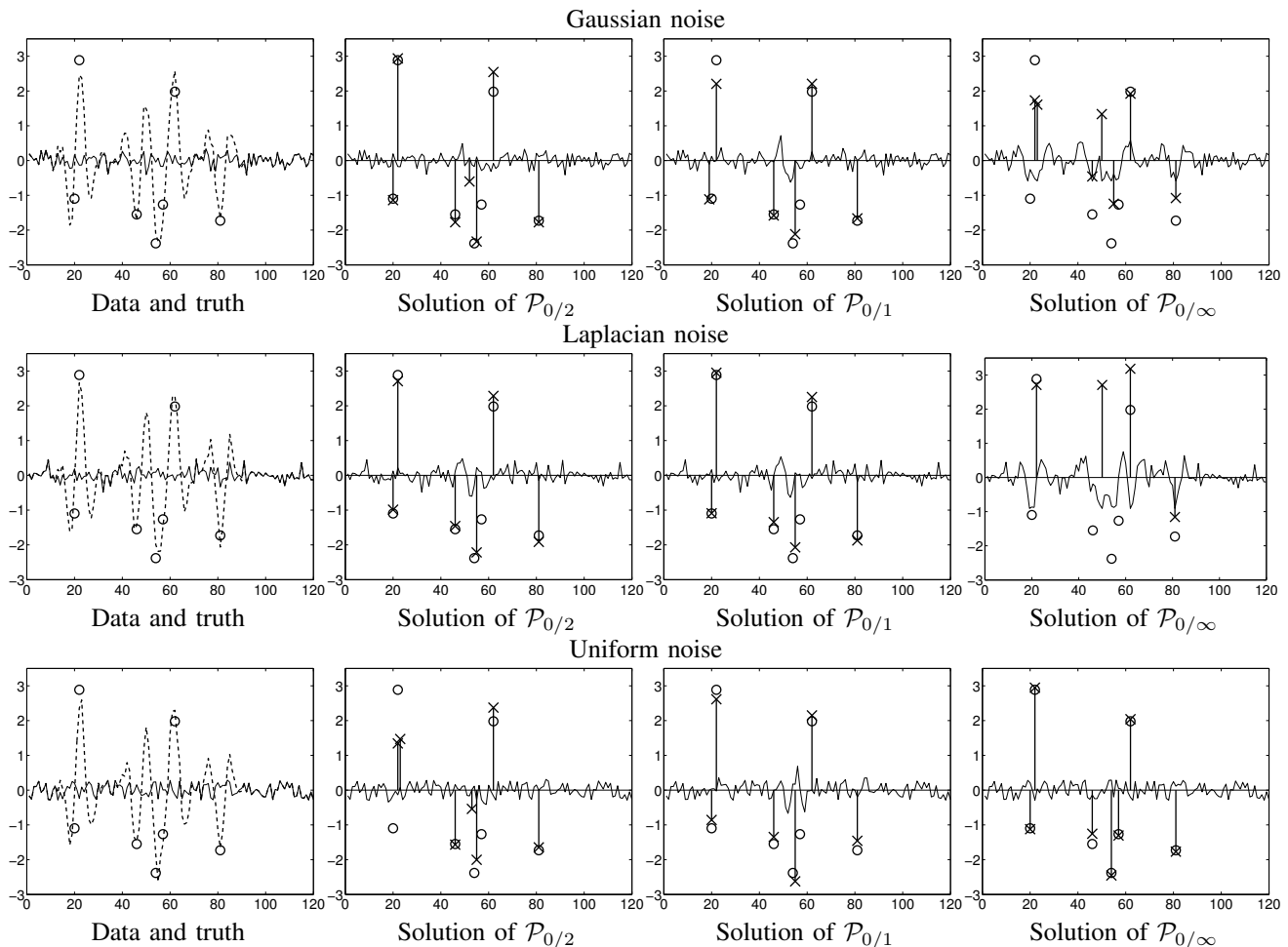


Fig. 5. Solutions of deconvolution problems $\mathcal{P}_{0/p}$ with Gaussian (top), Laplacian (center) and uniform (bottom) noises, for one particular noise realization, with SNR = 15 dB. Circles locate the true spikes. On the left column, the dashed (respectively, solid) line represents the data \mathbf{y} (respectively, the noise $\boldsymbol{\epsilon}$). On the three other columns, crosses show the obtained minimizer $\hat{\mathbf{x}}$ and the solid line represents the residual $\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}$.

constraint. All CPU times increase with the number of non-zero components in the true solution, and also with the amount of noise in the data. Our encouraging numerical results tend to indicate however that such optimization formulations may be appropriate for tackling sparse approximation problems with several hundreds of unknowns, as long as the solution is highly sparse and/or the noise level is low. In particular, they do represent an alternative to ℓ_1 -norm-based and greedy methods for difficult estimation problems with highly correlated dictionaries, both of which are likely to fail. Simulations revealed, in particular, that exact solutions of ℓ_0 problems almost always achieve perfect support recovery for underdetermined noise-free problems, whereas classical methods perform relatively badly. In the presence of noise, the MIP solutions still outperform that of classical methods (in both over- and underdetermined cases), although the required computing time for obtaining exact solutions dramatically increases.

The ℓ_0 sparse approximation problem with ℓ_2 data misfit measure has been used in a huge quantity of works in signal processing, statistics, machine learning, *etc.* To the best of our knowledge, the methods presented in this paper are the only guaranteed-optimality alternatives to exhaustive search that do not rely on any strong assumption on the dictionary structure.

With the introduced MIP reformulations, we also proposed to solve exactly less common sparse optimization problems based on ℓ_1 and ℓ_∞ misfits. Such problems may be of interest from an informational point of view. Simulations illustrated this point, and confirmed a rather intuitive fact: choosing an ℓ_p misfit with $p = 2$ (respectively, $p = 1$ and $p = \infty$) is relevant if the noise distribution is Gaussian (respectively, Laplacian and uniform) *as far as support identification is concerned*. In particular, with uniformly distributed noise, introducing an ℓ_∞ misfit constraint frequently achieves correct support identification, which is not the case for any other combination of data misfit and noise distribution.

Several points in the MIP reformulations could be considered in order to improve computational efficiency. First, as acknowledged in previous works on MIP reformulations of sparsity-based problems [24], [27], tuning the value, M , in the “big- M ” reformulation impacts algorithmic performance. For a given problem, statistical rules may be used in order to infer reasonable M values. Then, new constraints in the optimization formulations may be added in order to reduce the feasible domain. For example, in [27], an upper bound on the ℓ_0 norm of the solution sought is considered. Furthermore, many signal processing problems naturally involve linear constraints such

as positivity or sum-to-one requirements. The proposed MIP-based approaches can easily be adapted to such cases, for which exact solutions can still be obtained. Adding such extra constraints may also contribute to reducing the computational time, whereas it generally penalizes the efficiency of classical (convex or greedy) sparse approximation algorithms. One may also consider directly the bi-objective optimization problem with multi-criterion optimization methods [44] in order to propose a whole range of trade-off (sparsity vs. data fitting) solutions.

Global optimization of criteria involving *structured sparsity* would also be worth being studied, where (possibly overlapping) subsets of coefficients are jointly zero or non-zero. Such problems are generally tackled by convex optimization approaches involving mixed norms [45] or by extensions of greedy algorithms [46]. Both suffer from similar limitations than their scalar ℓ_1 -norm relaxation and greedy counterparts, as far as optimality with respect to the ℓ_0 -based problem is concerned. We believe that exact optimization of such problems through MIP should also be possible for moderate-size problems. For example, MIP-like formulations of some structured sparsity problems are shown in [25]—although the authors finally resort to (inexact) continuous relaxation of the binary variables—and in [47], where specific structured sparsity problems defined through totally unimodular systems allow exact optimization in polynomial time.

REFERENCES

- [1] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. IEEE*, vol. 98, no. 6, pp. 948–958, Jun. 2010.
- [2] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Feb. 2009.
- [3] S. Chen, S. Billings, and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. J. Control*, vol. 50, no. 5, pp. 1873–1896, Nov. 1989.
- [4] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [5] Y. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 1993, pp. 40–44 vol.1.
- [6] A. J. Miller, *Subset Selection in Regression*, 2nd ed. London, UK: Chapman and Hall, 2002.
- [7] C. Soussen, J. Idier, D. Brie, and J. Duan, "From Bernoulli-Gaussian deconvolution to sparse signal restoration," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4572–4584, Oct. 2011.
- [8] N. Karahanoglu and H. Erdogan, "A* orthogonal matching pursuit: Best-first search for compressed sensing signal recovery," *Digit. Signal Process.*, vol. 22, no. 4, pp. 555–568, Jul. 2012.
- [9] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. A.*, vol. 26, no. 3, pp. 301–321, May 2009.
- [10] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
- [11] C. Soussen, R. Gribonval, J. Idier, and C. Herzet, "Joint k -step analysis of orthogonal matching pursuit and orthogonal least squares," *IEEE Trans. Inf. Theory*, vol. 59, pp. 3158–3174, May 2013.
- [12] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 289–301, Jan. 2009.
- [13] K. Herrity, A. Gilbert, and J. Tropp, "Sparse approximation via iterative thresholding," in *Proc. IEEE ICASSP*, vol. 3, 2006, pp. 624–627.
- [14] T. Blumensath and M. Davies, "Normalized iterative hard thresholding: Guaranteed stability and performance," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 298–309, Apr. 2010.
- [15] Z. Lu and Y. Zhang, "Sparse approximation via penalty decomposition methods," *SIAM J. Optimization*, vol. 23, no. 4, pp. 2448–2478, 2013.
- [16] M. S. O'Brien, A. N. Sinclair, and S. M. Kramer, "Recovery of a sparse spike time series by ℓ_1 norm deconvolution," *IEEE Trans. Signal Process.*, vol. 42, no. 12, pp. 3353–3365, Dec. 1994.
- [17] J. Mendel, "Some modeling problems in reflection seismology," *ASSP Magazine, IEEE*, vol. 3, no. 2, pp. 4–17, Apr. 1986.
- [18] S. Bourguignon, H. Carfantan, and J. Idier, "A sparsity-based method for the estimation of spectral lines from irregularly sampled data," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 575–585, Dec. 2007.
- [19] G. Tang, B. Bhaskar, and B. Recht, "Sparse recovery over continuous dictionaries - just discretize," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2013, pp. 1043–1047.
- [20] B. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comp.*, vol. 2, no. 24, pp. 227–234, Apr. 1995.
- [21] M.-D. Iordache, J. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.
- [22] A. Klein, H. Carfantan, D. Testa, A. Fasoli, J. Snipes, and JET EFDA Contributors, "A sparsity-based method for the analysis of magnetic fluctuations in unevenly-spaced Mirnov coils," *Plasma Phys. Contr. F.*, vol. 50, no. 12, p. 125005, 2008.
- [23] R. Bixby, "A brief history of linear and mixed-integer programming computation," *Doc. Math.*, vol. Optimization Stories, pp. 107–121, 2012.
- [24] S. Jokar and M. Pfetsch, "Exact and approximate sparse solutions of underdetermined linear equations," *SIAM J. Sci. Comp.*, vol. 31, no. 1, pp. 23–44, 2008.
- [25] I. Tosic and S. Drewes, "Learning joint intensity-depth sparse representations," *IEEE Trans. Image Process.*, vol. 23, no. 5, May 2014.
- [26] X. Sun, X. Zheng, and D. Li, "Recent advances in mathematical programming with semi-continuous variables and cardinality constraint," *J. Oper. Res. Soc. China*, vol. 1, no. 1, pp. 55–77, Mar. 2013.
- [27] N. B. Karahanoglu, H. Erdogan, and S. I. Birbil, "A mixed integer linear programming formulation for the sparse recovery problem in compressed sensing," in *Proc. IEEE ICASSP*, 2013, pp. 5870–5874.
- [28] I. Das and J. Dennis, "A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems," *Struct. Optimization*, vol. 14, no. 1, pp. 63–69, Aug. 1997.
- [29] R. J. Marks, G. L. Wise, D. S. Haldeman, and J. L. Whited, "Detection in Laplace noise," *IEEE Trans. Aerospace Electronic Syst.*, vol. 14, no. 6, pp. 866–872, Nov. 1978.
- [30] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, USA: Cambridge University Press, 2004.
- [31] C. Clason, " L^∞ fitting for inverse problems with uniform noise," *Inverse Probl.*, vol. 28, no. 10, p. 104007, Oct. 2012.
- [32] P. Stoica and Y. Selen, "Model-order selection: A review of information criterion rules," *IEEE Sig. Proc. Mag.*, vol. 21, no. 4, pp. 36–47, Jul. 2004.
- [33] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, Mar. 1997.
- [34] M. Nikolova, "Description of the minimizers of least squares regularized with ℓ_0 -norm. Uniqueness of the global minimizer," *SIAM J. Imaging Sciences*, vol. 6, no. 2, pp. 904–937, 2013.
- [35] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numer.*, vol. 22, pp. 1–131, May 2013.
- [36] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, USA: W. H. Freeman & Co., 1979.
- [37] A. Lodi, "Mixed integer programming computation," in *50 Years of Integer Programming 1958–2008*, M. Jünger *et al.*, Ed. Springer Berlin Heidelberg, 2010, pp. 619–645.
- [38] R. Gomory, "Outline of an algorithm for integer solutions to linear programs," *Bull. Amer. Math. Soc.*, vol. 64, no. 5, pp. 275–278, 1958.
- [39] F. Benhamou, F. Goualard, L. Granvilliers, and J.-F. Puget, "Revising hull and box consistency," in *Proc. ICLP*, 1999, pp. 230–244.
- [40] S. Mallat, *A wavelet tour of signal processing: The sparse way*. Elsevier, Academic Press, 2009.

- [41] D. M. Malioutov, M. Cetin, and A. S. Willsky, "Homotopy continuation for sparse signal representation," in *Proc. IEEE ICASSP*, vol. 5, Philadelphia, USA, Mar. 2005, pp. 733–736.
- [42] E. Carcreff, S. Bourguignon, J. Idier, and L. Simon, "Resolution enhancement of ultrasonic signals by up-sampled sparse deconvolution," in *Proc. IEEE ICASSP*, Vancouver, Canada, May 2013, pp. 6511–6515.
- [43] M. C. W. Van Rossum, "A novel spike distance," *Neural Computation*, vol. 13, no. 4, pp. 751–763, April 2001.
- [44] M. Ehrgott, *Multicriteria optimization*, 2nd ed. Springer, 2005.
- [45] R. Jenatton, J.-Y. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," *J. Mach. Learn. Res.*, vol. 12, pp. 2777–2824, Nov. 2011.
- [46] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," in *Proc. ICML*, New York, USA, 2009, pp. 417–424.
- [47] L. Baldassarre, N. Bhan, V. Cevher, and A. Kyrillidis, "Group-sparse model selection: Hardness and relaxation," *IEEE Trans. Inf. Theory*, to appear. arXiv preprint arXiv:1303.3207v4.