TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

institute of telecommunications

Generalized LASSO in Array Signal Processing

diploma thesis

Erich Zöchmann

matriculation number: 0925702

Vienna, April 14, 2015

advisors Prof. Christoph F. Mecklenbräuker
Institute of Telecommunications, TU Wien

Prof. Peter Gerstoft
Scripps Institution of Oceanography, University of California, San Diego

Abstract

Radar, seismic and and wireless communication systems observe waves (hidden in noise) by sensor arrays. Those systems infer the originating (spatially-sparse) set of sources, within a minimum prescribed resolution and with as few sensors as feasible. These requirements lead to under-determined systems of equations.

Compressed sensing is an active research field which treats the recovery of a set of sources from an under-determined system of equations exploiting sparsity. Naturally two main questions arise in this context. Firstly, what is the minimum number of equations/sensors for which reconstruction can be guaranteed and secondly, how to achieve efficient reconstruction regarding the sensors' observations?

Solutions to the first question provide insights into the design of linear measurements and use quantities like the restricted isometry property or coherence to describe well-behaved matrices, such as equiangular tight frames. Algorithms as answers to the second question mostly rely on the assumption, that the measurements were made in accordance to the reconstruction guarantees.

Suppose that one has to work with existing data acquisition systems, the measurement matrix is given a priori and algorithms studied under too idealistic assumptions are prone to failure.

This thesis shows that well known greedy algorithms like orthogonal matching pursuit are not suited for array processing problems. We devise an algorithm based on the generalized Least Absolute Shrinkage and Selection Operator (LASSO), which is a penalized least squares problem. The heuristically chosen ℓ_1 penalty term ensures strict convexity and that strong duality holds.

The corresponding dual problem is interpretable as a weighted conventional beamformer acting on the residuals of the LASSO. Based on physical insights provided by the dual problem's solution, three procedures for single snapshot reconstruction and one for sequential online reconstruction are proposed and analysed.

The sequential procedure assumes a weighted Laplace-like prior for the sources such that the maximum a posteriori source estimate at the current time step is the solution to a generalized LASSO problem. For the sequential implementation, the posterior distribution is fitted to the Laplace-like density by use of the dual solution.

Kurzfassung

Sensorgruppensignalverarbeitung für Wellenfelder erfolgt in vielen technischen Systemen wie der Funkortung und -abstandsmessung, der Seismologie und im Mobilfunk. Aus den Messungen möglichst weniger Sensoren soll Rückschluss über die räumliche Verteilung der Quellen gezogen werden. Werden weniger Sensoren als aufzulösende Quellpunkte verwendet, so führt dies zu unterbestimmten Gleichungssystemen.

Komprimiertes Abtasten (Compressed Sensing) bezeichnet ein immer noch aktuelles Teilgebiet der Linearen Algebra, in dem unterbestimmte lineare Gleichungssysteme unter einer Spärlichkeitsannahme (die meisten Einträge des Lösungvektors sind Null) gelöst werden.
Dabei stellen sich folgende Fragen: Was ist die minimale Anzahl an Gleichungen und wie kann, ausgehend von diesen, der Datenvektor rekonstruiert werden?

Die Beantwortung der ersten Frage liefert Einblicke in die Struktur der linearen Messungen. Die Qualität der Messmatrizen wird mittels der "Restricted Isometry Property" (RIP) oder der Kohärenz beschrieben. Algorithmen, die aus der Beantwortung der zweiten Frage resultieren, beruhen auf den zuvor gewonnenen Ergebnissen (Frage 1).

Werden existierende Datenerfassungssysteme verwendet, kann auf die Gestalt der Messmatrix nur wenig Einfluss genommen werden; oftmals zeigen die Matrizen unerwünschte Eigenschaften. Dadurch verringert sich die Effizienz jener Algorithmen, die eine niedrige RIP Konstante oder niedrige Kohärenz aufweisen.

Diese Arbeit zeigt, dass bereits bekannte "greedy algorithms", wie "orthogonal matching pursuit", nicht für Sensorgruppensignalverarbeitung geeignet sind. Unsere Minimierungsaufgabe wird durch ein verwandtes Problem - Hinzunahme eines ℓ_1 Terms in der Kostenfunktion statt der Spärlichkeitsannahme - ersetzt. Dieses verwandte, konvexe Problem ist bekannt als "generalized LASSO".

Das zugehörige duale Problem kann als gewichteter Beamformer interpretiert werden. Aus den dadurch gewonnenen physikalischen Einsichten können Algorithmen für die Rekonstruktion von Einzelmessungen und für die sequentielle Rekonstruktion gewonnen werden.

Für die sequentielle Rekonstruktion wird die a-priori Wahrscheinlichkeitdichte jedes Elementes als Laplace-ähnlich angenommen. Dadurch einsteht erneut ein "generalized LASSO" als maximum-a-posteriori Schätzer. Wiederum ist es die duale Variable, mittels der die posteriori Verteilung als Laplace-ähnlich genähert werden kann.

Contents

1. Introduction

1.1. Notation

The following incomplete table introduces the mathematical notation and variables which are extensively used throughout the thesis.

Table 1.1.: Table of mathematical notation and frequently used variables.

$\boldsymbol{M}, \boldsymbol{v}$	$\triangleq$	matrix and vector
$\boldsymbol{M}_{\mathcal{I}}$	$\triangleq$	matrix consisting of the columns of $\boldsymbol{M}$ indexed by $\mathcal{I}$
$\boldsymbol{v}[k]$	$\triangleq$	time - k dependent vector
$\mathcal{N}(\boldsymbol{M})$	$\triangleq$	nullspace of the matrix $\boldsymbol{M}$, i.e., $\{\boldsymbol{v} \big\vert \boldsymbol{M}\boldsymbol{v} = \boldsymbol{0}\}$
$\tilde{(\bullet)}$	$\triangleq$	linear transformation such as Fourier transform or matrix multiplication
$(\bullet)^{+}$	$\triangleq$	Moore-Penrose inverse
$(\bullet)^{-1}$	$\triangleq$	matrix inverse
$(\bullet)^{H}$	$\triangleq$	conjugate transposition
$(\bullet)^{-H}$	$\triangleq$	inverse of the conjugate transposed matrix
$\|\bullet\|_p$	$\triangleq$	ℓ_p norm, including the zero (pseudo) norm
$\|\bullet\|$	$\triangleq$	element-wise magnitude of a vector (scalar is a special case) or cardinality of a set
		Variables
$\boldsymbol{a}$		steering vector / atoms
$\boldsymbol{A}$		dictionary consisting of steering vectors $\boldsymbol{a}$
$\boldsymbol{D}$		generalization matrix, restricted to be a real, positive and diagonal matrix
k		time index
$\boldsymbol{k}$		wave vector
λ		wavelength or source prior hyperparameter
M		number of hypothetical source directions
$\mathcal{M}$		active set, support of $\boldsymbol{x}$
μ		regularization parameter
N		number of sensors
s		sparsity level, cardinality of $\mathcal{M}$
θ_m		direction corresponding to the m^{th} steering vector / column of $\boldsymbol{A}$
$\boldsymbol{u}$		Lagrange multiplier / dual vector
$\boldsymbol{x}$		complex source amplitudes (sparse vector)
$\boldsymbol{y}$		observations

1.2. Research Question

The aim of this diploma thesis is to find a computationally tractable program to solve an inverse array processing problem. By observing noisy wave signals we want to infer a spatially sparse set of directions of arrivals (DOA). We formalize this more mathematically:

Given $s \in \mathbb{N}$ spatially distributed far-field narrowband point sources with complex amplitudes $\boldsymbol{x} \in \mathbb{C}^M$ and an array of sensors modelled by a matrix (dictionary) $\boldsymbol{A} \in \mathbb{C}^{N \times M}$ ($N < M$), find a program to solve

$$\boldsymbol{x}_{\ell_0} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad \|\boldsymbol{x}\|_0 \leq s \,. \tag{P0}$$

The ℓ_2 norm as the typical least squares observation error minimizer is constraint to sparseness with the pseudo norm $\|\boldsymbol{x}\|_0$, which is simply a counter of non-zero entries in $\boldsymbol{x}$. Checking a variable in $\mathbb{C}$ for zero is numerically very insecure, so that we re-define (P0) such that it becomes a special case of a numerically stable version.
The active set $\mathcal{M}$ is defined as the set of all indices m with $|x_m| > \delta$, $\delta \geq 0 \ \in \mathbb{R}$

$$\mathcal{M} = \{m \mid |x_m| > \delta\} \,. \tag{1.1}$$

The zero norm constraint is now replaced by the cardinality of the active set and (P0) is rewritten to

$$\boldsymbol{x}_{\ell_0} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad |\mathcal{M}| \leq s \,. \tag{P0'}$$

For the choice of $\delta = 0$ we recover the original statement. In a second stage, this plain minimization problem should be modified such that it is able to model prior knowledge and allows for a sequential implementation.

1.3. Greedy Methods

Although the formulation of (P0) and (P0') appear quite intuitive they are NP-hard combinatorial problems [Nat95] and thus intractable already for small dimensions. Greedy algorithms which approximate the solution by incremental/iterative accumulation of the active set divide into two main classes.

Orthogonal Matching Pursuit (OMP) adds the index, which best fits the measurements, to the active set at each iteration and regresses onto the dictionary spanned by the active columns [TG07]. An extension to OMP is Compressive Sampling Matching Pursuit (CoSaMP), which allows to add multiple fitting indices at a time and hard thresholds after the regression to keep only s active indices [NT09].
The main problem of OMP (and CoSaMP) is the incremental reconstruction. The indices already found remain in the active set. If two source waves are impinging under nearby angles, they are interpreted as one joint source in the first iteration and the algorithm has no change to separate the sources later on. "Nearby" will be specified in terms of coherence. For the sake of completeness both algorithms are provided in Appendix A.1 and A.2.

Besides these matching pursuits there are iterative thresholding algorithms which basically soft or hard threshold the output of the matched filter $\boldsymbol{A}^H$. They are derived by minimizing a surrogate objective function [Ela10]. Thresholding algorithms are even more prone to failure for high coherence and will not be part of this thesis.

1.4. Penalized Least Squares

The most prominent idea to solve (P0) is to relax the zero norm to the closest convex p norm, i.e., the ℓ_1 norm [CRT06; Tro06]

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad \|\boldsymbol{x}\|_1 \leq \varepsilon \,. \tag{P1}$$

This relaxation suffers from the drawback that the dependency $\varepsilon(s)$ is non-linear and unknown. The explicit norm constraint of (P1) can be made implicit with Lagrange multipliers and was first discussed by Robert Tibshirani [Tib96] who gave it the name Least Absolute Shrinkage and Selection Operator (LASSO)

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{x}\|_1 \,. \tag{P1'}$$

The relation between μ and ε is known to be [Lor+09]

$$\mu = 2 \left\| \boldsymbol{A}^H \left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1}(\varepsilon) \right) \right\|_\infty \,, \qquad \varepsilon = \|\boldsymbol{x}_{\ell_1}(\mu)\|_1 \,. \tag{1.2}$$

The objective function of (P1') is a penalized least squares problem, where μ trades off the fidelity to the measurements and the sparsity level. It can be understood as Tikhonov regularization [Tik95] or in the context of variational analysis as proximity operator [RW09]. In a Bayesian framework this penalization could also be recognized as a maximum a posteriori (MAP) estimate [Gri11].
The idea of relaxing to the ℓ_1 norm is illustrated in Figure 1.1. For noise free measurements ($\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$) the solution vector is given for the smallest norm ball which explains the observation and depending on the selected $p \leq 1$ norm, the solution vector $\boldsymbol{x}_{\ell_p}$ may be sparse.

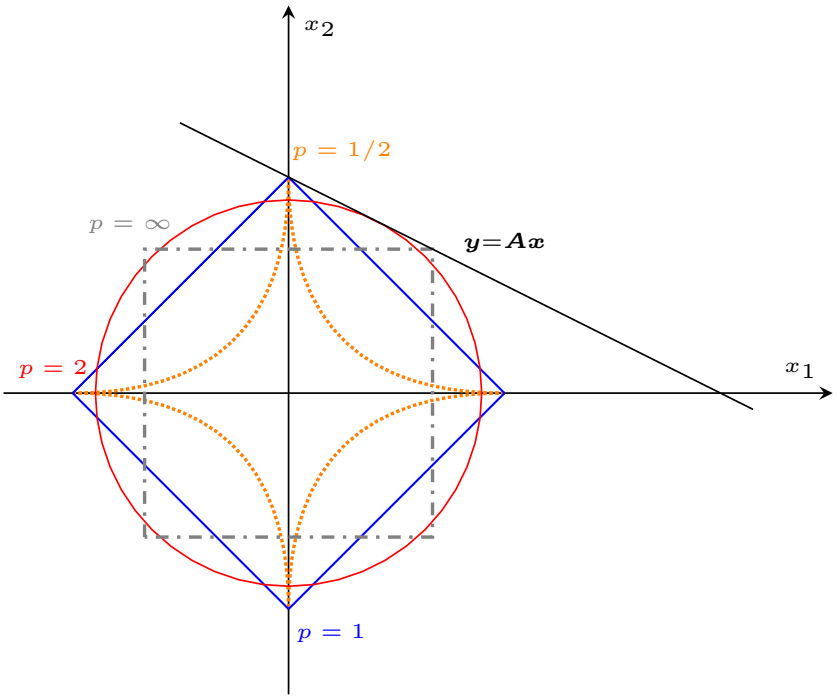


Figure 1.1.: Solution to an (noise free) under-determined system of equation $\boldsymbol{A}$ for different penalization norms. The norm balls of $p = 1/2$, $p = 1$, $p = 2$ and $p = \infty$ are shown in dotted-orange, blue, red and dash-dotted-black respectively. The sparse solution is only achieved with $p \leq 1$ norms.

1.5. Outline

Chapter 2 provides the mathematical model for the array processing setup. Through the way common sensor arrays, e.g., Uniform Linear Arrays (ULA), are built, the columns of $\boldsymbol{A}$ are almost parallel, which is termed coherent in compressed sensing (CS) literature. Most recovery guarantees for CS are assuming low coherence and we will show that OMP is prone to failure.

Chapter 3 therefore focusses on the LASSO since ℓ_1 relaxation methods are generally more robust to coherence. With ℓ_1 approaches we encounter new problems. Firstly a systematic error between the ℓ_0 and ℓ_1 solution will be described and secondly an estimator which relates the desired sparsity level to the regularization parameter is needed. Our workhorse to analyse the LASSO will be its dual problem.

Chapter 4 will then be devoted to the regularization parameter estimator. After explaining a necessary approximation we derive a slow OMP-like and a faster CoSaMP-like procedure. In return for a lengthy analysis of the dual problem to the generalized LASSO we are able to formulate a procedure which works entirely in the dual domain.

Chapter 5 presents the re-interpretation of ℓ_1 norm constraints as a (Laplacian) prior in a Bayesian estimation framework. By approximating the posterior distribution with a Laplacian, a sequential Bayesian estimation task is performed and a sequential algorithm is explained in detail.

Chapter 6 concludes the thesis with a short discussion on the results and on more advanced - atomic norm based - approaches.

2. Array Signal Processing

Array processing is a kind of space-time processing. Many sensors located at different positions in space observe waveforms at different times in order to draw conclusions about the spatial distribution of the wave sources.

2.1. Narrowband Far-Field Description

This section follows the arguments and mostly the notation of [VT02]. We assume the propagation medium to be linear, homogeneous, isotropic, time-invariant, deterministic, infinitely extended and lossless. The N uncoupled, linear, isotropic sensors sample the the incident field at positions $\boldsymbol{p}_n,\ n \in \{0, \dots, N-1\}$. The signal output of the N sensors at time t are stacked into one vector and scaled by $\sqrt{N}$ [1]

$$\tilde{\boldsymbol{y}}(\boldsymbol{p}, t) = \frac{1}{\sqrt{N}} \begin{pmatrix} \tilde{y}(\boldsymbol{p}_0, t) \\ \tilde{y}(\boldsymbol{p}_1, t) \\ \vdots \\ \tilde{y}(\boldsymbol{p}_{N-1}, t) \end{pmatrix} . \tag{2.1}$$

As the sensors and the propagation medium are assumed to be linear, we first analyse $\tilde{\boldsymbol{y}}(\boldsymbol{p}, t)$ for one impinging source wave. Many sources simply cause a superposition at the sensors output. Assuming that the sources are sufficiently far away, the incident field has a plane wave front, so that the sampled signal $\tilde{\boldsymbol{y}}$ consists of delayed copies of the same signal carried by the wave with wave vector $\boldsymbol{k}$. We are able to write

$$\tilde{\boldsymbol{y}}(\boldsymbol{p}, t, \boldsymbol{k}) = \frac{1}{\sqrt{N}} \begin{pmatrix} \tilde{y}(t - \tau_{\boldsymbol{p}_0}) \\ \tilde{y}(t - \tau_{\boldsymbol{p}_1}) \\ \vdots \\ \tilde{y}(t - \tau_{\boldsymbol{p}_{N-1}}) \end{pmatrix} \tag{2.2}$$

$$\overset{\mathcal{FT}}{\Leftrightarrow} \boldsymbol{y}(\boldsymbol{p}, \omega, \boldsymbol{k}) = \frac{1}{\sqrt{N}} \begin{pmatrix} e^{-j\omega\tau_{\boldsymbol{p}_0}} \\ e^{-j\omega\tau_{\boldsymbol{p}_1}} \\ \vdots \\ e^{-j\omega\tau_{\boldsymbol{p}_{N-1}}} \end{pmatrix} y(\omega) = \underbrace{\frac{1}{\sqrt{N}} \begin{pmatrix} e^{-j\boldsymbol{k}^T\boldsymbol{p}_0} \\ e^{-j\boldsymbol{k}^T\boldsymbol{p}_1} \\ \vdots \\ e^{-j\boldsymbol{k}^T\boldsymbol{p}_{N-1}} \end{pmatrix}}_{\boldsymbol{a}(\boldsymbol{k})} y(\omega) \overset{\text{monochr.}}{=} \boldsymbol{a}(\boldsymbol{k})$$

with delay $\tau_{\boldsymbol{p}_n} = \frac{\boldsymbol{k}^T\boldsymbol{p}_n}{\omega}$. Here $\tau_{\boldsymbol{p}_n}$ measures the travel time in propagation direction of the wavefront to sensor n, although it is very doubtful to assign a planar wave to a spatially concentrated source point; the reformulation with the wave vector $\boldsymbol{k}$ is therefore a more reasonable representation. The vector $\boldsymbol{a}$ captures all spatial characteristics of the spatial array and is called array manifold vector / steering vector. A monochromatic wave with unit amplitude is described as $\tilde{y}_n(t, \boldsymbol{p}_n) = e^{j\left(\omega t - \boldsymbol{k}^T\boldsymbol{p}_n\right)} = e^{j\omega t}\boldsymbol{a}_n(\boldsymbol{k})$, and its Fourier transform at frequency ω is $\boldsymbol{y}(\boldsymbol{p}, \omega, \boldsymbol{k}) = \boldsymbol{a}(k)$. The only remaining parameters for source

[1] The normalization step is not necessary at this point, but it will be beneficial later on as the measurement matrix have unit column ℓ_2-norm.

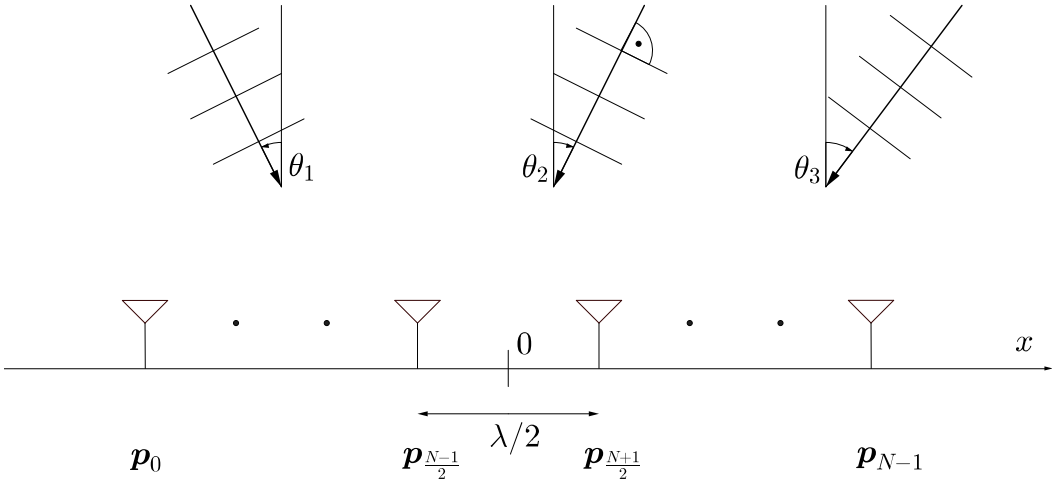


Figure 2.1.: Schematic illustration of the array processing problem with three impinging narrowband far-field sources and N sensors. W.l.o.g. the ULA is drawn for N even. The parallel lines reflect the plane wave fronts.

position inferring are the delay differences $\tau_{\boldsymbol{p}_n}$ to each sensor element. In Figure 2.1, three sources are illustrated, the sources corresponding to θ_2 and θ_3 cause a similar observation $\boldsymbol{y}$ and constitute the difficult, coherent estimation task.

2.2. Matrix Description

As the set of possible wavevectors is infinite, we have to discretize the angular space, such that we allow only for finite number of possible $\boldsymbol{k}$. For a discrete set of wavevectors $\boldsymbol{k}_m,\ m \in \{1, \dots, M\}$ the corresponding steering vectors form the matrix model

$$\boldsymbol{y} = \underbrace{\frac{1}{\sqrt{N}}\big(\boldsymbol{a}(\boldsymbol{k}_1), \boldsymbol{a}(\boldsymbol{k}_2), \dots, \boldsymbol{a}(\boldsymbol{k}_M)\big)}_{\boldsymbol{A} \in \mathbb{C}^{N \times M}} \begin{pmatrix} x_1(\boldsymbol{k}_1) \\ x_2(\boldsymbol{k}_2) \\ \vdots \\ x_M(\boldsymbol{k}_M) \end{pmatrix} . \tag{2.3}$$

In case that we use more discretization points M than sensor elements N, $\boldsymbol{A}$ is under-determined and there is no unique solution to (2.3); the sparsity constraint allows us to retrieve it.

Signal to Noise Ratio

The matrix model is augmented by additive noise. For this sub-section we explicitly express the time dependency in all relevant variables with square brackets (immediately thereafter, the brackets are dropped until Section 5.2)

$$\boldsymbol{y}[k] = \boldsymbol{A}\boldsymbol{x}[k] + \boldsymbol{n}[k] . \tag{2.4}$$

The noise is then assumed to be both spatially and temporally white Gaussian

$$\mathrm{E}\left\{\boldsymbol{n}[k]\boldsymbol{n}^H[k+l]\right\} = \begin{cases} \sigma^2 \boldsymbol{I}, & \text{for } l = 0, \\ \boldsymbol{0} & \text{otherwise} . \end{cases} \tag{2.5}$$

We define the Signal to Noise Ratio (SNR) as the ratio of the expected receive power in the absence of noise and the expected receive power for no signal present. With the noise definition and the assumption of a deterministic signal $\boldsymbol{x}$ the SNR at time step k becomes

$$\mathrm{SNR}[k] = 10\log_{10}\left(\frac{\mathrm{E}\left\{\left\|\boldsymbol{y}\big|_{\boldsymbol{n}\equiv\boldsymbol{0}}\right\|_2^2\right\}}{\mathrm{E}\left\{\left\|\boldsymbol{y}\big|_{\boldsymbol{x}\equiv\boldsymbol{0}}\right\|_2^2\right\}}\right) = 10\log_{10}\left(\frac{\mathrm{E}\left\{\|\boldsymbol{A}\boldsymbol{x}[k]\|_2^2\right\}}{\mathrm{E}\left\{\|\boldsymbol{n}[k]\|_2^2\right\}}\right) = 10\log_{10}\left(\frac{\|\boldsymbol{A}\boldsymbol{x}[k]\|_2^2}{N\sigma^2}\right) . \tag{2.6}$$

2.3. Uniform Linear Array

Throughout the whole thesis a $\lambda/2$ spaced horizontal ULA with $N = 30$ elements is utilized. Iff isotropic sensors are $n\frac{\lambda}{2},\ n \in \mathbb{N}$ spaced, their measurement are decoupled; in all other cases the model should consider coupling [IN10]. The sensors are placed on the x axis, centred around the origin

$$\boldsymbol{p}_{n,y} = \boldsymbol{p}_{n,z} = 0, \qquad \boldsymbol{p}_{n,x} = \left(n - \frac{N-1}{2}\right)\frac{\lambda}{2} . \tag{2.7}$$

The angular space is sampled linearly and the wavevectors in a homogeneous medium are expressed with unit vectors $\boldsymbol{e}$ [2]

$$\boldsymbol{k}_m = \frac{2\pi}{\lambda}\boldsymbol{e}_{\theta_m} , \qquad \boldsymbol{e}_{\theta_m} = \begin{pmatrix} \sin(\theta_m) \\ \times \\ \times \end{pmatrix}, \quad \theta_m = \frac{\pi(m-1)}{M} - \frac{\pi}{2} . \tag{2.8}$$

With all previous assumptions the elements of the steering vector $\boldsymbol{a}(\boldsymbol{k}_m) = \boldsymbol{a}(\theta_m)$ derive to

$$a_n(\theta_m) = \frac{1}{\sqrt{N}} e^{-j\pi\sin(\theta_m)\left(n-\frac{N-1}{2}\right)} , \tag{2.9}$$

and the matrix-vector description is given as

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} = \frac{1}{\sqrt{N}}\exp\left\{-j\pi\left(\begin{bmatrix} -\frac{N-1}{2} \\ \vdots \\ \frac{N-1}{2} \end{bmatrix}\left[\sin(\theta_1)\ \ \sin(\theta_2)\ldots\sin(\theta_M)\right]\right)\right\}\boldsymbol{x} . \tag{2.10}$$

2.4. Coherence

From (2.10) it is apparent that the measurement matrix $\boldsymbol{A}$ has almost parallel columns (atoms $\boldsymbol{a}$) near endfire $-90°$ and $90°$. The compressed sensing community titles this effect as coherence.
The columns of $\boldsymbol{A}$ are ℓ_2 normalized, i.e., $\|\boldsymbol{a}_m\|_2 = 1$, thereby the coherence measures the correlation between two steering vectors by means of their inner product

$$\mathrm{coh}(\boldsymbol{a}_i, \boldsymbol{a}_j) = \boldsymbol{a}_i^H \boldsymbol{a}_j , \tag{2.11}$$

and the mutual coherence is the maximum of all possible coherences

$$\text{mutual coh}(\boldsymbol{A}) = \left\|\boldsymbol{A}^H\boldsymbol{A} - \boldsymbol{I}\right\|_{\max} = \left\|\mathrm{vec}\left(\boldsymbol{A}^H\boldsymbol{A} - \boldsymbol{I}\right)\right\|_\infty , \tag{2.12}$$

where the $\max$ matrix norm is equal to the element of largest magnitude which is identical to the ℓ_∞ norm of the vectorized matrix.

[2] The y and z values of the unit vectors do not enter in the formula. They "don't care" $\Rightarrow \times$.

The mutual coherence serves as measure for the compressive sensing suitability of $\boldsymbol{A}$. OMP for instance has recovery guarantees based on the mutual coherence. Figure 2.2 depicts the coherence of our 30 sensor ULA and degree-wise sampling between $-90°$ and $90° \Rightarrow \boldsymbol{A} \in \mathbb{C}^{30\times 181}$.

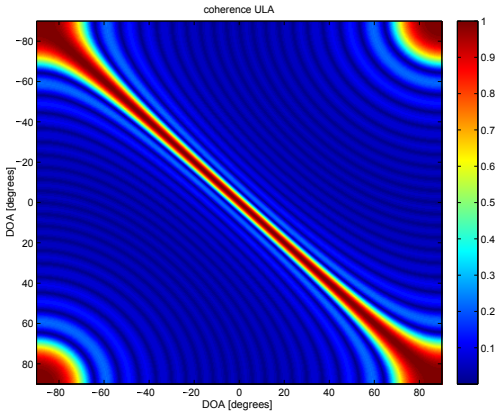


Figure 2.2.: Magnitude of the coherence $\boldsymbol{A}^H\boldsymbol{A}$ of the 30 sensors ULA and a degree-wise discritization between $-90°$ and $+90°$. The regions of strongest coherence displayed in red are broader at endfire.

From Figure 2.2 we are able to conclude that coherence is really dependent on the source angle. Near endfire ($\pm 90°$), nearby sources always look strongly coherent; they produce very similar measurements on an ULA. At broadside they can come much closer and still produce more distinguishable measurements. Hence, *whenever we talk about coherence later on, we implicitly mean the mutual coherence of the restricted dictionary* $\boldsymbol{A}_{\mathcal{M}}$.

2.4.1. OMP Recovery Guarantees

A further insight from (2.10) is the dependency of the coherence on the angular sampling. The finer the grid the stronger the coherence will be! Next we analyse this relationship with well known OMP recovery guarantees, similar to [TGA14].
OMP yields successful recovery for s-sparse signals if [Tro04]

$$\text{mutual coh}(\boldsymbol{A}) \leq \frac{1}{2s-1}\,. \tag{2.13}$$

Due to the structure of the dictionary $\boldsymbol{A}$ we know that the coherence is maximum between neighbouring columns which are $\Delta\theta = \frac{c\pi}{M}$, $c \ll 1$, spaced. W.l.o.g. we choose $\Delta\theta = \frac{1}{M}$ and the mutual coherence becomes

$$\begin{aligned}\text{mutual coh}(\boldsymbol{A}) &= \max_{\boldsymbol{a}(\theta_m)} |\operatorname{coh}(\boldsymbol{a}(\theta_m), \boldsymbol{a}(\theta_m+\Delta\theta))| \\ &= \max_{\theta_m} \frac{1}{N}\left| \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} e^{j\pi n(\sin(\theta_m+\Delta\theta)-\sin(\theta_m))}\right| \approx \frac{1}{N}\left| \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} e^{j\pi n\Delta\theta}\right|.\end{aligned} \tag{2.14}$$

In the last step the small-angle approximation ($c \ll 1$!) was used. The right hand side of (2.14) is known as Dirichlet kernel which is approximated by a sinc function

$$\frac{1}{N}\left| \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} e^{j\pi n\Delta\theta}\right| = \frac{1}{N}\left|\frac{\sin(\pi\Delta\theta\frac{N}{2})}{\sin(\pi\Delta\theta)}\right| \approx \left|\operatorname{sinc}\left(\Delta\theta\frac{N}{2}\right)\right|. \tag{2.15}$$

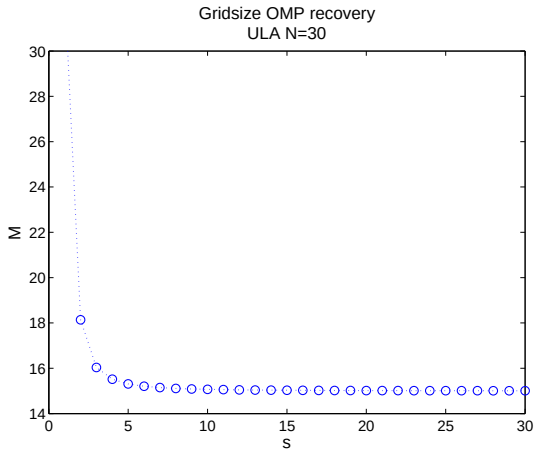


Figure 2.3.: Maximum gridsize for guaranteed OMP recovery of a 30 sensor ULA.

Whenever the angular discritization is smaller than $\frac{2}{N}$ (this is safely fulfilled as we would like to discritize such that an under-determined system of equations $N < M \rightarrow \frac{1}{2}\frac{N}{M} < \frac{1}{2}$ results) the sinc function is positive and has a well defined inverse

$$\text{mutual coh}(\boldsymbol{A}) \approx \mathrm{sinc}\left(\Delta\theta\frac{N}{2}\right) \leq \frac{1}{2s-1} \Leftrightarrow \frac{1}{M} = \Delta\theta \geq \frac{2}{N}\,\mathrm{sinc}^{-1}\left(\frac{1}{2s-1}\right). \tag{2.16}$$

Because $\mathrm{sinc}(\xi),\ \xi \in (0,1)$ is monotonically decreasing, the greater equal sign flips if (2.16) is inverted.

$$M \leq \frac{N}{2}\frac{1}{\mathrm{sinc}^{-1}\left(\frac{1}{2s-1}\right)} \tag{2.17}$$

The bound of inequality (2.17) is drawn in Figure 2.3. For $s = 1$ there is a pole and arbitrarily fine sampling is possible, but for $s \geq 2$ sparse recovery is not guaranteed. The result of this simple analysis is quite remarkable. It shows that greedy algorithms such as OMP, CoSaMP and thresholding algorithms (which we disqualified from the beginning due to their even stronger sensitivity to coherence) are not suited for array processing problems in their plain form.

2.4.2. Offset Regions due to Coherence

Another negative effect of coherence was demonstrated in [XGM14]. For $\mathrm{SNR} > 6\,\mathrm{dB}$, coherence defines a region of possible offsets around the true DOA θ_m associated with steering vector $\boldsymbol{a}_m$. All possible outcomes are from the set

$$\mathcal{O} = \left\{\boldsymbol{a_i} \mid \mathrm{coh}\left(\boldsymbol{a_i}, \boldsymbol{a_m}\right) \geq \sqrt{1 - 4 \cdot 10^{-\frac{\mathrm{SNR}}{10}}}\right\}. \tag{2.18}$$

The offset region $\mathcal{O}$ will be shown with circles around the true DOA in the figures of the next chapter. An intuitive argument is: Due to high coherence in relation to the SNR, a steering vector close to the correct one leads to a stronger decrease of the objective function and is therefore a valid solution of the optimization program.

3. Generalized LASSO

In the previous chapter we have seen that coherence limits the performance of greedy algorithms. The current chapter is devoted to ℓ_1 relaxation methods. Various works have shown the robustness of ℓ_1 methods to coherence or more frequently to the Restricted Isometry Property (RIP). The RIP constant δ_s for sparsity degree s is related to the mutual coherence

$$\delta_s(\boldsymbol{A}) \leq s\,\text{mutual coh}(\boldsymbol{A})\;. \tag{3.1}$$

Basis Pursuit (BP) for instance allows for more than 3.7 times higher RIP constants than OMP [FR13].
In literature there are several different names given to a related approach. BP minimizes $\|\boldsymbol{x}\|_1$ while keeping the measurement mismatch bounded $\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2 \leq \eta$, Basis Pursuit De-Noising (BPDN) minimizes the measurement mismatch under an ℓ_1 constraint. Both approaches yield the same Lagrangian formulation and can be converted into each other by inverting the regularization parameter $\mu_{BP} = {\mu_{BPDN}}^{-1}$.

The Lagrangian formulation of BP and BPDN - called LASSO - was originally intended as a statistical tool, which naturally selects a subset of variables of strongest effect and therefore rather proposed as a regression program for over-determined system of equations.

3.1. Generalization

A matrix $\boldsymbol{D}$ of arbitrary shape and structure was added to the penalty term by Ryan Tibshirani [TT11]. He named the new program generalized LASSO. Throughout the whole text we assume the matrix $\boldsymbol{D}$ to be quadratic and non-singular. This severe constraint actually means that the generalized LASSO could be considered as the plain LASSO program

$$\min_{\boldsymbol{x}} \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\underbrace{\boldsymbol{D}\boldsymbol{x}}_{\tilde{\boldsymbol{x}}}\|_1 = \min_{\tilde{\boldsymbol{x}}} \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{D}^{-1}\tilde{\boldsymbol{x}}\|_2^2 + \mu\|\tilde{\boldsymbol{x}}\|_1\;. \tag{3.2}$$

As most of the approaches in this thesis are inspired by the work of Ryan Tibshirani, we keep the name generalized LASSO.
Our workhorse to solve (P0) is the following (still non-convex) constrained generalized LASSO problem

$$\begin{aligned}\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\text{argmin}} &\left(\min_{\mu>0}\left(\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{D}\boldsymbol{x}\|_1\right)\right)\\ &\text{subject to}\quad \|\boldsymbol{x}\|_0 \leq s\;.\end{aligned} \tag{P2}$$

The zero norm of the generalized LASSO is piecewise constant as illustrated in Figure 3.1 so that the constraint of (P2) is translated to a regularization parameter interval

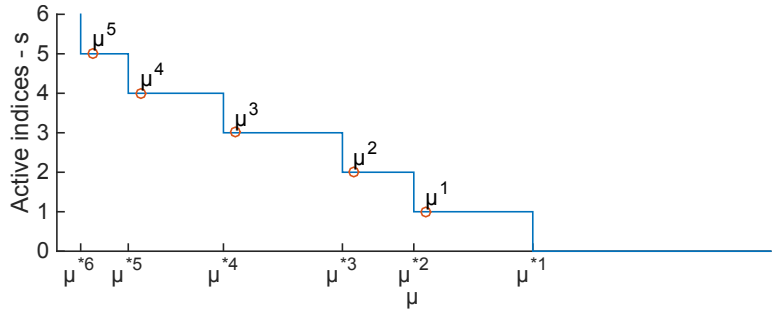


Figure 3.1.: The number of active indices as a function of the regularization parameter μ. The active set changes at candidate points μ^{*i} and is constant in between. The red circles indicate excellent regularization parameter μ^i for an active set size i.

$$\begin{aligned} \boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \left(\min_{\mu>0} \left(\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{D}\boldsymbol{x}\|_1 \right) \right) \\ \text{subject to} \quad \mu^{*(s+1)} < \mu < \mu^{*s} \,, \end{aligned} \tag{P2'}$$

which yields an unconstrained generalized LASSO problem if the regularization parameter would be known

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \underbrace{(1+\epsilon)\mu^{*(s+1)}}_{\mu} \|\boldsymbol{D}\boldsymbol{x}\|_1 \,. \tag{P2''}$$

Consequently, the aim of the current and the following chapter is to motivate and derive the regularization parameter estimator. The dual problem of the generalized LASSO will be our tool of choice.

3.2. Lagrangian and Dual Problem

The discussion of the dual problem follows the notation introduced in [BV04]. A primal complex-valued convex problem in standard form reads

$$\begin{aligned} &\min_{\boldsymbol{x}} f_0(\boldsymbol{x}) \\ &\text{subject to } f_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m \\ &\qquad\qquad h_j(\boldsymbol{x}) = 0, \quad j = 1, \ldots, q \,, \end{aligned} \tag{3.3}$$

with a convex objective function $f_0 : \mathbb{C}^M \to \mathbb{R}$, convex inequality constraint $f_i : \mathbb{C}^M \to \mathbb{R}$ and affine equality constraints $h_j : \mathbb{C}^M \to \mathbb{C}$. The primal problem (P2″) does not have any constraint but pursuing as in [TT11], we introduce trivial equality constraints

$$\min_{\boldsymbol{x},\boldsymbol{z}} \left(\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{z}\|_1 \right) \quad \text{subject to} \quad \boldsymbol{z} = \boldsymbol{D}\boldsymbol{x} \,. \tag{3.4}$$

The real Lagrangian to the standard form is formulated via Lagrangian multipliers $\{u_i \in \mathbb{R} | u_i \geq 0\}$ and $v_i \in \mathbb{C}$

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m} u_i f_i(\boldsymbol{x}) + \operatorname{Re}\left[\sum_{j=1}^{q} v_j h_j(\boldsymbol{x}) \right] \,. \tag{3.5}$$

The Lagrangian to (P2″) develops into

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{z}\|_1 + \mathrm{Re}\left[\boldsymbol{u}^H(\boldsymbol{D}\boldsymbol{x} - \boldsymbol{z})\right] = \mathcal{L}_1(\boldsymbol{x}, \boldsymbol{u}) + \mathcal{L}_2(\boldsymbol{z}, \boldsymbol{u})\,. \tag{3.6}$$

Note that the Lagrangian is a sum of two Lagrangians, the first depending on the primal vector $\boldsymbol{x}$ and the dual vector $\boldsymbol{u}$ and the second on the dual vector and our auxiliary vector $\boldsymbol{z}$ only.
The dual function is defined as infimum of the Lagrangian over the primal variables

$$g(\boldsymbol{u}, \boldsymbol{v}) = \inf_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v})\,. \tag{3.7}$$

For the generalized LASSO we have to distinguish between two cases. The dual function is only unbounded below if the elements of the dual vector are bounded and within a subspace of $\boldsymbol{A}$ given by the basis $\boldsymbol{U}$. The next sub-section will be devoted to the derivation of the dual function

$$g(\boldsymbol{u}) = \begin{cases} \boldsymbol{y}^H\boldsymbol{y} - \|\boldsymbol{A}\boldsymbol{A}^+\boldsymbol{y} - \frac{1}{2}(\boldsymbol{A}^H)^+\boldsymbol{D}^H\boldsymbol{u}\|_2^2 & \text{if} \quad \|\boldsymbol{u}\|_\infty \leq \mu \quad \text{and} \quad (\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u} = \boldsymbol{0}, \\ -\infty \quad \text{otherwise}\,. \end{cases} \tag{3.8}$$

As the Lagrangian multipliers of the inequality constraints are non-negative, the maximum of the dual function, i.e., the dual problem, is always less or equal to the minimum of the primal problem.

$$\begin{aligned} &\max_{\boldsymbol{u},\boldsymbol{v}} g(\boldsymbol{u}, \boldsymbol{v}) \quad \left\{ \leq \min_{\boldsymbol{x}} f_0(\boldsymbol{x}) \right\} \\ &\text{subject to} \quad u_i \geq 0, \;\; i = 1, \ldots, m \end{aligned} \tag{3.9}$$

Our primal problem is unconstrained (we only introduced trivial constraints), therefore strong duality holds. Strong duality implies the equality of the objective function value of the primal and dual problem. Strong duality is a nice property, but is not exploited as we are only interested in the minimizers $\boldsymbol{x}, \boldsymbol{u}$ and not in the objective function value itself.
The dual problem of the generalized LASSO is hence a magnitude and subspace constraint least squares (LS) problem

$$\max_{\boldsymbol{u} \in \mathbb{C}^M} \boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2 \tag{3.10}$$

$$\text{subject to} \quad \|\boldsymbol{u}\|_\infty \leq \mu, \tag{3.11}$$

$$(\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u} = \boldsymbol{0}, \tag{3.12}$$

where $\boldsymbol{U}$ is a basis of $\mathcal{N}(\boldsymbol{A})$ and the following abbreviations

$$\tilde{\boldsymbol{D}} = \tfrac{1}{2}\boldsymbol{D}\boldsymbol{A}^+, \tag{3.13}$$

$$\boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{P}_{\mathrm{span}(\boldsymbol{A})}, \tag{3.14}$$

$$\tilde{\boldsymbol{y}} = \boldsymbol{P}_{\mathrm{span}(\boldsymbol{A})}\,\boldsymbol{y}\,, \tag{3.15}$$

were introduced. Projection matrices such as $\boldsymbol{P}_{\mathrm{span}(\boldsymbol{A})}$ are discussed in Appendix B.2.

Derivation of the Dual Function

To derive the dual function, the Lagrangian is minimized over $\boldsymbol{x}$ and $\boldsymbol{z}$. The Lagrangian partitions nicely into a $\boldsymbol{x}$ dependent and a $\boldsymbol{z}$ dependent part so that the minimization is done separately. The two

partitions are

$$\mathcal{L}_1(\boldsymbol{x}, \boldsymbol{u}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mathrm{Re}\left[\boldsymbol{u}^H \boldsymbol{D}\boldsymbol{x}\right] \tag{3.16}$$

$$\mathcal{L}_2(\boldsymbol{z}, \boldsymbol{u}) = \mu\|\boldsymbol{z}\|_1 - \mathrm{Re}[\boldsymbol{u}^H \boldsymbol{z}] \,. \tag{3.17}$$

Partition $\mathcal{L}_1$ is differentiable and the minimizer $\hat{\boldsymbol{x}}$ is found by differentiation, which gives

$$\boldsymbol{D}^H \boldsymbol{u} = 2\boldsymbol{A}^H \left(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}}\right) \,. \tag{3.18}$$

If $\boldsymbol{D}^H \boldsymbol{u} \in \mathrm{span}(\boldsymbol{A}^H)$, i.e., it is orthogonal to $\mathcal{N}(\boldsymbol{A}) \Rightarrow \boldsymbol{U}^H \boldsymbol{D}^H \boldsymbol{u} = \boldsymbol{0}$, where $\boldsymbol{U}$ is a basis of $\mathcal{N}(\boldsymbol{A})$, the minimizer solution $\hat{\boldsymbol{x}}$ derives to

$$\hat{\boldsymbol{x}} = \underbrace{\boldsymbol{A}^+ \boldsymbol{y} + \boldsymbol{\xi}}_{\hat{\boldsymbol{x}}_{\mathrm{LS}}} - \frac{1}{2}(\boldsymbol{A}^H \boldsymbol{A})^+ \boldsymbol{D}^H \boldsymbol{u} \,, \tag{3.19}$$

with $\boldsymbol{\xi} \in \mathcal{N}(\boldsymbol{A})$. The first part of $\mathcal{L}_1$ gets expanded to

$$\|\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 = \|\boldsymbol{y}\|_2^2 + \|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 - 2\,\mathrm{Re}[\boldsymbol{y}^H \boldsymbol{A}\hat{\boldsymbol{x}}] \,, \tag{3.20}$$

and the second part is expressed solely by $\hat{\boldsymbol{x}}$ using (3.18),

$$\mathrm{Re}\left[(\boldsymbol{D}^H \boldsymbol{u})^H \hat{\boldsymbol{x}}\right] = \mathrm{Re}\left[2(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}})^H \boldsymbol{A}\hat{\boldsymbol{x}}\right] = 2\,\mathrm{Re}\left[\boldsymbol{y}^H \boldsymbol{A}\hat{\boldsymbol{x}}\right] - 2\|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 \,. \tag{3.21}$$

Equations (3.20) and (3.21) are summed and $\hat{\boldsymbol{x}}$ is replaced by (3.19) to arrive at

$$\mathcal{L}_1(\hat{\boldsymbol{x}}, \boldsymbol{u}) = \|\boldsymbol{y}\|_2^2 - \|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 = \boldsymbol{y}^H \boldsymbol{y} - \|\boldsymbol{A}\boldsymbol{A}^+ \boldsymbol{y} - \tfrac{1}{2}(\boldsymbol{A}^H)^+ \boldsymbol{D}^H \boldsymbol{u}\|_2^2 = \boldsymbol{y}^H \boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H \boldsymbol{u}\|_2^2 \,. \tag{3.22}$$

To minimize $\mathcal{L}_2$ we first write the ℓ_1 norm and the inner product as a sum and transform the complex entries to polar coordinates $\mathrm{Re}[u_m^* z_m] = \mathrm{Re}[|u_m|\,|z_m|\,e^{i\phi}]$

$$\mathcal{L}_2(\boldsymbol{z}, \boldsymbol{u}) = \mu\|\boldsymbol{z}\|_1 - \mathrm{Re}[\boldsymbol{u}^H \boldsymbol{z}] = \sum_{m=1}^{M} \left(\mu|z_m| - \mathrm{Re}[u_m^* z_m]\right) = \sum_{m=1}^{M} \underbrace{\left(\mu - |u_m|\cos\phi\right)}_{c_m \geq 0\,?} |z_m| \,. \tag{3.23}$$

If all coefficients $c_m \geq 0$ for all choices $z_m \in \mathbb{C}$ then

$$\inf_{\boldsymbol{z}} \left(\mu\|\boldsymbol{z}\|_1 - \mathrm{Re}[\boldsymbol{u}^H \boldsymbol{z}]\right) = 0, \tag{3.24}$$

otherwise (3.23) is unbounded below. As $\cos(\phi)$ is always less or equal to one is suffices that all $|u_m|$ are bounded by μ.

3.3. Theorems and Corollaries to the Dual Problem

The following theorem and corollaries are a direct quotation of [MGZ15] and [ZGM15].

3.3.1. Beamforming on the LASSO residuals

Theorem 1. *If $\boldsymbol{D}$ is non-singular, the dual vector $\boldsymbol{u}$ is the output of a weighted conventional beamformer acting on the vector of residuals, i.e.*

$$\boldsymbol{u} = 2\boldsymbol{D}^{-H} \boldsymbol{A}^H (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1}) \,, \tag{3.25}$$

where $\boldsymbol{x}_{\ell_1}$ is such that the box constraint

$$\|\boldsymbol{u}\|_\infty \leq \mu \tag{3.26}$$

is fulfilled.

Proof. This is derived straight forward from the results of Section 3.2 by knowing that the minimizer of the Lagrangian $\hat{\boldsymbol{x}}$ is also the minimizer of (P2″) at a unique Lagrange multiplier $\boldsymbol{u}$, i.e. $\boldsymbol{x}_{\ell_1}$. To facilitate reading we name the nullspace part $\boldsymbol{x}_{\ell_1}^{\text{null}}$ instead of $\boldsymbol{\xi}$ in some latter figures. □

3.3.2. Implications of the Active Set on the Dual Vector

Corollary 1. *If the m^{th} primal coordinate is active, i.e. $x_{\ell_1,m} \neq 0$ then the box constraint (3.26) is tight in the m^{th} dual coordinate. Formally, for any choice of $\delta > 0$*

$$|x_{\ell_1,m}| > \delta \quad \Rightarrow \quad |u_m| = \mu, \qquad (m = 1, \ldots, M). \tag{3.27}$$

Informally, we say that the m^{th} dual coordinate hits the boundary when the m^{th} primal coordinate becomes active. Conversely, when the bound on $|u_m|$ is loose (i.e. the constraint on u_m is inactive), the corresponding primal variable x_m is zero (the m^{th} primal coordinate is inactive). The active set $\mathcal{M}$ is

$$\mathcal{M} \;=\; \big\{m \,\big|\, |x_{\ell_1,m}| > \delta\big\} \;\subseteq\; \big\{m \,\big|\, |u_m| = \mu\big\} \;=\; \mathcal{U}\,. \tag{3.28}$$

Proof. In [MGZ15], Corollary 1 was proven with complex valued sub-gradients. Here, a different longer but more insightful approach based on the dual problem of the dual problem is presented. We first have to express the inequality (3.26) in a different fashion

$$\|\boldsymbol{u}\|_\infty \leq \mu \Leftrightarrow \underbrace{u_m}_{|u_m|e^{i\phi_m}} e^{-i\phi_m} - \mu \leq 0, \quad \forall m\,. \tag{3.29}$$

We incorporate the explicit constraints of the dual problem into the objective function via the Lagrangian multipliers $\boldsymbol{\lambda} \in \mathbb{R}^M,\ \lambda_i \geq 0,\ \forall i = 1 \ldots M$ and $\boldsymbol{\nu} \in \mathbb{C}^M$

$$\mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2 + \boldsymbol{\lambda}^H\left(\boldsymbol{u}e^{-i\boldsymbol{\phi}} - \mu\right) + \mathrm{Re}\left\{\boldsymbol{\nu}^H(\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u}\right\}\,. \tag{3.30}$$

The dual problem is concave, thus the dual-dual must solve the Karush-Kuhn-Tucker (KKT) conditions. We already have "primal feasibility" (here it should be termed dual feasibility) and "dual feasibility" (dual-dual feasibility). Next the gradient must evaluate to zero

$$\nabla_{\boldsymbol{u}}\mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = 2\tilde{\boldsymbol{D}}(\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}) + \boldsymbol{\lambda}e^{i\boldsymbol{\phi}} + (\boldsymbol{D}\boldsymbol{U})\boldsymbol{\nu} = \boldsymbol{0}\,. \tag{3.31}$$

Now we write out our abbreviations for $\tilde{\boldsymbol{D}}$ and $\tilde{\boldsymbol{y}}$ to express $\boldsymbol{\lambda}$ in terms of the remaining components

$$-\boldsymbol{\lambda}e^{i\boldsymbol{\phi}} = 2\tfrac{1}{2}\boldsymbol{D}\boldsymbol{A}^+(\boldsymbol{A}\boldsymbol{A}^+\boldsymbol{y} - \tfrac{1}{2}(\boldsymbol{A}^+)^H\boldsymbol{D}^H\boldsymbol{u}) + (\boldsymbol{D}\boldsymbol{U})\boldsymbol{\nu} \tag{3.32}$$

$$\boldsymbol{D}^{-1}\boldsymbol{\lambda}e^{i\boldsymbol{\phi}} = \underbrace{\boldsymbol{A}^+\boldsymbol{A}\boldsymbol{A}^+}_{\boldsymbol{A}^+}\boldsymbol{y} - \tfrac{1}{2}\underbrace{\boldsymbol{A}^+(\boldsymbol{A}^+)^H}_{(\boldsymbol{A}^H\boldsymbol{A})^+}\boldsymbol{D}^H\boldsymbol{u} + \underbrace{\boldsymbol{U}\boldsymbol{\nu}}_{\boldsymbol{\xi}} \tag{3.33}$$

$$\boldsymbol{D}^{-1}\boldsymbol{\lambda}e^{i\boldsymbol{\phi}} = \boldsymbol{A}^+\boldsymbol{y} + \boldsymbol{\xi} - \tfrac{1}{2}(\boldsymbol{A}^H\boldsymbol{A})^+\boldsymbol{D}^H\boldsymbol{u} \tag{3.34}$$

comparing with Equation (3.19)

$$\boldsymbol{D}^{-1}\boldsymbol{\lambda}e^{i\boldsymbol{\phi}} \stackrel{!}{=} \hat{\boldsymbol{x}} \Leftrightarrow \boldsymbol{\lambda} = \boldsymbol{D}|\hat{\boldsymbol{x}}|\,, \tag{3.35}$$

where $|\hat{\boldsymbol{x}}|$ is short for the vector of all magnitudes.
The last missing KKT condition is *complementary slackness*

$$\boldsymbol{\lambda}^H \left(\boldsymbol{u}e^{-i\boldsymbol{\phi}} - \mu\right) = \mathbf{0} \tag{3.36}$$

$$|\hat{\boldsymbol{x}}|^H \left(\boldsymbol{u}e^{-i\boldsymbol{\phi}} - \mu\right) = \mathbf{0}\,. \tag{3.37}$$

We are now able to conclude the following statements[1]

$$|\hat{x}_m| > 0 \quad \Rightarrow \quad u_m = \mu e^{i\phi} \tag{3.38}$$

$$u_m < \mu e^{i\boldsymbol{\phi}} \quad \Rightarrow \quad |\hat{x}_m| = 0\,. \tag{3.39}$$

The nullspace part $\boldsymbol{\xi}$ may differ from $\boldsymbol{x}_{\ell_1}^{\text{null}}$ but all terms which lie in $\text{span}(\boldsymbol{A}^H)$ are equal. As there is always a part of the solution in $\text{span}(\boldsymbol{A}^H)$, simply because the noise is not zero there, we safely replace $\hat{\boldsymbol{x}}$ by $\boldsymbol{x}_{\ell_1}$ in all conclusions above.

□

3.3.3. Phase Equality of the Primal and the Dual Solution

Corollary 2. *If matrix $\boldsymbol{D}$ is diagonal with real-valued positive diagonal entries, then the phase angles of the corresponding entries of the dual and primal solution vectors are equal.*

$$\arg(u_m) = \arg(x_{\ell_1,m}), \quad \forall m \in \mathcal{M} \tag{3.40}$$

Proof. The objective function of the complex-valued generalized LASSO problem (P2″) is assigned to the symbol f_0

$$f_0(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{D}\boldsymbol{x}\|_1\,. \tag{3.41}$$

The subderivative $\partial f_0(\boldsymbol{x})$ as introduced in [Ber99] is the set of all subgradients $\boldsymbol{g}$. [BST12] extended this concept to complex valued functions by use of Wirtinger's calculus. A vector $\boldsymbol{g}$ is said to be subgradient at $\boldsymbol{x}_0$ if the inequality

$$f_0(\boldsymbol{x}) \geq f_0(\boldsymbol{x}_0) + \boldsymbol{g}^H\left(\boldsymbol{x} - \boldsymbol{x}_0\right)\,, \tag{3.42}$$

is fulfilled. A subgradient equals the gradient if $f_0(\boldsymbol{x})$ is differentiable at $\boldsymbol{x}$ and exists even if the function f_0 is not differentiable at $\boldsymbol{x}_0$; $\boldsymbol{g}$ is not unique then. With this definition the subderivate is written as

$$\partial f_0(\boldsymbol{x}_0) = \left\{\boldsymbol{g} \middle| f_0(\boldsymbol{x}) \geq f_0(\boldsymbol{x}_0) + \boldsymbol{g}^H\left(\boldsymbol{x} - \boldsymbol{x}_0\right)\right\}\,. \tag{3.43}$$

The subderivative for the LASSO objective function is thus expressible as

$$\partial f_0(\boldsymbol{x}) = -\underbrace{2\boldsymbol{A}^H(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})}_{\boldsymbol{D}^H\boldsymbol{u}} + \mu\,\partial\|\boldsymbol{D}\boldsymbol{x}\|_1\,. \tag{3.44}$$

Next, with the restrictions for $\boldsymbol{D}$ - diagonal matrix with positive real-valued diagonal entries - the subderivative of the penalty term becomes

$$\partial\|\boldsymbol{D}\boldsymbol{x}\|_1 = \begin{cases} D_{mm}\frac{x_m}{|x_m|} & \text{for} \quad D_{mm}x_m \neq 0 \\ \\ \{z \in \mathbb{C}, |z| \leq 1\} & \text{for} \quad D_{mm}x_m = 0\,. \end{cases} \tag{3.45}$$

[1] Unfortunately, the very useful statement $|u_m| = \mu \Rightarrow |\hat{x}_m| > 0$ cannot be concluded this way!

The minimality condition for $f_0(\boldsymbol{x})$ is equivalent to setting (3.44) to zero. For all m with $D_{mm}x_m \neq 0$ it follows that

$$D^*_{mm}\frac{u_m}{\mu} = D_{mm}\frac{x_m}{|x_m|}\ . \tag{3.46}$$

Corollary 1 allows us to replace μ by $|u_m|$ at the active set, and in the last step we cancel $D_{mm} = D^*_{mm}$

$$\frac{u_m}{|u_m|} = \frac{x_m}{|x_m|}\ . \tag{3.47}$$

This also proves Corollary 1 as the right hand side of (3.47) has unit magnitude after cancelling D_{mm}. It readily follows that $|u_m| = \mu$. □

3.3.4. Linearity in the Regularization Parameter

Corollary 3. *The primal and the dual solutions to the complex-valued generalized LASSO problem are continuous and piecewise linear in the regularization parameter $\mu > 0$. The changes in slope occur at those values for μ where the set of active indices $\mathcal{M}$ changes.*

Proof. This corollary has been proven for various assumptions in different works, please have a look into Appendix B in [MGZ15] and the references therein. □

3.4. Systematic Error of the ℓ_1 Relaxation

The solution of the generalized LASSO is known up to the nullspace part $\boldsymbol{x}^{\text{null}}_{\ell_1}$ which does not affect the objective function's value of (P0) or the ℓ_2 part of the LASSO objective function, the nullspace component enters via the ℓ_1 penalty only. Next we assume that the problems (P0) and (P2) yield the same sparsity pattern $\mathcal{M}$ and look at the difference of the solution vectors $\boldsymbol{x}_{\ell_0}$ and $\boldsymbol{x}_{\ell_1}$.
The gradient of (P0) at the active set must be the zero vector

$$\left(\nabla \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_0}\|_2^2\right)_{\mathcal{M}} = -2\boldsymbol{A}^H_{\mathcal{M}}\left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}}\boldsymbol{x}_{\ell_0,\mathcal{M}}\right) = \boldsymbol{0}\ . \tag{3.48}$$

For (P2″) the right hand side of (3.48) cannot be zero as it is known from Theorem 1 to be the scaled dual variable

$$2\boldsymbol{A}^H_{\mathcal{M}}\left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}}\boldsymbol{x}_{\ell_1,\mathcal{M}}\right) \quad = \quad \boldsymbol{D}^H_{\mathcal{M}}\boldsymbol{u}_{\mathcal{M}}\ , \tag{3.49}$$

and Corollary 1 asserts the value of the dual variable at the active set

$$2\boldsymbol{A}^H_{\mathcal{M}}\left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}}\boldsymbol{x}_{\ell_1,\mathcal{M}}\right) \quad = \quad \boldsymbol{D}^H_{\mathcal{M}}\mu e^{j\boldsymbol{\theta}_{\mathcal{M}}}\ . \tag{3.50}$$

To express this discrepancy we set

$$\boldsymbol{x}_{\ell_0,\mathcal{M}} = \boldsymbol{x}_{\ell_1,\mathcal{M}} + \boldsymbol{\Delta}\ , \tag{3.51}$$

and use (3.48) to arrive at

$$2\boldsymbol{A}^H_{\mathcal{M}}\left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}}\left(\boldsymbol{x}_{\ell_0,\mathcal{M}} - \boldsymbol{\Delta}\right)\right) \quad = \quad \boldsymbol{D}^H_{\mathcal{M}}\mu e^{j\boldsymbol{\theta}_{\mathcal{M}}} \tag{3.52}$$

$$2\boldsymbol{A}^H_{\mathcal{M}}\boldsymbol{A}_{\mathcal{M}}\boldsymbol{\Delta} \quad = \quad \mu\boldsymbol{D}^H_{\mathcal{M}}e^{j\boldsymbol{\theta}_{\mathcal{M}}} \tag{3.53}$$

The systematic error (in the span of $\boldsymbol{A}$) of the ℓ_0 and ℓ_1 solution is thus a function of the regularization parameter μ, the dictionary $\boldsymbol{A}$ and the generalization matrix $\boldsymbol{D}$

$$\boldsymbol{\Delta} = \frac{\mu}{2} \left(\boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{A}_{\mathcal{M}} \right)^{-1} \boldsymbol{D}_{\mathcal{M}}^H e^{j\boldsymbol{\theta}_{\mathcal{M}}} \, . \tag{3.54}$$

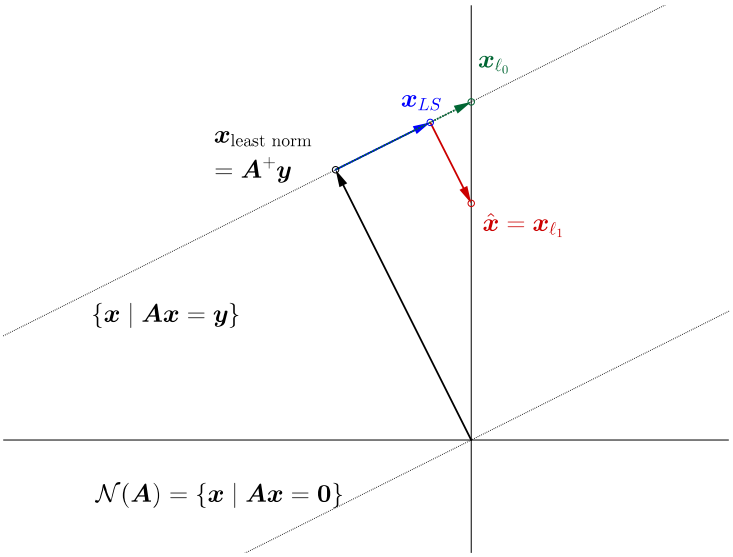


Figure 3.2.: Constructive summation of the solution parts. The least norm solution together with the nullspace part form a least squares solution which is deteriorated by the last part containing the dual vector. The last part in red explains the systematic error between (P0) and (P2″).

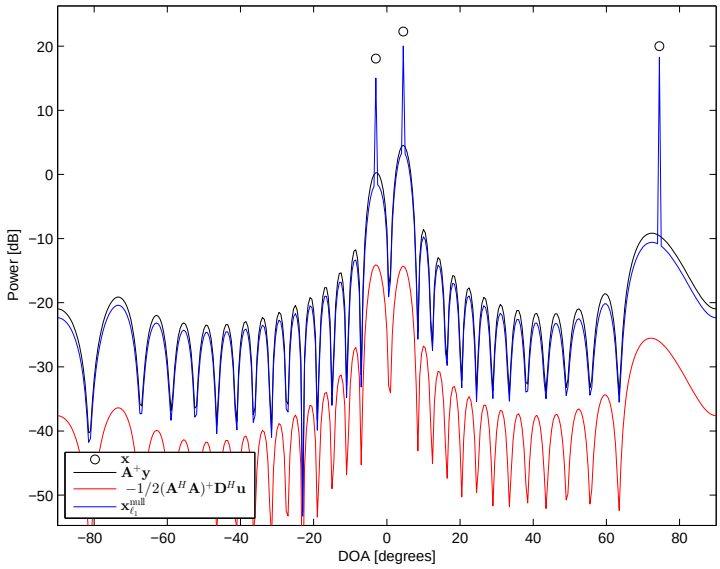


Figure 3.3.: Solution parts for a DOA problem. The minimum norm solution in black is able to resolve all sources but suffers from the high sidelobe levels, where other sources could hide. The nullspace illustrated in blue gives the minimization enough degrees of freedom to produce a sparse solution, as it sums destructively outside the active set and constructively within.

Figure 3.2 illustrates the individual parts of $\boldsymbol{x}_{\ell_1} = \boldsymbol{A}^+\boldsymbol{y} + \boldsymbol{x}_{\ell_1}^{\text{null}} - \frac{1}{2}(\boldsymbol{A}^H\boldsymbol{A})^+\boldsymbol{D}^H\boldsymbol{u}$. $\boldsymbol{A}^+\boldsymbol{y}$ is the minimum norm solution of the under-determined system of equations which is generally non-sparse. The nullspace gives the necessary degrees of freedom to generate a sparse solution, which corresponds to the intersection of the line $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ with the x or y axis. Both intersection points are valid solutions for

(P0) but for (P2″) only an intersection with the y axis is a possible solution as the ℓ_1 norm solution is at a minimum there. Due to the inherent $\boldsymbol{\Delta}$ term the amplitude of the ℓ_1 solution is a shrunk version of the true solution. Figure 3.3 shows the magnitude of the solution parts for a DOA estimation problem. The true (ℓ_0) solution is marked with circles. At the active set, the three summands add up constructively and destructively outside. The minimum norm solution is obviously non-sparse and the blue spikes of the nullspace solution part indicate already, that the sparsity originates from the nullspace.

Reduction of the Systematic Error

Equation (3.54) reveals possible countermeasures against the systematic error:

μ A lower regularization parameter leads to a lower systematic error, therefore an additional minimization over μ was included in our "workhorse" (P2).

$\boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{A}_{\mathcal{M}}$ In case we allow for dictionary learning, e.g., a sensor sub-set selection of a larger ULA, the coherence of the sources can be minimized. We do not consider this option and stay with the 30 sensor ULA.

$\boldsymbol{D}_{\mathcal{M}}$ If the generalization matrix elements D_{mm} are close to zero for $m \in \mathcal{M}$ the systematic error disappears. But the "prior" knowledge of the active set is not obtained by running the LASSO program once. One either has to iterate [CWB08] with the same measurements or one has to take observations sequentially on static sources [Mec+13; ZGM15].

Item 1 is automatically fulfilled by the construction of our problem, but item 2 and 3 are either not possible or often not considered. Another way to reduce the systematic error is to assume the active set obtained by (P2″) and (P0) is equal and use the columns associated with the active set to perform a LS regression

$$\boldsymbol{x}_{\ell_0} = \boldsymbol{A}_{\mathcal{M}}^{+} \boldsymbol{y} \,. \tag{3.55}$$

3.5. Beampattern of the Dual Vector

For the next sections, we use a dictionary with $M = 81$ hypothetical source locations $\theta_m \in [-20°, 20°]$ with $0.5°$ spacing. Theorem 1 and Corollary 1 allow now a marvellous physical interpretation of the solution to the dual problem - the dual vector. Without technical proofs we realize the improved resolution capabilities of LASSO over conventional beamforming by inspection of the dual vector's beampattern. Figure 3.4 depicts beampatterns of the dual vector, where subfigure a) starts by a large regularization parameter μ which is decreased until we find all sources as they are given in in Table 3.1 .

No.	DOA (°)	Power (lin.)
1	−6.0	4.0
2	−1.0	7.0
3	4.0	9.0
4	9.0	7.0
5	14.0	12.0
6	19.0	5.0

Table 3.1.: Source parameters for the first simulation scenario.

The different sub-figures of Figure 3.4 show:

a) $\mu > \|2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}\|_\infty$: The box constraint (3.26) is without action and the solution $\boldsymbol{u} = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$ gives the maximum dual objective value as

$$\boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2 = \boldsymbol{y}^H\boldsymbol{y} - \|\boldsymbol{A}\boldsymbol{A}^+\boldsymbol{y} - \underbrace{\frac{1}{2}(\boldsymbol{A}^+)^H\boldsymbol{D}^H 2\boldsymbol{D}^{-H}\boldsymbol{A}^H}_{(\boldsymbol{A}^+)^H\boldsymbol{A}^H=(\boldsymbol{A}\boldsymbol{A}^+)^H=\boldsymbol{A}\boldsymbol{A}^+}\boldsymbol{y}\|_2^2 = \boldsymbol{y}^H\boldsymbol{y}\,. \tag{3.56}$$

Clearly this implies that $\boldsymbol{x}_{\ell_1} \equiv \boldsymbol{0}$. The dual vector is the output of a re-weighted ($\boldsymbol{D}$) matched filter (MF). By the maxima of this re-weighted MF one could already guess the directions of 3 sources which amounts basically to the resolution capability of conventional beam forming, i.e., matched filtering.

b) By Corollary 1 we know that it is necessary that an entry of the dual vector hits the boundary (its magnitude becomes μ) to get a primal coordinate active. The primal coordinate starts to grow from 0 and if $\mu \to 0$ its magnitude is no longer a shrunk version. Its limiting value is calculated by the LS regression on the dictionary consisting of the steering vectors of the active bins $\boldsymbol{x}_{\ell_0} = \boldsymbol{A}_{\mathcal{M}}^+\boldsymbol{y}$.

c) By lowering the regularization parameter more and more, the beampattern of the re-weighted MF and the dual vector start to disagree. A strong source has not more impact on the dual vector beampattern than a weak source as the contributions are limited by μ. This explains the improved resolution.

d-e) More and more sources get active till

f) finally all six sources are detected.

The circles around the true DOA are a graphical representation of the offset regions due to coherence. Remember, anything within this region is a valid solution.

3.6. Solution Path

The foregoing discussion is now treated more formally. As already mentioned, LASSO was initially proposed as a statistical tool for variable selection. In that sense statisticians plot solution paths to depict the magnitude of all elements of the solution vector (both the primal and the dual) as a function of the regularization parameter μ. For the real valued LASSO, the piece-wise linearity as shown in Corollary 3 is used to formulate an efficient algorithm for producing the solution path [Efr+04]. For the real valued *generalized* LASSO a solution path generating algorithm was formulated in the dual domain [TT11]. For the complex valued (weighted) LASSO ($\boldsymbol{D}$ is a real, positive and diagonal matrix) a solution path algorithm in the primal domain is formulated in [PV12] and [Mec+13]. For the following figures the solution path was generated by individual LASSO runs for each μ. We are showing the magnitude of the dual and primal vector only, as Corollary 2 assures the equality of the phases anyway.

3.6.1. Over-determined Case

For the first illustration of a solution path we reduce the size of the dictionary $\boldsymbol{A}$ such that we keep only the steering vectors corresponding to sources. The dictionary is thus over-determined $\boldsymbol{A} \in \mathbb{C}^{30\times 6}$ and the solution vector $\boldsymbol{x}_{\ell_1}$ has just six entries which makes Figure 3.5 clearly laid out.
Although six sources are active all the time, depending on the chosen μ, some of the weaker sources are forced to zero, thus LASSO selects those sources of largest impact. The shrinkage as an inherent

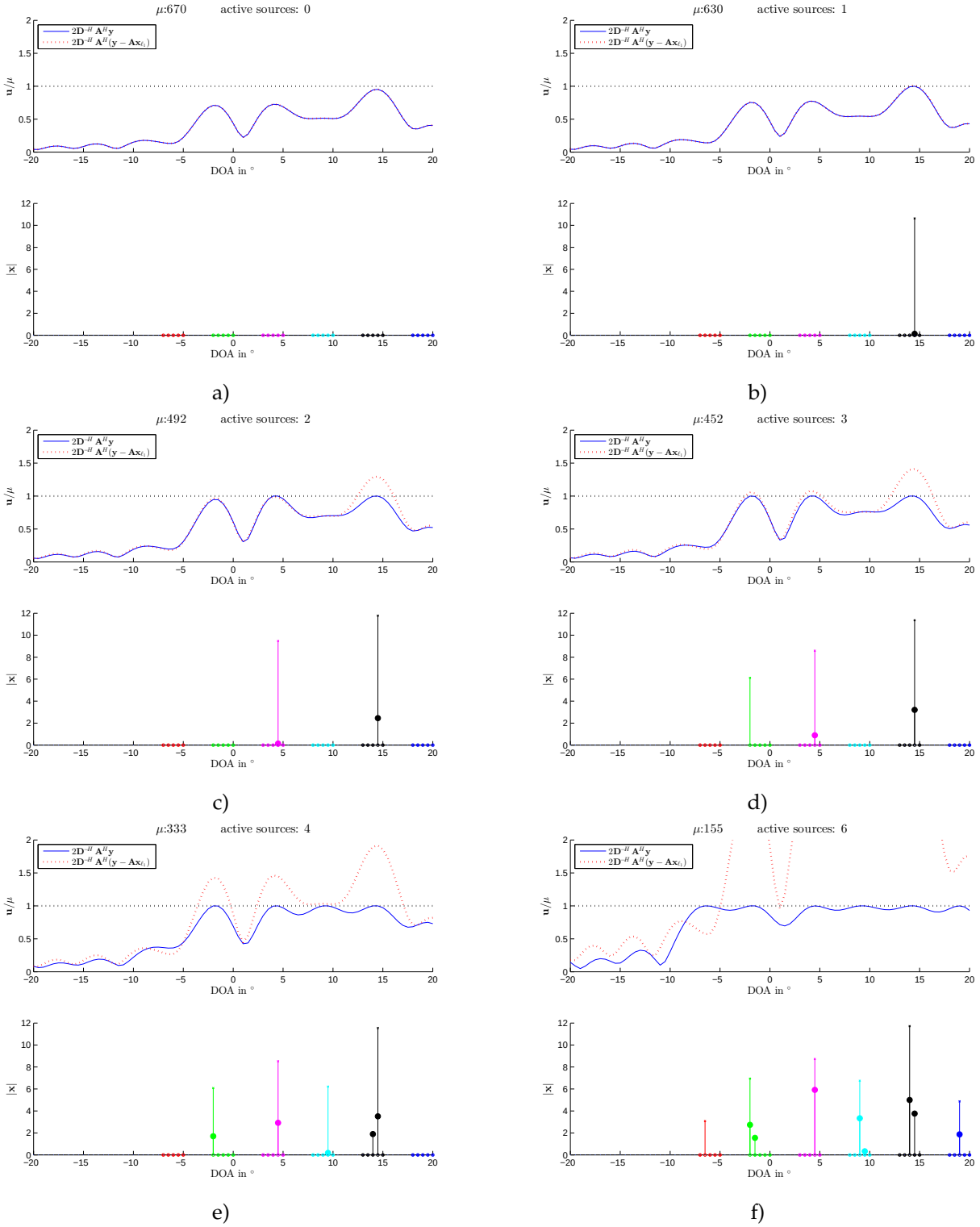


Figure 3.4.: Scaled beampattern of the conventional beamformer (spatial matched filter) $2\boldsymbol{D}^{-H}\boldsymbol{A}^{H}\boldsymbol{y}$ compared to the dual vector for different regularization parameter μ (a-f) at 30dB SNR. The dual vector starts to differ from the conventional beamformer as the first source gets active. As the impact of the stronger sources is limited weaker sources are detectable as depicted in the last sub-figure f).

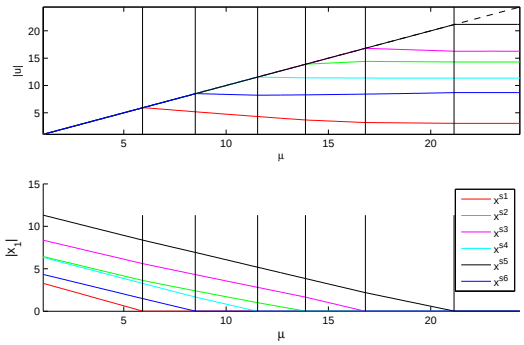
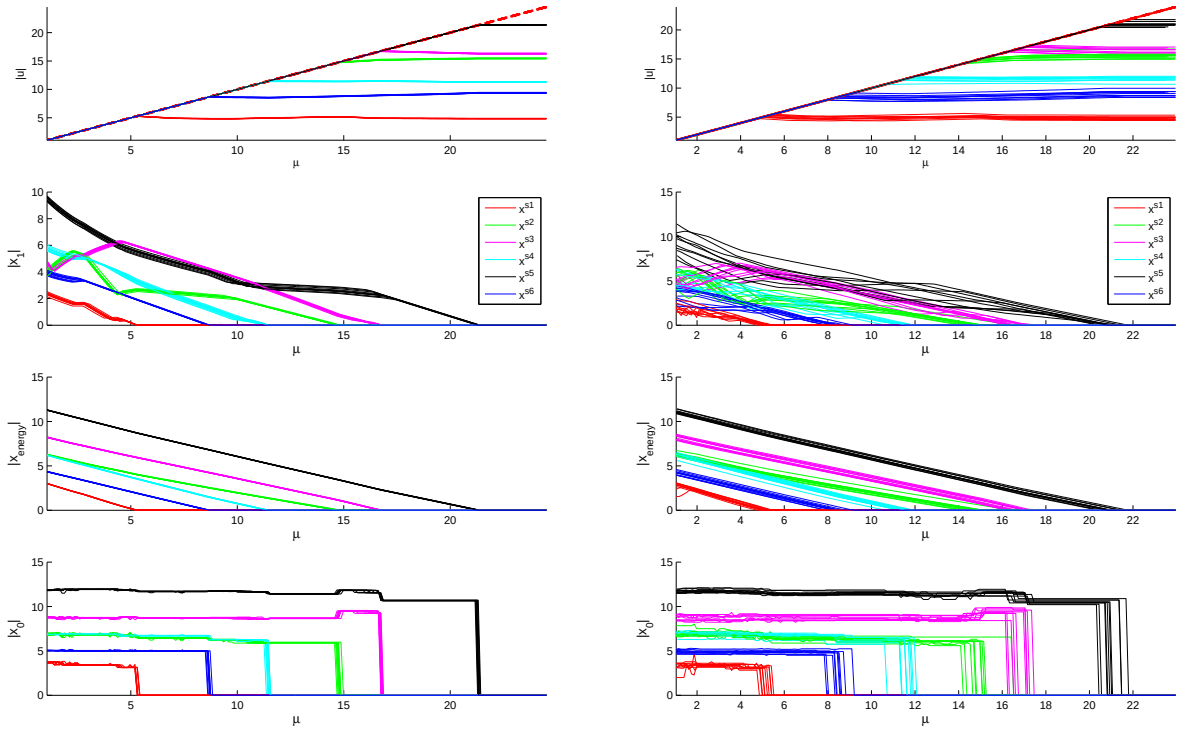


Figure 3.5.: Solution path of an over-determined system of equations at 40dB SNR. The bound $|u_m| < \mu$ is shown with the dashed black line.

problem of LASSO is also well depicted, as the source magnitude raises till its true value at $\mu \to 0$ (which amounts to plain LS). The black vertical bars illustrate Corollary 1, as for each active primal coordinate the corresponding dual coordinate is on the boundary. From Figure 3.5 we motivate the procedure for the regularization parameter selection (introduced in the next chapter). Before the first source becomes active the entries of the dual vector are constant and given by $2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$, the maximum of these entries will hit the boundary μ first and the corresponding entry of the primal vector is non-zero, i.e. active. From that μ level on, the magnitudes of $\boldsymbol{u}$ vary with μ, but one may already guess when the next dual entry hits the boundary.

3.6.2. Under-determined Case

The solution path of the over-determined system of equations was simple and intuitive. Now we are interested in under-determined systems of equations. We use the same dictionary as in the beampattern section ($\boldsymbol{A} \in \mathbb{C}^{30\times 81}$). Instead of plotting all 81 entries of the primal and dual solution vector, we restrict the solution path to the source coordinates. The two top sub-figures of Figure 3.6 are showing the same parameters as Figure 3.5. The top sub-figure showing the dual variable seems to be identical to the over-determined case and pretty robust to noise, cf. the right top sub-figure at higher noise level. The second sub-figure from the top needs a more careful explanation. At the first glance it seems that the kinks of the primal variable do not coincide with the hitting times of the dual variable, therefore violating Corollary 3, which states that the changes of the slope occur whenever the active set changes. In fact, at each kink the set of non-zero entries in $\boldsymbol{x}$ changes. As mentioned in Section 2.4.2, due to the high coherence there is an offset region around the true DOA and any outcome within this offset region is a valid solution. Figure 3.4f) shows that there could be more than one active bin within that region and that there is a trade-off between the power of the bins. If the primal entries within the offset region are summed coherently, the energy is conserved, and the equivalent magnitude $x_{\text{engergy}} \propto \|\boldsymbol{A}_{\mathcal{O}}\boldsymbol{x}_{\ell_1,\mathcal{O}}\|_2$ rises linearly as in Figure 3.5, cf. the second sub-figure from below. The bottom sub-figure shows the ℓ_0 solution - the regression with the restricted dictionary $\boldsymbol{A}_{\mathcal{M}}$ - which is much more robust in terms of coherence.



$\mathrm{SNR} = 40dB$ $\mathrm{SNR} = 20dB$

Figure 3.6.: Solution paths of 10 realizations of an under-determined system of equations at $40\mathrm{dB}$ and $20\mathrm{dB}$ SNR. The dual vector at the corresponding source positions is fairly robust to noise and coherence. Quite the contrary to the dual vector is the primal solution which is effected strongly by the offset regions (2.18). The "energy detector" is a reasonable workaround for illustration purposes.

4. Procedures

The active set of the LASSO solution is constant on an μ interval and the ℓ_0 norm changes at certain μ^{*i}, the candidate points [PV12]. The superscript i denotes the sparsity degree for $\mu^{*i} \geq \mu > \mu^{*(i+1)}$. The $i - \mu$ curve therefore looks staircase-shaped, cf. Figure 3.1. The aim of this chapter is to derive a procedure which estimates the candidate points based on the insights given by Theorem 1 and Corollary 1. Although there are (fast) approaches like [PV12; Mec+13], our approach uses an approximation to save computational complexity significantly. The coarsest approximation runs a MF once and uses that output for regularization parameter prediction/estimation.

4.1. Approximations

From Theorem 1 we know that the magnitude of the entries of the dual vector are bounded by the regularization parameter and furthermore we know that the dual vector's shape is formed essentially by the conjugate transposed dictionary $\boldsymbol{A}^H$, i.e., "beamforming on the residuals". Corollary 1 tells us that there are at least $|\mathcal{M}|$ entries in the dual vector with magnitude μ at those entries where the source magnitudes are active ($|x_m| > \delta$).
In other words, **we expect to see a peak, formed by $\boldsymbol{A}^H$, of height μ, for each active source**. We define the $\mathrm{peak}(\boldsymbol{u}, p)$ function which returns the p^{th} largest local peak in magnitude of the vector $\boldsymbol{u}$ equally to MATLAB's implementation [MAT13]

> *PKS = findpeaks(X) finds local peaks in the data vector X. A local peak is defined as a data sample which is either larger than the two neighboring samples or is equal to Inf.*

Consequently, if two sources have at least one bin between each other (otherwise the definition of a local peak is not fulfilled) the following fix point equation holds true

$$\mu^i = \mathrm{peak}\left(\left|\boldsymbol{u}(\mu^i)\right|, i\right) = \mathrm{peak}\left(\boldsymbol{u}(\mu^i), i\right) \ . \tag{4.1}$$

Equation (4.1) states that the i^{th} local peak of the dual vector, with $\mu^{*i} \geq \mu^i > \mu^{*(i+1)}$, has magnitude μ^i. This kind of fix point equations are unhandy as we need a lot of LASSO runs for convergence.

The approximations of this section are motivated by the following expansion of the dual vector for a possible new source position n

$$u_n(\mu^i) = \frac{2}{D^*_{n,n}} \boldsymbol{a}_n^H \left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1}(\mu^i)\right) = \frac{2}{D^*_{n,n}} \boldsymbol{a}_n^H \left(\boldsymbol{y} - \sum_{m \in \mathcal{M}_i} \boldsymbol{a}_m \boldsymbol{x}_{\ell_1,m}(\mu^i)\right) \tag{4.2}$$

$$\overset{|\boldsymbol{a}_n^H \boldsymbol{a}_m| \ll 1}{\approx} \frac{2}{D^*_{n,n}} \boldsymbol{a}_n^H \left(\boldsymbol{y} - \sum_{m \in \mathcal{M}_{i-1}} \boldsymbol{a}_m \boldsymbol{x}_{\ell_1,m}(\mu^{i-1})\right) \qquad \text{order-rec. procedure} \tag{4.3}$$

$$\overset{\boldsymbol{a}_n^H \boldsymbol{a}_m \approx 0}{\approx} \frac{2}{D^*_{n,n}} \boldsymbol{a}_n^H \boldsymbol{y} \qquad \text{fast procedure} \tag{4.4}$$

An example showing the coherence $\boldsymbol{A}^H\boldsymbol{A}$ was depicted in Figure 2.2. For certain DOAs the coherence $\boldsymbol{a}_n^H\boldsymbol{a}_m$ is zero or very small. This is very fortunate as $\boldsymbol{a}_m\boldsymbol{x}_{\ell_1,m}(\mu^i)$ has little impact on u_n then and the chosen regularization parameter μ has no or little influence on the magnitude of u_n; we replace $\boldsymbol{x}_{\ell_1,m}(\mu^i)$ by $\boldsymbol{x}_{\ell_1,m}(\mu^{i-1})$. From (4.3) we conclude for weakly coherent sources

$$\mu^i = \mathrm{peak}(\boldsymbol{u}(\mu^i), i) \approx \mathrm{peak}(\boldsymbol{u}(\mu^{i-1}), i)\ . \tag{4.5}$$

Through Equation (4.5) we have found a rule for predicting μ^i, the regularization parameter such that $\boldsymbol{u}$ has i peaks of magnitude μ^i. Remember, this is the necessary condition to find i active sources. In fact, the approximation in (4.3) or better in (4.4) is always perfect for the first active source as $\boldsymbol{x}_{\ell_1}\ (\mu > 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}) \equiv \boldsymbol{0}$ and the candidate point estimate for two active sources is obtained from $\mu^2 = \mathrm{peak}(\boldsymbol{u}(\mu = \|2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}\|_\infty), 2)$. By developing this idea further, we obtain the order recursive procedure, which produces all order solutions $p = |\mathcal{M}|$ up to the desired sparsity degree s. This procedure is actually "OMP like" as it adds one source per iteration, but as a key difference, the active indices may change from iteration to iteration.
What happens to u if all sources are incoherent? We could have predicted the i^{th} level of the peaks in $\boldsymbol{u}$ from the re-weighted matched filter $\mu^i \approx \mathrm{peak}(2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}, i)$ as indicated in (4.4). This amounts to a really computationally cheap matrix multiplication. Indeed it is identified as a very delicate approximation of the dual vector

$$\mu^i \approx \mathrm{peak}(\boldsymbol{u}(\mu^{i-1}), i) \approx \mathrm{peak}(\boldsymbol{u}(\mu^{i-1}), i) \approx \cdots \approx \mathrm{peak}(\boldsymbol{u}(\mu > 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}), i) = \mathrm{peak}(2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}, i)\ . \tag{4.6}$$

Maybe we do not end up with the desired sparsity level and need to iterate some steps, but overall this approach is much faster if we are not interested in all order solutions p. Anything in between, e.g. to predict based on $\boldsymbol{u}(\mu^2)$ the regularization parameter to obtain four active sources, i.e., μ^4, could also be considered. To stick to the metaphor from before, this faster approach is "CoSaMP like".

4.2. Order Recursive Procedure

As already mentioned, the order recursive procedure proposed in this section can be intuitively understood as a kind of "high coherence OMP". Instead of choosing the index with highest correlation to the dictionary, we run LASSO (at a specific μ level) to detect the active set. The main advantage is simply a possible change of the active set from order to order. A toy-example is the case of two nearby impinging sources which are seen as a strong joint source for $p = 1$ and as two separated sources for $p = 2$.
Figure 4.1 shows the signal flow chart of the algorithm described in Table 4.1. Based on the beamformer on the residuals, i.e., the dual vector, the regularization parameter is estimated according to the approximation (4.3) and passed to the generalized LASSO.
If the generalized LASSO is solved via primal-dual interior point methods [Wri97], the dual variable comes at no computationally cost directly from the solver and is used for the estimation of the next order regularization parameter. This feedback path is represented by a dotted line. In case the generalized LASSO solver does not provide the dual vector, we need to evoke Theorem 1 and use the primal vector to calculate the vector of residuals $\boldsymbol{r}$.
From the LASSO's solution the active set is determined such that small spurious peaks are ignored.
Depending on the size of the active set

- the peaksearcher must adopt μ to get another peak in $\boldsymbol{u}$ as the active set is still too low - a peak of the dual vector does not belong to an active source.

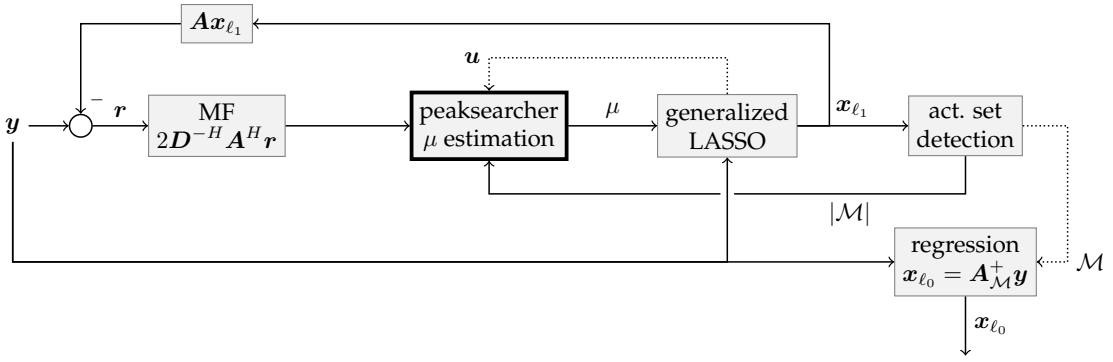


Figure 4.1.: Signal flow diagram for the primal based procedures. The main difference between the order recursive and the fast procedure is hidden inside the bold framed peaksearcher.

- the regularization is bisected as there are too many sources (this case is not shown in Table 4.1).
- the $\boldsymbol{x}_{\ell_0}$ solution is calculated by a regression on the restricted dictionary $\boldsymbol{A}_{\mathcal{M}}$.

The informal description of the order recursive procedure is now formalized in Table 4.1.

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$.
1:	Initialize $i = 0$, $p = 1$, $\boldsymbol{x}^0_{\ell_1} = \boldsymbol{0}$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $p < s$
	$i = i + 1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, i\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, i+1\big)$
4a:	$\boldsymbol{x}^i_{\ell_1} =$ solution to LASSO problem for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
4b	$\boldsymbol{u}^i = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^i_{\ell_1}\right)$
5:	$\mathcal{M}_i = \{m \Big\vert\, \lvert x^i_{\ell_1,m}\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}^i_{\ell_1}\rVert_\infty$
6:	if $\lvert\mathcal{M}_i\rvert < p$
7:	$i = i + 1$
8:	else
9:	$\boldsymbol{x}^p_{\ell_0} = \boldsymbol{A}^+_{\mathcal{M}_i}\boldsymbol{y}$
10:	$\mathcal{M}_p = \mathcal{M}_i$
11:	$p = p + 1$
12:	end
13:	end
	Output: $\boldsymbol{x}^p_{\ell_0}, \mathcal{M}_p \quad \forall p = 1 \ldots s$

Table 4.1.: Order recursive procedure to approximate (P2″).

line 1 The procedure starts with the initialization of the loop-counter i, the peak-counter p and the residual vector $\boldsymbol{r} = \boldsymbol{y}$.

line 2 As long as the desired sparsity degree is not reached we keep iterating and count up the loop-counter i.

line 3 The approximation of Section 4.1 is now used to choose the i^{th} regularization parameter as convex combination of the estimated candidate points of i and $i + 1$ sources active and is traded

off by the parameter $0 < F < 1$. $F \ll 1$ means that the μ is chosen higher as needed and the systematic error/shrinkage is bigger than necessary. $F \gg 0$ means we may risk to lower μ too fast and get two additional sources active. For the simulations we have chosen $F = 0.9$.

line 4 The solution of the generalized LASSO is now used to calculate the dual vector for the next iteration (as already mentioned, this is omitted for primal-dual interior point methods).

line 5 The active set is determined and the threshold δ is chosen such that numerical artefacts (small spurious peaks) are not considered as an active source.

line 6 Based on the size of the active set, the procedure branches off.

line 7 If the cardinality of the active set is less than the current output order p, the regularization parameter estimator searches for the next peak in $\boldsymbol{u}$ (μ is decreased).

line 8 If the cardinality equals the intended order the output variables (***line 9***, ***line 10***) are generated.

line 11 Finally, the procedure aims for the next order solution and p is counted up.

The effectiveness of the proposed procedure is demonstrated on a synthetic example with eight sources at direction and powers given in Table 4.2. The noise is zero-mean complex-valued circularly symmetrically normally distributed, i.e., $\boldsymbol{n} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I})$.

No.	DOA+90 (°)	Power (dB)
1	45	-5
2	60	10
3	76	5
4	99	0
5	107	11
6	120	12
7	134	9
8	162	25

Table 4.2.: Source parameters for the second simulation scenario.

4.3. Fast Procedure

The main drawback of the order recursive method is its run time. It needs at least s LASSO runs to converge. We use the same idea as in CoSaMP [NT09], to add more active indices at a time and risk an imprecise regularization parameter estimation. The approximation (4.4) allows us to predict to regularization parameter for the s^{th} order already without any LASSO run, simply by looking at the peak levels of the re-weighted matched filter. As mentioned above, if the sources are coherent the first estimated μ may produce the wrong sparsity order. If the order is too low, the dual vector is used for the next improved estimation. If the order is too high, bisection is used until the desired sparsity order is met.
The signal flow diagram is still equal to Figure 4.1 since entire knowledge is hidden in the μ - estimator / peaksearcher. The runtime of the algorithm is hard to forecast well, but simulations have shown a speed-up factor of approximately three. The new procedure is formulated in Table 4.3.
The main differences is shown in the following line:

line 3 We try to find the regularization parameter μ such that we obtain the desired sparsity level s in one shot. The initial $\boldsymbol{u} = 2\boldsymbol{D}^{-H}\boldsymbol{A}^{H}\boldsymbol{y}$ is very likely produce a wrong sparsity order. On that account we make the next, more precise, μ estimate based on the dual vector $\boldsymbol{u}$.

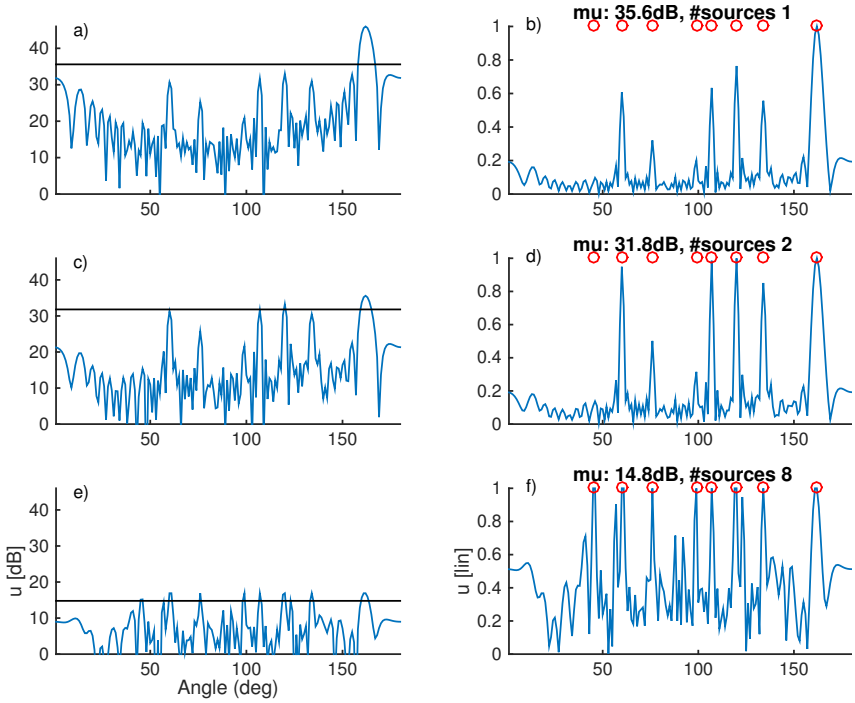


Figure 4.2.: Dual vector for order-recursive approach corresponding to step $p = 1$ (a and b), $p = 2$ (c and d), and $p = 8$ (e and f). Left column: Dual (dB) for the previous step which is used for predicting μ (horizontal line). Right column: Dual (lin) normalized with μ (maximum is 1), the true source locations are marked with $\circ$, and the actual value of μ and number of sources found is also indicated.

4.4. Fully Dual Approach

Equation (3.28) of Corollary 1 states that searching for peaks in the dual vector amounts to a relaxation of the active set ($\mathcal{M} \subseteq \mathcal{U}$). Primal active entries enforce the bound on the corresponding dual entry to be tight, but the opposite must not hold true. The equality $|u_m| = \mu$ is only a necessary condition for x_m to be in the active set. Despite this fact, we chose μ such that there is a new peak in $\boldsymbol{u}$ in each iteration (as before), but check with regression on the relaxed set $\mathcal{U}$ whether the peak belongs to a true active source or not.

At this point it may appear ill-advised to solve the relaxed problem as no gains are visible. The fully dual formulation for the LASSO is of academic interest only, but for grid-free DOA estimation, e.g. [Tan+13], where no discritization of the angular space takes place, the dimensions of the primal problem are infinite while the dimension of the dual problem are still bounded. It is essential to formulate a similar procedure then.

Both primal-based procedures are re-formulable to work fully in the dual domain. In this thesis only the fast procedure is repeated and the key steps of the procedure given in Table 4.4 are:

line 3 Find the regularization parameter exactly as in the primal procedure.

line 4 Solve the dual problem to the generalized LASSO.

line 5 Find the peaks of magnitude μ within a reasonable numerical precision.

line 6 Perform the LS regression with the columns indexed by $\mathcal{U}$ and check if all indices after the regression are really active.

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$
1:	Initialize $i = 0$, $\boldsymbol{x}_{\ell_1}^0 = \boldsymbol{0}$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $\lvert\mathcal{M}_i\rvert < s$
	$i = i + 1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s+1\big)$
4a:	$\boldsymbol{x}_{\ell_1}^i =$ solution to LASSO problem for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
4b	$\boldsymbol{u}^i = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1}^i\right)$
5:	$\mathcal{M}_i = \{m \,\big\vert\, \lvert x_{\ell_1,m}^i\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}_{\ell_1}^i\rVert_\infty$
6:	end
7:	$\boldsymbol{x}_{\ell_0}^s = \boldsymbol{A}_{\mathcal{M}_i}^{+}\boldsymbol{y}$
8:	$\mathcal{M}_s = \mathcal{M}_i$
	Output: $\boldsymbol{x}_{\ell_0}^s, \mathcal{M}_s$

Table 4.3.: Fast iterative primal-based procedure to approximate (P2″).

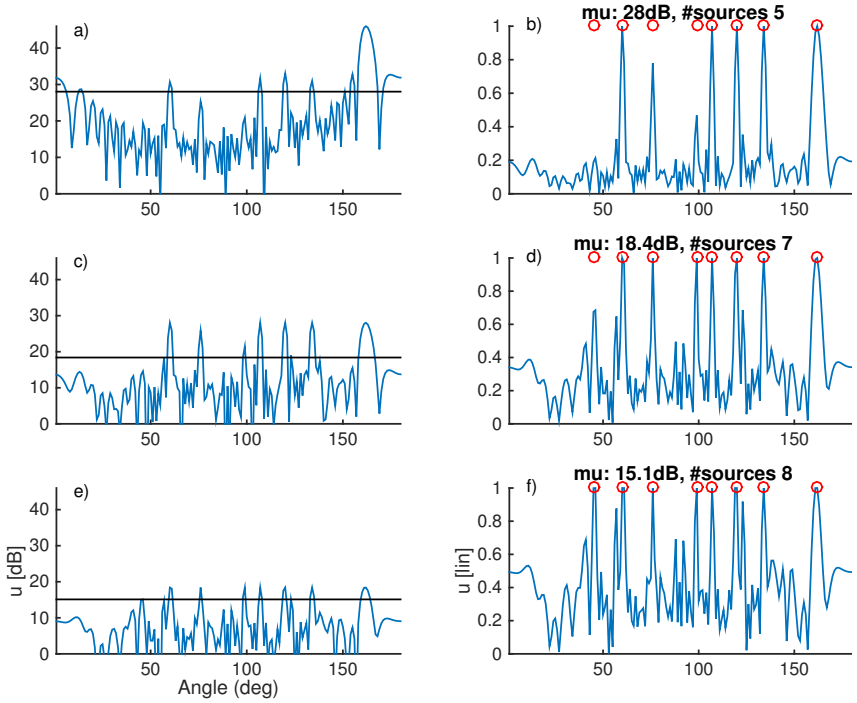


Figure 4.3.: Dual vector for the fast iterative approach for localizing $s = 8$ sources for step $i = 1$ (a and b), $i = 2$ (c and d), and $i = 3$ (e and f). Left column: Dual (dB) for the previous step which is used for selecting μ (horizontal line). Right column: Dual (lin) normalized with μ (maximum is 1), the true source locations are marked with $\circ$, and the actual value of μ and number of sources found is also indicated.

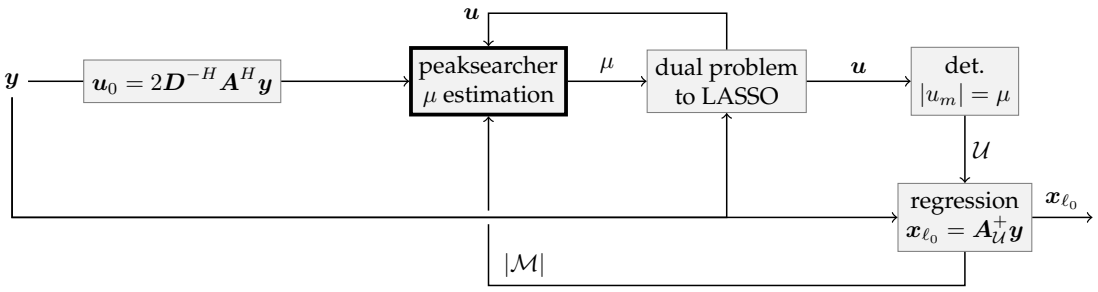


Figure 4.4.: Signal flow diagram for the fully dual procedure. The size of the active set is determined from the regression on the relaxed active set $\mathcal{U}$, so that a pseudo-inverse must be calculated each iteration.

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$
1:	Initialize $i = 0$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $\lvert\mathcal{M}_i\rvert < s$
	$i = i + 1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s+1\big)$
4:	$\boldsymbol{u}^i =$ solution to dual problem for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
5:	$\mathcal{U}_i = \{m \Big\vert 1 - \frac{\lvert u_m^i\rvert}{\mu} < \epsilon_\mu\}$
6a:	$\boldsymbol{x}_{\ell_0}^i = \boldsymbol{A}_{\mathcal{U}_i}^{+}\boldsymbol{y}$
6b:	$\mathcal{M}_i = \{m \Big\vert \lvert x_{\ell_0,m}^i\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}_{\ell_0}^i\rVert_\infty$
7:	end
8:	$\boldsymbol{x}_{\ell_0}^s = \boldsymbol{A}_{\mathcal{M}_i}^{+}\boldsymbol{y}$
9:	$\mathcal{M}_s = \mathcal{M}_i$
	Output: $\boldsymbol{x}_{\ell_0}^s, \mathcal{M}_s$

Table 4.4.: Iterative dual-based procedure to approximate (P2″).

5. Sequential Bayesian Estimation

This chapter will be published at the 3rd *Workshop on Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing* [ZGM15] and is presented in a more or less unchanged fashion. At some points more detailed explanations and derivations are added.
The sparse Kalman filters of Namrata Vaswani [Vas09; Vas10] are well performing algorithms which were not derived by optimality conditions and run CS algorithms on the LS or Kalman filtered residuals, cf. Section A.3. Our approach is likewise not deduced by optimality but it was titled density evolution to emphasise the physical motivation behind.

5.1. Bayesian Inference

For the observation $\boldsymbol{y}$ according to the linear model (2.4), the conditional probability density given the source vector $\boldsymbol{x}$ is

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2\right)}{(\pi\sigma^2)^N} \,. \tag{5.1}$$

For the source vector $\boldsymbol{x}$, a prior probability density is assumed in form of a multivariate complex Laplace-like density [He+07],

$$p(\boldsymbol{x}) = \prod_{m=1}^{M} p_m(x_m), \text{ with } p_m(x) = \frac{(\lambda_m)^2}{2\pi}\mathrm{e}^{-\lambda_m|x_m|} \,, \tag{5.2}$$

with associated hyperparameters $\lambda_m > 0$ modelling the source signal strength at location θ_m. $x_m = |x_m|\mathrm{e}^{j\phi_m}$ is the complex source signal at hypothetical source location θ_m. Note that (5.2) defines the joint distribution for $|x_m|$ and ϕ_m with the phases uniformly distributed on $[0, 2\pi)$, for $m = 1, \ldots, M$. The prior mean and variances are

$$\mathbb{E}\{\boldsymbol{x}\} = \boldsymbol{0}, \qquad \mathbb{E}\{\boldsymbol{x}\boldsymbol{x}^H\} = 6\,\mathrm{diag}\left(\lambda_1^{-2}, \ldots, \lambda_M^{-2}\right) \,. \tag{5.3}$$

The covariance factor seems unfamiliar but follows directly from a polar coordinate transformation $x_m = re^{i\varphi}$ and integration over r from 0 to ∞ and φ from $-\pi$ to π

$$\mathbb{E}\{|x_m|^2\} = \underbrace{\int_{-\pi}^{\pi} \frac{1}{2\pi}\mathrm{d}\varphi}_{1} \int_0^{\infty} r^2\, \lambda_m^2\, e^{-\lambda_m r}\, r\,\mathrm{d}r = \frac{6}{\lambda_m^2} \,. \tag{5.4}$$

Taking the logarithm of (5.2) gives

$$-\ln p(\boldsymbol{x}) = \sum_{m=1}^{M} \lambda_m|x_m| - 2\sum_{m=1}^{M} \ln\lambda_m + M\ln 2\pi \,. \tag{5.5}$$

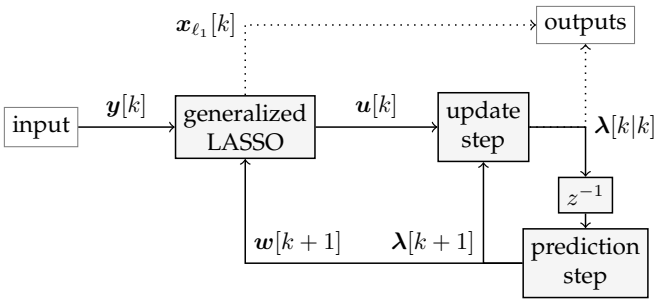


Figure 5.1.: Signal flow diagram for sequential Bayesian estimation at step k. The generalized LASSO is implemented according (4.3). The update step is based on a mean fit of the posterior distribution and the prediction step utilizes a diffusion-like motion model.

For the posterior probability density function (pdf) $p(\boldsymbol{x}|\boldsymbol{y})$, Bayes' rule is used for obtaining the generalized LASSO Lagrangian [TT11; Mec+13]

$$\frac{1}{\sigma^2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2+\mu\,\|\boldsymbol{W}\boldsymbol{x}\|_1\ , \tag{5.6}$$

with the weights $0<w_m\leq 1$

$$\boldsymbol{W}=\frac{1}{\mu}\,\mathrm{diag}(\boldsymbol{\lambda})=\mathrm{diag}(\boldsymbol{w})\ . \tag{5.7}$$

Equivalently to (5.6), this is reformulated as

$$\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2+\mu\,\|\boldsymbol{D}\boldsymbol{x}\|_1\ , \tag{5.8}$$

with

$$\boldsymbol{D}=\sigma^2\boldsymbol{W}\ . \tag{5.9}$$

The minimization of (5.8) constitutes a strictly convex optimization problem. Minimizing the generalized LASSO Lagrangian (5.8) with respect to $\boldsymbol{x}$ for given μ, and $\boldsymbol{w}=(w_1,\ldots,w_M)^T$, $\boldsymbol{\lambda}=\mu\boldsymbol{w}$, gives a sparse MAP source estimate $\boldsymbol{x}_{\ell_1}$. This minimization problem promotes sparse solutions in which the ℓ_1 constraint is weighted by giving every source amplitude its own hyperparameter w_m.

5.2. Sequential Estimation

[Mec+13] proposed and analysed a sequential Bayesian sparse source reconstruction which is now interpretable as solving both the generalized LASSO problem (3.4) and its dual (3.10)–(3.12) at step k. Remember, the dependency of time is denoted explicitly in all relevant variables, e.g. $\boldsymbol{y}[k]$ to denote the data at step k.
First, the history of all previous array observations is summarized in $\boldsymbol{Y}[k-1]=(\boldsymbol{y}[1],\ldots,\boldsymbol{y}[k-1])$. Given the history $\boldsymbol{Y}[k-1]$ and the new data $\boldsymbol{y}[k]$, we seek the MAP source estimate $\boldsymbol{x}_{\ell_1}[k]$ for the linear model (2.4) at step k under the ℓ_1–constraint.

5.2.1. Update Step

In [Mec+13] two approximations were introduced in order to relate the (non-Laplace-like) posterior weight vector $\boldsymbol{\lambda}[k|k]$ to the prior weight vector $\boldsymbol{\lambda}[k]$ in (5.2). By means of Theorem 1 and Corollary 1, both approximations for the posterior weight vector are expressible by the dual solution.
The preferable approximation, based on a mean fit, leads to following relation between posterior and

prior weight vector in the complement of the active set

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{\left|\boldsymbol{a}_m^H \frac{2}{\sigma^2}(\boldsymbol{y}[k] - \boldsymbol{A}\boldsymbol{x}_{\ell_1})\right|^2}{\lambda_m^2[k]}\right) \qquad \forall m \notin \mathcal{M}[k], \tag{5.10}$$

and in the active set the posterior weight vector must be zero

$$\lambda_m[k|k] = 0, \quad \forall m \in \mathcal{M}[k]\,. \tag{5.11}$$

By Theorem 1 we express the numerator of (5.10) by the dual vector $\boldsymbol{u}$ and the weights $\boldsymbol{w}$. Corollary 1 links Equations (5.10) and (5.11), as (5.10) is zero for $|u_m| = \mu$.

Theorem 2. *With the mean fit approximation, the posterior weight vector $\boldsymbol{\lambda}[k|k]$ is related to the prior weight vector $\boldsymbol{\lambda}[k]$ by the dual solution $\boldsymbol{u}[k]$ at step k,*

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{|u_m[k]|^2}{\mu^2[k]}\right)\,. \tag{5.12}$$

Due to Theorem 1 and Corollary 1, μ is equal to the max-norm of $\boldsymbol{u}$ and Theorem 2 is expressible solely by the dual vector $\boldsymbol{u}$

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{|u_m[k]|^2}{\|\boldsymbol{u}[k]\|_\infty^2}\right)\,. \tag{5.13}$$

Equation (5.13) shows that the dual coordinate equals μ and the posterior weights become zero at source positions $m \in \mathcal{M}$. Outside the active set, the probability of finding a source depends on the relative sidelobe power level of the beamformer of the LASSO residuals, cf. Theorem 1.

5.2.2. Prediction Step

In sequential estimation, typically the prior for the upcoming step $k+1$ is calculated from the current posterior and a state-transition probability density function ("motion model"). In a Markovian stochastic framework this is based on the Chapman-Kolmogorov equation [RAG04]. For Brownian motion the state-transition probability density satisfies the diffusion equation. Our prediction step is therefore based on a diffusion model. Where diffusion occurs just in the neighbourhood of active sources.

Neighborhood of an active source

The index neighborhood of m is denoted as $\mathcal{N}_m = \{j \mid m-l, \ldots, m+l\}$. If any $\lambda_j[k|k] \in \mathcal{N}_m$ is less than the threshold λ_0 then a source is active in the neighborhood of m with high probability.
The motion model is defined via complementary cumulative distribution functions (ccdf) of the neighborhood magnitudes,

$$\mathrm{P}\left\{|x_m[k+1]| > \delta \middle| \boldsymbol{x}[k]\right\} = \sum_{j=-l}^{l} \alpha_j \,\mathrm{P}\left\{|x_{m+j}[k]| > \delta\right\}\,, \tag{5.14}$$

with non-negative coefficients α_j and $\sum_j \alpha_j = 1$. The ccdf, after a polar coordinate transformation $x_m = r_m e^{i\varphi_m}$, evaluates to

$$\begin{aligned} \mathrm{P}\left\{ |x_m[k+1]| > \delta \middle| \boldsymbol{x}[k] \right\} &= \sum_{j=-l}^{l} \alpha_j \int_{-\pi}^{\pi} \frac{\mathrm{d}\varphi_{m+j}}{2\pi} \int_{\delta}^{\infty} \lambda_{m+j}^2[k|k] e^{-\lambda_{m+j}[k|k] r_{m+j}} \; r_{m+j} \; \mathrm{d}r_{m+j} \\ &= \sum_{j=-l}^{l} \alpha_j \left(\lambda_{m+j}[k|k] \; \delta + 1\right) e^{-\lambda_{m+j}[k|k] \; \delta} \; , \end{aligned}$$

and by taking the negative derivative w.r.t. δ, we obtain the magnitude's probability density

$$-\frac{\partial}{\partial \delta} \mathrm{P}\left\{ |x_m[k+1]| > \delta \middle| \boldsymbol{x}[k] \right\} = 2\pi\delta \sum_{j=-l}^{l} \alpha_j \frac{(\lambda_{m+j}[k|k])^2}{2\pi} \; e^{-\lambda_{m+j}[k|k] \; \delta} \; , \tag{5.15}$$

which is a mixture of Erlang-2 distributions with variances $2(\lambda_{m+j}[k|k])^{-2}$, cf. Eq. (5.2).
We approximate the mixture by a Laplace-like density of the form (5.2). We choose to fit the variance of the Laplace-like density such that

$$\frac{1}{(\lambda_m[k+1])^2} = \sum_{j=-l}^{l} \alpha_j \frac{1}{(\lambda_{m+j}[k|k])^2} \; . \tag{5.16}$$

We note that (5.16) is ill-behaved whenever a posterior weight $\lambda_{m+j}[k|k] = 0$. In this case, a small offset $\varepsilon > 0$ is added to stabilize (5.16) numerically. The predicted $\boldsymbol{\lambda}[k+1]$ is the product of the regularization parameter $\mu[k+1]$ and the weights $\boldsymbol{w}[k+1]$. As $\mu[k+1]$ is not yet known at step k, we need to assume that the regularization parameter remains constant between k and $k+1$, i.e.,

$$\frac{1}{(\lambda_m[k+1])^2} = \frac{1}{(\mu[k+1] w_m[k+1])^2} \approx \frac{1}{(\mu[k] w_m[k+1])^2} \; . \tag{5.17}$$

The predicted weights $w_m[k+1]$ are then calculated from the weighted harmonic mean, i.e.,

$$(w_m[k+1])^2 = \left(\sum_{j=-l}^{l} \frac{\alpha_j}{(w_{m+j}[k])^2} \right)^{-1} \; . \tag{5.18}$$

The weights are bounded, $0 < w_m[k+1] \leq 1$. The weighted harmonic mean is a special instance of the weighted Hölder mean [Bul03], see Section 5.2.3. To express the uncertainty of the prediction, the weights are increased by an offset $w_0 > 0$, similar to process noise in Kalman filtering.

Not in the neighborhood of an active source

The posterior $\lambda_j[k|k]$ exceeds the threshold λ_0 which indicates that it is improbable for a source to be near DOA θ_j. At step $k+1$, we penalize the DOA θ_j by adding a multiple of weight uncertainty w_0, i.e., $w_m[k+1] = w_m[k] + c\, w_0$ with $c > 1$. In the simulations, $w_0 = 0.01$ and $c = 10$. To guarantee that the weights remain upper bounded by one, the weighting vector is normalized to $\|\boldsymbol{w}\|_\infty = 1$.

5.2.3. Conservative Prediction

The weighted harmonic mean (5.18) is a pessimistic mean as low values have stronger impact on the mean. Generally, it tends to broaden the low weight region. To mitigate this undesirable effect, we investigate alternative rules for the predicted weights.

A weighted Hölder mean is defined as [Bul03]

$$M_p(w_1^2, \ldots, w_n^2) = \left(\sum_{j=l}^{l} \alpha_j \left(w_j^2 \right)^p \right)^{\frac{1}{p}}, \qquad \sum_{j=l}^{l} \alpha_j = 1. \tag{5.19}$$

For the choice of power $p = -1$, the weighted Hölder mean coincides with the weighted harmonic mean (5.18). The following inequality holds for weighted Hölder means,

$$M_p < M_q, \quad \text{for} \quad p < q \, . \tag{5.20}$$

Any Hölder mean with $p > 0$ is not dominated by lower weights and the arithmetic mean ($p = 1$) is the tightest conservative choice of weighting coefficients for Laplace-like prior. Notice [PV11] has used a max-log approximation instead of (5.16) which amounts to picking $M_{+\infty}$, the least tight bound.

5.3. Bayesian Procedure

The Bayesian procedure is formalized in Table 5.1 as a loop over time step k which processes the single snapshot array observation $\boldsymbol{y}[k]$ when it becomes available.

line 3 The weighting coefficients for the generalized LASSO problem (3.4) are defined for the current step k.

line 4 The s-sparse solution is implemented via the procedure given in Table 4.3 and the dual solution is evaluated by weighted beamforming of the residuals.

line 6 The posterior weighting coefficients are evaluated in line 6 ,

line 7 which are needed for the prediction step.

	Implementation of density evolution procedure: Given constants: $\boldsymbol{A} \in \mathbb{C}^{N \times M}$, $\boldsymbol{w}[1] \in [0,1]^M$, $s \in \mathbb{N}$
1:	for $k = 1, 2, 3, \ldots$
2:	Input: $\boldsymbol{y}[k] \in \mathbb{C}^N$ $\boldsymbol{w}[k] = \boldsymbol{w}[k] / \|\boldsymbol{w}[k]\|_\infty$
3:	$\boldsymbol{D}[k] = \sigma^2 \operatorname{diag}(\boldsymbol{w}[k])$
4a:	$\boldsymbol{x}_{\ell_1}[k] = s$-sparse solution to generalized LASSO (3.4) at k
4b:	$\boldsymbol{u}[k] =$ corresponding dual solution via Theorem 1
5:	Update $\boldsymbol{\lambda}[k\|k]$ via Theorem 2
6:	$\boldsymbol{w}[k+1] = \text{motion model prediction}(\boldsymbol{\lambda}[k\|k])$
7:	Output: $\boldsymbol{x}_{\ell_1}[k] \in \mathbb{C}^M$, $\boldsymbol{\lambda}[k\|k] \in \mathbb{C}^M$
8:	end

Table 5.1.: Primal/Dual formulation of sequential Bayesian sparse signal reconstruction.

We investigate the weight evolution, for 361 DOAs which are half-degree spaced, from step $k = 1$ to $k = 100$, where the generalized LASSO of Table 5.1 is solved by CVX [GBY08] at each step. In Figure 5.2 the weight evolution of sources with trivial motion model, $l = 0$ and $\alpha_0 = 1$ is shown. In Figure 5.3 movement is modelled with a uniform motion model ($l = 2,\ \alpha_j = \frac{1}{2l+1} = 0.2$). We can observe the trade off between having precise estimates for the static sources and a good quality estimate of the moving source.

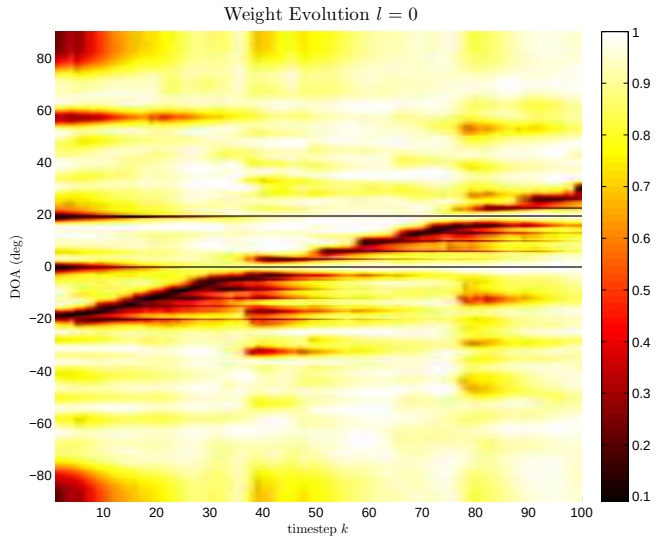


Figure 5.2.: Weight evolution for three sources at DOA $20°, 0°, -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 1$, SNR $= 20$dB .

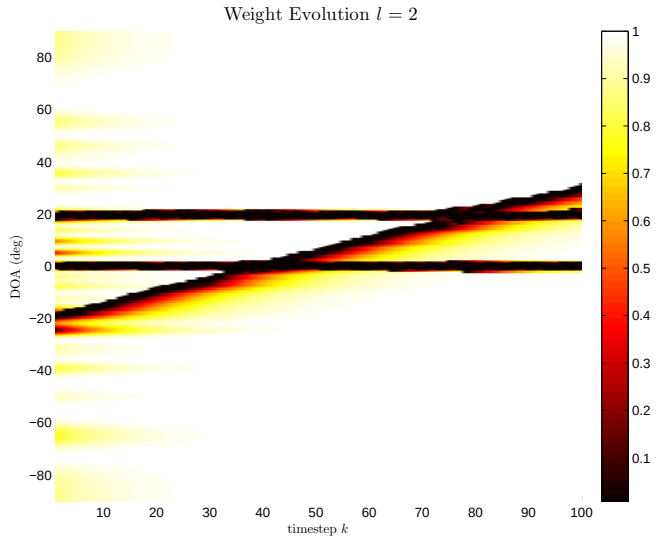


Figure 5.3.: Weight evolution for three sources at DOA $20°, 0°, -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 10$, SNR $= 20$dB .

A reasonable compromise of capturing the motion of a source while still improving the estimate of the static sources is to use an $l > 0$ and a conservative choice of the weights. Figure 5.4 uses the same motion model as in Figure 5.3, but the weighted arithmetic mean is used, i.e. Equation (5.19) for power parameter $p = 1$. For the arithmetic mean, the low weight region of the static sources is narrower than for the harmonic mean. This comes at the expense of the traceability of the moving source.

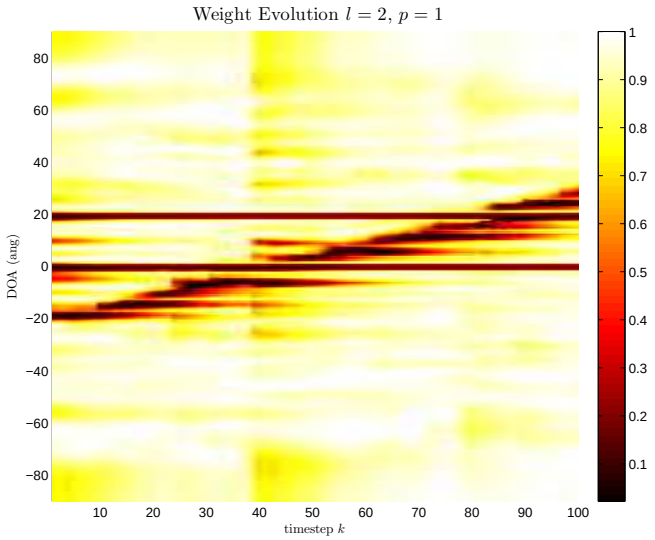


Figure 5.4.: Conservative weight evolution for three sources at DOA $20°; 0°; -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 10$, SNR $= 20$dB .

The proposed DOA tracking procedure from Table 5.1 is compared to "Compressive Sensing on Kalman filtered residuals (KF-CS)" [Vas09] in Figure 5.5. For KF-CS μ is chosen non-adaptively analogous to the value given in [QLV09]:Algorithm 1. The density evolution approach with $p = -1$ mean recovers the static sources worse than the Kalman filter and the conservative ($p = 1$) approach, but in return the moving source is traced well.

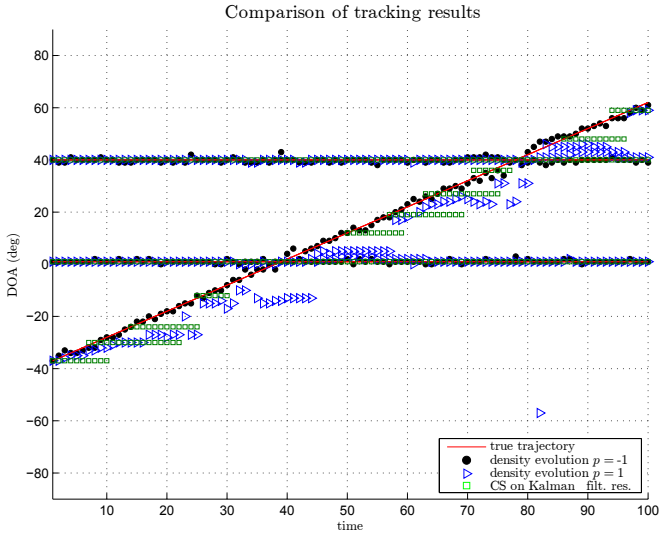


Figure 5.5.: Active set $\mathcal{M}$ of different tracking algorithms.

6. Conclusions

6.1. Results

The dual problem to the generalized LASSO is interpretable as a weighted conventional beamformer acting on the LASSO residuals. Due to an affine one-to-one relation between the dual and primal vectors there is no need to solve the dual problem and any results formulated for the primal problem are readily extendible to the dual problem.

The s-sparse reconstruction solution $\boldsymbol{x}_{\ell_0}$ is generally different from the LASSO solution $\boldsymbol{x}_{\ell_1}$. Assuming the active set is equal for (P0′) and (P2″), the difference between $\boldsymbol{x}_{\ell_0}$ and $\boldsymbol{x}_{\ell_1}$ is characterized via the dual vector and shows its strong linear dependence on the regularization parameter μ and the basis coherence of the active sources, cf. Equation (3.54).

Based on mathematical and physical insights, an order-recursive and a faster (iterative) LASSO-based procedure are proposed and evaluated. These procedures use the dual variable of the generalized LASSO for regularization parameter estimation. We predict the candidate points where the active set changes. Further, a fully dual-based procedure is formulated which solves only the dual problem.

At last, a sequential reconstruction procedure was proposed. The dual variable is propagated to the update step, which approximates the posterior distribution with a Laplace-like distribution (see Figure 5.1). From the approximated posterior together with a motion model, the prior for the next step is derived and the procedure is ready for the next step. Without the prediction step using a trivial motion model, the proposed procedure is fully equivalent to the procedure in [Mec+13]. By including a non-trivial motion model we have shown superior performance by means of a synthetic example.

6.2. Discussion and Outlook

The generalized LASSO framework seems very promising because efficient dedicated solvers, e.g., alternating direction method of multipliers [Boy+12], are available. The main difficulty is the limited angular sampling. The finer the angular sampling the higher the coherence will be. On the other hand if the sampling is too coarse many sources will not be localized on (or at least close to) a discretization bin to which we refer as basis mismatch [Chi+11].

One way to circumvent this problem is adaptive grid refinement [TGA14; MÇW05] in which the grid is refined at regions where sources were found. Obviously this leads to many reconstruction runs as the problem is just postponed.

A more promising way to eliminate all basis mismatch problems is to use a continuous, primal parameter space. This is a huge step, like from signal detection (hypothesis testing) to parameter estimation and with infinite dimension / continuous space new problems arise. The continuous

problems are formulated with atomic norms [BTR13] and these are solvable in the dual domain which has finite dimensionality [Tan+13]. In this sense, our approach in Table 4.4 is a discrete approach for the atomic norm problem.

At an early stage, in Equation (3.2), we restricted the generalization matrix $\boldsymbol{D}$ to be real, positive and diagonal shaped. This is a severe limitation as the generalized LASSO can be transformed into a plain LASSO problem if $\boldsymbol{D}$ is regular. [TT11] has indicated that the use of $\boldsymbol{D}_{\text{fused}}$ leads to the fused LASSO problem [Tib+05].

$$\boldsymbol{D}_{\text{fused}} = \begin{pmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ & & & \dots & & \\ 0 & 0 & 0 & \dots & -1 & 1 \end{pmatrix} \tag{6.1}$$

Through the smoothness introduced with $\boldsymbol{D}_{\text{fused}}$ it might be possible to improve the estimation of distributed sources [VCK95]. But as $\boldsymbol{D}_{\text{fused}}$ is not regular all theorems and corollaries developed in this thesis do not hold for the fused LASSO.

A. Algorithms

A.1. Orthogonal Matching Pursuit

Table A.1 gives the Orthogonal Matching Pursuit (OMP) algorithm as introduced in [TG07] adapted to our notation.

	Implementation of (P0) via OMP: Given constants: $\boldsymbol{A} \in \mathbb{C}^{N \times M}$, $s \in \mathbb{N}$
1:	for $k = 1, 2, \ldots, s$
2:	Input: $\boldsymbol{y} \in \mathbb{C}^N$
3:	$m^k = \underset{m=\{1,\ldots,M\} \setminus \mathcal{M}^{k-1}}{\operatorname{argmax}} \left\{ \left(\boldsymbol{A}^H (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_0}^{k-1} \right)_m \right\}$
4:	$\mathcal{M}^k = \mathcal{M}^{k-1} \cup m^k$
5:	$\boldsymbol{x}_{\ell_0,\mathcal{M}}^k = \boldsymbol{A}_{\mathcal{M}^k}^+ \boldsymbol{y}$
6:	end
7:	Output: $\boldsymbol{x}_{\ell_0,\mathcal{M}^k}^s$

Table A.1.: OMP implementation

The main difference of OMP and our LASSO based approach is found in line 3. In OMP one uses the regression output $\boldsymbol{x}_{\ell_0}$ instead of $\boldsymbol{x}_{\ell_1}$ and detects the active set directly from the matched filter output.

A.2. Compressive Sampling Matching Pursuit

Table A.2 presents the implementation of the Compressive Sampling Matching Pursuit (CoSaMP) as proposed by [NT09].

	Implementation of (P0) via CoSaMP: Given constants: $\boldsymbol{A} \in \mathbb{C}^{N \times M}$, $s \in \mathbb{N}$
1:	iterate until stopping criterion met
2:	Input: $\boldsymbol{y} \in \mathbb{C}^N$
3:	$\boldsymbol{m}^k = L_{2s} \left\{ \boldsymbol{A}^H \left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^{k-1} \right) \right\}$
4:	$\mathcal{M}^k = \mathcal{M}^{k-1} \cup \boldsymbol{m}^k$
5:	$\boldsymbol{x}_{\ell_0,\mathcal{M}}^k = \boldsymbol{A}_{\mathcal{M}^k}^+ \boldsymbol{y}$
6:	$\boldsymbol{x} = H_s(\boldsymbol{x}_{\ell_0,\mathcal{M}}^k)$
7:	end
8:	Output: $\boldsymbol{x}_{\ell_0,\mathcal{M}}^s$

Table A.2.: CoSaMP implementation

The operator L_{2s} takes the $2s$ largest elements and the hard thresholding operator H_s sets all except the s largest elements zero. Obviously, the two operators are closely related as $H_s(\boldsymbol{x}) = \boldsymbol{x}_{L_s(\boldsymbol{x})}$, see, e.g., [FR13] for details.

A.3. Compressive Sensing on Kalman Filtered Residual

The algorithm introduced in [Vas09] is re-written and adapted to our notation in Table A.3. After an initial Kalman filtering phase, CS is ran on the residuals $\boldsymbol{y}_{\mathrm{res}}[k] = \boldsymbol{y}[k] - \boldsymbol{A}\boldsymbol{x}_{\mathrm{init}}[k]$. The initial and the residual solution determine a temporary active set on which LS regression takes place. To account for vanished sources, the regressed vector is controlled for activity and a final LS is applied.

	Implementation of sequential estimation via Kalman filtering:
	Given constants: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{P}_0$, σ^2_{sys}, μ_{res}
2:	Input: $\boldsymbol{y}[k] \in \mathbb{C}^N$
	Initial Kalman filtering
3:	$\boldsymbol{P}[k\|k-1] = \boldsymbol{P}[k-1] + \sigma^2_{\mathrm{sys}}\boldsymbol{I}_{\mathcal{M}}$
4:	$\boldsymbol{K}[k] = \boldsymbol{P}[k\|k-1]\boldsymbol{A}^H \left(\boldsymbol{A}\boldsymbol{P}[k\|k-1]\boldsymbol{A}^H + \sigma\boldsymbol{I}\right)^{-1}$
5:	$\boldsymbol{P}[k] = (\boldsymbol{I} - \boldsymbol{K}[k]\boldsymbol{A})\,\boldsymbol{P}[k\|k-1]$
6:	$\boldsymbol{x}_{\mathrm{init}}[k] = (\boldsymbol{I} - \boldsymbol{K}[k]\boldsymbol{A})\,\boldsymbol{x}[k-1] + \boldsymbol{K}[k]\boldsymbol{y}[k]$
7:	$\boldsymbol{y}_{\mathrm{res}}[k] = \boldsymbol{y}[k] - \boldsymbol{A}\boldsymbol{x}_{\mathrm{init}}[k]$
	Compressed sensing on the residuals
8:	$\boldsymbol{x}_{\mathrm{res}}[k] = \mathrm{LASSO}\,(\boldsymbol{A}, \mu_{\mathrm{res}})$
9:	$\boldsymbol{x}[k] = \boldsymbol{x}_{\mathrm{init}}[k] + \boldsymbol{x}_{\mathrm{res}}[k]$
	Active set detection and regression
10:	$\mathcal{M}^{\mathrm{temp}}[k] = \{m \Big\vert \, \lvert x_m[k]\rvert > \delta\}$, $\delta = \epsilon\lVert\boldsymbol{x}[k]\rVert_\infty$
11:	$\boldsymbol{x}'_{\ell_0,\mathcal{M}^{\mathrm{temp}}}[k] = \boldsymbol{A}^+_{\mathcal{M}^{\mathrm{temp}}[k]}\boldsymbol{y}[k]$
	Deletion and final LS
12:	$\mathcal{M}[k] = \{m \Big\vert \, \lvert x'_{\ell_0,m}[k]\rvert > \delta\}$, $\delta = \epsilon\lVert\boldsymbol{x}[k]\rVert_\infty$
13:	$\boldsymbol{x}_{\ell_0,\mathcal{M}}[k] = \boldsymbol{A}^+_{\mathcal{M}[k]}\boldsymbol{y}[k]$
	Covariance update
14:	$\boldsymbol{P}_{\mathcal{M},\mathcal{M}}[k] = \sigma_2 \left(\boldsymbol{A}^H_{\mathcal{M}}\boldsymbol{A}_{\mathcal{M}}\right)^{-1}$
15:	end
	Output: $\boldsymbol{x}_{\ell_0}[k]$

Table A.3.: Compressive Sensing on Kalman filtered residuals - Implementation

B. Moore-Pensrose Inverse and Projections

B.1. Moore-Penrose Inverse

This section is a short excerpt of [BH12]. The Moore-Penrose Inverse provides a solution to the least squares problem $\boldsymbol{x} = \operatorname{argmin}_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|$. All possible minimizers are within the set

$$\left\{ \boldsymbol{A}^{+}\boldsymbol{y} + \underbrace{\left(\boldsymbol{I} - \boldsymbol{A}^{+}\boldsymbol{A}\right)}_{\boldsymbol{P}_{\mathrm{null}(\boldsymbol{A}^{H})}} \boldsymbol{z}, \boldsymbol{z} \in \mathbb{C}^{M} \right\} . \tag{B.1}$$

For over-determined system of equations the projection onto the left nullspace vanishes and for under-determined ones this nullspace term makes the solution non-unique.
The common approach for describing the Moore-Penrose inverse makes use of the singular value decomposition of $\boldsymbol{A}$. The singular value decomposition factors a matrix into a product of unitary and diagonal matrices

$$\boldsymbol{A} = (\underbrace{\boldsymbol{U}_1\, \boldsymbol{U}_2}_{\boldsymbol{U}})\, \boldsymbol{\Sigma} \left(\underbrace{\begin{array}{c} \boldsymbol{V}_1 \\ \boldsymbol{V}_2 \end{array}}_{\boldsymbol{V}} \right)^{H} . \tag{B.2}$$

The matrix $\boldsymbol{\Sigma}$ contains descending sorted non-negative singular values which usually leads to some zero blocks belonging to nullspaces.

$$\boldsymbol{\Sigma} = \left(\begin{array}{ccc|ccc} \sigma_1 & & & & \vdots & \\ & \ddots & & \cdots & 0 & \cdots \\ & & \sigma_r & & \vdots & \\ \hline & \vdots & & & \vdots & \\ \cdots & 0 & \cdots & \cdots & 0 & \cdots \\ & \vdots & & & \vdots & \end{array}\right) \quad \Leftrightarrow \quad \boldsymbol{\Sigma}^{+} = \left(\begin{array}{ccc|ccc} \frac{1}{\sigma_1} & & & & \vdots & \\ & \ddots & & \cdots & 0 & \cdots \\ & & \frac{1}{\sigma_r} & & \vdots & \\ \hline & \vdots & & & \vdots & \\ \cdots & 0 & \cdots & \cdots & 0 & \cdots \\ & \vdots & & & \vdots & \end{array}\right) \tag{B.3}$$

With $\boldsymbol{\Sigma}^{+}$ the Moore-Penrose inverse is now defined as

$$\boldsymbol{A}^{+} = \boldsymbol{V}\boldsymbol{\Sigma}^{+}\boldsymbol{U}^{H} , \tag{B.4}$$

as the following conditions for a pseudo-inverse are fulfilled

$$\boldsymbol{A}\boldsymbol{A}^{+}\boldsymbol{A} = \boldsymbol{A} \qquad \boldsymbol{U}\boldsymbol{\Sigma}\underbrace{\boldsymbol{V}^{\boldsymbol{H}}\boldsymbol{V}}_{\boldsymbol{I}}\boldsymbol{\Sigma}^{+}\underbrace{\boldsymbol{U}^{H}\boldsymbol{U}}_{\boldsymbol{I}}\boldsymbol{\Sigma}\boldsymbol{V}^{\boldsymbol{H}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\boldsymbol{H}} \tag{B.5}$$

$$\boldsymbol{A}^{+}\boldsymbol{A}\boldsymbol{A}^{+} = \boldsymbol{A}^{+} \qquad \boldsymbol{V}\boldsymbol{\Sigma}^{+}\boldsymbol{U}^{H}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\boldsymbol{H}}\boldsymbol{V}\boldsymbol{\Sigma}^{+}\boldsymbol{U}^{H} = \boldsymbol{V}\boldsymbol{\Sigma}^{+}\boldsymbol{U}^{H} . \tag{B.6}$$

Properties of the Moore-Penrose Inverse

Table B.1.: Table of Moore-Penrose inverse properties

$(\boldsymbol{A}^+)^+$	$=$	$\boldsymbol{A}$	$(\boldsymbol{A}^+)^H$	$=$	$(\boldsymbol{A}^H)^+$
$(c\boldsymbol{A})^+$	$=$	$c^{-1}(\boldsymbol{A})^+$	$\forall$	$c \in \mathbb{C},$	$c \neq 0$
$\boldsymbol{A}^+$	$=$	$\boldsymbol{A}^+(\boldsymbol{A}^+)^H\boldsymbol{A}^H$	$\boldsymbol{A}$	$=$	$\boldsymbol{A}\boldsymbol{A}^H(\boldsymbol{A}^+)^H$
$\boldsymbol{A}^H$	$=$	$\boldsymbol{A}^H\boldsymbol{A}\boldsymbol{A}^+$	$\boldsymbol{A}^+$	$=$	$\boldsymbol{A}^H(\boldsymbol{A}^+)^H\boldsymbol{A}^+$
$\boldsymbol{A}$	$=$	$(\boldsymbol{A}^+)^H\boldsymbol{A}^H\boldsymbol{A}$	$\boldsymbol{A}^H$	$=$	$\boldsymbol{A}^+\boldsymbol{A}\boldsymbol{A}^H$

B.2. Projections

Throughout this thesis at several positions, products of the dictionary and its Moore-Penrose inverse occur. All these products are projections on the four fundamental subspaces. Our dictionary $\boldsymbol{A}$ is over complete (under-determined) therefore lower zero blocks in $\boldsymbol{\Sigma}$ are non existing and the non-zero singular values are shown in figure B.1.

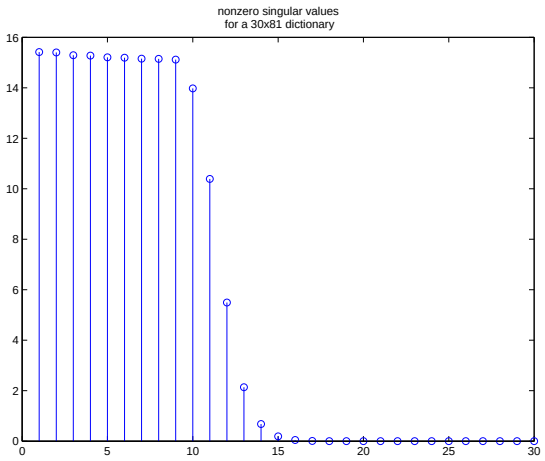


Figure B.1.: Non-zero singular values of the 30×81 dictionary $\boldsymbol{A}$. Some of the singular values are close to zero which makes $\boldsymbol{A}$ numerically bad conditioned.

Projection onto the Column Space

$$\boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H\boldsymbol{V}\boldsymbol{\Sigma}^+\boldsymbol{U}^H = \boldsymbol{U}\begin{pmatrix}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\boldsymbol{U}^H = \boldsymbol{U}_1\boldsymbol{U}_1^H = \boldsymbol{P}_{\mathrm{span}(\boldsymbol{A})} \tag{B.7}$$

Projection onto the Right Nullspace

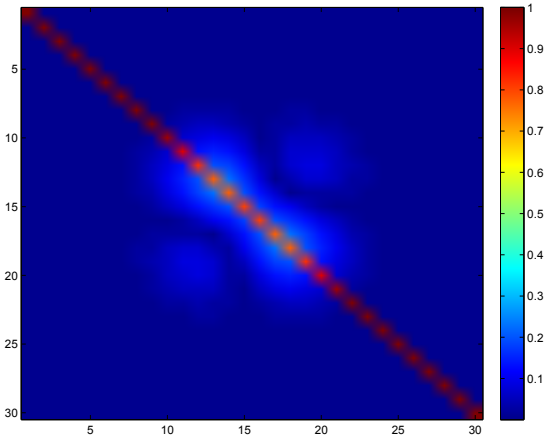
$$\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{U}\boldsymbol{I}\boldsymbol{U}^H - \boldsymbol{U}\begin{pmatrix}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\boldsymbol{U}^H = \boldsymbol{U}\begin{pmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{I}\end{pmatrix}\boldsymbol{U}^H = \boldsymbol{U}_2\boldsymbol{U}_2^H = \boldsymbol{P}_{\mathrm{null}(\boldsymbol{A})} \tag{B.8}$$

Projection onto the Rowspace

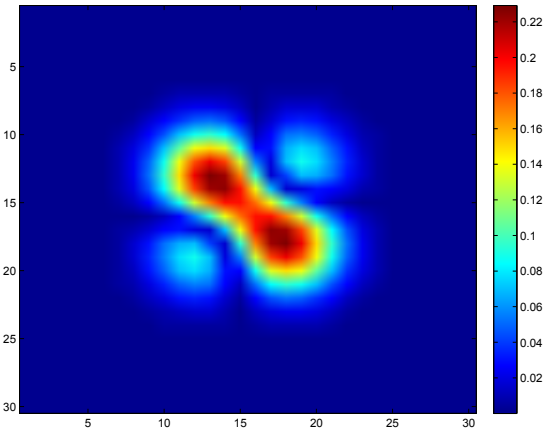
$$\boldsymbol{A}^+\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Sigma}^+\boldsymbol{U}^H\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H = \boldsymbol{V}_1\boldsymbol{V}_1^H = \boldsymbol{P}_{\mathrm{span}(\boldsymbol{A}^H)} \tag{B.9}$$

Projection onto the Left Nullspace

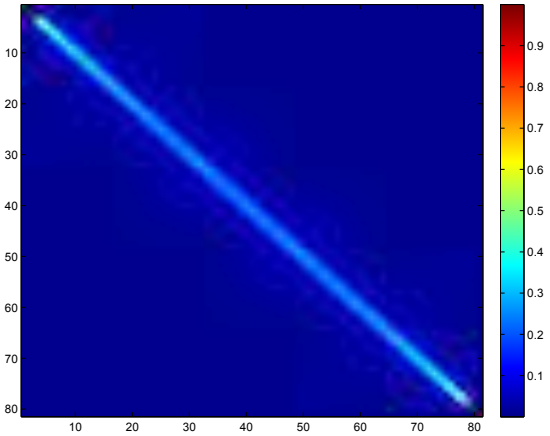
$$\boldsymbol{I} - \boldsymbol{A}^{+}\boldsymbol{A} = \boldsymbol{V}\boldsymbol{V}^{H} - \boldsymbol{V}_1\boldsymbol{V}_1^{H} = \boldsymbol{V}_2\boldsymbol{V}_2^{H} = \boldsymbol{P}_{\mathrm{null}(\boldsymbol{A}^{H})} \tag{B.10}$$



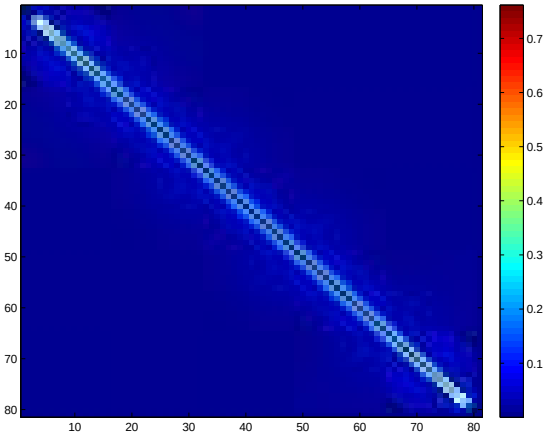
Projection onto the column space.



Projection onto the right nullspace.



Projection onto the rowspace



Projection onto the left nullspace.

Figure B.2.: Magnitude of the different projection matrices.

C. Matlab Code

C.1. Single Snapshot Procedures

C.1.1. Configuration and Simulation Start

```
%% Single time sample
close all;
clear all; %clc;

%% generate A
ULA;

%% generate sources
coherence = 'strong';       %'strong','small','none','super_strong'
SOURCES;

%% single shot measurement
noiseless = false;
snr = 20; % signal-to-noise ratio in dB
MEASUREMENTS;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%change settings here to obtain different algorithms
% iterative => fast, noniterative => order recursive
iterative = true;
detection = 'primal';   %'primal','dual'
dual_lasso = false;   % dual_lass = true leads to the algo in Table IV
plot_in_fig = true;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

x_l1=zeros(n,1);
x_l0=zeros(n,1);
active_set=[];

%==== line 1 ====
p=1;    %sparsity order counter
ii=0;   %peaks in dual counter
F=0.70;  %trade off parameter
ff=0;   %figure counter
u1=2*pinv(D)*A'*yn;

%==== line 2 ====
while p<=s
    ii=ii+1;

    if plot_in_fig
        ff=1;
    else
        ff=ff+1;
    end
```

```
46
47      if ii>2*s
48          break; %to prevent infinite loops
49      end
50
51      %==== line 3 ====
52      [mu_val,~] = findpeaks(abs(u1),'SORTSTR','descend','MINPEAKHEIGHT',0.001*max(abs(2*pinv(D)
            *A'*yn)) );
53
54      if iterative
55          mu=(1-F)*mu_val(s)+F*mu_val(s+1);
56      else
57          mu=(1-F)*mu_val(ii)+F*mu_val(ii+1);
58      end
59
60      if ~dual_lasso
61          %==== line 4a =====
62          cvx_begin
63          cvx_quiet true
64          variable x_l1(n) complex
65          variable z(n) complex
66          dual variable u
67          dual variable v
68          minimize (sum_square_abs(yn - A*x_l1) + mu* norm(z,1))
69          subject to
70          u :  real(z) == real(D*x_l1);
71          v :  imag(z) == imag(D*x_l1);
72          cvx_end
73
74          %==== line 4b =====
75          u1=complex(u,v);
76  %         u1 = 2*pinv(D)*A'*(y-A*x_l1);
77      else
78          if strcmp(detection,'primal')
79              error('for dual_lasso only dual detection is possible');
80          end
81
82          D_tilde=D*pinv(A);
83          P=A*pinv(A);
84          y_tilde=P*y;
85          U=null(A);
86          zerovector=zeros(n-m,1);
87
88          cvx_begin
89                          variable u1(n) complex;
90                                  maximize (sum_square_abs(y)- sum_square_abs(y_tilde-ctranspose
                                        (D_tilde)*u1));
91                          subject to
92                                  ctranspose(D*U)*u1==zerovector
93                                  norm(u1,Inf) <= mu
94          cvx_end
95      end
96
97      if plot_in_fig
98          clf
99      end
100     figure(ff)
101     subplot(211)
```

```
102     plot(theta,20*log10(abs(2*pinv(D)*A'*yn)/(max(abs(2*pinv(D)*A'*yn)))),'b',theta,20*log10(
            abs(u1)/(max(abs(u1)))),'k',theta(pf),zeros(size(pf)),'or');%,theta,20*log10(mu/max(
            abs(2*pinv(D)*A'*yn)))*ones(size(theta)),'k:' );
103     l=legend('$2 \mathbf{D}^{-H} \mathbf{A}^H \mathbf{y}$','$2 \mathbf{D}^{-H} \mathbf{A}^H (\
            mathbf{y}-\mathbf{A}\mathbf{x})$','${\rm sources}$','Location','SouthWest');
104     set(l,'Interpreter','Latex');
105     xlabel('DOA [degrees]'); ylabel('normalized power (dB)'); set(gca,'xlim',[theta(1) theta(
            end)] ,'ylim',[-40 1])
106
107     %==== line 5 ====
108     switch detection
109         case 'primal'
110             eps_x = 10^-4;
111             [~,active_set] = findpeaks(abs(x_l1),'SORTSTR','descend','MINPEAKHEIGHT',max
                    ([10^-4 eps_x*norm(x_l1,'inf')]) );
112         case 'dual'
113             eps_u = 10^-2;
114             [~,active_set] = findpeaks(abs(u1)/mu,'SORTSTR','descend','MINPEAKHEIGHT',1-eps_u
                    );
115     end
116     if (length(active_set) > p) && (~iterative)
117         active_set = active_set(1:p);
118     end
119
120     size_active=length(active_set);
121
122     %==== line 6ff ====
123     if ~iterative                %order recursive
124         switch detection
125             case 'primal'
126                 if size_active<p
127                     ii=ii+1;
128                 else
129                     x_active=pinv(A(:,active_set))*yn;
130                     x_l0=zeros(size(u));
131                     x_l0(active_set)=x_active;
132                     p=p+1;
133                 end
134             case 'dual'
135                 x_active=pinv(A(:,active_set))*yn;
136                 eps_l0 = 10^-4;
137                 l0_norm=sum(abs(x_active)>eps_l0); %check 0 norm
138                 if l0_norm<p
139                     ii=ii+1;
140                 else
141                     x_l0=zeros(size(u));
142                     x_l0(active_set)=x_active;
143                     p=p+1;
144                 end
145         end
146     else                         %fast algo
147         x_active=pinv(A(:,active_set))*yn;
148         x_l0=zeros(size(u));
149         x_l0(active_set)=x_active;
150
151         switch detection
152             case 'primal'
153                 if size_active >= s
154                     break;
```

```
                end
            case 'dual'
                eps_l0 = 10^-4;
                l0_norm=sum(abs(x_active)>eps_l0); %check 0 norm
                if l0_norm >= s
                    break;
                end
        end
    end

    figure(ff)
    subplot(212)
    hold on;
    stem(theta,20*log10(abs(x_l0)),'b','filled');
    stem(theta,20*log10(abs(fk)),'r');
    title([num2str(p-1) ' sparse vector estimate (real sparsity degree ' num2str(s) ')  |M| ='
        num2str(length(active_set))]);
    xlabel('DOA [degrees]');ylabel('power (dB)'); set(gca,'xlim',[theta(1) theta(end)] ,'ylim'
        ,[0 20*log10(max(abs(fk)))+10])
    l=legend('${\ell_0 } \;\; {\rm estimates}$','${\rm true}\;\; {\rm sources}$','Location','
        SouthWest');
    set(l,'Interpreter','Latex');

    if plot_in_fig
        waitforbuttonpress
    end
end

%to plot in the iterative case after the break
figure(ff)
subplot(212)

hold on;
stem(theta,20*log10(abs(x_l0)),'b','filled');
stem(theta,20*log10(abs(fk)),'r');
title([num2str(p-1) ' sparse vector estimate (real sparsity degree ' num2str(s) ')  |M| ='
    num2str(length(active_set))]);
xlabel('DOA [degrees]');ylabel('power (dB)'); set(gca,'xlim',[theta(1) theta(end)] ,'ylim',[0
    20*log10(max(abs(fk)))+10])
l=legend('${\ell_0 } \;\; {\rm estimates}$','${\rm true}\;\; {\rm sources}$','Location','
    SouthWest');
set(l,'Interpreter','Latex');
```

C.1.2. ULA Model Generation

```
% Parameters
c = 1500;        % speed of sound
f = 200;         % frequency
lambda = c/f;    % wavelength
k = 2*pi/lambda;% wavenumber

% ULA-horizontal
N=30;
d = 1/2*lambda; % intersensor spacing
q = 0:1:(N-1);% sensor numbering
xq = (q-(N-1)/2)*d; % sensor locations

theta = -90:0.5:90;
theta_r = theta*pi/180;
```

```
u = sin(theta_r);

% Dimensions of the problem
n = length(u);  % total length of the grid (long signal)
m = length(xq); % number of measurements m<n

% Represenation matrix (steering matrix)
A = (1/sqrt(n))*exp(1i*2*pi/lambda*xq'*u);
```

C.1.3. Source Settings

```
switch coherence
    case 'strong'
        pf(1) = floor(n/2)-5;
        pf(2) = floor(n/2)+10;
        pf(3) = floor(n/2)+17;
        pf(4) = floor(n/2)+150;
        s = 4;          % sparsity level= # of sources s<<n

        % Source vector
        fk = zeros(n,1);
        fk(pf(1)) = -8;
        fk(pf(2)) = +13i;
        fk(pf(3)) = +10*(1i+1)/sqrt(2);
        fk(pf(4)) = (-10*(1i+2)/sqrt(5));

    case 'super_strong'
        pf(1) = floor(n/2)-5;
        pf(2) = floor(n/2)+10;
        pf(3) = floor(n/2)+13;
        pf(4) = floor(n/2)+150;
        s = 4;          % sparsity level= # of sources s<<n

        % Source vector
        fk = zeros(n,1);
        fk(pf(1)) = -8;
        fk(pf(2)) = +13i;
        fk(pf(3)) = +10*(1i+1)/sqrt(2);
        fk(pf(4)) = (-10*(1i+2)/sqrt(5));

    case 'small'
        pf(1) = floor(n/2)-5;
        pf(2) = floor(n/2)+10;
        pf(3)= floor(n/2)+150;
        s = 3;          % sparsity level= # of sources s<<n

        % Source vector
        fk = zeros(n,1);
        fk(pf(1)) = -8;
        fk(pf(2)) = +13i;
        fk(pf(3)) = (-10*(1i+2)/sqrt(5));

    case 'none'
        pf(1) = floor(n/2)-38;
        pf(2) = floor(n/2)+1;
        pf(3) = floor(n/2)+40;
        s = 3;          % sparsity level= # of sources s<<n

        % Source vector
```

```
        fk = zeros(n,1);
        fk(pf(1)) = -7;
        fk(pf(2)) = +13i;
        fk(pf(3)) = +10*(1i+1)/sqrt(2);
end
```

C.1.4. Measurement Generation

```
% Measurements (sensor signals)
y = A*fk;

% Noisy measurements
rnl = 10^(-snr/20)*norm(y);          % relative noise level

% Specify rand stream
stream = RandStream('mt19937ar','Seed',1);

% Generate noise
nwhite= randn(stream,m,2);
nwhite=complex(nwhite(:,1),nwhite(:,2))/sqrt(2);
e = nwhite * rnl/m;                    % error vector

if noiseless
    yn = y ;
else
    yn = y+ e;                          % adding noise to the data vector
end

Weights = eye(n);

if noiseless
    D= Weights;
else
    D=rnl^2*Weights;
end
```

C.2. Sequential Procedures

C.2.1. Configuration and Simulation Start

```
%% Single time sample
 close all; % cab('last');       %close all or keep last figure open
clear all; %clc;

%% generate A
ULA;

%%%%%%%%%%%%%%%%%%%
timesteps=3;
realizations =1;
%%%%%%%%%%%%%%%%%%%

%fix random seed for fair comparison
seed = RandStream('mcg16807','Seed',5);
RandStream.setGlobalStream(seed);

size_act = nan(realizations,timesteps);
l2_res = nan(realizations,timesteps);
```

```
l2_ref = nan(realizations,timesteps);
mu_history = nan(realizations,timesteps);
active_pattern = nan(30,realizations,timesteps);
source_pattern = nan(30,timesteps);
TOT=0;  FA =0;
Weights = eye(n);
dens_evo=nan(n,timesteps);

approach = 'density';  %density,kalman

for r=1:realizations

    for t=1:timesteps

        %% generate sources
        pf(1) = floor(n/2)-38;
        pf(1)= pf(1)+t;          %move to the right each timestep
        pf(2) = floor(n/2)+1;
        pf(3) = floor(n/2)+40;

        s = length(pf);          % sparsity level= # of sources s<<n
        source_pattern(1:s,t)=pf-floor(n/2);

        % Source vector
        fk = zeros(n,1);
        fk(pf(1)) = -7;
        fk(pf(2)) = +13i;
        fk(pf(3)) = +10*(1i+1)/sqrt(2);

        %% generate measurements
        SNR = 20;
        y = A*fk;

        rnl = 10^(-SNR/20)*norm(y);          % relative noise level
        nwhite = complex(randn(m,1),randn(m,1))/sqrt(2);  % complex i.i.d. ~ N(0,1)
        e = nwhite * rnl;                    % error vector

        yn = y+ e;                           % adding noise to the data vector
        D=rnl^2*Weights;

        %% sequential approaches
        switch approach
            case 'density'
                DENSITY_EVO;
                l2_ref(r,t) = norm(yn-A*x_ref);      %reference solution with W=I
            case 'kalman'
                SPARSE_KALMAN;
        end

        %active set pattern
        active_pattern(1:length(active_set),r,t)=active_set-floor(n/2);

        %benchmark parameters
        size_act(r,t) = length(active_set);
        l2_res(r,t) = norm(yn-A*x_hat);
        mu_history(r,t)=mu(end);

        %% TIME ON TARGET? FALSE ALARM?
        delta=2;
```

```
78          x_false=x_hat;
79          for ii=1:s
80              if norm(x_hat(pf(ii)-delta:pf(ii)+delta)) > (2*delta+1)*rnl  % ith source detected
                    ?
81                  TOT=TOT+1;
82              end
83              x_false((pf(ii)-delta:pf(ii)+delta))=zeros(size((pf(ii)-delta:pf(ii)+delta)));
84          end
85          %how many false alarms?
86          FA=FA+sum(x_false>rnl);
87      end
88  end
89
90  figure(2)  % benchmark parameters
91  hold on
92  subplot(3,1,1)
93  stem(mean(size_act,1));
94  ylabel('$| \mathcal{M} |$','interpreter','latex');
95  subplot(3,1,2)
96  plot(1:timesteps,mean(mu_history,1));
97  ylabel('$\mu$','interpreter','latex');
98  subplot(3,1,3)
99  plot(1:timesteps,mean(l2_res,1)./norm(e));
100 ylabel('$\frac{\|\mathbf{y}[k]-\mathbf{A}\mathbf{x}[k]\|_2}{\|\mathbf{n}\|_2}$','interpreter',
        'latex');
101 xlabel('time');
102
103 figure(3) % outcome of the tracker
104 hold on;
105 plot(source_pattern(:,:)','r');
106 for rr=1:realizations
107     if strcmp(approach,'kalman')
108         marker='gs';
109     elseif p==-1
110         marker='ko';
111     elseif p==1
112         marker='b>';
113     end
114     plot(1:timesteps,squeeze(active_pattern(1:20,rr,:)),marker,'markersize',4);
115 end
116 set(gca,'xlim',[1 timesteps],'ylim',[theta(1) theta(end)])
117
118 %dummy objects for legend
119 p = plot(1:timesteps,nan(size(1:timesteps)),'r',1:timesteps,nan(size(1:timesteps)),'ko',1:
        timesteps,nan(size(1:timesteps)),'b>',1:timesteps,nan(size(1:timesteps)),'gs');
120 leg=legend(p,'true trajectory','density evolution $p=-1$','density evolution $p=1$','CS on
        Kalman filt. res.');
121 set(leg,'Interpreter','latex','location','southeast');
122
123 %TOT probability
124 P_TOT=TOT/(s*realizations*timesteps);
125
126 %FA probability
127 P_FA = FA/(s*realizations*timesteps);
```

C.2.2. CS on Kalman Filtered Residuals

```
1  sigma_sys=rnl/100;
2  P=zeros(n,n);
```

```
3
4  if t==1        %at time 0 just CS
5      %taken from "Real-time dynamic MR image recontruction... , Vaswani"
6      mu=sqrt(2*log2(n))*rnl;
7
8      cvx_begin quiet
9          variable x_init(n) complex
10                         minimize (sum_square_abs(yn - A*x_init) + mu * norm(x_init,1))
11     cvx_end
12
13     % detect active set
14     active_set=[];
15     eps_x = 10^-4;
16     [~,active_set]=findpeaks(abs(x_init),'SORTSTR','descend','MINPEAKHEIGHT',max([10^-4 eps_x*
           norm(x_init,'inf')]) );
17 else
18     %Kalman filtering
19     I_T = zeros(n,1);
20     I_T(active_set)=1;
21     I_T=diag(I_T);
22
23     P_pred = P + sigma_sys * I_T;
24     K_gain = P_pred*A'*pinv(A*P_pred*A'+ rnl*eye(m));
25     P = (eye(n)-K_gain*A)*P_pred;
26
27     x_init=(eye(n)-K_gain*A)*x_hat+K_gain*yn;
28 end
29
30 %% residuals (creation)
31 y_res = yn-A*x_init;
32
33 mu_res = 0.6*mu;        %heuristically chosen
34
35 cvx_begin quiet
36         variable x_res(n) complex
37                         minimize (sum_square_abs(y_res - A*x_res) + mu_res* norm(1*x_res,1))
38 cvx_end
39
40 x_help = x_init+x_res;
41
42 % detect active set
43 active_set=[];
44 eps_x = 10^-4;
45 [~,active_set]=findpeaks(abs(x_help),'SORTSTR','descend','MINPEAKHEIGHT',max([10^-4 eps_x*norm
       (x_help,'inf')]) );
46
47 % LS regression on active set
48 x_hat = zeros(size(x_help));
49 x_hat(active_set)=pinv(A(:,active_set))*yn;
50
51 %% deletion (annihilation)
52 [~,active_set]=findpeaks(abs(x_hat),'SORTSTR','descend','MINPEAKHEIGHT',1 );
53 x_hat = zeros(size(x_hat));
54 x_hat(active_set)=pinv(A(:,active_set))*yn;
55
56 %% update error covariance matrix
57 P=zeros(n,n);
58 P(active_set,active_set)=pinv(A(:,active_set)'*A(:,active_set))*rnl^2;
```

C.2.3. Density Evolution

```
%% produce reference with W=eye() weights
GEN_LASSO_non_seq;                              % x_ref

% detect active set for non seq
active_set_non=[];
eps_x = 10^-4;
[~,active_set_non]=findpeaks(abs(x_ref),'SORTSTR','descend','MINPEAKHEIGHT',max([10^-4 eps_x*
    norm(x_ref,'inf')]) );

x_ref_l0=zeros(size(x_ref));
x_ref_l0(active_set_non)=pinv(A(:,active_set_non))*yn;

%============================
%% sequential implementation
GEN_LASSO;

% detect active set
active_set=[];
eps_x = 10^-4;
[~,active_set]=findpeaks(abs(x_hat),'SORTSTR','descend','MINPEAKHEIGHT',max([10^-4 eps_x*norm(
    x_hat,'inf')]) );

x_hat_l0=zeros(size(x_hat));
x_hat_l0(active_set)=pinv(A(:,active_set))*yn;

lambda = diag(Weights * mu(end));

%% update step
% mean fit
lambda_up = lambda.*(1 - (1/mu(end)^2)*abs(u1).^2);
W_up = lambda_up/mu(end);
W_up= max(W_up,100*eps);        % to prevent undefined division

%% prediction step
W_pre = nan(size(W_up));

% uniform motion model
l=2;
Motion = diag(ones(1,2*l+1)/(2*l+1));        %uniform motion model

%%%%%%
p= -1;
w_0 = 0.01;
factor = 10;
%%%%%%

for zz=1:n
    if sum(zz==[1:(l+1), n-(l+1):n])>0     % no motion model for the border region
        W_pre(zz) = W_up(zz)+factor*w_0;
    else
        if min(W_up(zz-l:zz+l))<eps_x           % neighbourhood of an active source?
            if p==0          % weighted geometric mean
                W_pre(zz) = exp(sum(Motion*log(W_up(zz-l:zz+l))))+w_0;
            else             % all other means besides min and max
                W_pre(zz)  = (sum(Motion*W_up(zz-l:zz+l).^(2*p)))^(1/(2*p))+w_0;
            end
        else
```

```
56                W_pre(zz)=W_up(zz)+factor*w_0;
57            end
58        end
59    end
60
61    Weights = diag(W_pre/norm(W_pre,'inf'));  %rescale
62    lambda = diag(Weights * mu(end));
63
64    dens_evo(:,t,r)=diag(Weights);
65    if t == timesteps && r == realizations
66        figure(1)
67        imagesc(1:t,theta,mean(dens_evo,3));
68        set(gca,'xlim',[1 timesteps] ,'ylim',[theta(1) theta(end)])
69        xlabel('time step $k$','interpreter','latex');
70        ylabel('DOA','interpreter','latex');
71        title('Weight Evolution','interpreter','latex');
72        colormap('hot');
73        colorbar;
74    end
```

C.2.4. Generalized LASSO Implementation

```
1   mu=[];
2   u1=2*pinv(D)*A'*yn;
3   mu_save = norm(u1,'inf');
4   ii=0;        %loopcounter
5   F=0.9;
6   p=0;
7
8   while 1
9       ii=ii+1;
10
11      if p<s     % do not in case of bisection
12          [mu_val,~] = findpeaks(abs(u1),'SORTSTR','descend','MINPEAKHEIGHT',0.000001*max(abs(2*
                pinv(D)*A'*yn)) );
13          mu(ii)=(1-F)*mu_val(ii)+F*mu_val(ii+1);
14      end
15
16      %Here the generalized LASSO is solved via Primal-Dual Interior-Point Methods
17      x_hat=zeros(n,1);
18      cvx_begin quiet
19          variable x_hat(n) complex
20          variable z(n) complex
21          dual variable u
22          dual variable v
23              minimize (sum_square_abs(yn - A*x_hat) + mu(ii)* norm(z,1))
24          subject to
25              u :  real(z) == real(D*x_hat);
26              v :  imag(z) == imag(D*x_hat);
27      cvx_end
28      u1=complex(u,v);
29
30      %detection of the active set in the primal domain
31      eps_x = 10^-4;
32      [~,active_set] = findpeaks(abs(x_hat),'SORTSTR','descend','MINPEAKHEIGHT',max([10^-4 eps_x
            *norm(x_hat,'inf')]) );
33
34      p=length(active_set);
35
```

```
36      if p == s
37          break
38      elseif p > s   %start bisection
39          mu(ii+1) = (mu(ii)+mu_save)/2;
40      elseif p<s
41          if ii>2*s
42              break;  %to prevent infinite loops
43          end
44          mu_save=mu(ii); %save mu for a possibly occuring bisection
45      end
46  end
```

List of Figures

List of Tables

Bibliography

[Ber99] Dimitri P Bertsekas. *Nonlinear programming*. Athena Scientific, 1999.

[BH12] João Carlos Alves Barata and Mahir Saleh Hussein. "The Moore–Penrose pseudoinverse: A tutorial review of the theory". In: *Brazilian Journal of Physics* 42.1-2 (2012), pp. 146–165.

[Boy+12] Stephen Boyd et al. *Matlab scripts for alternating direction method of multipliers*. Tech. rep. http://www. stanford. edu/boyd/papers/admm, 2012.

[BST12] Pantelis Bouboulis, Konstantinos Slavakis, and Sergios Theodoridis. "Adaptive learning in complex reproducing kernel Hilbert spaces employing Wirtinger's subgradients". In: *IEEE Transactions on Neural Networks and Learning Systems* 23.3 (2012), pp. 425–438.

[BTR13] Badri Narayan Bhaskar, Gongguo Tang, and Benjamin Recht. "Atomic norm denoising with applications to line spectral estimation". In: *IEEE Transactions on Signal Processing* 61.23 (2013), pp. 5987–5999.

[Bul03] Peter S Bullen. *Handbook of means and their inequalities*. Springer Science & Business Media, 2003.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Chi+11] Yuejie Chi et al. "Sensitivity to basis mismatch in compressed sensing". In: *IEEE Transactions on Signal Processing* 59.5 (2011), pp. 2182–2195.

[CRT06] Emmanuel J Candès, Justin Romberg, and Terence Tao. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information". In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 489–509.

[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.

[Efr+04] Bradley Efron et al. "Least angle regression". In: *The Annals of statistics* 32.2 (2004), pp. 407–499.

[Ela10] Michael Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media, 2010.

[FR13] Simon Foucart and Holger Rauhut. *A mathematical introduction to compressive sensing*. Springer, 2013.

[GBY08] Michael Grant, Stephen Boyd, and Yinyu Ye. *CVX: Matlab software for disciplined convex programming, version 1.21*. 2008.

[Gri11] Rémi Gribonval. "Should penalized least squares regression be interpreted as maximum a posteriori estimation?" In: *IEEE Transactions on Signal Processing* 59.5 (2011), pp. 2405–2410.

[He+07] Zhaoshui He et al. "Convolutive blind source separation in the frequency domain based on sparse representation". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.5 (2007), pp. 1551–1563.

[IN10] Michel T Ivrlac and Josef A Nossek. "Toward a circuit theory of communication". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 57.7 (2010), pp. 1663–1683.

[Lor+09] Ignace Loris et al. "Accelerating gradient projection methods for ℓ_1-constrained signal recovery by steplength selection rules". In: *Applied and computational harmonic analysis* 27.2 (2009), pp. 247–254.

[MAT13] MATLAB. *version 8.2.0.701 (R2013b)*. Natick, Massachusetts: The MathWorks Inc., 2013.

[Mec+13] Christoph F Mecklenbräuker et al. "Sequential Bayesian sparse signal reconstruction using array data". In: *IEEE Transactions on Signal Processing* 61.24 (2013), pp. 6344–6354.

[MGZ15] Christoph F Mecklenbräuker, Peter Gerstoft, and Erich Zöchmann. "Beamforming of the Residuals is the LASSO's Dual". In: *arXiv preprint arXiv:1502.04643* (2015).

[MÇW05] Dmitry Malioutov, Müjdat Çetin, and Alan S Willsky. "A sparse signal reconstruction perspective for source localization with sensor arrays". In: *IEEE Transactions on Signal Processing* 53.8 (2005), pp. 3010–3022.

[Nat95] Balas Kausik Natarajan. "Sparse approximate solutions to linear systems". In: *SIAM journal on computing* 24.2 (1995), pp. 227–234.

[NT09] Deanna Needell and Joel A Tropp. "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples". In: *Applied and Computational Harmonic Analysis* 26.3 (2009), pp. 301–321.

[PV11] Ashkan Panahi and Mats Viberg. "Fast LASSO based DOA tracking". In: *4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE. 2011, pp. 397–400.

[PV12] Ashkan Panahi and Mats Viberg. "Fast candidate points selection in the LASSO path". In: *IEEE Signal Processing Letters* 19.2 (2012), pp. 79–82.

[QLV09] Chenlu Qiu, Wei Lu, and Namrata Vaswani. "Real-time dynamic MR image reconstruction using Kalman filtered compressed sensing". In: *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2009, pp. 393–396.

[RAG04] Branko Ristic, Sanjeev Arulampalam, and Neil Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Vol. 685. Artech house Boston, 2004.

[RW09] Tyrrell R Rockafellar and Roger J-B Wets. *Variational analysis*. Vol. 317. Springer Science & Business Media, 2009.

[Tan+13] Gongguo Tang et al. "Compressed sensing off the grid". In: *IEEE Transactions on Information Theory* 59.11 (2013), pp. 7465–7490.

[TG07] Joel A Tropp and Anna C Gilbert. "Signal recovery from random measurements via orthogonal matching pursuit". In: *IEEE Transactions on Information Theory* 53.12 (2007), pp. 4655–4666.

[TGA14] Oguzhan Teke, Ali Cafer Gurbuz, and Orhan Arikan. "A Recursive Way for Sparse Reconstruction of Parametric Spaces". In: *2014 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. IEEE. 2014.

[Tib+05] Robert Tibshirani et al. "Sparsity and smoothness via the fused lasso". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.1 (2005), pp. 91–108.

[Tib96] Robert Tibshirani. "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1996), pp. 267–288.

[Tik95] Andrei Nikolaevich Tikhonov. *Numerical methods for the solution of ill-posed problems*. Vol. 328. Springer Science & Business Media, 1995.

[Tro04] Joel A Tropp. "Greed is good: Algorithmic results for sparse approximation". In: *IEEE Transactions on Information Theory* 50.10 (2004), pp. 2231–2242.

[Tro06] Joel A Tropp. "Just relax: Convex programming methods for identifying sparse signals in noise". In: *IEEE Transactions on Information Theory* 52.3 (2006), pp. 1030–1051.

[TT11] Ryan J Tibshirani and Jonathan Taylor. "The Solution Path of the Generalized LASSO". In: *Annals of statistics* 39.3 (2011), pp. 1335–1371.

[Vas09] Namrata Vaswani. "KF-CS: compressive sensing on Kalman filtered residual". In: *arXiv preprint arXiv:0912.1628* (2009).

[Vas10] Namrata Vaswani. "LS-CS-residual (LS-CS): compressive sensing on least squares residual". In: *IEEE Transactions on Signal Processing* 58.8 (2010), pp. 4108–4120.

[VCK95] Shahrokh Valaee, Benoit Champagne, and Peter Kabal. "Parametric localization of distributed sources". In: *IEEE Transactions on Signal Processing* 43.9 (1995), pp. 2144–2153.

[VT02] Harry L Van Trees. *Optimum array processing: part IV of detection, estimation, and modulation*. 2002.

[Wri97] Stephen J Wright. *Primal-dual interior-point methods*. Siam, 1997.

[XGM14] Angeliki Xenaki, Peter Gerstoft, and Klaus Mosegaard. "Compressive beamforming". In: *The Journal of the Acoustical Society of America* 136.1 (2014), pp. 260–271.

[ZGM15] Erich Zöchmann, Peter Gerstoft, and Christoph F Mecklenbräuker. "Density Evolution of Sparse Source Signals". In: *3rd International Workshop on Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing*. accepted. IEEE. 2015.

Hiermit erkläre ich, dass die vorliegende Arbeit gemäß dem Code of Conduct - Regeln zur Sicherung guter wissenschaftlicher Praxis (in der aktuellen Fassung des jeweiligen Mitteilungsblattes der TU Wien), insbesondere ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel, angefertigt wurde. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Wien, April 14, 2015

Erich Zöchmann

Submitted Publications

Beamforming of the Residuals is the LASSO's Dual

Christoph F. Mecklenbräuker, *Senior Member, IEEE,* Peter Gerstoft, *Member, IEEE,* Erich Zöchmann

***Abstract*—Waves hidden in additive noise are observed by a sensor array. The waves are assumed to originate from a sparse set of sources. We treat the estimation of the sparse set of sources as a generalized complex-valued LASSO problem. The generalized complex-valued LASSO problem is strictly convex and strong duality holds. The corresponding dual problem is interpretable as a weighted conventional beamformer acting on the residuals of the LASSO. Moreover, this establishes a simple linear-affine relation between the dual and primal vectors. The solution path of the complex-valued LASSO is analyzed and three procedures for signal processing are proposed and evaluated which are based on the generalized LASSO and its dual: An order-recursive procedure and two iterative procedures which are based on a further approximation.**

***Index Terms*—sparsity, generalized LASSO, duality theory**

I. Introduction

The modification of quadratic optimization criteria by introducing suitable constraints leads to sparse solutions. This idea has opened new directions in many fields of signal processing, e.g., in linear regression, compressive sensing, channel identification, and equalization. The use of (possibly weighted) ℓ_1-norms induces sparsity of the solution.

The early results for sparse signals [1], [2], [3], [4] have been extended to compressible (approximately sparse) signals and sparse signals buried in noise [5], [6], [7], [8], [9] which renders the framework applicable to problems in array processing.

Tibshirani [10] noted that the generalized LASSO problem is difficult to analyze directly because the nondifferentiable ℓ_1 penalty is composed with a linear transformation. He showed that it is more intuitive to derive a solution path to the corresponding Lagrangian dual problem. Moreover, the dual approach comes without additional computational cost.

Fortunati et al. [11] found that super-resolution beyond the Rayleigh limit is achievable for compressed sensing based beamformers even for the single snapshot case. Xenaki et al. [14] noted that such techniques are implementable with single-snapshot data and irregular array configurations. Their bias and resolution analysis indicates that sparse signal reconstruction enjoys robust performance in most of the angular spectrum.

Weiss and Zoubir use an upper bound of the mean squared residuals to jointly mitigate the error contributions from noise and sensing matrix mismatch [12]. Compressed sensing can be formulated without a discrete sensing matrix, gridless compressed sensing [13], [15], [16], [17]. In this approach an atomic norm is used in the primal domain with a continuous primal variable. The support of the primal variable found in the dual domain. While more complicated, their approach have many similarities with our dual solution where we use a discrete sensing matrix and ℓ_1 norm.

Similar to [18], the generalized LASSO is formulated for complex-valued observations acquired from a sensor array. It is shown here that the corresponding dual vector is interpretable as the output of a weighted conventional beamformer (CBF) acting on the residuals of the linear observation model, cf. [19]. The maximum magnitudes of the dual vector is used for selecting the regularization parameter of the generalized LASSO. This is the basis for an efficient order-recursive procedure to solve the sparse signal reconstruction problem [20], [21], [22] at the desired sparsity level.

Further, we discuss and interpret the solutions to the generalized LASSO problem in the primal and dual domain and show its relation to the ℓ_0-constrained solution.

A. Notation

Matrices $\boldsymbol{A}, \boldsymbol{B}, \ldots$ and vectors $\boldsymbol{a}, \boldsymbol{b}, \ldots$ are complex-valued and denoted by boldface letters. The zero vector is $\boldsymbol{0}$. The Hermitian transpose, inverse, and Moore-Penrose pseudo inverse are denoted as $\boldsymbol{X}^H, \boldsymbol{X}^{-1}, \boldsymbol{X}^+$ respectively. We abbreviate $\boldsymbol{X}^{-H} = (\boldsymbol{X}^H)^{-1}$. The complex vector space of dimension N is written as $\mathbb{C}^N$. $\mathcal{N}$ is the null space of $\boldsymbol{A}$ and $\mathrm{span}(\boldsymbol{A})$ denotes the linear hull of $\boldsymbol{A}$. The projection onto $\mathrm{span}(\boldsymbol{A})$ is $\boldsymbol{P}_{\boldsymbol{A}}$. The ℓ_p-norm is written as $\|\cdot\|_p$.

II. Problem formulation

We start from the following problem formulation: Let $\boldsymbol{y} \in \mathbb{C}^N$ and $\boldsymbol{A} \in \mathbb{C}^{N\times M}$. Find the best sparse solution $\boldsymbol{x}_{\ell_0} \in \mathbb{C}^M$ with sparsity level $s \in \mathbb{N}$ such that the squared measurement residuals are minimal,

$$\boldsymbol{x}_{\ell_0} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad \|\boldsymbol{x}\|_0 \le s \ , \quad \text{(P0)}$$

where $\|\cdot\|_p$ denotes the ℓ_p-norm. The problem (P0) is known as ℓ_0-reconstruction. It is *non-convex* and hard to solve, cf. [23]. Therefore, the ℓ_0-constraint in (P0) is commonly relaxed to an ℓ_1 constraint which renders the problem (P1) to be *convex*. Further, a matrix $\boldsymbol{D}$ is introduced in the formulation of the constraint which gives flexibility in the problem definition. Several variants are discussed in [10]. This gives

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 \quad \text{subject to} \quad \|\boldsymbol{D}\boldsymbol{x}\|_1 \le \varepsilon \ . \quad \text{(P1)}$$

In the following, this problem is referred to as the complex-valued generalized LASSO problem. Incorporating the ℓ_1 norm constraint into the objective function results in the equivalent formulation (P1′),

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \left(\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{D}\boldsymbol{x}\|_1 \right) \ . \quad \text{(P1′)}$$

December 11, 2014

Christoph F. Mecklenbräuker and Erich Zöchmann are with Institute of Telecommunications, Vienna University of Technology 1040 Vienna, Austria, cfm@ieee.org

Peter Gerstoft is with University of California San Diego, La Jolla, CA 92093-0238, USA http://www.mpl.ucsd.edu/people/pgerstoft

The problems (P0) and (P1′) yield the same sparsity level $s = \|\boldsymbol{x}\|_0$ of their solutions if the regularization parameter μ in (P1′) is suitably chosen.

Strong duality does not hold for ℓ_0-reconstruction (P0) and studying its dual problem does not seem a useful exercise as the duality gap is infinite. The real-valued generalized LASSO problem and its dual were analyzed in [10].

In this contribution, a similar analysis is carried out for the complex-valued problem (P1′) which has straightforward applications in array signal processing. Based on this analysis, we propose and analyze an efficient procedure for approximating the solution to (P0) for a given sparsity level s which is based on the dual to (P1′). To formalize this idea, we introduce the following (non-convex) problem

$$\boldsymbol{x}_{\ell_1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \left(\min_{\mu>0} \left(\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{D}\boldsymbol{x}\|_1 \right) \right) \quad \text{subject to} \quad \|\boldsymbol{x}\|_0 \leq s \ . \tag{P2}$$

for which we seek an efficient solver.

III. Dual problem to the generalized LASSO

The generalized LASSO problem [10] is written in constraint form, all vectors and matrices are assumed to be complex-valued. The following discussion is valid for arbitrary $N, M \in \mathbb{N}$: both the over-determined and the under-determined cases are included. Following [24], [25], a new vector $\boldsymbol{z} \in \mathbb{C}^M$ and a new equality constraint $\boldsymbol{z} = \boldsymbol{D}\boldsymbol{x}$ are introduced , to obtain the equivalent problem

$$\min_{\boldsymbol{x},\boldsymbol{z}} \left(\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{z}\|_1 \right) \quad \text{subject to} \quad \boldsymbol{z} = \boldsymbol{D}\boldsymbol{x} \ . \tag{1}$$

The complex-valued dual vector $\boldsymbol{u} = (u_1, \ldots, u_M)^T$ is introduced and associated with the new equality constraint. The corresponding Lagrangian is

$$\begin{aligned} \mathcal{L}(\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}) &= \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{z}\|_1 + \operatorname{Re}\left[\boldsymbol{u}^H(\boldsymbol{D}\boldsymbol{x}-\boldsymbol{z})\right] &(2)\\ &= \mathcal{L}_1(\boldsymbol{x},\boldsymbol{u}) + \mathcal{L}_2(\boldsymbol{z},\boldsymbol{u}). &(3)\end{aligned}$$

To derive the dual problem, the Lagrangian is minimized over $\boldsymbol{x}$ and $\boldsymbol{z}$. The terms involving $\boldsymbol{x}$ are

$$\mathcal{L}_1(\boldsymbol{x},\boldsymbol{u}) = \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 + \operatorname{Re}\left(\boldsymbol{u}^H\boldsymbol{D}\boldsymbol{x}\right). \tag{4}$$

The terms in (2) involving $\boldsymbol{z}$ are

$$\mathcal{L}_2(\boldsymbol{z},\boldsymbol{u}) = \mu\|\boldsymbol{z}\|_1 - \operatorname{Re}(\boldsymbol{u}^H\boldsymbol{z}) \ . \tag{5}$$

The value $\hat{\boldsymbol{x}}$ minimizing (4) is found by differentiation. This gives

$$\boldsymbol{D}^H\boldsymbol{u} = 2\boldsymbol{A}^H\left(\boldsymbol{y}-\boldsymbol{A}\hat{\boldsymbol{x}}\right) \tag{6}$$

and

$$\boldsymbol{A}^H\boldsymbol{A}\hat{\boldsymbol{x}} = \boldsymbol{A}^H\boldsymbol{y} - \frac{1}{2}\boldsymbol{D}^H\boldsymbol{u} \ . \tag{7}$$

If $\boldsymbol{D}^H\boldsymbol{u} \in \operatorname{span}(\boldsymbol{A}^H)$ the solution to (7) becomes,

$$\hat{\boldsymbol{x}} = \underbrace{\boldsymbol{A}^+\boldsymbol{y} + \boldsymbol{\xi}}_{\hat{\boldsymbol{x}}_{\mathrm{LS}}} - \frac{1}{2}(\boldsymbol{A}^H\boldsymbol{A})^+\boldsymbol{D}^H\boldsymbol{u} \ , \tag{8}$$

where $(\cdot)^+$ denotes the Moore-Penrose pseudo inverse of $\boldsymbol{X}$, i.e. $\boldsymbol{X}^+ = (\boldsymbol{X}^H\boldsymbol{X})^+\boldsymbol{X}^H$. Here, $\boldsymbol{\xi} \in \mathcal{N}(\boldsymbol{A})$ is a nullspace term which enables $\hat{\boldsymbol{x}}$ to deviate from the least norm solution $\boldsymbol{A}^+\boldsymbol{y}$. The nullspace $\mathcal{N}(\boldsymbol{A})$ is $\{\boldsymbol{\xi} \in \mathbb{C}^M | \boldsymbol{A}\boldsymbol{\xi} = \boldsymbol{0}\}$. By identifying $\boldsymbol{\xi} = \boldsymbol{x}_{\ell_1}^{\mathrm{null}}$, we specialize (8) to the solution of (P1′),

$$\boldsymbol{x}_{\ell_1} = \boldsymbol{A}^+\boldsymbol{y} + \boldsymbol{x}_{\ell_1}^{\mathrm{null}} - \tfrac{1}{2}(\boldsymbol{A}^H\boldsymbol{A})^+\boldsymbol{D}^H\boldsymbol{u}. \tag{9}$$

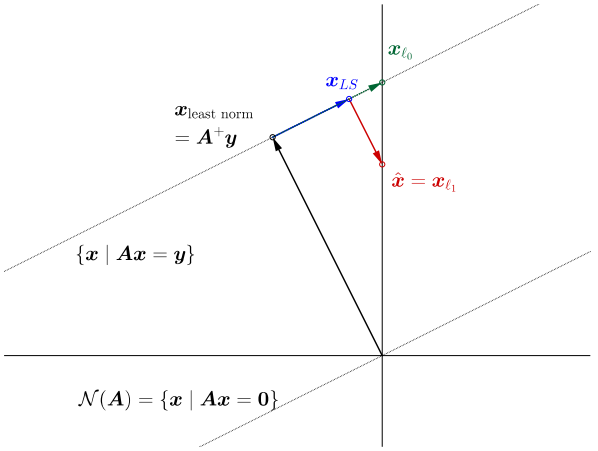


Fig. 1. Sketch of the relations between the primal solution and the terms in (9): least norm solution $\boldsymbol{x}_{\text{least norm}}$, least squares solution $\boldsymbol{x}_{\mathrm{LS}}$, and the sparse solutions $\boldsymbol{x}_{\ell_0}$, $\boldsymbol{x}_{\ell_1}$. The nullspace term $\boldsymbol{\xi}$ is any vector along the line perpendicular to $\boldsymbol{A}^+$. The red arrow represents the last term in (9) which is perpendicular to $\boldsymbol{\xi}$.

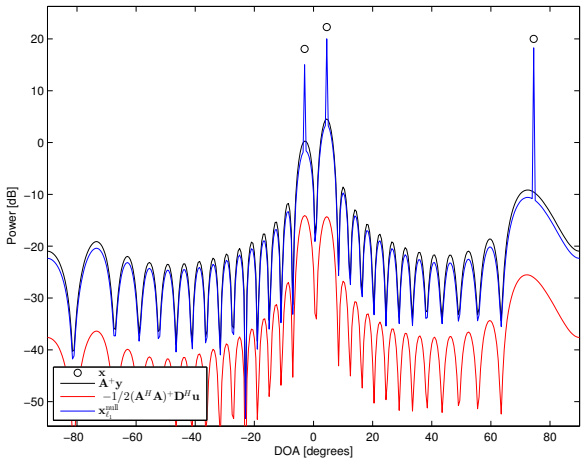


Fig. 2. Numerical example solution terms in Eq. (9) versus direction of arrival (DOA).

Thus, the solution to the generalized LASSO problem (8) consists of three terms. These are illustrated schematically in Fig. 1. The first two terms are the least norm solution $\boldsymbol{A}^+\boldsymbol{y}$ and the nullspace solution $\boldsymbol{\xi}$ which together form the unconstrained least squares (LS) solution $\hat{\boldsymbol{x}}_{\mathrm{LS}}$. The third term in (8) is associated with the dual solution. Fig. 2 shows the three terms of (9) individually for a simple array-processing scenario. The continuous angle θ is discretized uniformly in $[-90, 90]$ using 361 samples and the wavefield is observed by 30 sensors resulting in a complex-valued 30×361 $\boldsymbol{A}$ matrix (see section IV-A). At those primal coordinates m which correspond to directions of arrival at $-5°$, $10°$ and $150°$

in Fig. 2, the three terms sum up constructively resulting in a non-zero x_m ("the mth source position is active"), while for all other entries they interfere destructively. The case of constructive interference is illustrated in Fig. 1 which is in constrast to the case of destructive interference when the three terms in (8) sum up to zero. This is formulated rigorously in Corollary 1.

We evaluate (4) at the minimizing solution $\hat{\boldsymbol{x}}$ and express the result solely by the dual $\boldsymbol{u}$. Firstly, we expand

$$\|\boldsymbol{y}-\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 = \|\boldsymbol{y}\|_2^2 + \|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 - 2\,\mathrm{Re}\{\boldsymbol{y}^H\boldsymbol{A}\hat{\boldsymbol{x}}\} \tag{10}$$

Secondly using (6),

$$\begin{aligned}(\boldsymbol{D}^H\boldsymbol{u})^H\hat{\boldsymbol{x}} &= 2(\boldsymbol{y}-\boldsymbol{A}\hat{\boldsymbol{x}})^H\boldsymbol{A}\hat{\boldsymbol{x}} \\ &= 2\boldsymbol{y}^H\boldsymbol{A}\hat{\boldsymbol{x}} - 2\|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 \end{aligned} \tag{11}$$

Eq.(10) and the real part of (11) are summed. This results in

$$\begin{aligned}\mathcal{L}_1(\hat{\boldsymbol{x}},\boldsymbol{u}) &= \|\boldsymbol{y}\|_2^2 - \|\boldsymbol{A}\hat{\boldsymbol{x}}\|_2^2 \\ &= \boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2\ , \end{aligned} \tag{12}$$

where we used (8) and introduced the abbreviations

$$\tilde{\boldsymbol{D}} = \tfrac{1}{2}\boldsymbol{D}\boldsymbol{A}^+, \tag{13}$$

$$\boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{P}_{\boldsymbol{A}}, \tag{14}$$

$$\tilde{\boldsymbol{y}} = \boldsymbol{P}_{\boldsymbol{A}}\,\boldsymbol{y}\ . \tag{15}$$

Formally, the subspace constraint $\boldsymbol{D}^H\boldsymbol{u} \in \mathrm{span}(\boldsymbol{A}^H)$ is formulated as $(\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u} = \boldsymbol{0}$ where $\boldsymbol{U}$ is a unitary basis of the null space $\mathcal{N}$. This results in

$$\inf_{\boldsymbol{x}}\mathcal{L}_1(\boldsymbol{x},\boldsymbol{u}) = \begin{cases} \boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2\ , & \text{if } (\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u} = \boldsymbol{0}, \\ -\infty, & \text{otherwise.}\end{cases} \tag{16}$$

Next (5) is minimized with respect to $\boldsymbol{z}$, see Appendix A,

$$\inf_{\boldsymbol{z}}\mathcal{L}_2(\boldsymbol{z},\boldsymbol{u}) = \begin{cases} 0, & \text{if } \|\boldsymbol{u}\|_\infty \le \mu \\ -\infty, & \text{otherwise.}\end{cases} \tag{17}$$

Combining (16) and (17), the dual problem to the generalized LASSO (1) is,

$$\max_{\boldsymbol{u}\in\mathbb{C}^M} \boldsymbol{y}^H\boldsymbol{y} - \|\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{D}}^H\boldsymbol{u}\|_2^2 \tag{18a}$$

$$\text{subject to} \quad \|\boldsymbol{u}\|_\infty \le \mu, \tag{18b}$$

$$(\boldsymbol{D}\boldsymbol{U})^H\boldsymbol{u} = \boldsymbol{0}. \tag{18c}$$

Equation (6) can be solved for $\boldsymbol{u}$ if the row space constraint (18c) is fulfilled. The result is summarized in the following

Theorem 1. *If $\boldsymbol{D}$ is non-singular, the dual vector $\boldsymbol{u}$ is the output of a weighted CBF acting on the vector of residuals, i.e.*

$$\boldsymbol{u} = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}_{\ell_1})\ , \tag{19}$$

where $\boldsymbol{x}_{\ell_1}$ is such that the box constraint (18b) is fulfilled.

The dual vector $\boldsymbol{u}$ gives an indication of the sensitivity of the primal solution to small changes in the constraints of the primal problem [24]. In [10], it was worked out for the real-valued case that the solution to (P1′) can be more easily constructed and better understood via the dual problem. Theorem 1 asserts a linear one-to-one relation between the corresponding dual and primal solution vectors also in the complex-valued case. Thus, any results formulated in the primal domain are readily applicable in the dual domain. This allows a more fundamental interpretation of sequential Bayesian approaches to density evolution for sparse source reconstruction [20], [21]: they can be rewritten in a form that shows that they solve a generalized complex-valued LASSO problem and its dual. The posterior probability density is actually related to the dual solution.

The following corollaries clarify useful element-wise relations between the primal and dual solutions: Corollary 1 relates the *magnitudes* of the corresponding primal and dual coordinates. Further, Corollary 2 certifies what conditions on $\boldsymbol{D}$ are sufficient for guaranteeing that the *phase angles* of the corresponding primal and dual coordinates are equal. Finally, Corollary 3 states that *both* the primal and the dual solutions to (P1′) are piecewise linear in the regularization parameter μ.

Corollary 1. *If the mth primal coordinate is active, i.e. $x_{\ell_1,m} \neq 0$ then the box constraint (18b) is tight in the mth dual coordinate. Formally,*

$$x_{\ell_1,m} \neq 0 \quad \Rightarrow \quad |u_m| = \mu, \qquad (m = 1,\ldots,M). \tag{20}$$

Informally, we say that the mth dual coordinate hits the boundary when the mth primal coordinate becomes active. Conversely, when the bound on $|u_m|$ is loose (i.e. the constraint on u_m is inactive*), the corresponding primal variable x_m is zero (the mth primal coordinate is* inactive*). The proof is given in Appendix B. The active set $\mathcal{M}$ is*

$$\mathcal{M} \;=\; \big\{m\,\big|\,x_{\ell_1,m}\neq 0\big\} \;\subseteq\; \big\{m\,\big|\,|u_m| = \mu\big\} \;=\; \mathcal{U}\ . \tag{21}$$

The active set $\mathcal{M}$ implicitly depends on the choice of μ in problem (P1′). Let $\mathcal{M}$ contain exactly s indices,

$$\mathcal{M} = \{m_1,\, m_2, \ldots,\, m_s\}. \tag{22}$$

Corollary 2. *If matrix $\boldsymbol{D}$ is diagonal with real-valued positive diagonal entries, then the phase angles of the corresponding entries of the dual and primal solution vectors are equal.*

$$\arg(u_m) = \arg(x_{\ell_1,m}), \quad \forall m \in \mathcal{M} \tag{23}$$

Corollary 3. *The primal and the dual solutions to the complex-valued generalized LASSO problem (P1′) are continuous and piecewise linear in the regularization parameter $\mu > 0$. The changes in slope occur at those values for μ where the set of active indices $\mathcal{M}$ changes.*

The proofs for these corollaries are given in Appendix B.

A. Relation to the ℓ_0 solution

It is now assumed that $\mathcal{M}$ defines the indices of the s non-zero elements of the corresponding ℓ_0 solution. In other words: the ℓ_1 and ℓ_0 solutions share the same sparsity pattern. The ℓ_0 solution with sparsity s is then obtained by regressing the s *active* columns of $\boldsymbol{A}$ to the data $\boldsymbol{y}$ in the least-squares sense. Let

$$\boldsymbol{A}_{\mathcal{M}} = [\boldsymbol{a}_{m_1},\, \boldsymbol{a}_{m_2},\, \ldots,\, \boldsymbol{a}_{m_s}]\ , \tag{24}$$

where $\boldsymbol{a}_m$ denotes the mth column of $\boldsymbol{A}$. The ℓ_0 solution becomes (cf. Appendix C)

$$\boldsymbol{x}_{\ell_0} = \boldsymbol{A}_{\mathcal{M}}^{+}\boldsymbol{y} \ . \tag{25}$$

Here, $\boldsymbol{A}_{\mathcal{M}}^{+} = (\boldsymbol{A}_{\mathcal{M}}^{H}\boldsymbol{A}_{\mathcal{M}})^{-1}\boldsymbol{A}_{\mathcal{M}}^{H}$ is the left inverse of $\boldsymbol{A}_{\mathcal{M}}$. By subtracting (9) from (25) and restricting the equations to the contracted basis $\boldsymbol{A}_{\mathcal{M}}$ yields

$$\begin{aligned} \boldsymbol{A}_{\mathcal{M}}(\boldsymbol{x}_{\ell_0} - \boldsymbol{x}_{\ell_1}) &= \tfrac{1}{2}\boldsymbol{A}_{\mathcal{M}}\left(\boldsymbol{A}_{\mathcal{M}}^{H}\boldsymbol{A}_{\mathcal{M}}\right)^{+}\boldsymbol{D}_{\mathcal{M}}^{H}\boldsymbol{u}_{\mathcal{M}} & (26)\\ &= \tilde{\boldsymbol{D}}_{\mathcal{M}}^{H}\mu e^{j\boldsymbol{\theta}} \ . & (27)\end{aligned}$$

In the image of $\boldsymbol{A}$, the ℓ_0-reconstruction problem (P0) and the generalized LASSO (P1′) coincide if the LASSO problem is pre-informed (prior knowledge) by setting $\boldsymbol{D}_{mm},\ m \in \mathcal{M}$ to zero. The prior knowledge is obtainable by an iterative re-weighting process [26] or by a sequential procedure on stationary sources [21].

IV. Direction of Arrival Estimation

For the numerical examples, we model a uniform linear array (ULA), which is described with its steering vectors representing the incident wave for each array element.

A. Array Data Model

Let $\boldsymbol{x} = (x_1, \ldots, x_M)^{\mathrm{T}}$ be a vector of complex-valued source amplitudes. We observe time-sampled waveforms on an array of N sensors which are stacked in the vector $\boldsymbol{y}$. The following linear model for the narrowband sensor array data $\boldsymbol{y}$ at frequency ω is assumed,

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n} \ . \tag{28}$$

The mth column of the transfer matrix $\boldsymbol{A}$ is the array steering vector $\boldsymbol{a}_m$ for hypothetical waves from direction of arrival (DOA) θ_m. To simplify the analysis all columns are normalized such that their ℓ_2 norm is one. The transfer matrix $\boldsymbol{A}$ is constructed by sampling all possible directions or arrival, but only very few of these correspond to real sources. Therefore, the dimension of $\boldsymbol{A}$ is $N \times M$ with $N \ll M$ and $\boldsymbol{x}$ is sparse. In our setting, the number of hypothetical source locations M is much larger than the number of sensors N, i.e. $N \ll M$. The linear model equations (28) are underdetermined.

The nmth element of $\boldsymbol{A}$ is modeled by

$$A_{nm} = \frac{1}{\sqrt{N}} \exp\left[\mathrm{j}(n-1)\pi \sin\theta_m\right]. \tag{29}$$

Here $\theta_m = \frac{\pi(m-1)}{M} - \pi/2$ is the DOA of the mth hypothetical DOA to the nth array element.

The additive noise vector $\boldsymbol{n}$ is assumed spatially uncorrelated and follows a zero-mean complex normal distribution with diagonal covariance matrix $\sigma^2\boldsymbol{I}$.

For the observation $\boldsymbol{y}$ according to the linear model (28), the conditional probability density given the source vector $\boldsymbol{x}$ is

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2\right)}{(\pi\sigma^2)^N} \ . \tag{30}$$

For the source vector $\boldsymbol{x}$, a prior probability density is assumed in form of a multivariate complex Laplace-like density [27],

$$p(\boldsymbol{x}) = \prod_{m=1}^{M}\left(\frac{\lambda_m}{\sqrt{2\pi}}\right)^2 \mathrm{e}^{-\lambda_m|x_m|} \ , \tag{31}$$

with associated hyperparameters $\lambda_m > 0$ modeling the source signal strength at location θ_m. $x_m = |x_m|\mathrm{e}^{j\phi_m}$ is the complex source signal at hypothetical source location θ_m. Note that (31) defines a joint distribution for $|x_m|$ and ϕ_m for all $m = 1, \ldots, M$. Taking the logarithm gives

$$-\ln p(\boldsymbol{x}) = \sum_{m=1}^{M}\lambda_m|x_m| - 2\sum_{m=1}^{M}\ln\lambda_m + M\ln 2\pi \ . \tag{32}$$

For the posterior probability density function (pdf) $p(\boldsymbol{x}|\boldsymbol{y})$, Bayes' rule is used for obtaining the generalized LASSO Lagrangian [10], [21]

$$\frac{1}{\sigma^2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\,\|\boldsymbol{W}\boldsymbol{x}\|_1 \tag{33}$$

with

$$\boldsymbol{W} = \frac{1}{\mu}\,\mathrm{diag}(\boldsymbol{\lambda}) = \mathrm{diag}(\boldsymbol{w}). \tag{34}$$

Minimizing the generalized LASSO Lagrangian (35) with respect to $\boldsymbol{x}$ for given μ, and $\boldsymbol{w} = (w_1, \ldots, w_M)^T$, $\boldsymbol{\lambda} = \mu\boldsymbol{w}$, gives a sparse MAP source estimate $\boldsymbol{x}_{\ell_1}$. This minimization problem promotes sparse solutions in which the ℓ_1 constraint is weighted by giving every source amplitude its own hyperparameter w_m.

Equivalently to (33), this is reformulated as

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\,\|\boldsymbol{D}\boldsymbol{x}\|_1 \ , \tag{35}$$

with

$$\boldsymbol{D} = \sigma^2\boldsymbol{W} \ . \tag{36}$$

The minimization of (35) constitutes a strictly convex optimization problem.

B. Basis coherence

The chosen examples feature different levels of basis coherence in order to examine the solution's behavior. As described in [14], the *basis coherence* is a measure of correlation between two steering vectors and defined as the inner product between atoms, i.e. the columns of $\boldsymbol{A}$. The maximum of these inner products is called mutual coherence and is customarily used for performance guarantees of recovery algorithms. To state the difference formally:

$$\mathrm{coh}\,(\boldsymbol{a_i}, \boldsymbol{a_j}) = \boldsymbol{a_i}^H\boldsymbol{a_j} \tag{37}$$

$$\text{mutual coh}(\boldsymbol{A}) = \left\|\boldsymbol{A}^H\boldsymbol{A} - \boldsymbol{I}\right\|_\infty \tag{38}$$

The mutual coherence is bounded between 0 and 1 and [14] has shown that the coherence defines a region of possible offsets around the true DOA θ_m associated with steering vector $\boldsymbol{a}_m$. The region of possible offsets is marked in Fig. 7 as circles ("○").

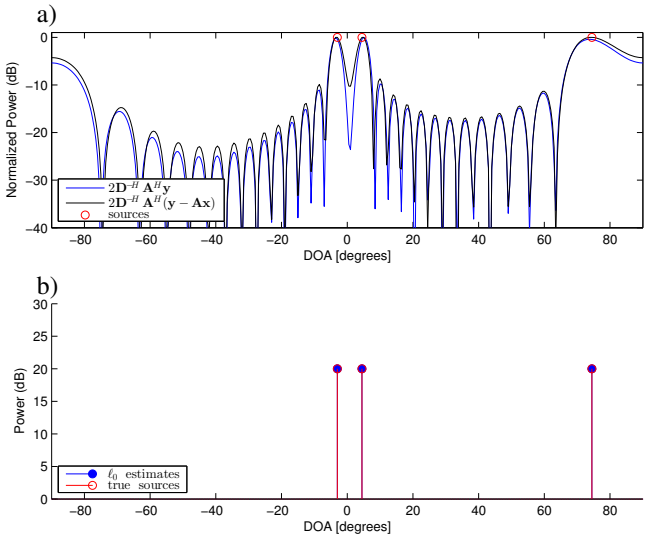


Fig. 3. Dual (a) and primal (b) coordinates for 3 well separated sources with low basis coherence.

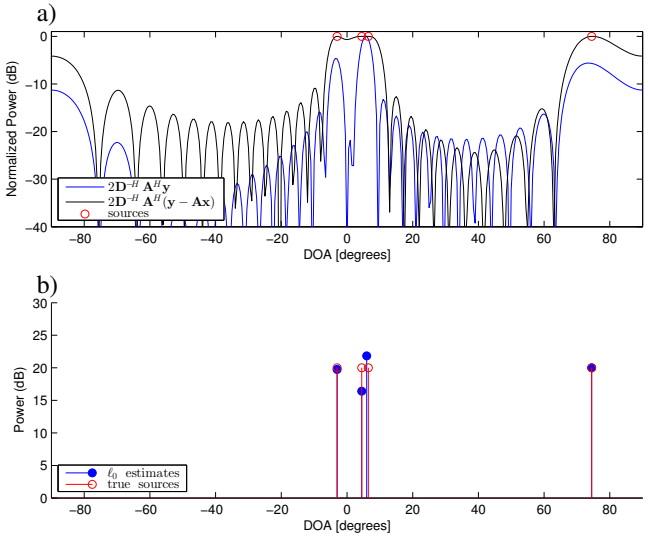


Fig. 4. Dual (a) and primal (b) coordinates for 4 sources with higher basis coherence.

C. Numerical Example

The following numerical example in Figs. 3 and 4 demonstrates the dual solution and the classical CBF output for $N = 30$ and $M = 361$. The example is noiseless and the diagonal matrix $\boldsymbol{D} = \boldsymbol{I}$ is non-informative in the Bayesian sense. In Fig. 3, the LASSO with $\mu = 1$ is solved for a scenario with three sources at DOA $-5°, 10°, 150°$ and all sources have same power level (see Fig. 3b), whereas in Fig. 4, an additional fourth source at $14°$ is included into the scenario.

1) CBF resolvable regime: In Fig. 3, we investigate the performance when the steering vectors of the active sources have small basis coherence. The basis of source 1 is weakly coherent with source 2, $\mathrm{coh} \approx 0.02$ using (37).

Figure 3a shows the normalized beampattern of the CBF (blue) and the pattern of the dual vector (black). This figure shows that the true source parameters (DOA and power) are well estimated.

2) Beyond the CBF capabilities: Figure 4a shows that the sources are not separable with the CBF, because the steering vectors belonging to source 2 and 3 are coherent, $\mathrm{coh} = 0.61$ using (37). The (generalized) LASSO approach is still capable of resolving all 4 sources. Figure 4b shows that the true source locations (DOA) are still well estimated, in constrast to the source powers.

V. Solution Path

The problems (P0)–(P2) are complex-valued and the corresponding solution paths behave differently from what is described in Ref. [10]. In the following figures, only the magnitudes of the primal and dual solution coordinates are illustrated. Note that Corollary 1 guarantees that the phases of the active primary solution elements and their duals are identical and independent from μ.

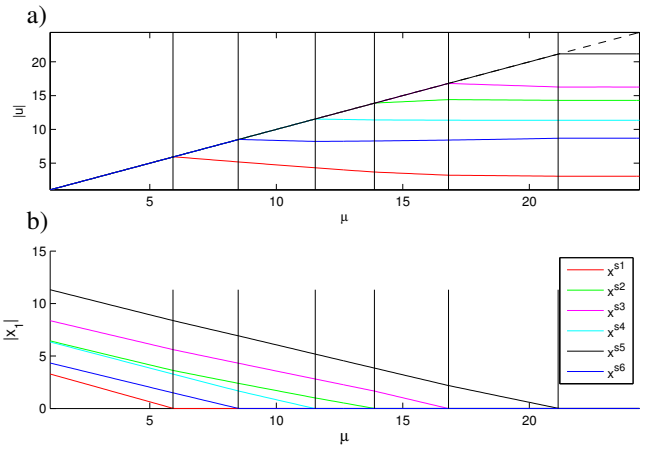


Fig. 5. Magnitudes of the solution paths versus μ for the simulation parameters in Table I and $\mathrm{SNR} = 40\,\mathrm{dB}$: (a) dual, and (b) primal vectors for the case of the *complete basis*.

A. Complete Basis

First (Fig. 5) discusses the dual and primal solution for a complete basis with $M = 6$, sparsity level $s = 6$, and $N = 30$ sensors linearly spaced with half wavelength spacing. This simulation scenario is not sparse and all steering vectors $\boldsymbol{a}_m$ for $1 \le m \le M$ will eventually be used to reconstruct the data for small μ. The source parameters that are used in the simulation scenario are given in Table I. The signal to noise ratio is defined as

$$\mathrm{SNR} = 10\log_{10}\left(\mathrm{E}\|\boldsymbol{A}\boldsymbol{x}\|_2^2 \,/\, \mathrm{E}\|\boldsymbol{n}\|_2^2\right)\,\mathrm{dB}, \qquad (39)$$

and chosen to be $\mathrm{SNR} = 40\,\mathrm{dB}$. The diagonal matrix $\boldsymbol{D} = \boldsymbol{I}$ is again chosen to be non informative.

We discuss the solution paths in Figs. 5–9 from right ($\mu = \infty$) to left ($\mu = 0$). Initially all dual solution paths are horizontal (slope = 0), since the primal solution $\boldsymbol{x}_{\ell_1} = \boldsymbol{0}$ for

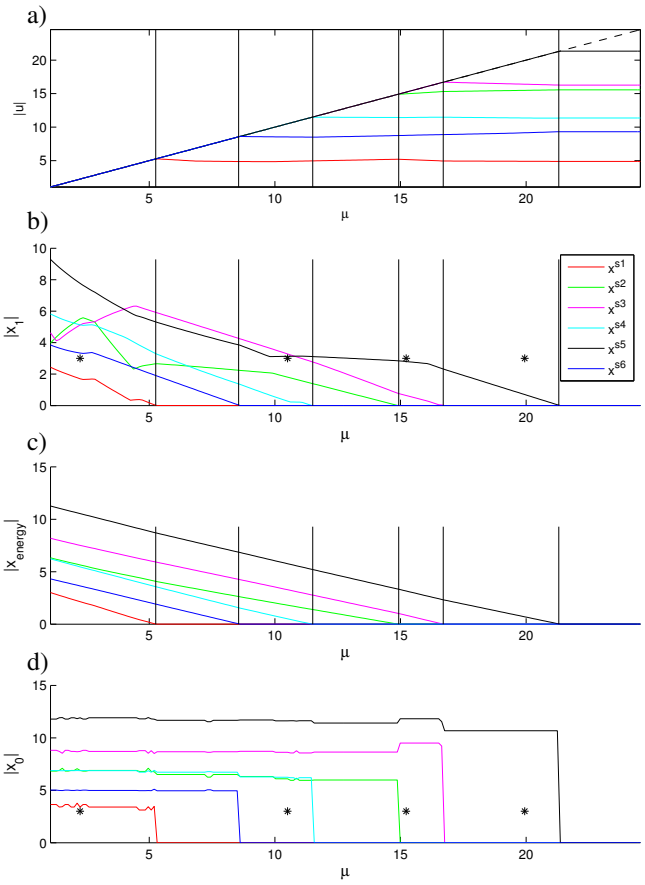


Fig. 6. Magnitudes of the solution paths versus μ for the simulation parameters in Table I and SNR $= 40$ dB: (a) dual, and (b, c and d) primal vectors for the case of an 80-vector *overcomplete basis*. For the primal coordinates the peak within ± 2 bins from the true bin is tracked based on (b) maximum (c) energy. The magnitudes of the corresponding elements of $\boldsymbol{x}_{\ell_0}$ are shown in (d).

No.	DOA (°)	Power (lin.)
1	-6.0	4.0
2	-1.0	7.0
3	4.0	9.0
4	9.0	7.0
5	14.0	12.0
6	19.0	5.0

TABLE I
SOURCE PARAMETERS FOR SIMULATION SCENARIO

$\mu > 2\|\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}\|_\infty$. In this strongly penalized regime, the dual vector is the output of the weighted CBF $\boldsymbol{u} = \boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$ which does not depend on μ.

At the point $\mu^1 = 2\|\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}\|_\infty$ the first dual coordinate hits the boundary (18b). This occurs at $\mu^1 = 21$ in Fig. 5a and the corresponding primal coordinate becomes active. As long the active set $\mathcal{M}$ does not change, the magnitude of the corresponding dual coordinate is μ, due to Corollary 1. The remaining dual coordinates change slope relative to the basis coherence level of the active set.

As μ decreases, the source magnitudes at the primal active indices increase since the ℓ_1-constraint in (P1′) becomes less

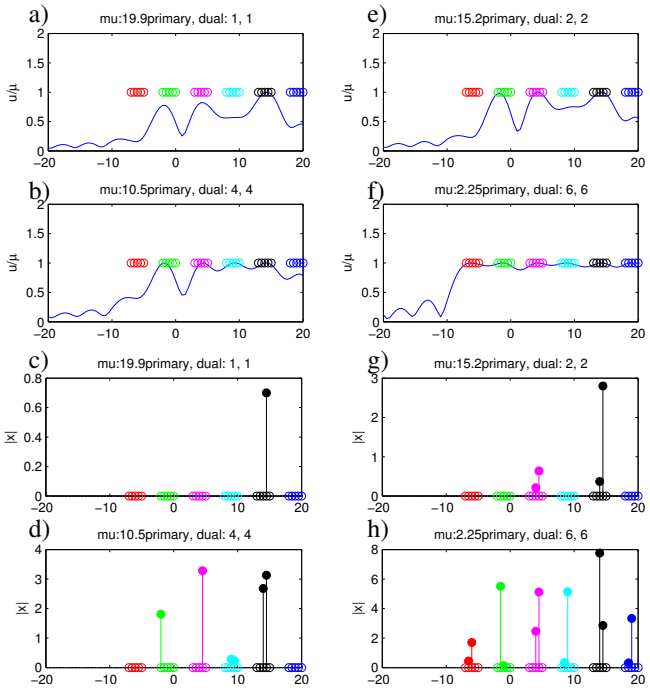


Fig. 7. Dual and primal coordinates at selected values of μ for 81-vector overcomplete basis for SNR $= 40$ dB.

important, see Fig. 5b. The second source will become active when the next dual coordinate hits the boundary (at $\mu^1 = 17$ in Fig. 5).

When the active set is constant, the primary and dual solution is piecewise linearly with μ, as proved in Corollary 3. The changes in slope are quite gentle, as shown for the example in Fig. 5 . Finally, at $\mu = 0$ the problem (P1′) degenerates to an unconstrained (underdetermined) least squares problem. Its primal solution $\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}}_{\mathrm{LS}}$, see (8), is not unique and the dual vector is trivial, $\boldsymbol{u} = \boldsymbol{0}$.

B. Overcomplete Basis

We now enlarge the basis to $M = 81$ with hypothetical source locations $\theta_m \in [-20°, 20°]$ with $0.5°$ spacing, and all other parameters as before. The solution is now sparse.

The LASSO path [28] is illustrated in Fig. 6 where we expect the source location estimate to be up to ± 2 bins from the true source location. The dual Fig. 6a appears to be quite similar to Fig. 5a.

Corollary 3 gives that the primary solution should change linearly, as demonstrated for the complete basis in Fig. 5)b. Here we explain why this is not the case for the overcomplete basis primary solution in Fig. 6b . This can be understood be examining the full solution at selected values of μ (stars (*) in Fig. 6). At $\mu = 20$ just one solution is active, only the black source (source 5) is active though one bin to the left, as shown in Fig. 7. Between $\mu = 16$ and $\mu = 11$, the black source appears constant, this is because at large values the source is initially located in a neighboring bin. As μ decreases, the correct bin receive more power, see Fig7 for $\mu = 15$ and $\mu = 10$. When it is stronger than the neighbor bin at ($\mu = 11$),

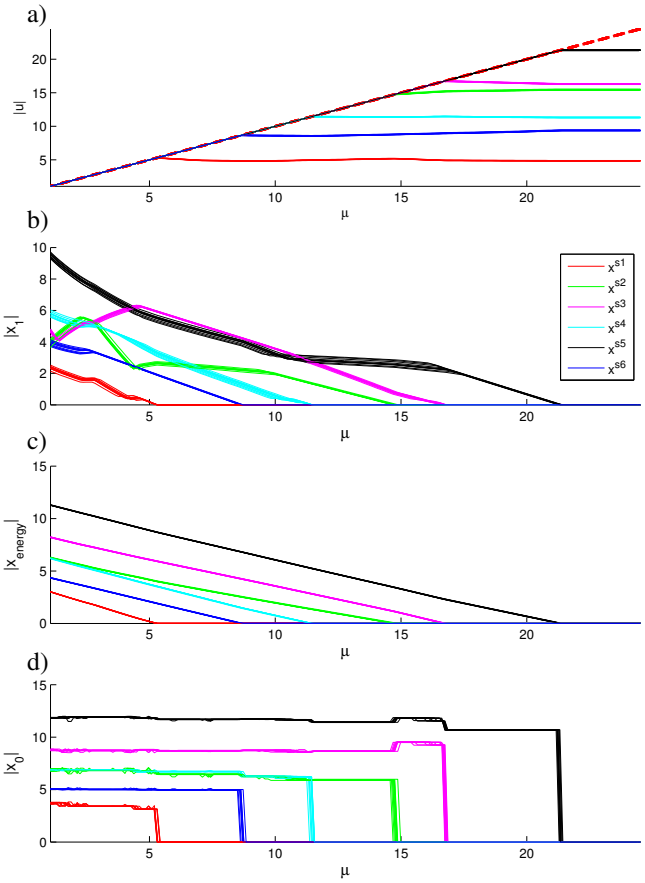


Fig. 8. For 10 noise realizations, magnitudes of the solution paths versus μ for the simulation parameters in Table I and $\mathrm{SNR} = 40\,\mathrm{dB}$: (a) dual, and (b, c and d) primal vectors for the case of an 80-vector *overcomplete basis*. For the primal coordinates the peak within ± 2 bins from the true bin is tracked based on (b) maximum (c) energy. The magnitudes of the corresponding elements of $\boldsymbol{x}_{\ell_0}$ are shown in (d).

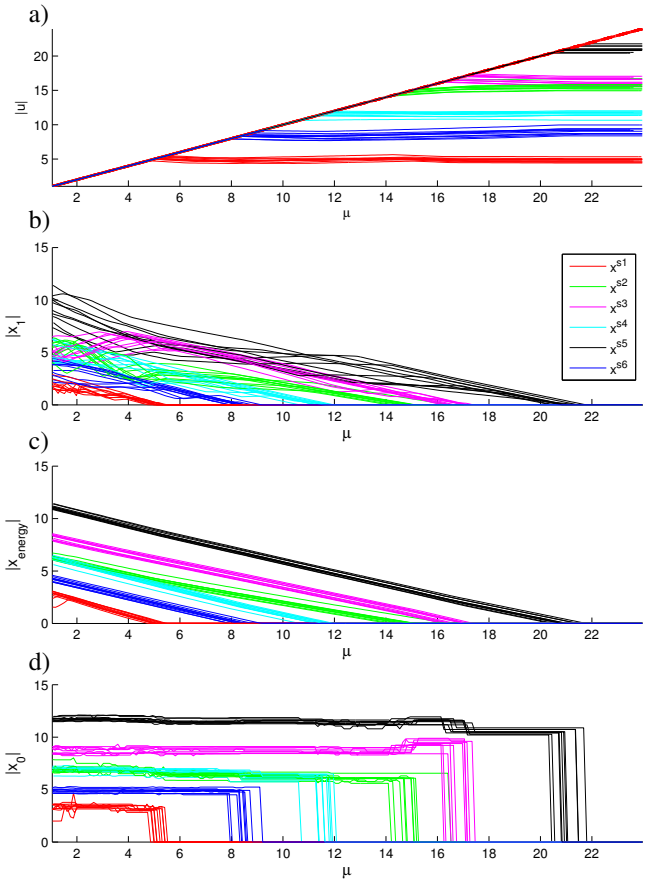


Fig. 9. As Fig. 8, but with $\mathrm{SNR} = 20\,\mathrm{dB}$:

this source power will start increase again. This trading in source power causes the fluctuations in Fig. 6b.

One way to correct for this fluctuation is to sum the coherent energy for all bins near a source, i.e., multiplying the source vector with the corresponding neighbor columns of $\boldsymbol{A}$ and then compute the energy based on the average received power at each sensor. This gives a steady rise in power as shown in Fig. 6c.

We motivate solving (P1′) as a substitute for ℓ_0-reconstruction (P0)—finding the active indexes of the ℓ_1 solution, see Fig. 6d. The ℓ_0 primal can be found with the restricted basis and the value of the ℓ_1 primal from (8), which depends on μ, or by just solving (25).

To investigate the sensitivity to noise, 10 LASSO paths are simulated for 10 noise realizations for both $\mathrm{SNR} = 40\,\mathrm{dB}$ (Fig. 8) and $\mathrm{SNR} = 20\,\mathrm{dB}$ (Fig 9). The dual (Figs. 8a and 9a), appears quite stable to noise, but the primal $|\mathbf{x}_{\ell 1}|$ (Figs. 8b and 9b) show quite large variation with noise. This is because the noise causes the active indexes to shift and thus the magnitude to vary. The mapping to energy $|\mathbf{x}_{\mathrm{energy}}|$ (Figs. 8c and 9c) or the $|\mathbf{x}_{\ell 0}|$ solution (Figs. 8d and 9d) makes the solution much more stable.

VI. Solution Procedures

Motivated by Theorem 1 and Corollary 1, we propose the order-recursive procedure in Table II for approximating the solution to problem (P2), a faster iterative procedure in Table III, and a dual-based iterative procedure in Table IV.

As shown by Theorem 1, the dual vector is actually a CBF acting on the LASSO residuals which will hit the boundary ($|u_m| = \mu$) if the corresponding primal coordinate x_m is active. This insight leads to an estimator for the regularization parameter μ for a given sparsity order.

The LASSO path [28] is illustrated in Fig. 10. Starting from a large choice of regularization parameter μ and then decreasing, we observe incremental changes in the active set at specific values μ^{*p} of the regularization parameter. The active set remains constant within the interval $\mu^{*p} > \mu > \mu^{*p+1}$. Assuming the sequence μ^{*1}, μ^{*2}, ... known, we follow a path of regularization parameters μ^1, μ^2, ... where μ^p is slightly higher than the lower end μ^{*p+1} of the regularization interval. Specifically, $\mu^p = (1-F)\mu^{*p} + F\mu^{*p+1}$ with $F < 1$. For the numerical examples $F = 0.9$ is used. This F is chosen because the primal solution $\boldsymbol{x}_{\ell_1}$ is closest to $\boldsymbol{x}_{\ell_0}$) at the lower end of the interval.

The order recursive procedure in Table II finds one source at a time as μ is lowered, the iterative procedure in Table III iterates on the finding the number of active sources, the dual iterative procedure in Table IV is entirely in the dual domain.

In the following we focus on the1 order recursive procedure, and indicate the differences to the other approaches.

A. Primary-based

The procedure starts with the all-zero solution $\boldsymbol{x}_{\ell_1}^0 = \boldsymbol{0}$ in line 1. This corresponds to a large value of $\mu = \mu^0 > 2\|\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}\|_\infty$, but is not used directly. The first value of μ, μ^1 is chosen based on the first peak in $\boldsymbol{u}$ for $\boldsymbol{x}_{\ell_1} = \boldsymbol{0}$. The ℓ_∞-norm is implemented by calling the $\mathrm{peak}(\boldsymbol{u}, p)$–function for $p = 1$ in line 3. For this purpose, we define the $\mathrm{peak}(\boldsymbol{u}, p)$–function which returns the pth largest local peak in magnitude of the vector $\boldsymbol{u}$. A local peak is defined as an element which is larger than its adjacent elements.

In lines 4–5 in Table II, the generalized LASSO problem (P1′) is solved for $\mu = \mu^1$ and the corresponding active set is detected by thresholding (at this point, the active set contains only a single active index). Then, by assuming that the active set is also valid for the solution to the ℓ_0-problem (P0), the solution $\boldsymbol{x}_{\ell_0}^p$ is computed. Next, we start an iteration loop with counter i which reduces the regularization parameter μ by inserting the updated dual variable $\boldsymbol{u}_1$ into line 3.

The next choice of regularization parameter μ^2 is selected in the interval $[\mu^{*2}, \mu^{*3}[$ which we estimate by the 2^{nd} and 3^{rd} largest peaks of the dual variable. This is illustrated in Fig. 10. This is implemented in line 3. Then, we solve (P1′) for $\mu = \mu^2 < \mu^1$ and continue the iteration until the desired sparsity level s is reached.

In line 5, the active set $\mathcal{M}$ is approximated by thresholding of the primal solution.

If the number of elements in $\mathcal{M}$ equals the loop counter i then the basis is restricted to $\mathcal{M}$ and the corresponding ℓ_0-solution is computed in lines 9–13. Otherwise, at least one peak of the dual vector is a sidelobe artifact. Then an *additional* peak in $\boldsymbol{u}$ must be included for estimating the next μ. This is implemented by incrementing the loop counter i in line 8, which keeps track of the number of peaks in $\boldsymbol{u}$.

The order-recursive procedure in Table II employs an approximation of the height of the ith local peak given the $(i-1)$th solution. Theorem 1 together with Corollary 1 gives $\mu^i = \mathrm{peak}(\boldsymbol{u}^i, i)$. The underlying assumption is that the next source will become active at the location corresponding to the dual coordinate of the next peak. This assumption is not universally valid as discussed below (44).

From the box constraint (18b), it is concluded that the level of the ith peak in $\boldsymbol{u}$ does not change much during the iteration over i: It is bounded by the difference in regularization parameter. For $\mu' < \mu''$,

$$\mathrm{peak}(\boldsymbol{u}(\mu''), i) - \mathrm{peak}(\boldsymbol{u}(\mu'), i) \leq \mu'' - \mu'. \tag{40}$$

This allows to approximate

$$\mu^i \quad = \quad \mathrm{peak}(\boldsymbol{u}^i, i) \approx \mathrm{peak}(\boldsymbol{u}^{i-1}, i)\ . \tag{41}$$

This approximation is not limited to a single iteration. Therefore, (41) can be extended further to

$$\mu^i \approx \mathrm{peak}(\boldsymbol{u}^{i-1}, i) \approx \mathrm{peak}(\boldsymbol{u}^{i-2}, i) \approx \cdots \approx \mathrm{peak}(\boldsymbol{u}^0, i)\ . \tag{42}$$

This observation motivates the iterative procedure in Table III.

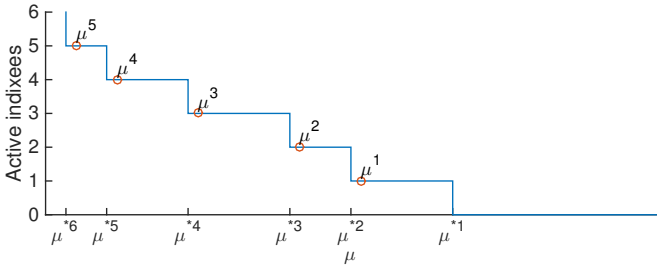


Fig. 10. Illustration of the LASSO path: Number of active indices versus the regularization parameter μ. Increments in the active set occur at μ^{*p}

For $i = 0$ there is no previous basis, and the relation (42) is exact. If $(\boldsymbol{A}^H\boldsymbol{A})$ were a diagonal matrix, the relation would also be exact. This follows from

$$\begin{aligned} \boldsymbol{u}^i \quad &= \quad 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \boldsymbol{A}\,\boldsymbol{x}_{\ell_1}^i\right) \\ &= \quad 2\boldsymbol{D}^{-H}(\boldsymbol{A}^H\boldsymbol{y} - (\boldsymbol{A}^H\boldsymbol{A})\,\boldsymbol{x}_{\ell_1}^i). \end{aligned} \tag{43}$$

A diagonal coherence matrix implies the mutual coherence is zero, which is not possible for $M > N$. From (43), it is seen that the coherence matrix of the basis $(\boldsymbol{A}^H\boldsymbol{A})$ is re-weighted by the primal solution vector $\boldsymbol{x}_{\ell_1}^i$ in the ith iteration. The accuracy of the approximation (42) depends on the magnitudes of the off-diagonal elements of the coherence matrix. For discussing the approximation accuracy, note that μ^i depends on the peaks in the magnitude of $|\boldsymbol{u}^i|$

$$\boldsymbol{u}^i \quad = \quad 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \sum_{m\in\mathcal{M}_i} \boldsymbol{a}_m \boldsymbol{x}_{\ell_1,m}^i\right) \tag{44}$$

As long as the *next* active column is orthogonal to *all* active columns $\boldsymbol{a}_m$ with $m \in \mathcal{M}_i$, the approximation is exact.

Although (44) appears to be quite complicated, Corollary 3 assures that the dual coordinate is linear in the regularization parameter for all $\mu^{*(p+1)} < \mu < \mu^{*p}$. It may happen that the coordinate corresponding to the $(i+1)$st peak becomes active first, although $\mathrm{peak}(\boldsymbol{u}^{i-1}, i) > \mathrm{peak}(\boldsymbol{u}^{i-1}, i+1)$. In this case, two sources become active as the regularization parameter is chosen too low. This is not treated in the tables to keep them simple, but this exception can be handled by, e.g., bisection.

B. Dual-based

As asserted by (21), searching for active indices in the dual domain is effectively a form of relaxation of the primal problem (P1′). This amounts to peak finding in the output of a beamformer acting on the residuals, cf. Theorem 1. In line 5, the active set $\mathcal{M}$ is effectively approximated by the relaxed set $\mathcal{U}_i$.

Instead of the primal (P1′), the dual (18a)–(18c) is solved exclusively. As a demonstrative example, we provide the fast iterative algorithm formulated in the dual domain in Table IV. Note that the gird-free atomic norm solutions [13], [15], [16], [17] follows a solution approach similar to this.

VII. Simulation

In this section, the performance of the proposed dual estimation procedures is evaluated based on numerical simulation.

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$.
1:	Initialize $i=0$, $p=1$, $\boldsymbol{x}^0_{\ell_1}=\boldsymbol{0}$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $p < s$
	$i = i+1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, i\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, i+1\big)$
4a:	$\boldsymbol{x}^i_{\ell_1} =$ solution to problem (P1′) for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
4b	$\boldsymbol{u}^i = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^i_{\ell_1}\right)$
5:	$\mathcal{M}_i = \{m \Big\vert\, \lvert x^i_{\ell_1,m}\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}^i_{\ell_1}\rVert_\infty$
	which ensures that $\lvert\mathcal{M}_i\rvert \le i$
7:	if $\lvert\mathcal{M}_i\rvert < p$
8:	$i = i+1$
9:	else
10:	$\boldsymbol{x}^p_{\ell_0} = \boldsymbol{A}^+_{\mathcal{M}_i}\boldsymbol{y}$
11:	$\mathcal{M}_p = \mathcal{M}_i$
12:	$p = p+1$
13:	end
14:	end
15:	Output: $\boldsymbol{x}^p_{\ell_0}, \mathcal{M}_p \quad \forall p = 1\ldots s$

TABLE II
ORDER-RECURSIVE PROCEDURE TO APPROXIMATE (P2).

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$
1:	Initialize $i=0$, $\boldsymbol{x}^0_{\ell_1}=\boldsymbol{0}$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $\lvert\mathcal{M}_i\rvert < s$
	$i = i+1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s+1\big)$
4a:	$\boldsymbol{x}^i_{\ell_1} =$ solution to problem (P1′) for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
4b	$\boldsymbol{u}^i = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^i_{\ell_1}\right)$
5:	$\mathcal{M}_i = \{m \Big\vert\, \lvert x^i_{\ell_1,m}\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}^i_{\ell_1}\rVert_\infty$
	which ensures that $\lvert\mathcal{M}_i\rvert \le i$
7:	end
8:	$\boldsymbol{x}^s_{\ell_0} = \boldsymbol{A}^+_{\mathcal{M}_i}\boldsymbol{y}$
9:	$\mathcal{M}_s = \mathcal{M}_i$
10:	Output: $\boldsymbol{x}^s_{\ell_0}, \mathcal{M}_s$

TABLE III
ITERATIVE PRIMAL-BASED PROCEDURE TO APPROXIMATE (P2).

	Given: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{D} \in \mathrm{diag}\mathbb{R}^{M}$, $\boldsymbol{y} \in \mathbb{C}^{N}$
	Given: $s \in \mathbb{N}$, $F \in]0,1[$
1:	Initialize $i=0$, $\boldsymbol{u}^0 = 2\boldsymbol{D}^{-H}\boldsymbol{A}^H\boldsymbol{y}$
2:	while $\lvert\mathcal{M}_i\rvert < s$
	$i = i+1$
3:	$\mu^i = (1-F)\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s\big) + F\,\mathrm{peak}\big(\boldsymbol{u}^{i-1}, s+1\big)$
4:	$\boldsymbol{u}^i =$ solution to (18a) − (18c) for $\boldsymbol{A}, \boldsymbol{D}, \boldsymbol{y}, \mu = \mu^i$
5:	$\mathcal{U}_i = \{m \Big\vert\, 1 - \frac{\lvert u^i_m\rvert}{\mu} < \epsilon_\mu\}$
6a:	$\boldsymbol{x}^i_{\ell_0} = \boldsymbol{A}^+_{\mathcal{U}_i}\boldsymbol{y}$
6b:	$\mathcal{M}_i = \{m \Big\vert\, \lvert x^i_{\ell_0,m}\rvert > \delta_i\}$, $\delta_i = \epsilon\lVert\boldsymbol{x}^i_{\ell_0}\rVert_\infty$
7:	end
8:	$\boldsymbol{x}^s_{\ell_0} = \boldsymbol{A}^+_{\mathcal{M}_i}\boldsymbol{y}$
9:	$\mathcal{M}_s = \mathcal{M}_i$
10:	Output: $\boldsymbol{x}^s_{\ell_0}, \mathcal{M}_s$

TABLE IV
ITERATIVE DUAL-BASED PROCEDURE TO APPROXIMATE (P2).

We use synthetic data from a uniform linear array with $N = 64$ elements with half-wavelength spacing. The angular domain is discretized by $\theta_m = (m-1)\frac{180^\circ}{M}$ with $m = 1,\ldots,M$ and $M = 180$. The simulation scenario includes $s = 8$ far-field sources modelled by plane waves (28). The uncorrelated noise $\boldsymbol{n}$ is zero-mean complex-valued circularly symmetric normally distributed $\sim \mathcal{N}(\boldsymbol{0},\, \boldsymbol{I})$, i.e. 0 dB power. Eight sources are stationary at $\theta^{\mathrm{T}} = [45,\ 60,\ 76,\ 99,\ 107,\ 120,\ 134,\ 162]$ degrees relative to endfire with constant power level (PL) $[-5,\ 10,\ 5,\ 0,\ 11,\ 12,\ 9,\ 25]$ dB [21].

The dual solution for the order-recursive approach, Table II, corresponds to the results shown in Fig. 11. The faster iterative approach, Table III, yields the results in Fig. 12. The dual solution using the primal solution from the previous iteration is interpreted as a weighted CBF and used for the selection of μ (left column). Next, the convex optimization is carried out for that value of μ giving the dual solution. We plot the dual solution on a linear scale and normalized to a maximum value of 1 which is customary in implementations of the dual for compressed sensing [13], [15], [16]. The number of active sources (see right column in Figs. 11 and 12) are determined according to line 5 in Tables II and III.

For the order-recursive approach step 1, the μ is selected based on the main peak $\theta = 162^\circ$ and a large side lobe at $\theta = 170^\circ$. The solution progresses steadily down the LASSO path. Figure 12 shows the faster iterative approach in Table III for the 8-source problem. In the first iteration we use a μ between the 8th and 9th peak based on the CBF solution (Fig. 12a). There are many sidelobes associated with the source at $\theta = 162^\circ$. As soon as the dominant source is determined, the sidelobes in the residuals are reduced and only 5 sources are observed. After two more iterations, all 8 sources are found at their correct locations.

For both procedures, the main CPU time is used in solving the convex optimization problem. Thus the iterative procedure is a factor 8/3 faster in this case than the straightforward approach which strictly follows the LASSO path. The approach described in Table II has approximately the same CPU time usage as the approach in Ref. [21], but it is conceptually simpler and provides deeper physical insight into the problem.

VIII. CONCLUSION

The complex-valued generalized LASSO problem is strictly convex and strong duality holds. The corresponding dual problem is interpretable as a weighted CBF acting on the residuals of the LASSO. There is a linear one-to-one relation between the dual and primal vectors. Any results formulated for the primal problem are readily extendable to the dual problem. Thus, the sensitivity of the primal solution to small changes in the constraints can be easily assessed.

While the LASSO solution gives the $\boldsymbol{x}_{\ell_1}$ solution it is usually the $\boldsymbol{x}_{\ell_0}$ solution that is of interest. The difference between the $\boldsymbol{x}_{\ell_0}$ and the $\boldsymbol{x}_{\ell_1}$ is characterized via the dual vector and show its strong linear dependence on the regularization

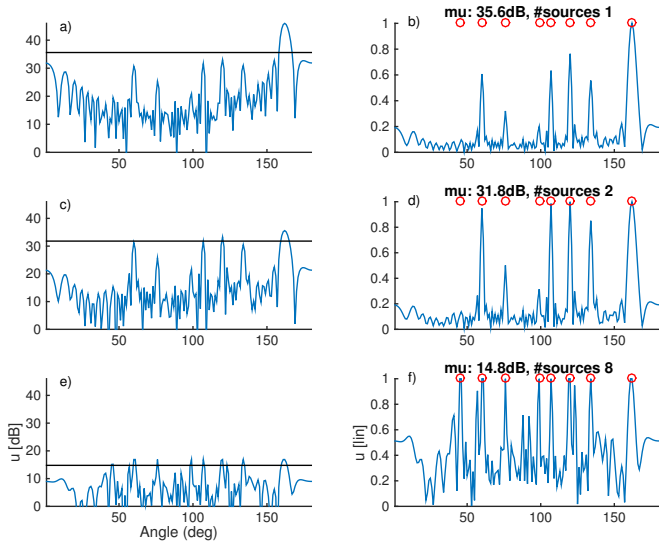


Fig. 11. Dual coordinates for order-recursive approach corresponding to step $p = 1$ (a and b), $p = 2$ (c and d), and $p = 8$ (e and f). Left column: Dual (dB) for the previous step which is used for selecting μ (horizontal line). Right column: Dual (lin) normalized with μ (maximum is 1), the true source locations are marked with $\circ$, and the actual value of μ and number of sources found is also indicated.

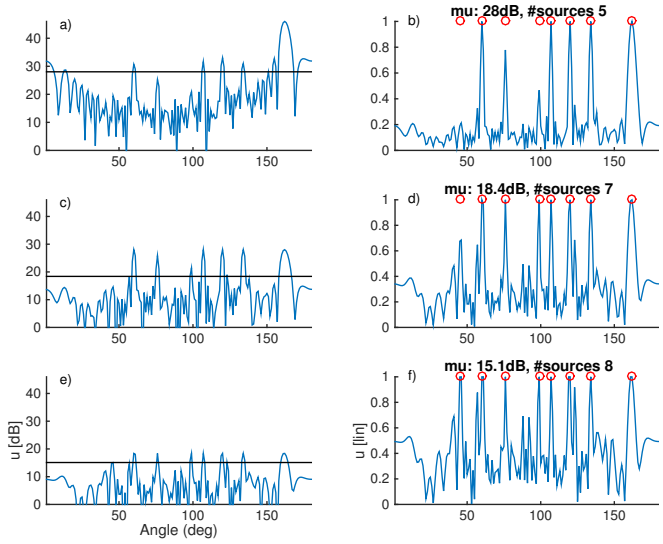


Fig. 12. Dual coordinates iterative approach corresponding for localizing $s = 8$ sources for step $i = 1$ (a and b), $i = 2$ (c and d), and $i = 3$ (e and f). Left column: Dual (dB) for the previous step which is used for selecting μ (horizontal line). Right column: Dual (lin) normalized with μ (maximum is 1), the true source locations are marked with $\circ$, and the actual value of μ and number of sources found is also indicated..

parameter μ and the basis coherence of the active sources, Eq. (26).

Based on mathematical and physical insight, an order-recursive and a faster iterative LASSO-based procedure are proposed and evaluated. These procedures use the dual variable of the generalized LASSO for regularization parameter selection, which greatly facilitates computation of the path as we can predict the changes in the active indexes. Further, a dual-based procedure is formulated which solves only the dual problem. The examples demonstrate the procedures, confirming that the dual and primal coordinates are piecewise linear in the regularization parameter μ.

Appendix A

Proof of (17): Set $\boldsymbol{u} = (u_1, u_2, \ldots, u_M)^T \in \mathbb{C}^M$.

$$\mu\|\boldsymbol{z}\|_1 - \mathrm{Re}(\boldsymbol{u}^H\boldsymbol{z}) = \sum_{m=1}^{M} \left(\mu|z_m| - \mathrm{Re}(u_m^* z_m)\right) \tag{A1}$$

Further, $u_m^* z_m = |u_m|\,|z_m|\,\mathrm{e}^{j\phi_{mm}}$, where the phase difference ϕ_{mm} depends on both u_m and z_m. We continue from (A1)

$$= \sum_{m=1}^{M} \underbrace{\left(\mu - |u_m|\cos\phi_{mm}\right)}_{\tilde{\mu}_m \geq 0\ ?} |z_m|. \tag{A2}$$

If all coefficients $\tilde{\mu}_m \geq 0$ for all choices $z_m \in \mathbb{C}$ then

$$\min_{\boldsymbol{z}} \left(\mu\|\boldsymbol{z}\|_1 - \mathrm{Re}(\boldsymbol{u}^H\boldsymbol{z})\right) = 0, \tag{A3}$$

otherwise there is no lower bound on the minimum. Therefore, all $|u_m|$ must be bounded, i.e. $|u_m| \leq \mu\,\forall\, m = 1, \ldots, M$ to ensure that all $\tilde{\mu}_m \geq 0$ for all possible phase differences $-1 \leq \cos\phi_{mm} \leq 1$. Finally, we note that $\|\boldsymbol{u}\|_\infty = \max_m |u_m|$.

Appendix B: Proofs of Corollaries 1, 2, and 3

Proof for Corollary 1

Let the objective function of the complex-valued generalized LASSO problem (P1′) be

$$\mathscr{L} = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\|\boldsymbol{D}\boldsymbol{x}\|_1\ . \tag{B1}$$

In the following, we evaluate the subderivative $\partial\mathscr{L}$ [29] as the set of all complex subgradients as introduced in [30]. First, we observe

$$\partial\mathscr{L} = -2\boldsymbol{A}^H(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \mu\,\partial\|\boldsymbol{D}\boldsymbol{x}\|_1\ . \tag{B2}$$

Next, it is assumed that $\boldsymbol{D}$ is a diagonal matrix with positive real-valued diagonal entries. It follows that

$$\partial\|\boldsymbol{D}\boldsymbol{x}\|_1 = \begin{cases} \frac{D_{mm}^* x_m}{|D_{mm}x_m|} & \text{for} \quad D_{mm}x_m \neq 0 \\ \{z \in \mathbb{C}, |z| \leq 1\} & \text{for} \quad D_{mm}x_m = 0 \end{cases} \tag{B3}$$

The minimality condition for $\mathscr{L}$ is equivalent to setting (B2) to zero. For all m with $D_{mm}x_m \neq 0$ and with (19), this gives

$$\mu\frac{D_{mm}^* x_m}{|D_{mm}x_m|} = u_m\ . \tag{B4}$$

It readily follows that $|u_m| = \mu$.

Proof for Corollary 2

For matrices $\boldsymbol{D}$ with positive diagonal entries, we conclude for the active set, $m \in \mathcal{M}$,

$$\mu\mathrm{e}^{j\arg(x_m)} = \frac{2}{D_{mm}}\boldsymbol{e}_m^H\boldsymbol{A}^H(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) = u_m\ , \tag{B5}$$

where $\boldsymbol{e}_m$ is the mth standard basis vector. This concludes the proof of Corollary 2.

Proof for Corollary 3

For the primal vector, this was shown in the real-valued case by Tibshirani [1]. For the complex-valued case, this is a direct consequence of Appendix B in [21]. For the dual vector, this was shown in the real-valued case by Tibshirani [10]. For the complex-valued case, this readily follows from Theorem 1.

APPENDIX C: ℓ_0 SOLUTION

The gradient (cf. Appendix B) of the (P0)– and (P1)–objective functions is

$$\nabla \left\| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \right\|_2^2 = -2\boldsymbol{A}^H \left(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \right) \tag{C1}$$

For the active signal components, x_m with $m \in \mathcal{M}$, the ℓ_0-constraint of (P0) is without effect and the solution results from setting the gradient to zero, i.e. solving the normal equations.

$$\boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{y} = \boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{A}_{\mathcal{M}} \boldsymbol{x}_{\ell_0,\mathcal{M}} \quad \Rightarrow \quad \boldsymbol{x}_{\ell_0,\mathcal{M}} = \boldsymbol{A}_{\mathcal{M}}^+ \boldsymbol{y} \tag{C2}$$

We set

$$\boldsymbol{x}_{\ell_0,\mathcal{M}} = \boldsymbol{x}_{\ell_1,\mathcal{M}} + \boldsymbol{\Delta} \ . \tag{C3}$$

This is inserted into (C1),

$$\nabla \left\| \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1,\mathcal{M}} \right\|_2^2 = -2\boldsymbol{A}^H \left(\boldsymbol{y} - \boldsymbol{A}(\boldsymbol{x}_{\ell_0,\mathcal{M}} - \boldsymbol{\Delta}) \right). \tag{C4}$$

Using (6) gives

$$\begin{aligned} 2\boldsymbol{A}_{\mathcal{M}}^H \left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}} \boldsymbol{x}_{\ell_1,\mathcal{M}} \right) &= \boldsymbol{D}_{\mathcal{M}}^H \boldsymbol{u}_{\mathcal{M}} && \text{(C5)} \\ 2\boldsymbol{A}_{\mathcal{M}}^H \left(\boldsymbol{y} - \boldsymbol{A}_{\mathcal{M}} \left(\boldsymbol{x}_{\ell_0,\mathcal{M}} - \boldsymbol{\Delta} \right) \right) &= \boldsymbol{D}_{\mathcal{M}}^H \mu e^{j\boldsymbol{\theta}_{\mathcal{M}}} && \text{(C6)} \\ 2\boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{A}_{\mathcal{M}} \boldsymbol{\Delta} &= \mu \boldsymbol{D}_{\mathcal{M}}^H e^{j\boldsymbol{\theta}_{\mathcal{M}}} && \text{(C7)} \end{aligned}$$

This results in

$$\boldsymbol{\Delta} = \frac{\mu}{2} \left(\boldsymbol{A}_{\mathcal{M}}^H \boldsymbol{A}_{\mathcal{M}} \right)^{-1} \boldsymbol{D}_{\mathcal{M}}^H e^{j\boldsymbol{\theta}_{\mathcal{M}}} \tag{C8}$$

which depends on μ both explicitly and implicitly through $\mathcal{M}$. If the set of nonzero elements of (P0) is equal to the active set of (P1′), the solutions of (P0) and (P1′) differ by (C8).

REFERENCES

[1] R. Tibshirani: Regression Shrinkage and Selection via the Lasso, J. R. Statist. Soc. B, vol. 58, no. 1, pp. 267–288, 1996.

[2] 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, 1997.

[3] E. J. Candès, J. Romberg, T. Tao: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[4] D. Donoho: Compressed sensing, IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[5] J. J. Fuchs: Recovery of exact sparse representations in the presence of bounded noise, IEEE Trans. Inf. Theory, vol. 51, pp. 3601–3608, 2005.

[6] D. M. Malioutov, C. Müjdat, A. S. Willsky: A sparse signal reconstruction perspective for source localization with sensor arrays, IEEE Trans. Signal Process., vol. 53, no. 8, pp. 3010–3022, Aug. 2005.

[7] D. L. Donoho, M. Elad, V. N. Temlyakov: Stable recovery of sparse overcomplete representations in the presence of noise, IEEE Trans. Inf. Theory, vol. 52, pp. 6–18, 2006.

[8] J. A. Tropp: Just relax: Convex programming methods for identifying sparse signals in noise, IEEE Trans. Inf. Theory, vol. 52, pp. 1030–1051, 2006.

[9] M. Elad: Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing, Springer; 2010

[10] R. J. Tibshirani and J. Taylor: The Solution Path of the Generalized Lasso, Ann. Statist., vol. 39, no. 3, pp. 1335–1371, Jun. 2011.

[11] S. Fortunati, R. Grasso, F. Gini, M. Greco, and K. LePage: Single-snapshot DOA estimation by using Compressed Sensing, EURASIP J. Advances in Signal Processing ,Vol 2014, p 120, 2014.

[12] C. Weiss and A. Zoubir: DOA estimation in the presence of array imperfections: A sparse regularization parameter selection problem, in Proc. IEEE Workshop on Statistical Signal Processing (SSP), pp. 348–351, Gold Coast, Australia, Jun. 29–Jul. 2, 2014.

[13] G. Tang and B. N. Bhaskar, B.N. P. Shah, B. Recht, Compressed Sensing Off the Grid, IEEE Transactions on Information Theory, vol. 59, no. 11, pp. 7465-7490, Nov. 2013

[14] A. Xenaki, P. Gerstoft and K. Mosegaard: Compressive beamforming, The Journal of the Acoustical Society of America, vol. 136, no. 1, pp. 260–271, 2014

[15] E. J. Candès and C. Fernandez-Granda, Super-resolution from noisy data, Journal of Fourier Analysis and Applications, vol. 19, no. 6, pp. 1229–1254, Springer, 2013.

[16] E. J. Candès and C. Fernandez-Granda, Towards a Mathematical Theory of Super-resolution, Communications on Pure and Applied Mathematics, vol. 67, no. 6, pp. 906–956, 2014.

[17] A. Panahi and M. Viberg: Gridless Compressive Sensing, in Proc. ICASSP 2014, Florence, Italy, 2014.

[18] J. F. de Andrade Jr, M.L.R. de Campos, and J.A. Apolinário Jr.: A complex version of the LASSO algorithm and its application to beamforming, in Proc. 7$^{\text{th}}$ Int. Telecommun. Symp. (ITS 2010), Manaus, Brazil, Sep 6–9, 2010.

[19] D. L. Donoho, Y. Tsaig, I. Drori, J.-L. Starck: Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit, IEEE Transactions on Information Theory, vol. 58, no. 2, pp.1094–1121, Feb. 2012

[20] A. Panahi, M. Viberg: Fast LASSO based DOA tracking, in Proc. Fourth International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (IEEE CAMSAP 2011), San Juan, Puerto Rico, Dec. 2011.

[21] C. F. Mecklenbräuker, P. Gerstoft, A. Panahi, M. Viberg: Sequential Bayesian Sparse Signal Reconstruction Using Array Data, IEEE Trans. Signal Process., vol. 61, no. 24, pp. 6344–6354, Dec. 2013.

[22] Z. Koldovský, P. Tichavsky, A Homotopy Recursive-in-Model-Order Algorithm for Weighted LASSO, in Proc. ICASSP 2014, Florence, Italy, 2014.

[23] D. Needell and J. A. Tropp, CoSaMP: Iterative signal recovery from incomplete and inaccurate samples, Appl. Comput. Harmon. Anal., vol. 26, pp. 301-321, 2009.

[24] S. P. Boyd, L. Vandenberghe: *Convex optimization*, Chapters 1–7, Cambridge University Press, 2004.

[25] S. J. Kim, K. Koh, S. Boyd, D. Gorinevsky, l_1 trend filtering. SIAM Rev., vol. 51, pp. 339–360. MR2505584.

[26] E. J. Candès, and M. B. Wakin and S. P. Boyd: Enhancing Sparsity by Reweighted l1 Minimization, Journal of Fourier Analysis and Applications, vol. 14, no. 5–6, pp. 877–905, 2008.

[27] Z. He, S. Xie, S. Ding, A. Cichocki: Convolutive blind source separation in the frequency domain based on sparse representation, IEEE Trans. Audio, Speech, and Language Proc., vol. 15, no. 5, Jul. 2007.

[28] A. Panahi, M. Viberg: Fast candidate points selection in the LASSO path, IEEE Signal Process. Lett., vol. 19, no. 2, pp. 79–82, Feb. 2012.

[29] D. P. Bertsekas: Nonlinear programming, Athena Scientific, 1999.

[30] P. Bouboulis, K. Slavakis and S. Theodoridis: Adaptive Learning in Complex Reproducing Kernel Hilbert Spaces Employing Wirtinger's Subgradients, IEEE Trans. Neural Networks and Learning Systems, vol. 23, no. 3, pp.425–438, Mar. 2012.

Density Evolution of Sparse Source Signals

Erich Zöchmann
Institute of Telecommunications
Vienna University of Technology
1040 Vienna, Austria
ezoechma@nt.tuwien.ac.at

Peter Gerstoft
Scripps Institution of Oceanography
La Jolla, CA 92093-0238, USA
gerstoft@ucsd.edu

Christoph F. Mecklenbräuker
Christian Doppler Lab Wireless
Technologies for Sustainable Mobility
Vienna University of Technology
1040 Vienna, Austria

***Abstract*—A sequential Bayesian approach to density evolution for sparse source reconstruction is proposed and analysed which alternatingly solves a generalized LASSO problem and its dual. Waves are observed by a sensor array. The waves are emitted by a spatially-sparse set of sources. A weighted Laplace-like prior is assumed for the sources such that the maximum a posteriori source estimate at the current time step is the solution to a generalized LASSO problem. The posterior Laplace-like density at step k is approximated by the corresponding dual solution. The posterior density at step k leads to the prior density at $k+1$ by applying a motion model. Thus, a sequence of generalized LASSO problems is solved for estimating the temporal evolution of a sparse source field.**

***Index Terms*—sequential estimation, Bayesian estimation, sparsity, generalized LASSO.**

I. Introduction

In this contribution, the online estimation of sparse signals is solved from noisy data samples that become available sequentially in time [2], [3], [4]. The proposed online estimator alternatingly solves a generalized LASSO problem and its dual. Besides the actual reconstruction, we are also interested in the probability density of the source amplitudes and their temporal evolution.

Previously, the Bayesian approach [5], [6], [7] was extended to *sequential* Maximum A Posteriori (MAP) estimation for sparse signals [3], [1]. A sequential MAP filter which preserves sparsity was approximated based on [9]. It uses a single new measurement snapshot in each step.

The theory is formulated so that it is applicable to sparse source estimation in higher spatial dimensions.

II. Dual problem to the generalized LASSO

The generalized LASSO as introduced in [8] penalizes a weighted sum of the optimization variables

$$\min_{\boldsymbol{x}} \left(\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{D}\boldsymbol{x}\|_1 \right) \ , \tag{1}$$

where $\boldsymbol{A}$ is the complex-valued dictionary, $\boldsymbol{x}, \boldsymbol{y}$ are complex-valued vectors and $\mu > 0$. We seek solutions $\boldsymbol{x}$ with given sparsity degree $s \in \mathbb{N}$,

$$\|\boldsymbol{x}\|_0 = s \ . \tag{2}$$

The regularization parameter μ is chosen to satisfy (2).

Following [11], the generalized complex-valued LASSO problem is re-written as

$$\min_{\boldsymbol{x},\boldsymbol{z}} \left(\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu \|\boldsymbol{z}\|_1 \right) \quad \text{subject to} \quad \boldsymbol{z} = \boldsymbol{D}\boldsymbol{x} \ , \tag{3}$$

but now we restrict $\boldsymbol{D}$ to be diagonal with real, positive entries. This substitution provides a Lagrangian multiplier for each element in $\boldsymbol{x}$, and in fact these Lagrangian multipliers will update the corresponding hyperparameters later on.

The dual problem to the generalized LASSO (3) is [8], [10],

$$\max_{\boldsymbol{u} \in \mathbb{C}^M} \boldsymbol{y}^H \boldsymbol{y} - \|\boldsymbol{A}\boldsymbol{A}^+ \boldsymbol{y} - \tfrac{1}{2} \left(\boldsymbol{D}\boldsymbol{A}^+ \right)^H \boldsymbol{u}\|_2^2 \tag{4a}$$

$$\text{subject to} \qquad \|\boldsymbol{u}\|_\infty \leq \mu, \tag{4b}$$

$$(\boldsymbol{D}\boldsymbol{U})^H \boldsymbol{u} \ = \ \boldsymbol{0} \ , \tag{4c}$$

where $\boldsymbol{A}^+$ denotes the Moore-Penrose inverse of the dictionary $\boldsymbol{A}$. Having solved the primal problem, the corresponding dual solution is easily computed with the help of the following theorem [10].

Theorem 1. *The dual vector $\boldsymbol{u}$ is the output of a weighted classical beamformer (weighted matched filter) acting on the vector of residuals, i.e.,*

$$\boldsymbol{u} = 2(\boldsymbol{D}^H)^{-1} \boldsymbol{A}^H (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_{\ell_1}) \ , \tag{5}$$

where $\boldsymbol{x}_{\ell_1}$ - the solution to the primal problem (17) - is such, that the box constraint

$$\|\boldsymbol{u}\|_\infty \leq \ \mu \tag{6}$$

is fulfilled.

Built upon Theorem 1 the following corollary was proven in [10].

Corollary 1. *If the mth primal coordinate is active, i.e. $x_{\ell_1,m} \neq 0$ then the box constraint (6) is tight in the mth dual coordinate. Formally, for any choice $\delta > 0$,*

$$|x_{\ell_1,m}| > \delta \quad \Rightarrow \quad |u_m| = \mu, \qquad (m = 1, \ldots, M) \ . \tag{7}$$

Informally, we say that the mth dual coordinate hits the boundary when the mth primal coordinate becomes active.

We define the active set $\mathcal{M}$ as the set of all indices m with $|x_{\ell_1,m}| > \delta$,

$$\mathcal{M} = \{m \mid |x_{\ell_1,m}| > \delta\}. \tag{8}$$

III. Direction of Arrival Estimation

For the numerical examples, we model a uniform linear array (ULA) which is described with its steering vectors representing the incident wave for each array element.

Let $\boldsymbol{x} = (x_1, \ldots, x_M)^{\mathrm{T}}$ be a vector of complex-valued source amplitudes. We observe time-sampled waveforms on an array of N sensors which are stacked in the vector $\boldsymbol{y}$. The following linear model for the narrowband sensor array data $\boldsymbol{y}$ at frequency ω is assumed,

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n} \ . \tag{9}$$

The mth column of the transfer matrix $\boldsymbol{A}$ is the array steering vector $\boldsymbol{a}_m$ for hypothetical waves from direction of arrival (DOA) θ_m. All columns are normalized such that their ℓ_2 norm is one. The transfer matrix $\boldsymbol{A}$ is constructed by sampling all possible directions or arrival, but only very few of these correspond to real sources. Therefore, the dimension of $\boldsymbol{A}$ is $N \times M$ with $N \ll M$ and $\boldsymbol{x}$ is sparse. The linear model equations (9) are under-determined.
The nmth element of $\boldsymbol{A}$ is modelled by

$$A_{nm} = \frac{1}{\sqrt{N}} \exp\left[-\mathrm{j}(n-1)\pi \sin\theta_m\right]. \tag{10}$$

Here $\theta_m = \frac{\pi(m-1)}{M} - \pi/2$ is the DOA of the mth hypothetical DOA to the nth sensor element of the sensor array.

The additive noise vector $\boldsymbol{n}$ is assumed to be spatially uncorrelated and follows the zero-mean complex normal distribution with diagonal covariance matrix $\sigma^2\boldsymbol{I}$.

For the observation $\boldsymbol{y}$ according to the linear model (9), the conditional probability density given the source vector $\boldsymbol{x}$ is

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp\left(-\frac{1}{\sigma^2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2\right)}{(\pi\sigma^2)^N} \ . \tag{11}$$

For the source vector $\boldsymbol{x}$, a prior probability density is assumed in form of a multivariate complex Laplace-like density [12],

$$p(\boldsymbol{x}) = \prod_{m=1}^{M} p_m(x_m), \text{ with } p_m(x) = \frac{(\lambda_m)^2}{2\pi} \mathrm{e}^{-\lambda_m|x_m|} \ , \tag{12}$$

with associated hyperparameters $\lambda_m > 0$ modelling the source signal strength at location θ_m. $x_m = |x_m|\mathrm{e}^{j\varphi_m}$ is the complex source signal at hypothetical source location θ_m. Note that (12) defines the joint distribution for $|x_m| = r_m$ and φ_m with the phases uniformly distributed on $[0, 2\pi)$, for $m = 1, \ldots, M$. The prior mean and variances are

$$\mathbb{E}\{\boldsymbol{x}\} = \boldsymbol{0}, \qquad \mathbb{E}\{\boldsymbol{x}\boldsymbol{x}^H\} = 6\,\mathrm{diag}\left(\lambda_1^{-2}, \ldots, \lambda_M^{-2}\right) \ . \tag{13}$$

Taking the logarithm of (12) gives

$$-\ln p(\boldsymbol{x}) = \sum_{m=1}^{M} \lambda_m|x_m| - 2\sum_{m=1}^{M} \ln\lambda_m + M\ln 2\pi \ . \tag{14}$$

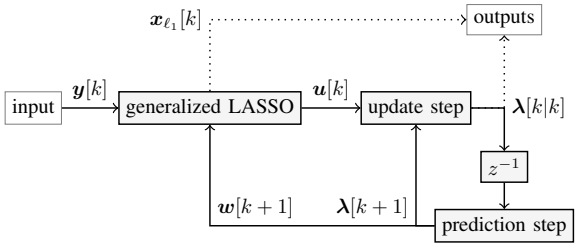


Fig. 1. Signal flow diagram for sequential Bayesian estimation at step k

For the posterior probability density function (pdf) $p(\boldsymbol{x}|\boldsymbol{y})$, Bayes' rule is used for obtaining the generalized LASSO Lagrangian [8], [1]

$$\frac{1}{\sigma^2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\,\|\boldsymbol{W}\boldsymbol{x}\|_1 \tag{15}$$

with bounded weights $\|\boldsymbol{w}\|_\infty = 1$

$$\boldsymbol{W} = \mathrm{diag}(\boldsymbol{w}) = \frac{1}{\mu}\,\mathrm{diag}(\boldsymbol{\lambda}) \ . \tag{16}$$

Equivalently to (15), this is reformulated as

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \mu\,\|\boldsymbol{D}\boldsymbol{x}\|_1 \ , \tag{17}$$

with

$$\boldsymbol{D} = \sigma^2\boldsymbol{W} \ . \tag{18}$$

The minimization of (17) constitutes a strictly convex optimization problem. Minimizing the generalized LASSO Lagrangian (17) with respect to $\boldsymbol{x}$ for given μ, and $\boldsymbol{w} = (w_1, \ldots, w_M)^T$, $\boldsymbol{\lambda} = \mu\boldsymbol{w}$, gives a sparse MAP source estimate $\boldsymbol{x}_{\ell_1}$. This minimization problem promotes sparse solutions in which the ℓ_1 constraint is weighted by giving every source amplitude its own hyperparameter w_m.

IV. Sequential Bayesian Estimation

In [1], a sequential Bayesian sparse source reconstruction was proposed and analyzed which is now interpreted as solving both the generalized LASSO problem (3) and its dual (4a)–(4c) at step k. In the following, the dependency of time is denoted explicitly in all relevant variables, e.g. $\boldsymbol{y}[k]$ to denote the data at step k.

First, the history of all previous array observations is summarized in $\boldsymbol{Y}[k-1] = (\boldsymbol{y}[1], \ldots, \boldsymbol{y}[k-1])$. Given the history $\boldsymbol{Y}[k-1]$ and the new data $\boldsymbol{y}[k]$, we seek the maximum a posteriori (MAP) source estimate $\boldsymbol{x}_{\ell_1}[k]$ for the linear model

$$\boldsymbol{y}[k] = \boldsymbol{A}\boldsymbol{x}[k] + \boldsymbol{n}[k] \ , \tag{19}$$

at step k under the ℓ_1–constraint. The additive noise $\boldsymbol{n}[k]$ is assumed to be both spatially and temporally white,

$$\mathrm{E}(\boldsymbol{n}[k]\boldsymbol{n}^H[k+l]) = \begin{cases} \sigma^2\boldsymbol{I}, & \text{for } l = 0, \\ \boldsymbol{0} & \text{otherwise.} \end{cases} \tag{20}$$

The algorithm in [1] is reformulated in terms of the vector of dual variables in Table I and shown schematically in Figure 1. It is actually the vector of dual variables which carries the sequential information from each step, and not the primal variables as customary in sequential filtering [14].

A. Update Step

In [1] two approximations were introduced in order to relate the posterior weight vector $\boldsymbol{\lambda}[k|k]$ to the prior weight vector $\boldsymbol{\lambda}[k]$ in the form of (12). By means of Theorem 1 and Corollary 1, both approximations for the posterior weight vector are expressible by the dual solution. In the sequel we express the superior approximation, the mean fit, by the dual vector.
In the complement of the active set, the relation between posterior and prior weight vector is given as

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{\left|\boldsymbol{a}_m^H \frac{2}{\sigma^2}(\boldsymbol{y}[k] - \boldsymbol{A}\boldsymbol{x}_{\ell_1})\right|^2}{\lambda_m^2[k]}\right) \quad \forall m \notin \mathcal{M}[k], \tag{21}$$

and in the active set the posterior weight vector must be zero.

$$\lambda_m[k|k] = 0, \quad \forall m \in \mathcal{M}[k] \tag{22}$$

By Theorem 1 we express the numerator of (21) by the dual vector $\boldsymbol{u}$ and the weights $\boldsymbol{w}$. Corollary 1 links Equations (21) and (22), as (21) is zero for $|u_m| = \mu$.

Theorem 2. *With the mean fit approximation, the posterior weight vector $\boldsymbol{\lambda}[k|k]$ is related to the prior weight vector $\boldsymbol{\lambda}[k]$ by the dual solution $\boldsymbol{u}[k]$ at step k,*

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{|u_m[k]|^2}{\mu^2[k]}\right) \ . \tag{23}$$

Due to Theorem 1 and Corollary 1, μ is equal to the max-norm of $\boldsymbol{u}$ and Theorem 2 is expressible solely by the dual vector $\boldsymbol{u}$

$$\lambda_m[k|k] = \lambda_m[k]\left(1 - \frac{|u_m[k]|^2}{\|\boldsymbol{u}[k]\|_\infty^2}\right) \ . \tag{24}$$

Equation (24) shows that the dual coordinate equals μ and the posterior weights become zero at source positions $m \in \mathcal{M}$. Outside the active set, the probability of finding a source depends on the relative sidelobe power level of the beamformer of the LASSO residuals, cf. Theorem 1.

B. Prediction Step

In sequential estimation, typically the prior for the upcoming step $k+1$ is calculated from the current posterior and a state-transition probability density function ("motion model"). In a Markovian stochastic framework this is based on the Chapman-Kolmogorov equation [14]. For Brownian motion the state-transition probability density satisfies the diffusion equation. Our prediction step is therefore based on a diffusion model. Where diffusion occurs just in the neighbourhood of active sources.

1) Neighborhood of an active source: The index neighborhood of m is denoted as $\mathcal{N}_m = \{j \mid m-l, \ldots, m+l\}$. If any $\lambda_j[k|k] \in \mathcal{N}_m$ is less than the threshold λ_0 then a source is active in the neighborhood of m with high probability.

The motion model is defined via the complementary cumulative distribution function (ccdf) of the neighborhood magnitudes,

$$\mathrm{P}\left\{|x_m[k+1]| > \delta \middle| \boldsymbol{x}[k]\right\} = \sum_{j=-l}^{l} \alpha_j \, \mathrm{P}\left\{|x_{m+j}[k]| > \delta\right\} \ , \tag{25}$$

with non-negative coefficients α_j and $\sum_j \alpha_j = 1$. The ccdf, after a polar coordinate transformation $x_m = r_m e^{i\varphi_m}$, evaluates to

$$\begin{aligned}
&\mathrm{P}\left\{|x_m[k+1]| > \delta \middle| \boldsymbol{x}[k]\right\} = \qquad (26)\\
&= \sum_{j=-l}^{l} \alpha_j \int_{-\pi}^{\pi} \frac{\mathrm{d}\varphi_{m+j}}{2\pi} \int_{\delta}^{\infty} \lambda_{m+j}^2[k|k] e^{-\lambda_{m+j}[k|k] r_{m+j}} \ r_{m+j} \ \mathrm{d}r_{m+j}\\
&= \sum_{j=-l}^{l} \alpha_j \left(\lambda_{m+j}[k|k] \ \delta + 1\right) e^{-\lambda_{m+j}[k|k] \ \delta} \ ,
\end{aligned}$$

and by taking the negative derivative w.r.t. δ, we obtain the magnitude's probability density

$$\begin{aligned}
&-\frac{\partial}{\partial \delta} \mathrm{P}\left\{|x_m[k+1]| > \delta \middle| \boldsymbol{x}[k]\right\} =\\
&= 2\pi\delta \sum_{j=-l}^{l} \alpha_j \frac{\left(\lambda_{m+j}[k|k]\right)^2}{2\pi} \ e^{-\lambda_{m+j}[k|k] \ \delta}
\end{aligned} \tag{27}$$

which is a mixture of Erlang-2 distributions with variances $2(\lambda_{m+j}[k|k])^{-2}$, cf. Eq. (12).

We approximate the mixture by a Laplace-like density of the form (12). We choose to fit the variance of the Laplace-like density such that

$$\frac{1}{(\lambda_m[k+1])^2} = \sum_{j=-l}^{l} \alpha_j \frac{1}{(\lambda_{m+j}[k|k])^2} \ . \tag{28}$$

We note that (28) is ill-behaved whenever a posterior weight $\lambda_{m+j}[k|k] = 0$. In this case, a small offset $\varepsilon > 0$ is added to stabilize (28) numerically. The predicted $\boldsymbol{\lambda}[k+1]$ is the product of the regularization parameter $\mu[k+1]$ and the weights $\boldsymbol{w}[k+1]$. As $\mu[k+1]$ is not yet known at step k, we need to assume that the regularization parameter remains constant between k and $k+1$, i.e.,

$$\frac{1}{(\lambda_m[k+1])^2} = \frac{1}{(\mu[k+1]w_m[k+1])^2} \approx \frac{1}{(\mu[k]w_m[k+1])^2} \ . \tag{29}$$

The predicted weights $w_m[k+1]$ are then calculated from the weighted harmonic mean, i.e.,

$$(w_m[k+1])^2 = \left(\sum_{j=-l}^{l} \frac{\alpha_j}{\left(w_{m+j}[k]\right)^2}\right)^{-1} \ . \tag{30}$$

The weighted harmonic mean is a special instance of the weighted Hölder mean [13], see Sec. V. To express the uncertainty of the prediction, the weights are increased by an offset $w_0 > 0$, similar to process noise in Kalman filtering.

	Implementation of density evolution procedure: Given constants: $\boldsymbol{A} \in \mathbb{C}^{N\times M}$, $\boldsymbol{w}[1] \in [0,1]^M$, $s \in \mathbb{N}$
1:	for $k = 1, 2, 3, \ldots$
2:	Input: $\boldsymbol{y}[k] \in \mathbb{C}^N$
	$\boldsymbol{w}[k] = \boldsymbol{w}[k]/\|\boldsymbol{w}[k]\|_\infty$
3:	$\boldsymbol{D}[k] = \sigma^2 \operatorname{diag}(\boldsymbol{w}[k])$
4:	$\boldsymbol{x}_{\ell_1}[k] = s$-sparse solution to generalized LASSO (3) at k
5:	$\boldsymbol{u}[k]$ = corresponding dual solution via Theorem 1
6:	$\mu[k] = \|\boldsymbol{u}[k]\|_\infty$
7:	Update $\boldsymbol{\lambda}[k\|k]$ via Theorem 2
8:	$\boldsymbol{w}[k+1]$ = motion model prediction$(\boldsymbol{\lambda}[k\|k])$
9:	Output: $\boldsymbol{x}_{\ell_1}[k] \in \mathbb{C}^M$, $\boldsymbol{\lambda}[k\|k] \in \mathbb{C}^M$
10:	end

TABLE I
PRIMAL/DUAL FORMULATION OF SEQUENTIAL BAYESIAN SPARSE SIGNAL RECONSTRUCTION

2) Not in the neighborhood of an active source: The posterior $\lambda_j[k|k]$ exceeds the threshold λ_0 which indicates that it is improbable for a source to be near DOA θ_j. At step $k+1$, we penalize the DOA j by adding a multiple of weight uncertainty w_0, i.e., $w_m[k+1] = w_m[k] + c\,w_0$ with $c > 1$. In the simulations, $w_0 = 0.01$ and $c = 10$.

To guarantee that the weights remain upper bounded by 1, the weighting vector is normalized to $\|\boldsymbol{w}\|_\infty = 1$.
The Bayesian procedure is formalized in Table I as a loop over time step k which processes the single snapshot array observation $\boldsymbol{y}[k]$ when it becomes available. In line 3, the weighting coefficients for the generalized LASSO problem (3) are defined for the current step k. The s-sparse solution in line 4 is implemented via the LASSO path [1], [10]. Next, the corresponding dual solution is evaluated by weighted beamforming of the residuals. Finally, the posterior weighting coefficients are evaluated in line 7 which are needed for the prediction step in line 8.

V. Conservative Choice of the Weights

The weighted harmonic mean (30) is a pessimistic mean as low values have stronger impact on the mean. Generally, it tends to broaden the low weight region. This broad low weight region leads to a jitter of the DOA estimate. To mitigate this undesirable effect, we investigate alternative rules for the predicted weights.
A weighted Hölder mean is defined as [13]

$$M_p(w_1^2, \ldots, w_n^2) = \left(\sum_{j=l}^{l} \alpha_j \left(w_j^2\right)^p \right)^{\frac{1}{p}}, \qquad \sum_{j=l}^{l} \alpha_j \quad = 1 \ . \tag{31}$$

For the choice of power $p = -1$, the weighted Hölder mean coincides with the weighted harmonic mean (30). The following inequality holds for weighted Hölder means,

$$M_p < M_q, \quad \text{for} \quad p < q \ . \tag{32}$$

Any Hölder mean with $p > 0$ will not be dominated by lower weights and the arithmetic mean ($p = 1$) is the tightest conservative choice of weighting coefficients for Laplace-like prior. [9] has used a max-log approximation instead of (28) which amounts to picking $M_{+\infty}$, the least tight bound.

VI. Simulations

A. Weight Evolution

We investigate the weight evolution from step $k = 1$ to $k = 100$, where the generalized LASSO of Table I is solved by CVX [15] at each step. The ULA is equipped with $N = 30$ sensors and the angular space is sampled equidistantly with half degree spacing between $-90°$ and $90°$.
In Figure 2 the weight evolution of sources with trivial motion model, $l = 0$ and $\alpha_0 = 1$ is shown. In Figure 3 movement is modelled with a uniform motion model ($l = 2$, $\alpha_j = \frac{1}{2l+1} = 0.2$). Observe the trade off between having precise estimates for the static sources and a good quality estimate of the moving source.

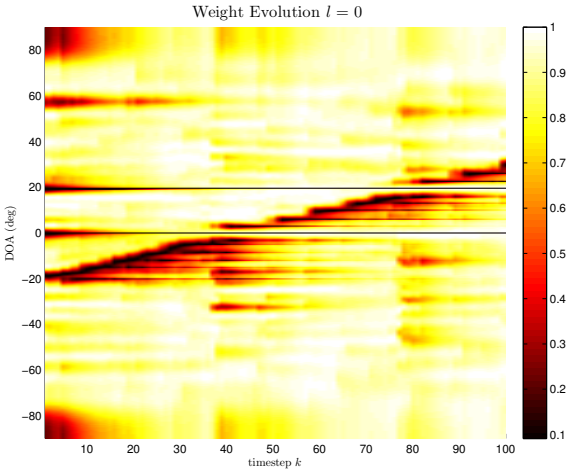


Fig. 2. Weight evolution for 3 sources at DOA $20°, 0°, -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 1$, SNR $= 20$dB

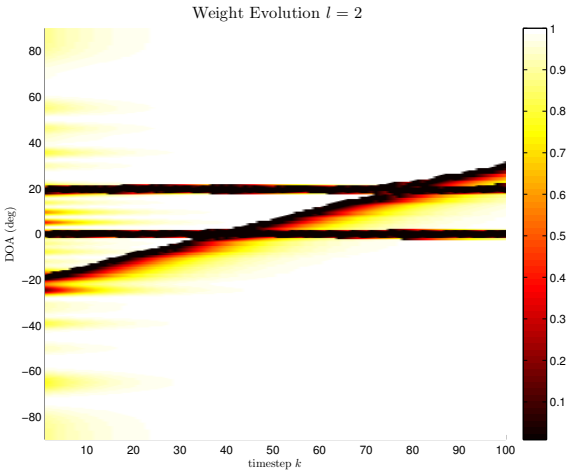


Fig. 3. Weight evolution for 3 sources at DOA $20°, 0°, -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 10$, SNR $= 20$dB

A reasonable compromise of capturing the motion of a source while still improving the estimate of the static sources is to use an $l > 0$ and a conservative choice of the weights. Figure 4 uses the same motion model as in Fig. 3, but the weighted arithmetic mean is used, i.e. Equation (31) for power parameter $p = 1$. For the arithmetic mean, the low weight region of the static sources is narrower than for the harmonic mean. This comes at the expense of the traceability of the moving source.

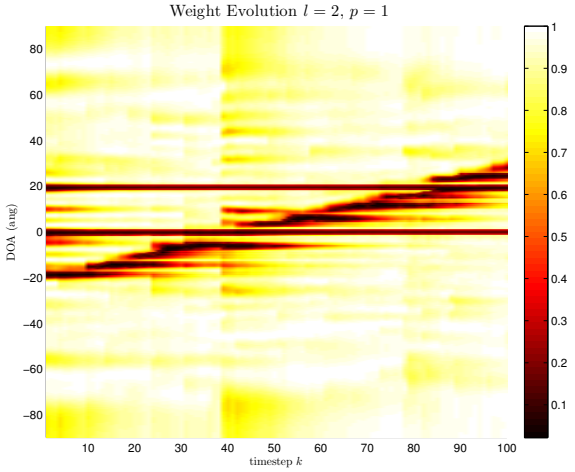


Fig. 4. Conservative weight evolution for 3 sources at DOA $20°; 0°; -20°$, the third source moves with $0.5°$ per time step; $w_0 = 0.01$, $c = 10$, SNR $=$ 20dB

B. Comparison of the Tracking Results

The proposed DOA tracking procedure from Table I is compared to "Compressive Sensing on Kalman filtered residuals (KF-CS)" [17] in Figure 5. For KF-CS μ is chosen non-adaptively analogous to the value given in [18]:Algorithm 1. The density evolution approach with $p = -1$ mean recovers the static sources worse than the Kalman filter and the conservative ($p = 1$) approach, but in return the moving source is traced well.

VII. Conclusion

A sequential reconstruction procedure was proposed which uses both the primal and the dual solution to the generalized LASSO. The dual variable is propagated to the update step, which approximates the posterior distribution with a Laplace-like distribution (see Fig. 1). From the approximated posterior and a motion model, the prior for the next step is derived and the procedure is ready for the next step. Without the prediction step, the proposed procedure is fully equivalent to the procedure in [1]. By including the motion model and prediction step, we show superior performance by means of a synthetic example.

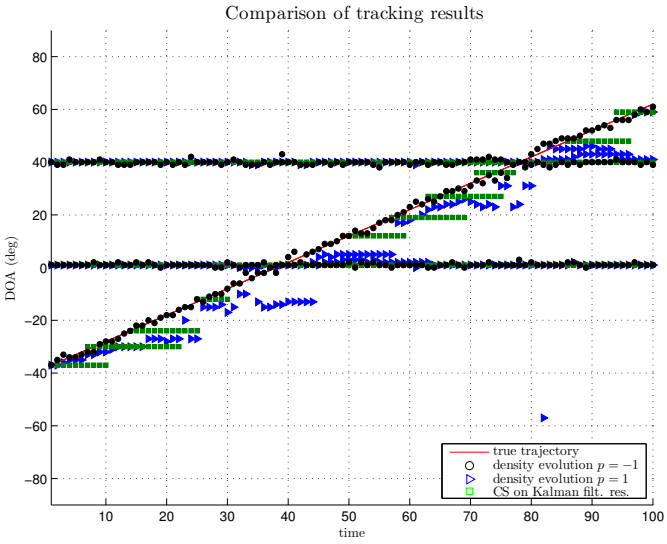


Fig. 5. Active set $\mathcal{M}$ of different tracking algorithms, SNR $=$ 20dB.

References

[1] C. F. Mecklenbräuker, P. Gerstoft, A. Panahi, M. Viberg: Sequential Bayesian Sparse Signal Reconstruction Using Array Data, IEEE Trans. Signal Process., vol. 61, no. 24, pp. 6344–6354, Dec. 2013.

[2] D. M. Malioutov, S. Sanghavi, A. S. Willsky: Sequential Compressed Sensing, IEEE J. Sel. Top. Sig. Proc, vol. 4, pp. 435–444, 2010.

[3] D. Angelosante, J. A. Bazerque, G. B. Giannakis: Online adaptive estimation of sparse signals: Where RLS meets the ℓ_1-norm, IEEE Trans. Signal Process., vol. 58, no. 7, pp. 3436–3447, 2010.

[4] D. Eiwen, G. Tauböck, F. Hlawatsch, H. G. Feichtinger: Compressive tracking of doubly selective channels in multicarrier systems based on sequential delay-Doppler sparsity, in Proc. ICASSP 2011, pp. 2928–2931, Prague, Czech Republic, May 2011.

[5] D. P. Wipf, B. D. Rao: An Empirical Bayesian Strategy for Solving the Simultaneous Sparse Approximation Problem, IEEE Trans. Signal Process., vol. 55, no. 7, pp. 3704–3716, Jul. 2007.

[6] S. Ji, Y. Xue, L. Carin: Bayesian compressive sensing, IEEE Trans. Signal Process., vol. 56, no. 6, pp. 2346–2356, Jun. 2008.

[7] S. D. Babacan, R. Molina, A. K. Katsaggelos: Bayesian compressive sensing using Laplace priors, IEEE Trans. Image Proc., vol. 19, no. 1, pp. 53–63, Jan. 2010.

[8] R. J. Tibshirani and J. Taylor: The Solution Path of the Generalized Lasso, Ann. Statist., vol. 39, no. 3, pp. 1335–1371, Jun. 2011.

[9] A. Panahi, M. Viberg: Fast LASSO based DOA tracking, in Proc. Fourth International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (IEEE CAMSAP 2011), San Juan, Puerto Rico, Dec. 2011.

[10] C. F. Mecklenbräuker, P. Gerstoft, E. Zöchmann: Beamforming of the Residuals is the LASSO's Dual, submitted to IEEE Trans. Signal Process., Dec. 2014, online: `http://arxiv.org/abs/1502.04643`

[11] S. J. Kim, K. Koh, S. Boyd, D. Gorinevsky, l_1 trend filtering. SIAM Rev., vol. 51, pp. 339–360. MR2505584.

[12] Z. He, S. Xie, S. Ding, A. Cichocki: Convolutive blind source separation in the frequency domain based on sparse representation, IEEE Trans. Audio, Speech, and Language Proc., vol. 15, no. 5, Jul. 2007.

[13] P.S. Bullen: *Handbook of means and their inequalities*, Springer, 2003.

[14] B. Ristic, S. Arulampalam, N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA: Artech House, 2004.

[15] M. Grant, S. Boyd, CVX: Matlab software for disciplined convex programming, version 1.21, http://cvxr.com/cvx, Apr. 2011.

[16] A. Xenaki, P. Gerstoft and K. Mosegaard: Compressive beamforming, The Journal of the Acoustical Society of America, vol. 136, no. 1, pp. 260–271, 2014

[17] N. Vaswani. "KF-CS: compressive sensing on Kalman filtered residual." arXiv:0912.1628, 2009.

[18] C. Qiu, W. Lu, and N. Vaswani: "Real-time dynamic MR image reconstruction using Kalman filtered compressed sensing." IEEE International Conference on Acoustics, Speech and Signal Processing, 2009.