Group-Sparse Regression

With Applications in Spectral Analysis and Audio Signal Processing

Kronvall, Ted

– CENTRUM SCIENTIARUM MATHEMATICARUM –

Group-Sparse Regression

with Applications in Spectral Analysis and Audio Signal Processing

TED KRONVALL

Lund University
Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Group-Sparse Regression

with Applications in Spectral Analysis and Audio Signal Processing

Ted Kronvall

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118
SE-221 00 Lund
Sweden

http://www.maths.lth.se/

Doctoral Theses in Mathematical Sciences 2017:7
ISSN 1404-0034

ISBN 978-91-7753-417-4
LUTFMS-1044-2017

Printed in Sweden by Media-Tryck, Lund 2017

Acknowledgements

This thesis marks the completion of my doctoral education in mathematical statistics at Lund University. It is the result of a five-year process in which I owe much to many. Foremost among these is my supervisor Prof. Andreas Jakobsson, who is one of these remarkable persons with both a great mind, and a great heart. During these years, he has been my mentor in matters big and small, colleague, co-author, travel companion, and friend. He has also been available on Skype during all hours of the day and night. I am also deeply grateful towards my co-authors, Dr. Stefan Ingi Adalbjörnsson, Dr. Johan Swärd, Filip Elvander, Maria Juhlin, Santhosh Nadig, Dr. Simon Burgess, and Prof. Kalle Åström for their much appreciated contribution to the papers in this thesis. I am likewise grateful to my colleagues in the statistical signal processing research group, and to our collaborating research groups around the world. Furthermore, I want to give thanks to the present and former administrational and technical staff at the mathematical statistics department, for all their invaluable help. Also, I am grateful to all of my colleagues at the department, for creating that friendly, supportive, and creative environment which I believe is fundamental to good research. I think that the friendly banter and occasional distractions from what I should be doing is what makes it all work. On a personal level, I wish to say thank you to my mother and father, Karin and Andrzej, for their unwavering belief in me, and to my friends, for their love and occasional admiration. Last but not least, thank you Hanna, the center of my existence, for always being on my team.

Lund, Sweden, September 2017 *Ted Kronvall*

Abstract

This doctorate thesis focuses on sparse regression, a statistical modeling tool for selecting valuable predictors in underdetermined linear models. By imposing different constraints on the structure of the variable vector in the regression problem, one obtains estimates which have sparse supports, i.e., where only a few of the elements in the response variable have non-zero values. The thesis collects six papers which, to a varying extent, deals with the applications, implementations, modifications, translations, and other analysis of such problems. Sparse regression is often used to approximate additive models with intricate, non-linear, non-smooth or otherwise problematic functions, by creating an underdetermined model consisting of candidate values for these functions, and linear response variables which selects among the candidates. Sparse regression is therefore a widely used tool in applications such as, e.g., image processing, audio processing, seismological and biomedical modeling, but is also frequently used for data mining applications such as, e.g., social network analytics, recommender systems, and other behavioral applications. Sparse regression is a subgroup of regularized regression problems, where a fitting term, often the sum of squared model residuals, is accompanied by a regularization term, which grows as the fit term shrinks, thereby trading off model fit for a sought sparsity pattern. Typically, the regression problems are formulated as convex optimization programs, a discipline in optimization where first-order conditions are sufficient for optimality, a local optima is also the global optima, and where numerical methods are abundant, approachable, and often very efficient. The main focus of this thesis is structured sparsity; where the linear predictors are clustered into groups, and sparsity is assumed to be correspondingly group-wise in the response variable.

The first three papers in the thesis, A-C, concerns group-sparse regression for temporal identification and spatial localization, of different features in audio signal processing. In Paper A, we derive a model for audio signals recorded on an array of microphones, arbitrarily placed in a three-dimensional space. In a two-step group-sparse modeling procedure, we first identify and separate the recorded audio sources, and then localize their origins in space. In Paper B, we examine the multi-pitch model for tonal audio signals, such as, e.g., musical tones, tonal

speech, or mechanical sounds from combustion engines. It typically models the signal-of-interest using a group of spectral lines, located at some integer multiple of a fundamental frequency. In this paper, we replace the regularizers used in previous works by a group-wise total variation function, promoting a smooth spectral envelope. The proposed combination of regularizers thereby avoids the common suboctave error, where the fundamental frequency is incorrectly classified using half of the fundamental frequency. In Paper C, we analyze the performance of group-sparse regression for classification by chroma, also known as pitch class, e.g., the musical note C, independent of the octave.

The last three papers, D-F, are less application-specific than the first three; attempting to develop the methodology of sparse regression more independently of the application. Specifically, these papers look at model order selection in group-sparse regression, which is implicitly controlled by choosing a hyperparameter, prioritizing between the regularizer and the fitting term in the optimization problem. In Papers D and E, we examine a metric from array processing, termed the covariance fitting criterion, which is seemingly hyperparameter-free, and has been shown to yield sparse estimates for underdetermined linear systems. In the paper, we propose a generalization of the covariance fitting criterion for group-sparsity, and show how it relates to the group-sparse regression problem. In Paper F, we derive a novel method for hyperparameter-selection in sparse and group-sparse regression problems. By analyzing how the noise propagates into the parameter estimates, and the corresponding decision rules for sparsity, we propose selecting it as a quantile from the distribution of the maximum noise component, which we sample from using the Monte Carlo method.

Keywords

sparse regression, group-sparsity, statistical modeling, regularization, hyperparameter-selection, spectral analysis, audio signal processing, classification, localization, multi-pitch estimation, chroma estimation, convex optimization, ADMM, cyclic coordinate descent, proximal gradient.

Contents

List of papers

This thesis is based on the following papers:

A Stefan Ingi Adalbjörnsson, Ted Kronvall, Simon Burgess, Kalle Åström, and Andreas Jakobsson, ”Sparse Localization of Harmonic Audio Sources”. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 117-129, November 2015.

B Filip Elvander, Ted Kronvall, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, ”An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization”, *Elsevier Signal Processing*, vol. 127, pp. 56-70, October 2016.

C Ted Kronvall, Maria Juhlin, Johan Swärd, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, ”Sparse Modeling of Chroma Features”, *Elsevier Signal Processing*, vol. 30, pp. 106-117, January 2017.

D Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, ”Group-Sparse Regression using the Covariance Fitting Criterion”, *Elsevier Signal Processing*, vol. 139, pp. 116-130, October 2017.

E Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, ”Online Group-Sparse Regression using the Covariance Fitting Criterion”, *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, Kos, Greece, August 28 - September 2, 2017.

F Ted Kronvall, and Andreas Jakobsson, ”Hyperparameter-Selection for Group-sparse Regression: A Probablistic Approach”, submitted for possible publication in *Elsevier Signal Processing*.

Additional papers not included in the thesis:

1. Ted Kronvall, and Andreas Jakobsson, ”Hyperparameter-Selection for Sparse Regression: A Probablistic Approach”, *Proceedings of the 51st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, October 29 - November 2, 2017.

2. Ted Kronvall, Andreas Jakobsson, Martin Weiss Hansen, Jesper Rindom Jensen, Mads Græsbøll Christensen, and Andreas Jakobsson, “Sparse Multi-Pitch and Panning Estimation of Stereophonic Signals”, *Proceedings of the 11th IMA International Conference on Mathematics in Signal Processing*, Birmingham, Great Britain, December 12-14 2016.

3. Ted Kronvall, Stefan Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, “Hyperparameter-free sparse linear regression of grouped variables”, *Proceedings of the 50th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, November 6-9 2016.

4. Ted Kronvall, Filip Elvander, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, “Multi-Pitch Estimation via Fast Group Sparse Learning”, *Proceedings of the 24th European Signal Processing Conference (EUSIPCO)*, Budapest, Hungary, August 28 - September 2 2016

5. Maria Juhlin, Ted Kronvall, Johan Swärd, and Andreas Jakobsson, ”Sparse Chroma Estimation for Harmonic Non-stationary Audio”, *Proceedings of 23rd European Signal Processing Conference (EUSIPCO)*, Nice, France, August 31 - September 4 2015.

6. Ted Kronvall, Maria Juhlin, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, ”Sparse Chroma Estimation for Harmonic Audio”, *Proceedings of the 40th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brisbane, Australia, April 19-24, 2015.

7. Stefan Ingi Adalbjörnsson, Johan Swärd, Ted Kronvall, and Andreas Jakobsson, “A Sparse Approach for Estimation of Amplitude Modulated Sinusoids”, *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, Asilomar, USA, November 2-5, 2014.

8. Ted Kronvall, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, ”Joint DOA and Multi-pitch Estimation using Block Sparsity”, *Proceedings of the*

39th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Florence, Italy, May 4-9, 2014.

9. Ted Kronvall, Naveed R. Butt, and Andreas Jakobsson, "Computationally Efficient Robust Widely Linear Beamforming for Improper Non-stationary Signals", *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, Marrakech, Morocco, September 9-13, 2013.

10. Ted Kronvall, Johan Swärd, Andreas Jakobsson, "Non-Parametric Data-Dependent Estimation of Spectroscopic Echo-Train Signals", *Proceedings of the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 26-31, 2013.

Popular scientific summary (in Swedish)

Denna avhandling syftar till att undersöka och vidareutveckla idé och metodik inom forskningsområdena matematisk statistik och signalbehandling. Som så ofta inom den tillämpade matematiken finns i denna avhandling en nära, men också ambivalent, relation mellan teorin och dess tillämpning. Om den matematiska metodiken inte har någon tillämpning försvinner en del av matematikens existensberättigande, i vart fall i det populärvetenskapliga sammanhanget. Men samtidigt, om det bara är tillämpningen som är av intresse, och inte med vilken teori som dess problem ska lösas, försvinner också det sammanhang i vilket den tillämpade matematikern kan verka framgångsrikt. Om det bara är de kortsiktiga resultaten som räknas; om huvudsaken är att det just nu aktuella problemet kan lösas, då kan man också gå miste om de långsiktiga, världsomvälvande resultaten. Den tillämpade matematikern arbetar därför i gränslandet mellan det kortsiktiga och det långsiktiga, hållandes den teoretiske matematikern i ena handen och den praktiske ingenjören i den andra. I denna avhandling beskrivs problemställningar inom några olika tillämpningar, men det är inte dessa som främst är av intresse. Tillämpningarna är valda eftersom de utgör exempel där liknande matematisk metodik kan användas, och det är just metodiken som utgör avhandlingens mittpunkt.

Avhandlingen tar upp begreppet *regressionsanalys*, som används för att undersöka samband mellan uppmätt data och olika faktorer som kan beskriva den. Den fokuserar på en relativt ny sorts regressionsanalys som kallas *sparse regression* (eller gles regressionsanalys på svenska). Metodiken används för att hitta samband i potentiellt enorma system av faktorer, eller *features*. I sådana system antas endast ett litet antal features behövas, vilket motsvarar en sparse variabelvektor. Sparse regression är en metodik för att ett antal finna ett litet antal nålar i en stor höstack. Det är en metodik med vilken man med små antaganden snabbt kan leta efter mönster i stora datamängder. Av denna anledning kallas också systemet av features för *dictionary* (eller ordbok på svenska), då den innehåller alla relevanta features. Forskning kring sparse regression har pågått i drygt två decennier. Metodiken har många tillämpningar, exempelvis talkodning, bildanalys, DNA-sekvensering, mönsterigenkänning och dataanalys för sociala medier. Fokus för

denna avhandling är system där features är klustrade. Det innebär att de mönster som eftersöks inte beskrivs av en, utan av flera features, vilka framträder i grupper.

Den tillämpning som undersökts mest i denna avhandling är tal- och musikigenkänning. Ljud består av förtunningar och förtätningar av ett medium, typiskt luft, vilka kan ses som longitudinella vågor. Beroende på vågornas frekvens (men även andra features) får ljudet sin karaktär och en noggrann frekvensanalys kan användas för att skilja olika ljudkällor från varandra. Tal och musik som är tonande, exempelvis vokalljud, har en frekvensinnehåll som består av ett antal av frekvenser. Dessa har ett särskilt matematiskt samband som är kopplat till ljudets tonhöjd. Metodiken group-sparse regression kan då användas för att identifiera en viss ljudkälla med hjälp av dess tonhöjd. Frekvenser som motsvarar viss tonhöjd placeras då tillsammans i en grupp, och dictionaryt utgörs av ett system av grupper för alla möjliga tonhöjder. För en kort sekvens ljud förväntar man sig inte att alla grupper finns närvarande, utan endast en fåtal, varför en group-sparse variabelvektor eftersöks.

Avhandlingen inleds med en introduktion av tidigare forskning inom sparse och grupp-sparse regression, samt en översikt av tillämpningarna. Därefter följer sex artiklar som publicerats i tidskrifter inom området signalbehandling. I artikel A härleds en metodik för att identifiera och lokalisera ljudkällor i ett rum. Dessa har spelats in av en uppsättning mikrofoner vilka godtyckligt ställts upp i rummet. Testscenariot är att två eller flera personer pratar i mun på varandra och går runt i ett rum. Rummet har en viss återklang, d.v.s. ljudet studsar i rummets väggar, tak och golv. Identifikationsmässigt är problemet väldigt svårt; inom forskningens anses det som ett delvis olöst problem. En svårighet är att bestämma hur personernas röster ska skiljas från varandra, särskilt när man inte vet hur många dessa är. Det är också svårt att bestämma personernas position i rummet när ljudet studsar. I artikeln angrips problemet genom en tvåstegsraket. Steg ett är att identifiera ljudkällornas tonhöjder genom att dela upp ljudet i små sekvenser och finna tonhöjderna i varje sekvens. I steg två fastställs sedan, för varje identifierad person i varje sekvens, en eller flera positioner för denne. Dessa kommer att motsvara både personens riktiga position, men också studsarnas positioner. I båda stegen används group-sparse regression; i steg ett används ett dictionary med olika tonhöjder, i steg två ett dictionary med olika positioner. Fördelarna med metodiken för detta problem är att huvudsakligen två; dels behöver man på förhand inte veta antal personer som finns i rummet, dels kan positionering ske trots att ljudet studsar.

För sparse regression finns det oftast en eller flera inställningsparametrar som måste optimeras, men detta kräver detta en hel del beräkningskraft och tid. För problemet med identifikation av ljudets tonhöjd, som också kallas *pitch*, krävs ibland tre sådana parametrar. I artikel B härleds en metodik för att reducera bort minst en av dessa. Detta görs genom att dess optimeringsproblem modifieras med hjälp av en funktion som ofta används inom matematisk bildanalys. I artikel C undersöks en feature som är vanlig inom musikteori; *chroma* (eller tonklass på svenska). Dessa är till exempel tonklasserna som används för att komponera musik, såsom tonen C, oavsett vilken oktav den spelas i. Som beskrivits ovan kan en ton modelleras som en grupp av frekvenser. Chroma blir då en feature som innehåller alla toner inom samma tonklass. Dictionaryt för chroma innehåller sedan alla relevanta chroma för ett visst musikstycke. I artikeln beskrivs en utveckling av group-sparsity, där innehållet i varje grupp också är sparse. Det passar väl problemet med att finna identifiera chroma, då en chroma-grupp innehåller alla möjliga oktaver för en ton, medan en inspelning med detta chroma antas innehålla endast ett fåtal oktaver.

I artikel D till F avses ingen särskild tillämpning, i dessa föreslås istället olika förbättringar och modifikationer för group-sparse regression. Artikel D utgår från ett optimeringsproblem som används för matchning av *kovariansmatriser*; ett vanligt statistisk sätt att mäta beroende i dataserier. Ur denna härleds en metodik för group-sparse regression där ingen inställningsparameter behöver anges. I artikeln härleds vidare sambandet mellan metoden för kovariansmatchning och gängse metoder för group-sparse regression, vilket visar hur inställningsparametrarna kan väljas i group-sparse regression. I artikel E vidareutvecklas metoden i artikel D för att kunna köras *online*, vilket innebär att man så beräkningseffektivt som möjligt vill uppdatera lösningen i takt med att ny data insamlas. Artikel F ägnas helt åt hur inställningsparametrarna väljs. Vanligtvis används en statistisk metod som kallas *kors-validering* för detta, där regressionsproblemet löses för en mängd olika värden på inställningsparametrarna. Dessutom görs detta flera gånger, där datat varje gång delas upp i två delar. Den ena delen används för att skatta lösningen, den andra för att utvärdera hur bra lösningsvektorn kan användas för prediktion. Inställningsparametern väljs sedan som det värde som gör prediktionen så noggrann som möjligt. Denna metod har två nackdelar; först och främst att metoden är väldigt beräkningstung, men även även att metoden optimerar prediktion, inte specifikt urvalet av features, som ofta är det sökta problemet. I artikeln föreslås istället en metodik som med hjälp av sannolikhetsteori väljer inställningsparamet-

ern utifrån den statistiska fördelningen av det insamlade datats brus. I sparse regression anger inställningsparametern vad som är en legitim feature och vad som är mätfel och brus. Parametern skall därför typiskt skall väljas större än bruset, men mindre än den sökta signalen, vilken är okänd. Med den statistiska *Monte Carlo-metoden* kan man sedan numeriskt skatta fördelningen av den maximala brusnivån, från vilken man sedan kan välja inställningsparametern som en lämplig *kvantil* (eller risknivå). I numeriska jämförelser visar sig denna metodik vara både bättre på att välja features, men också mer beräkningseffektiv, än korsvalidering. Det är alltså tydligt att sparse regression är ett mycket mångsidigt verktyg. Det är också en relativt enkel matematisk metodik, som många ingenjörer och tekniker kan ta del av för att hitta mönster i data.

Det kan också avslutningsvis nämnas att för många problem, däribland flera av problemen i avhandlingen, kan sparse regression kombineras med *maskininlärning*. Maskininlärning är ett metodik inom datavetenskapen för automatisk mönsterigenkänning, där både features och modellparametrar tränas in istället för att väljas. Grundtanken är att insamlad data sällan beskriver isolerade fenomen; genom att låta systemet lära sig från tidigare insamlad data kan man bättre tolka ny data. Maskininlärning har inte undersökts i denna avhandling, men sambandet mellan sparse regression och maskininlärning passar utmärkt för framtida forskning.

List of abbreviations

ANLS	Approximative Non-linear Least Squares
ADMM	Alternating Directions Method of Multipliers
BEAMS	Block sparse Estimation of Amplitude Modulated Signals
CCA	Cross-Correlation Analysis
CCD	Cyclic Coordinate Descent
CEAMS	Chroma Estimation of Amplitude Modulated Signals
CEBS	Chroma Estimation using Block Sparsity
CRLB	Cramér-Rao Lower Bound
DFT	Discrete Fourier Transform
DFTBA	Don't Forget To Be Awesome
DOA	Direction-Of-Arrival
FAIL	First Attempt In Learning
HALO	Harmonic Audio LOcalization
KKT	Karush-Kuhn-Tucker
LAD-LASSO	Least Absolute Deviation LASSO
LARS	Least Angle RegreSsion
LASSO	Least Absolute Shrinkage and Selection Operator
LS	Least Squares
NLS	Non-linear Least Squares
MC	Monte Carlo
MIR	Music Information Retreival
ML	Maximum Likelihood
PEBS	Pitch Estimation using Block Sparsity
PEBSI-Lite	PEBS - Improved and Lighter
PEBS-TV	PEBS - Total Variation
PROSPR	PRObabilistic regularization approach for SParse Regression
RMSE	Root Mean Square Error
SFL	Sparse Fused LASSO
SGL	Sparse Group-LASSO
SNR	Signal-to-Noise Ratio

SOC	Second Order Cone
SR-LASSO	Square-Root LASSO
STFT	Short-Time Fourier Transform
TDOA	Time-Difference-Of-Arrival
TOA	Time-Of-Arrival
TR	Tikhonov Regularization
TV	Total Variation
ULA	Uniform Linear Array
YOLO	You Only Live Once

List of notation

Typical notational conventions

$\mathbf{a}, \mathbf{b}, \ldots$	boldface lower case letters denote column-vectors
$\mathbf{A}, \mathbf{B}, \ldots$	boldface upper case letters denote matrices
$A, a, \Delta, \alpha, \ldots$	non-bold letters generally denote scalars
$\boldsymbol{\Psi}, \boldsymbol{\psi}, \ldots$	bold-face greek letters generally denote parameter sets
$\mathcal{I}, \mathcal{J}, \ldots$	caligraphic upper case letters generally denote index sets
$(\cdot)^T$	vector or matrix transpose
$(\cdot)^H$	Hermitian (conjugate) transpose
$(\cdot)^\dagger$	Moore-Penrose pseudo-inverse
$\hat{(\cdot)}$	an estimated parameter
$(\cdot)_+$	positive threshold of real scalar, $(a)_+ = \max(0, a)$
$\{\cdot\}$	the set of elements or other entities
$\lvert \cdot \rvert$	magnitude of complex scalar
$\lvert\lvert \cdot \rvert\rvert$	the Euclidean norm of a vector, $\lvert\lvert a \rvert\rvert = \sqrt{\mathbf{a}^H \mathbf{a}}$
$\lvert\lvert \cdot \rvert\rvert_q$	the ℓ_q-norm of a vector, $\lvert\lvert a \rvert\rvert_q = \left(\sum_p \lvert a_p \rvert^q \right)^{1/q}$ but is not a proper norm for $p < 1$
$\lvert\lvert \cdot \rvert\rvert_0$	the ℓ_0-"norm" of a vector, $\lvert\lvert a \rvert\rvert_0 = \sum_p \lvert a_p \rvert^0$
$\lvert\lvert \cdot \rvert\rvert_\mathcal{F}$	the Frobenius norm of a matrix
abs$(\cdot)$	element-wise magnitude of (a vector or matrix)
arg$(\cdot)$	element-wise complex argument of
$\mathbb{R}^{n \times m}$	the real $n \times m$-dimensional space
$\mathbb{R}^n$	the real n-dimensional plane ($\mathbb{R}$ is used for $n = 1$)
$\mathbb{C}^{n \times m}$	the complex $n \times m$-dimensional space
$\mathbb{C}^n$	the complex n-dimensional space
$\mathbb{Q}$	the set of rational numbers
$\mathbb{Z}$	the set of integers
$\mathbb{N}$	the set of natural numbers
Im$(\cdot)$	the imaginary part of

$\mathrm{Re}(\cdot)$	the real part of
i	the imaginary unit, $\sqrt{-1}$, unless otherwise specified
$\forall$	for all (members in the set)
$\triangleq$	defined as
$\approx$	approximately equal to
$\times$	multiplied by, or dimensionality
$\otimes$	Kronecker product by
∂	differential of
$\in$	belongs to (a set)
$\subseteq$	is a subset of (a set)
$\sim$	has probability distribution
$P(\cdot)$	probability of event
$E(\cdot)$	expected value of a random variable
$V(\cdot)$	variance of a random variable
$D(\cdot)$	standard deviation of a random variable
$\mathcal{N}(\boldsymbol{\mu}, \mathbf{R})$	the multivariate Normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\mathbf{R}$
$\mathrm{Cov}(\cdot)$	the covariance matrix
$\arg\max(\cdot)$	the argument that maximizes
$\arg\min(\cdot)$	the argument that minimizes
$\mathrm{vec}(\cdot)$	column-wise vectorization of a matrix
$\mathrm{diag}(\cdot)$	diagonal matrix with specified diagonal vector
1-D, 2-D, ...	one-dimensional, two-dimensional, ...

Introduction

These lines introduce a doctoral thesis in the cross-section between the fields of mathematical statistics and signal processing. It takes the perspective of statistical signal processing, especially that of Kay (1993) [1] and Scharf (1991) [2], whose good practices hopefully will shine through in the analysis, solution, and execution done here. In line with this heritage, this thesis attempts to judge performance from a statistical point of view, i.e., whether estimation procedures are good or bad in terms of, e.g., *efficiency*, *consistency*, and *bias*. Many of the issues raised in the thesis concerns modeling; how to construct parametric models for different types of data, and how to estimate its parameters without unnecessary computational cost, to a desired precision in convergence. The main focus is modeling with sparse parameter supports; how very large linear systems can be used to model both linear and non-linear systems, and how to construct optimization problems to obtain estimates where the majority of the parameters become zero. The main formulation and analysis for sparse modeling derives from the work of Tibshirani (1996) [3], herein extended with a variety of criterions which enforce certain sparsity structures. Particularly, the thesis is concerned with linear models where the sought atoms exhibit some form of natural grouping behaviour. For these problems, different combination of regularizing the regression problem is used to promote suitably group-sparse solutions. Grouping of components often pose combinatorial issues, as the structural criteria may be implicitly defined, or as groups may have overlapping components, which the thesis will focus on dealing with. A benefit of using sparse modeling is that model orders, i.e., the number of groups and size of each group, are set implicitly, and so alleviates the need of model order estimation, which is a difficult problem necessary for parametric modeling. Many of the methods presented in this thesis are readily applicable to spectral estimation problems, and many fundamental results are based upon the standard reference of Stoica and Moses (2005) [4]. In the included works, the data is often modeled using a parametric sinusoidal model, where signals are assumed to be well described as super-positioned complex sinusoids, having both linear and non-linear parameters, corrupted by some additive noise. Using sparse estimation, these non-linear parameters are estimated using an overcomplete set

of candidate parameters, each activated by a linear parameter subject for estimation. Experience shows that a group of sinusoids can be used to describe the tonal part in acoustical signals, wherein the frequencies of the components in an audio source often exhibit a predetermined relationship, from which a cluster may be formed. Many of the papers in the thesis focus on one such relationship, termed pitch; a perception model for which describes the spectral content of many naturally occuring sounds, such as, e.g., from tonal voice, many musical instruments, and even from combustion engines. An other feature, herein modeled using grouped sinusoids, also closely related to pitch, is chroma; a musical property which is important in, for instance, music information retrieval (MIR) applications. Furthermore, this thesis will touch upon the field of array processing, where signals are also attributed with some spatial information. In fact, many results in spectral analysis may be used in array processing, and vice versa, as these fields are highly related. To give some fundamental context for the papers of which this thesis consists, some preliminaries from sparse modeling, spectral analysis, audio analysis, and array processing will constitute the bulk of this introductory chapter. Lastly, an overview of the papers in this thesis is given.

1 Modeling for sparsity

1.1 Preliminaries

This thesis deals with modeling of data variables using linear models. Given a measured or otherwise acquired sequence of N data variables stored in a vector $\mathbf{y}$, relationships on the form

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1}$$

are herein considered in order to identify some sought quantity, to encode, e.g., for transmission, or to reconstruct the data in some form. When, as in (1), the data is exactly modeled by the M parameters in $\mathbf{x}$ and the linear map $\mathbf{A}$, and the system is thus noiseless, whereas if

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{2}$$

for some non-zero noise component $\mathbf{e}$, the linear system is corrupted by noise and only captures a part of the data's variability. When the noise component is assumed to be stochastic with some addional imposed conditions, the noisy data model is often referred to as a linear regression model, where a trend is identified among the dependent variables, $\mathbf{y}$, described through the regressor matrix $\mathbf{A}$, such that an increase in $\mathbf{y}$ is proportional to an increase of the regression coefficients $\mathbf{x}$. For linear regression, two common assumptions are that $M < N$ and that the columns of $\mathbf{A}$ are pairwise independent. Also, it is typically assumed that the elements of $\mathbf{e}$ are independent and identically distributed, where, however, cases when the noise terms have different variances are sometimes considered. An objective of linear regression is to estimate the unknown regression coefficients given the observed data and known regressor matrix. Commonly, the estimator is formed by minimizing the ℓ_2-norm of the squared model residuals, i.e.,

$$\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 \tag{3}$$

which can be obtained using the Moore-Penrose pseudoinverse $\mathbf{A}^\dagger$, as

$$\hat{\mathbf{x}} = \mathbf{A}^\dagger \mathbf{y} \tag{4}$$

The Moore-Penrose pseudoinverse is a generalization of the matrix inverse, and exists for any system $\mathbf{A}$. If the assumptions stated above hold, may be obtained in closed form as

$$\mathbf{A}^\dagger \triangleq \left(\mathbf{A}^H \mathbf{A}\right)^{-1} \mathbf{A}^H \tag{5}$$

However, in this thesis, these assumptions are typically stretched or violated in some way, albeit with other assumptions made in their place. In particular, a recurring case is that $M \gg N$, such that the linear system is highly underdetermined with no unique solution. Furthermore, is it assumed that $\mathbf{x}$ has a sparse parameter support, meaning that only few of the elements in $\mathbf{x}$ are non-zero. In other words, it is assumed that the data is sparse in some high-dimensional domain, and that $\mathbf{A}$ is a linear map to that domain, i.e., $\mathbf{y} \overset{\mathbf{A}}{\mapsto} \mathbf{x}$.

The process of parameter estimation under some sparse constraint is often referred to as sparse modeling, where in particular, the constrained regression problem introduced in the next section is referred to as sparse regression. In the sparse modeling framework, $\mathbf{A}$ is also described as a dictionary or codebook, and its columns as atoms, due to the fact that the observed data may be seen figuratively as a combination of a small number of components from a vast library of candidate components.

1.2 Motivations

Sparse regression is an approach well suited for solving many problems in statistics and signal processing, depending on which the choice of dictionary, estimation approach, and numerical solver is deliberately made. In particular, problems often considered are

- How to reconstruct the data vector $\mathbf{y}$ using fewer than N data samples. Given some sparse encoding $\mathbf{A}$, only the non-zero parameters of $\mathbf{x}$ and their positions in the vector need to be stored or transmitted, from which a reconstruction can be made. This research subject is typically referred to as compressed sensing, see, e.g., [5, 6], and has attracted much attention during the last decades.

- Identifying and estimating the parameters of a non-linear system. When the data is a sum of non-linear functions with respect to some multidimensional parameter, sparse regression may be used to approximate each non-linear function using a set of linear functions, each representing a possible outcome of the sought parameter. The parameters for the linear system, $\mathbf{x}$, thus serves as activation and magnitude parameters, where the correct values of the sought non-linear parameters should be indicated by large magnitudes of the corresponding candidates in the linear model. By construction, the linear system becomes highly underdetermined and the use

of a sparse regression model is designed as to yield few linear parameters with significant magnitudes. The approach will identify the non-linear system on a grid of possible parameter outcomes, which is applicable for both discrete and continuous non-linear parameters. In the latter case, the dictionary may only represent a subset of possible outcomes of the continuous parameter space, for which a careful dictionary design must be made. The parameter estimates are often visualized as pseudo-spectra, for which a user may identify the number of components and their non-linear parameters. In particular, sparse regression is commonly used for estimation of line spectra, see, e.g., [7], where the estimated pseudo-spectra typically offers resolution capabilities far superior to the periodogram[1].

- How to separate and identify the components of mixed observations. When the data consists of a number of superimposed components, and the objective is to identify exactly which ones and how many, sparse regression can be primed for selection and model order estimation. Given a dictionary which exactly represents the data, but which is highly redundant, sparse regression can be used for identifying which ones are represented in the data, and, using careful statistical analysis, surmising precisely how many atoms the the observed data allows to model. This feature is often referred to as support recovery, or sparsistency [8].

1.3 Regularization and convexity

A system on the forms (1) or (2), where the number of unknowns outnumber the number of observations, either lacks or have infinitely many solutions. Such systems, termed ill-posed, are in this thesis solved using different regularized optimization approaches. Essentially, an optimization method seeks to minimize some criterion, also called objective or loss function, $f(\mathbf{x}) : \mathbb{C}^m \mapsto \mathbb{R}$ which goes to zero as $\mathbf{x}$ approaches its true value, say $\mathbf{x}^*$, such that $f(\mathbf{x}) \geq f(\mathbf{x}^*), \forall \mathbf{x} \in \mathbb{C}^m$. Typically, for the linear systems discussed here, the loss function is designed to measure the deviation from a perfect reconstruction using norms, i.e.,

$$f(\mathbf{x}) = \|\mathbf{y} - \mathbf{A}\mathbf{x}\| \tag{6}$$

In regularization methods, the loss function is balanced by a regularizer, $g(\mathbf{x}) : \mathbb{C}^m \mapsto \mathbb{R}$ which increases as the complexity of $f(\mathbf{x})$ increases. The regularizer can

[1] The periodogram is defined as the square magnitude of the discrete Fourier transform (DFT)

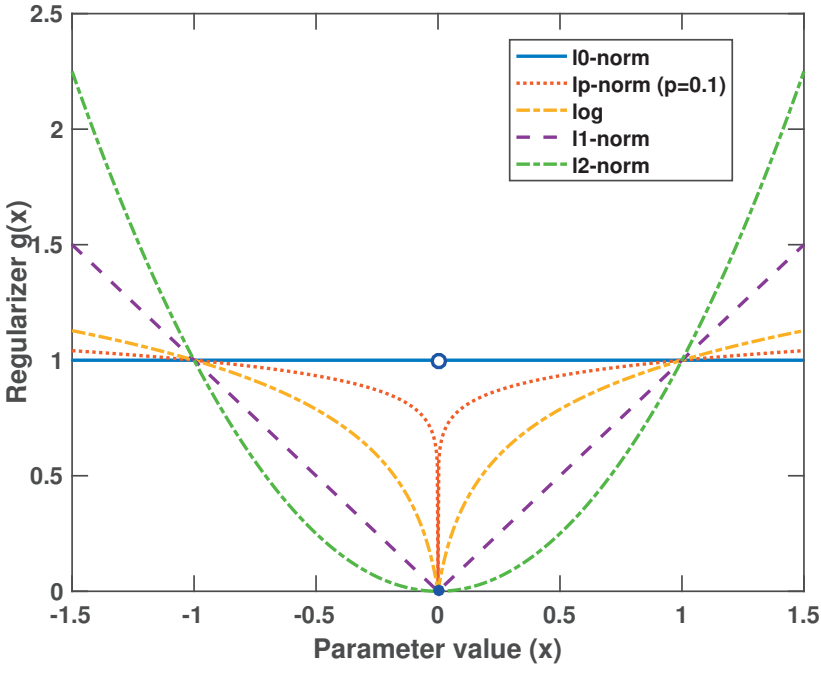


Figure 1: A comparison of different penalty functions for a scalar variable x. The ℓ_0 penalty is the most sparsity-enforcing, as any deviation from zero adds cost. Only the ℓ_1 and ℓ_2 functions are convex, whereof only the former enforces sparsity.

be seen as as a way of imposing Occam's razor to the solution, or alternatively the more contemporary KISS principle[2], and is designed to prevent overfitting the reconstruction quantity. The optimization problem sought to solve thus becomes

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) + \lambda g(\mathbf{x}) \tag{7}$$

where λ is a user-parameter controlling the degree of regularization. In the linear systems discussed here, the regularizer typically includes the norm of some function of $\mathbf{x}$. Figure 1

[2]The acronym spells out 'Keep it simple, stupid' and originates from the U.S. Navy forces in the 1960's.

shows an example of the regularization functions

$$\|\mathbf{x}\|_0 = \sum_{m=1}^{M} 1\{x_m \neq 0\} \tag{8}$$

$$\|\mathbf{x}\|_q = \left(\sum_{m=1}^{M} |x_m|^q \right)^{1/q} \tag{9}$$

$$\frac{1}{1+c} \sum_{m=1}^{M} \ln\left(1 + c|a_m|\right) \tag{10}$$

for $q = \{0.1, 1, 2\}$, and where c in (10) is a positive constant, which increases the absolute slope close to zero. In the figure, c is set to 20. A point of interest for imposing sparse solutions is at which rate a deviation from zero adds a regularizing penalty or cost. In this sense, the ℓ_0-norm[3] is optimal - even an infinitesmal non-zero value in an element adds a cost which must be justified by a significant decrease of the loss function. This regularizer is, however, impractical to use, as it requires solving an exhaustive search among all possible combinations of non-zero and zero elements of $\mathbf{x}$. To simplify estimation, regularized problems are typically designed to be convex, which in this example only the ℓ_1- and ℓ_2-norms are. Their respective effects on the solution are, however, completely different. Figure 2 illustrates the intuition behind their effects on the solution in $\mathbb{R}^2$. It shows the graphical representation of the equivalent constrained optimization problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) \tag{11}$$

$$\text{subject to} \quad g(\mathbf{x}) \leq \mu \tag{12}$$

where the left figure illustrates $g(\mathbf{x}) = \|\mathbf{x}\|_1$ and right figure illustrates $g(\mathbf{x}) = \|\mathbf{x}\|_2$. In both cases, the ellipse illustrates the level curves of the loss function, which has its unconstrained optimum in the center of the ellipse. The solutions can be found as the intersection points between the loss function and the regularizers' level curves for some μ. Here, one sees that the ℓ_1-norm intersects with the loss function at its edges, yielding zero elements. As a contrast, the smooth ℓ_2 norm is unlikely to intersect the loss function att precisely zero for some dimension. This example serves to introduce the reader as to why certain

[3] For correctness, is should be noted that the ℓ_0-norm is not a proper norm, as it is not homogeneously scalable, i.e., $\|a\mathbf{x}\| \neq |a| \, \|\mathbf{x}\|$. It is also sometimes termed the ℓ_0-"norm" [6]. Neither is ℓ_p, for $0 < p < 1$ a proper norm.

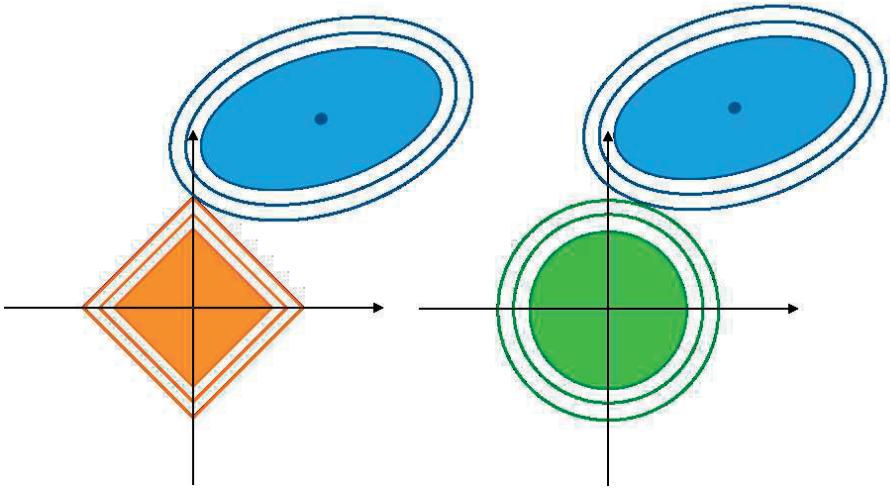


Figure 2: A comparison between the ℓ_1- and ℓ_2-norm constrained optimization problems on the left and right, respectively. The level curves in the center of the coordinate systems illustrate the regularizers, while the ellipses illustrate a smooth loss function.

regularizers promote sparse estimates and why others do not. In the next section, this is mathematically justified for a relevant selection of regularizers. All these have in common being convex, as for convex problems, there exists formal necessary and sufficient conditions for a solution to be optimal. These are termed the Karush-Kuhn-Tucker (KKT) conditions, which are easy to verify for most problems. Consider a constrained optimization problem

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) && (13)\\ \text{subject to} \quad & g(\mathbf{x}) \leq 0 && (14)\\ & h(\mathbf{x}) = 0 && (15) \end{aligned}$$

where the convex inequality constraints, $g(\cdot)$, and the linear equality constraints, $h(\cdot)$, are imposed on the convex loss function, $f(\cdot)$. For this problem, the Lagrangian is

$$\mathcal{L}(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \lambda g(\mathbf{x}) + \mu h(\mathbf{x}) \tag{16}$$

where $\lambda > 0$ and $\mu \in \mathcal{C}$ are the Lagrange multipliers. This convex problem has a unique minima, and $(\mathbf{x}, \lambda, \mu)$ is an optimal point for that minima if the KKT conditions are met. These are

$$\frac{\partial \mathcal{L}}{\partial \mathbf{x}} = \mathbf{0} \tag{17}$$

$$g(\mathbf{x}) \leq 0, \quad h(\mathbf{x}) = 0, \quad \mu > 0 \tag{18}$$

$$\lambda g(\mathbf{x}) = 0 \tag{19}$$

i.e., the optimal point is a stationary point of the Lagrangian, the solution is primal and dual feasible, and complementary slackness holds, respectively. The first two conditions mean that $\mathbf{x}$ is optimal only if it both minimizes the loss function and is a point in the feasible set, i.e., a point fulfilling the constraints. The last condition, complementary slackness, is more involved. It states that if the optimal point is in the interior of the feasible set, i.e., $h(\mathbf{x}) < 0$, then λ must be equal to zero. This implies that $h(\mathbf{x})$ vanishes from the Lagrangian, and the optimal point $\mathbf{x}$ only minimizes the loss function together with the equality constraint. The inequality constraint is thus only active for points on the boundary of the feasible set. As the equality constraint must always be active, it offers no complimentary slackness. For unconstrained problems, these conditions reduce to the first one, and the Lagrangian reduce to the loss function, for which the optimal point is a stationary point. The KKT conditions are often utilized to form numerical or (for simple problems) analytical solvers, some of which will be presented in later sections.

1.4 Complex-valued data

The outline for regularized optimization defined above describes real-valued functions taking complex-valued arguments. Most literature describing such problem typically operate in the domain of real-valued numbers. Due to the applications described in this thesis, it is natural to consider complex-valued parameters, for which some remarks are due.

Remark 1. Consider the example of $g(\mathbf{x}) = \|\mathbf{x}\|_1$ for complex-valued parameters. The regularizer is equivalent to

$$\sum_{m=1}^{M} |x_m| = \sum_{m=1}^{M} \left\| \begin{bmatrix} \operatorname{Re}(x_m) & \operatorname{Im}(x_m) \end{bmatrix}^\top \right\|_2 \tag{20}$$

i.e., a sum of the ℓ_2-norm for the real and imaginary part of each complex valued element in $\mathbf{x}$. It is worth noting that the sum of ℓ_2-norms is another common regularizer, which is central for this thesis and will be discussed at length in the next section. So, by stacking the real and imaginary parts of the parameters next to each other, modifying the loss function accordingly, and then adding the regularizer above, one obtains a real-valued function which takes real-valued arguments. The optimization problem is thus converted into $f(\mathbf{x})+\lambda g(\mathbf{x}) : \mathbb{R}^{2M} \mapsto \mathbb{R}$, which is be possible, however notationally tedious, for most problems described herein.

Remark 2. The common approach when solving the regularized optimization problems is to, at some point, form partial derivatives with respect to the complex-valued arguments. To that end, one may use Wirtinger derivatives, which permits a differential calculus much similar to the ordinary differential calculus for real-valued variables. Specifically, for the functions used herein, the complex derivative of $\mathbf{x}$ is formed by taking the ordinary derivative of $\mathbf{x}^H$, as if it was its own variable. Thus, for example, the derivative of a quadratic form becomes

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^H \mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{x} \tag{21}$$

For the works herein, depending on the implementation used, either one of these two approaches has been used when deriving solvers for the considered optimization problems.

2 Regularized optimization

Depending on which sparsity structure that is sought for a particular data model, one may use different regularizers to promote such structure. In this section, some commonly occurring regularizers will be introduced. For most of these, closed form solutions are derived using KKT, as it may give a qualitative understanding of the effect of regularization, as well as the effect of the hyperparameter λ. The problems introduced here are convex, which means that any numerical solver that is shown to converge will at some point do so for these problems. This furthermore means that if a particular iterative solver is used, the path it takes towards convergence, and the speed at which it reaches it, may differ from another converging solver, but in the end both will converge to the same point. These arguments justify the outline of this section, wherein a number of common sparsity-promoting regularized optimization problems are introduced. To mathematically illustrate how these problems promote sparse parameter solutions, the closed-form expressions for a cyclic coordinate descent (CCD) solver are presented. As the CCD will converge (although typically slow), the sparsifying effect it will illustrate will also be true for any other solver applied. See also Section (3.1) for an overview of the algorithm.

2.1 The underdetermined regression problem

As illustrated for the linear regression problem in the previous section, when the number of observations are far fewer than the number of modeling parameters, the system is underdetermined and the Moore-Penrose pseudoinverse does not have a closed-form expression. In this subsection, a standard approach for circumventing this issue is examined. As mentioned in Section 1.1, linear regression is the (unregularized) optimization problem where the loss function is equal to the ℓ_2-norm of the residual vector, i.e., the ordinary least squares (OLS) problem,

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 \tag{22}$$

which has solution[4] (4). For an underdetermined system, $\mathbf{A}^H\mathbf{A}$ has dimensionality $M \times M$ while only being rank $N < M$, and is therefore not invertible. The Tikhonov Regularization (TR), also known as ridge regression, is a common

[4] Obtained by solving the normal equations, i.e., taking the derivative of the loss function and setting it equal to zero.

method for solving such ill-posed problem; it is the regularized regression problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2+\gamma\|\mathbf{x}\|_2^2 \tag{23}$$

which has the closed-form solution

$$\hat{\mathbf{x}}=(\mathbf{A}^H\mathbf{A}+\gamma\mathbf{I})^{-1}\mathbf{A}^H\mathbf{y} \tag{24}$$

and always exists for a hyperparameter parameter $\gamma>0$. To examine the effects of this regularizer, consider the coordinate descent approach, where one optimizes one parameter at a time, while keeping the others fixed. To solve using KKT, and as (23) has no constraints, one only needs to set the loss function's derivative with respect to x_m equal to zero, yielding

$$-\mathbf{a}_m(\mathbf{y}-\mathbf{A}\mathbf{x})+x_m=0 \quad \Rightarrow \quad \hat{x}_m=\frac{\mathbf{a}_m^H\mathbf{r}_m}{\mathbf{a}_m^H\mathbf{a}_m+\gamma} \tag{25}$$

where $\mathbf{a}_m$ denotes the m:th atom of the dictionary and where $\mathbf{r}_m=\mathbf{y}-\sum_{i\neq m}\mathbf{a}_i\hat{x}_i$ is the residual where the reconstruction effect of the other estimated parameters have been removed. The iterative result in (24) has the following effects on the solution:

- For $\gamma=0$, the CCD solves the underdetermined OLS problem, but it will not converge to a unique solution.
- The denominator in (25) is always positive, and $\gamma>0$ shrinks $\hat{x}_m$ to have smaller magnitude than the OLS solution, thus leaving some of the explanatory potential in the dictionary atom to be utilized by another estimate.
- The explanatory capability of an atom in the dictionary depends on whether there exists linear dependence between the atom $\mathbf{a}_m$ and the data. As $N<M$, the atoms are not linearly independent and $\mathbf{a}_m^H\mathbf{a}_{m'}\neq 0$ for $m\neq m'$, i.e., there exists some redundancy in the dictionary such that a parameter may be replaced by another parameter.
- If the data has the form $\mathbf{y}=\mathbf{A}_{\mathcal{I}}\mathbf{x}_{\mathcal{I}}+\mathbf{e}$ for some subset of indices $\mathcal{I}$ in $\mathbf{A}$ and the TR problem is solved, there will in exist parameter estimates $\hat{x}_m\neq 0$, even though $m\notin\mathcal{I}$.
- TR estimates are not sparse, they are in fact the opposite, and are typically used to find smooth estimates for underdetermined problems.

2.2 Sparse regression: The LASSO

The classical approach to promote sparse estimates for a regression problem, using a statistical framework and convex analysis, was presented in the seminal work by Tibshirani et al. [3]. The method, termed the Least Absolute Shrinkage and Selection Operator (LASSO), solves the regularized optimization problem wherein the ℓ_2-norm loss function is paired with an ℓ_1-norm regularizer, i.e.,

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \tag{26}$$

The same optimization problem goes under different acronyms, and is also referred to as the Basis Pursuit De-Noising (BPDN) method [9]. It has been the constant focal point of much research during the last decades, and many prominent researchers have worked on the theoretical properties, solvers, applications, and extensions of the method. To illustrate the sparsifying effect of the LASSO, a coordinate-wise optimization scheme is derived, where for the m:th parameter, one wishes to solve

$$\underset{x_m}{\text{minimize}} \quad \|\mathbf{r}_m - \mathbf{a}_m x_m\|_2^2 + \lambda |x_m| \tag{27}$$

where $\mathbf{r}_m = \mathbf{y} - \sum_{i \neq m} \mathbf{a}_i \hat{x}_i$ is the residual where the reconstruction effect of the other estimated parameters have been removed. Examining (27), one may initially note that the regularizer is non-differentiable for $x_m = 0$. Using sub-gradient analysis, the KKT conditions for this unconstrained problem state that [10]

$$-\mathbf{a}_m^H (\mathbf{r}_m - \mathbf{a}_m x_m) + \lambda u_m = 0 \tag{28}$$

$$u_m = \begin{cases} \frac{x_m}{|x_m|} & x_m \neq 0 \\ \in [-1, 1] & x_m = 0 \end{cases} \tag{29}$$

where u_m is the m:th sub-gradient of the non-differentiable regularizer $\|\mathbf{x}\|_1$. Proceeding, consider the case $x_m \neq 0$ for which

$$\frac{x_m}{|x_m|} \left(\mathbf{a}_m^H \mathbf{a}_m |x_m| + \lambda \right) = \mathbf{a}_m^H \mathbf{r}_m \tag{30}$$

Applying the absolute value on both sides and solving for $|x_m|$ yields

$$|x_m| = \frac{\left| \mathbf{a}_m^H \mathbf{r}_m \right| - \lambda}{\mathbf{a}_m^H \mathbf{a}_m} \tag{31}$$

which inserted into (30) yields

$$x_m = \frac{\mathbf{a}_m^H \mathbf{r}_m}{|\mathbf{a}_m^H \mathbf{r}_m|} \frac{\left|\mathbf{a}_m^H \mathbf{r}_m\right| - \lambda}{\mathbf{a}_m^H \mathbf{a}_m} \tag{32}$$

Next, consider the case $x_m = 0$, which, using (29), results in the condition

$$\lambda u_m = \mathbf{a}_m^H \mathbf{r}_m \quad \Rightarrow \quad \left|\mathbf{a}_m^H \mathbf{r}_m\right| \leq \lambda \tag{33}$$

for the magnitude of the inner product between the dictionary and the residual, which, when combined with (32) yields the LASSO estimate

$$\hat{x}_m = \frac{\mathcal{S}\left(\mathbf{a}_m^H \mathbf{r}_m, \lambda\right)}{\mathbf{a}_m^H \mathbf{a}_m} \tag{34}$$

where

$$\mathcal{S}(z, \mu) = z/|z| \ \max(0, |z| - \mu) \tag{35}$$

is a shrinkage operator which reduces the magnitude of z by μ towards zero. The closed-form expression in (34) fulfills the KKT conditions and, when solved iteratively $\forall m$, yields the global optimum of (26). The solution also shows how the LASSO promotes sparsity. Just as with TR, all parameter estimates gets smaller magnitude than the unconstrained OLS would (compare with (25) where $\gamma = 0$). However, while the TR estimate is shrunk proportionally to the OLS estimate, the magnitude of the LASSO estimate is shrunk absolutely, which has the effect that when λ is large enough, that parameter estimate is completely zeroed out.

In some cases, it may be beneficial to replace the ℓ_2-norm in the LASSO's loss function with an ℓ_1-norm. Loosely laid out, an ℓ_1-norm will penalize the deviation in reconstruction fit less than the ℓ_2-norm for large deviations, and will thus be more lenient towards outlier samples. To that end, the Least Absolute Deviation (LAD) LASSO [11] is sometimes used, which solves the convex program

$$\underset{\mathbf{x}}{\text{minimize}} \ \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_1 + \lambda ||\mathbf{x}||_1 \tag{36}$$

However, producing an analytical coordinate-wise solution for the LAD-LASSO similar to the LASSO is not straight-forward. Instead, it will be shown in Paper D that the LAD-LASSO is equivalent to a particular covariance fitting problem, where the covariance matrix is parametrized using a heteroscedastic noise model, i.e., where the noise samples are allowed different variability.

2.3 Fused LASSO

A common variation of the LASSO, introduced in [12], is called the generalized LASSO, which use a regularizer on the form

$$g(\mathbf{x}) = \lambda||\mathbf{F}\mathbf{x}||_1 \tag{37}$$

where $\mathbf{F}$ is a linear transformation matrix, such that the ℓ_1-norm is imposed on a linear combination of the components in $\mathbf{x}$. A popular choice of $\mathbf{F}$ is the first-order difference matrix, defined as

$$\mathbf{F} = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & -1 \end{bmatrix} \tag{38}$$

which has dimension $(M-1)\times M$ and regularizes the absolute differences between adjacent parameters. This reguarlizer is often termed a Total Variation (TV) penalty, as it seeks to minimize the variation among parameters, often used for denoising images by removing spurious artifacts . To see this, consider a simplified solver where one does a change of variables, $\mathbf{z} = \mathbf{F}\mathbf{x}$, which yields the equivalent optimization problem

$$\underset{\mathbf{z}}{\text{minimize}} \ \ \|\mathbf{y} - \mathbf{B}\mathbf{z}\|_2^2 + \lambda||\mathbf{z}||_1 \tag{39}$$

where, for the dictionary, $\mathbf{B}$, $\mathbf{B}\mathbf{F} = \mathbf{A}$ is assumed to exist. The generalized LASSO is thus expressed in the standard LASSO form, where, from (34), sparsity in $\mathbf{z}$ is promoted. In terms of $\mathbf{x}$, as $\mathbf{z} = \mathbf{F}\mathbf{x}$ is underdetermined, there is no unique solution for $\hat{\mathbf{x}}$ given $\hat{\mathbf{z}}$. Parametrizing the solution by $\hat{x}_1 = u$, one obtains

$$\hat{x}_m = \hat{x}_{m-1} + \hat{z}_i, \quad m = 2, \dots, M \tag{40}$$

This implies that the parameter $\mathbf{x}$ can be seen as a sparse jump process; starting at u, the process evolves by taking its previous value, until a non-zero $\hat{z}_m$ comes along and adjusts $\hat{x}_m$ by this value. As the regularizer zeroes out insignificant jumps, the TV penalty ensures that the estimates are smooth; only to change when a significant saving in the loss function is gained by changing the parameter value. In practice, the generalized LASSO is solved for $\mathbf{x}$ directly, instead of $\mathbf{z}$

and u (see [12]) but (40) serves to illustrate the mechanics of the regularizer. As shown, the TV penalty does not promote sparse, but rather smooth, solutions. Therefore, TV may be used in tandem with the standard ℓ_1-norm, i.e.,

$$g = (1-\mu)\left\|\mathbf{x}\right\|_1 + \mu\left\|\mathbf{F}\mathbf{x}\right\|_1 \tag{41}$$

where $\mu \in [0,1]$ is a user-selected trade off parameter. The method is called the sparse fused LASSO (SFL), introduced in [13], and bestowes a grouping effect on the solution. If adjacent dictionary components have similar energy, they are fused into groups without a pre-defined structure. Simultaneously, if components are too weak, they are regularized to zero. Thus, SFL enforces both grouping and sparsity.

2.4 Elastic net regularization

In [14], a regularized regression problem is introduced which combines the ℓ_1- and ℓ_2-norm regularizers. It is called the elastic net and solves the problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad \left\|\mathbf{y}-\mathbf{A}\mathbf{x}\right\|_2^2 + \lambda_1\left\|\mathbf{x}\right\|_1 + \lambda_2\left\|\mathbf{x}\right\|_2^2 \tag{42}$$

When combining regularizers, the method imbibes some of the properties from both regularizers into the solution. As a combination of the LASSO and ridge regression, the elastic net promotes solutions which are, rather unintuitively, both sparse and smooth. The intuition for this combinations is that, in extreme cases of $M \gg N$, the atoms tend to have high degree of linear dependence (or coherence) i.e.,

$$\frac{\mathbf{a}_m^H\mathbf{a}_{m'}}{\sqrt{\mathbf{a}_m^H\mathbf{a}_m}\sqrt{\mathbf{a}_{m'}^H\mathbf{a}_{m'}}} \tag{43}$$

for two atoms m and m', and for certain dictionary designs, the linear dependence may be even further exaggerated. The LASSO then tends to only select one or a few of the coherent atoms, instead of all. Also, if N is very small, and the number of components which should be present in the solution, say K, approaches or surpasses the number of observations, the LASSO also tends to underestimate the model order. The elastic net therefore serves to smooth the LASSO solution somewhat, so that collinear dictionary atoms which are excluded from the LASSO

estimate get caught in the elastic net. Mathematically, this can be seen by initializing a coordinate descent solver. Similar to (28), the KKT conditions for the m:th parameter subproblem are

$$-\mathbf{a}_m^H(\mathbf{r}_m - \mathbf{a}_m x_m) + \lambda_1 u_m + \lambda_2 x_m = 0 \tag{44}$$

$$u_m = \begin{cases} \frac{x_m}{|x_m|} & x_m \neq 0 \\ \in [-1, 1] & x_m = 0 \end{cases} \tag{45}$$

where u_m is the sub-gradient of $|x_m|$. Solving for the two cases $x_m \neq 0$ and $x_m = 0$ separately, one obtains after some algebraic manipulation the closed-form solution

$$\hat{x}_m = \frac{\mathcal{S}\left(\mathbf{a}_m^H \mathbf{r}_m, \lambda_1\right)}{\mathbf{a}_m^H \mathbf{a}_m + \lambda_2} \tag{46}$$

which, similar to TR, reduces the magnitude of the estimate further than the LASSO estimate, giving the opportunity for coherent atoms to capture the remaining variability in the data.

2.5 Group-LASSO

This section introduces a method which is at the centre of this thesis, introduced in [15], in which sparsity is promoted among groups of dictionary atoms. By structuring the M atoms of the dictionary in K groups of L_k atoms each, such that

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \dots & \mathbf{A}_K \end{bmatrix} \tag{47}$$

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{a}_{k,1} & \dots & \mathbf{a}_{k,L_p} \end{bmatrix} \tag{48}$$

the group-LASSO solves the problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_{k=1}^{K} \sqrt{L_k} ||\mathbf{x}_k||_2 \tag{49}$$

For the LASSO, the ℓ_1-norm penalizes components based on their magnitudes, and similarly, the group-LASSO penalizes entire groups based on their magntiudes, quantified by the ℓ_2-norms of the parameter vectors. The effect is that

sparsity is promoted among the candidate groups, but not within them. To illustrate this mathematically, consider again a coordinate-wise approach, where estimates are sought for all parameters in a group, by solving

$$\underset{\mathbf{x}_k}{\text{minimize}} \quad \|\mathbf{r}_k - \mathbf{A}_k\mathbf{x}_k\|_2^2 + \lambda\sqrt{L_k}||\mathbf{x}_k||_2 \tag{50}$$

which is similar to the TR problem, except for the $(\cdot)^2$ in the regularizer. As will be shown, this difference has a substantial impact on the estimate. In (50), the regularizer is non-differentiable for $\mathbf{x}_k = \mathbf{0}$, and the KKT conditions for this unconstrained problem become

$$-\mathbf{A}_k^H(\mathbf{r}_k - \mathbf{A}_k\mathbf{x}_k) + \lambda\sqrt{L_k}\mathbf{u}_k = 0 \tag{51}$$

$$\mathbf{u}_k = \begin{cases} \frac{\mathbf{x}_k}{\|\mathbf{x}_k\|_2} & \mathbf{x}_k \neq \mathbf{0} \\ \in \{\mathbf{u}_k : \|\mathbf{u}_k\| \leq 1\} & \mathbf{x}_k = \mathbf{0} \end{cases} \tag{52}$$

which, similar to the LASSO, will be solved for the two cases in (52) separately. For $x_{k,\ell} \neq 0$, for any ℓ, one obtains

$$\left(\mathbf{A}_k^H\mathbf{A}_k\|\mathbf{x}_k\|_2 + \lambda\sqrt{L_k}\mathbf{I}\right)\frac{\mathbf{x}_k}{\|\mathbf{x}_k\|_2} = \mathbf{A}_k^H\mathbf{r}_k \tag{53}$$

where the approach is to solve for $\|\mathbf{x}_k\|_2$ and then insert the solution back into (53). While this equation could be solved numerically, in order to obtain a closed-form analytical expression, an assumption must be made. The dictionary group $\mathbf{A}_k$ has dimensions $N \times L_k$, which is typically a tall matrix (having more rows than columns). If assuming that the dictionary atoms are normalized, i.e., $\mathbf{a}_{k,\ell}^H\mathbf{a}_{k,\ell} = 1, \forall\ell$, and furthermore assuming that the atoms within each group are linearly independent, i.e., $\mathbf{A}_k^H\mathbf{A}_k = \mathbf{I}$, one obtains

$$\|\mathbf{x}_k\|_2 = \left\|\mathbf{A}_k^H\mathbf{r}_k\right\|_2 - \lambda\sqrt{L_k} \tag{54}$$

which plugged back into (53) yields

$$\mathbf{x}_k = \frac{\mathbf{A}_k^H\mathbf{r}_k}{\left\|\mathbf{A}_k^H\mathbf{r}_k\right\|_2}\left\|\mathbf{A}_k^H\mathbf{r}_k\right\|_2 - \lambda\sqrt{L_k} \tag{55}$$

Next, for the case when $x_{k,\ell} = 0$, for any ℓ, one obtains

$$\lambda\sqrt{L_k}\mathbf{u}_k = \mathbf{A}_k^H\mathbf{r}_k \quad \Rightarrow \quad \left\|\mathbf{A}_k^H\mathbf{r}_k\right\|_2 \leq \lambda\sqrt{L_k} \tag{56}$$

which, when combined with (55) yields the group-LASSO estimate, yields the closed-form solution

$$\hat{x}_m = \mathcal{T}\left(\mathbf{A}_k^H \mathbf{r}_k, \lambda\sqrt{L_k}\right) \tag{57}$$

where

$$\mathcal{T}(\mathbf{z}, \mu) = \mathbf{z}/\left\|\mathbf{z}\right\|_2 \max(0, \left\|\mathbf{z}\right\|_2 - \mu) \tag{58}$$

is an element-wise shrinkage function which reduces the magnitude of each parameter in the group proportionally to $\lambda\sqrt{L_k}$. From (57), one may see how group-sparse solutions is achieved; when the contribution from a candidate group is too small, i.e., $\left\|\mathbf{A}_k^H \mathbf{r}_k\right\|_2 \leq \lambda\sqrt{L_k}$, all the estimates in a group become zero, and similarly, when the inclusion of a candidate group may contribute enough explanatory power, the parameter estimates become non-zero. It should be noted that the assumption of linear independence within groups, made in order to obtain (54), is typically not very restrictive; for most cases, $L_k < N$ and if two atoms within a group become highly linearly dependent, one may consider pruning that group in order to remove such correlations. After all, the purpose of the group-LASSO is to make selection among groups, and not within groups.

Although, as noted in subsection 2.4, one may use a combination a regularizers in order to promote a specific sparsity structure, and a number of such combinations are introduced later in the thesis. In general, they solve convex optimization problems on the form

$$\underset{\mathbf{a}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_{j=1}^{J} g_j(\mathbf{x}, \mu_j) \tag{59}$$

where g_j denotes the j:th regularizer which promotes a certain sparsity structure, and with $\lambda_j\mu_j$ denoting its corresponding regularization level, which weighs the importance between the sparsity promoted by g_j and the model fit. In [16], Simon et al. introduce the sparse group-LASSO (SGL), which is a group-sparse method where sparsity is also introduced within groups. This is achieved by combining the regularizer in the group-LASSO with an ℓ_1-norm, i.e.,

$$g_1 + g_2 = \mu \left\|\mathbf{x}\right\|_1 + (1 - \mu) \sum_{k=1}^{K} \sqrt{L_k} \left\|\mathbf{x}_k\right\|_2 \tag{60}$$

for $0 \leq \mu \leq 1$. A closed-form expression for the group-wise optimization problem using this regularizer is not obtainable. However, using sub-gradient analysis similar to the one in (51) - (52), one may discern its sparsity patterns. Using algebraic manipulations, $\mathbf{x}_k = \mathbf{0}$ implies that

$$\left\| \left[\begin{array}{ccc} \mathcal{S}\left(\mathbf{a}_{k,\ell}^H \mathbf{r}_k, \lambda\mu\right) & \ldots & \mathcal{S}\left(\mathbf{a}_{k,\ell}^H \mathbf{r}_k, \lambda\mu\right) \end{array} \right]^\top \right\|_2 \leq (1-\mu)\lambda\sqrt{L_k} \qquad (61)$$

where each element in the right hand side vector is similar to the regular LASSO estimate, and the group-LASSO sets the entire group to zero if the ℓ_2-norm of these estimates is too small. For a component within a group, one similarly has

$$|\mathbf{a}_{k,\ell}^H \mathbf{r}_{k,\ell}| \leq \mu\lambda \qquad (62)$$

for $x_{k,\ell}$,where $\mathbf{r}_{k,\ell} = \mathbf{y} - \sum_{(k,i)\neq(k,\ell)} \mathbf{a}_{k,i}\hat{x}_{k,i}$ is the residual where the reconstruction of all other groups, as well as all other atoms within the current group, has been removed, implying that some form of CCD approach should also be used within the groups. Examining (61) and (62), it becomes clear that the SPL have two constraints on the parameters; that each individual parameter significantly improves the residual fit, and that each group significantly improves the residual fit, both of which must be fulfilled for the parameter estimate to become non-zero. In lack of closed-form expressions for solving the SPL, there are several numerical methods, some of which are introduced in section 3.

2.6 Regularization and model order selection

So far, little has been said about how to choose the regularization parameter(s) in sparse regression. As illustrated, these hyperparameters control the trade-off between reconstruction fit and sparsity, such that, e.g., for the LASSO,

$$\lambda \geq \mathbf{a}_m^H \mathbf{y} \geq \mathbf{a}_m^H \mathbf{r}_m \quad \Rightarrow \quad \hat{x}_m = 0 \tag{63}$$

i.e., setting that particular estimate to zero. The regularization parameter can thus be seen as an implicit model order selection; not one where an exact model order is selected, but as a minimum requirement on the linear dependence between the dictionary atom and the data. For notational simplicity in the following quantitative analysis, let's, without loss of generality, assume that the dictionary has standardized atoms, i.e., $\mathbf{a}_m^H \mathbf{a}_m = 1, \forall m$. Consider an observation $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where the parameter vector $\mathbf{x}$ is said to have support

$$\mathcal{I} = \{ i : x_i \neq 0 \} \tag{64}$$

i.e., a set of indices indicating the locations of the dictionary atoms included in the data. such that $\mathbf{A}\mathbf{x} = \mathbf{A}_{\mathcal{I}}\mathbf{x}_{\mathcal{I}}$. Moreover, let $|\mathcal{I}| = \|\mathbf{x}\|_0 = C$ be the size of the support, i.e., number of non-zero elements in $\mathbf{x}$; $\mathbf{x}$ is then said to be C/M-sparse. Then, consider a parameter $x_m \in \mathcal{I}$ which is up for estimation. The inner product between the m:th atom and its residual may be expressed as

$$\mathbf{a}_m^H \mathbf{r}_m = \mathbf{a}_m^H \left(\mathbf{a}_m x_m + \sum_{m' \neq m} \mathbf{a}_{m'} \left(x_{m'} - \hat{x}_{m'} \right) + \mathbf{e} \right) \approx x_m + \mathbf{a}_m^H \mathbf{e} \tag{65}$$

if assuming that the coherence between dictionary atoms in the support, $\mathbf{a}_m^H \mathbf{a}_{m'}$, $m, m' \in \mathcal{I}$, is negligable. It then follows that the estimate of x_m will be zero unless

$$\lambda < |x_m + \mathbf{a}_m^H \mathbf{e}| \leq |x_m| + |\mathbf{a}_m^H \mathbf{e}| \tag{66}$$

where the triangle inequality has been used in the last inequality. One may conclude that the regularization parameter operates in relation to the magnitude of the true parameters. An important consequence of this is related to model order estimation; the LASSO discriminates the estimated support based on magnitude, a larger parameter is always added before a smaller. Thus, if there are two components $m \in \mathcal{I}$ and $m' \notin \mathcal{I}$, but $|x_m| < |x_{m'}|$ due to noise artefacts, then one

can never obtain a LASSO solution where $\hat{x}_m$ is non-zero and $\hat{x}_{m'}$ is zero, i.e., it is impossible to recover the true support using the LASSO. This, however, is not an issue only restricted to sparse estimation methods. In order for support recovery to be possible,

$$\min_{m \in \mathcal{I}} |x_m| > \max_{m' \notin \mathcal{I}} |x_{m'}| \tag{67}$$

must be true, which is also typically the case for the chosen sparse encoding $\mathbf{A}$. The regularization parameter is always positive, but one must typically consider a narrower interval to obtain useful solutions. Let $\hat{\mathbf{x}}(\lambda)$ denote the LASSO solution as a function of regularization level. Starting from very high levels of λ, the cost of adding a non-zero parameter to the estimated support is much higher than its reduction in residual ℓ_2-norm, and $\hat{\mathbf{x}}(\lambda) \to \mathbf{0}$ as $\lambda \to \infty$. At some point, say λ_0, the first non-zero estimate enters the solution, at

$$\lambda_0 = \max_m |\mathbf{a}_m^H \mathbf{y}| \tag{68}$$

whereafter, when decreasing λ, more and more non-zero parameters are added until, as $\lambda \to 0$, the LASSO approaches the (generally ill-posed) least squares problem. Thus, let

$$\Lambda = \{ \lambda : \lambda \in (0, \lambda_{\max}] \} \tag{69}$$

denote the regularization path for which a non-zero solution path $\mathbf{x}(\Lambda)$ exists, on which an appropriate point is sought. Before proceeding, one may note how the function $\hat{\mathbf{x}}(\lambda)$ behaves. To that end, let λ^* be a regularization level where the estimated support $\hat{\mathcal{I}}$ is equal to the true support, and where for some small $\delta \in \mathbb{R}$, $\hat{\mathcal{I}}(\lambda^* + \delta) = \hat{\mathcal{I}}(\lambda^*)$. If furthermore the dictionary atoms in the true support are linearly independent, one obtains the LASSO solution (using (34))

$$\hat{x}_m(\lambda^* + \delta) = (x_m + \mathbf{a}_m^H \mathbf{e}) \left(1 - (\lambda^* + \delta) |x_m + \mathbf{a}_m^H \mathbf{e}|\right) \tag{70}$$

which is an affine function of δ that has a constant negative slope. The magnitudes of the LASSO estimates are thus reduced towards zero by a constant rate as λ increases.

Next, the concept of coherence, or collinearity between dictionary atoms is discussed. Let

$$\rho(m, m') = \mathbf{a}_m^H \mathbf{a}_{m'} \tag{71}$$

denote the linear dependence between two dictionary atoms. As the atoms are assumed to be standardized, i.e., $\rho(m,m) = 1$, then $|\rho(m,m')| \leq 1$. To see how a non-zero coherence affects the LASSO estimate, consider a much simplified one-component observation $\mathbf{y} = \mathbf{a}_m x_m + \mathbf{e}$ and one coherent noise component, $m' \notin \mathcal{I}$ where $\rho(m,m') = \rho < 1$, and no other coherence. First, one may note that including x_m is expected to be cheaper than including $x_{m'}$ in the optimization problem. To see this, consider the two options $\hat{x}_m = x_0$ and $\hat{x}_{m'} = x_0$ for some value x_0, all other parameters being equal. Comparing the expected optimization cost of these two solutions, one obtains after some algebra

$$\begin{aligned} &\mathrm{E}\left(f(\hat{\mathbf{x}}_m) + \lambda g(\hat{\mathbf{x}}_m) - f(\hat{\mathbf{x}}_{m'}) + \lambda g(\hat{\mathbf{x}}_{m'})\right) && (72)\\ =\ &-2\mathrm{E}\left(|x_0|^2(1-\rho) + \mathbf{a}_m^H\mathbf{e} - \mathbf{a}_{m'}^H\mathbf{e}\right) && (73)\\ =\ &-2|x_0|^2(1-\rho) < 0 && (74)\end{aligned}$$

which is always negative $\forall x_0$, as $\mathbf{e}$ is assumed to be a zero mean. Thus, assigning power to the correct atom is always preferable to assigning it to another atom to which it is coherent with $\rho < 1$. This, however, does unfortunately not mean that spurious estimates does not enter into the LASSO solution. Due to the shrinkage effect, a parameter has a bias which makes that atom unable to exploit its full explanatory potential, leaving some data structure to be modeled by coherent atoms. To see how, consider a CCD approach starting at m, and where λ is set small enough as to include x_m into the support, with estimate $\hat{x}_m = (x_m + \mathbf{a}_m^H\mathbf{e})\left(1 - \lambda/|x_m + \mathbf{a}_m^H\mathbf{e}|\right)$. If then turning to parameter m', to which there is coherence as above, $x_{m'}$ is only excluded from the support if

$$\hat{x}_{m'} = 0 \Leftrightarrow \lambda \leq \left|\frac{\rho\, x_m \lambda}{|x_m + \mathbf{a}_m^H\mathbf{e}|} + \rho\, \mathbf{a}_m^H\mathbf{e}\left(1 - \frac{\lambda}{|x_m + \mathbf{a}_m^H\mathbf{e}|}\right) + \mathbf{a}_{m'}^H\mathbf{e}\right| \quad (75)$$

from which it is difficult to discern a more precise conclusion. The important factors are, however, the regularization level, the coherence, the noise level, and the signal-to-noise ratio. For instance, if m' is not coherent to m', i.e., $\rho = 0$, then (75) reduces to

$$\hat{x}_{m'} = 0 \Leftrightarrow \lambda \leq \left|\mathbf{a}_{m'}^H\mathbf{e}\right| \quad (76)$$

i.e., the regularization level must be selected higher than the noise level as to not include a spurious estimate. On the other hand, if there is no noise, $\mathbf{e} = \mathbf{0}$, then

(75) reduces to

$$\hat{x}_{m'} = 0 \Leftrightarrow \lambda < \left| \frac{\rho x_m \lambda}{|x_m|} \right| \Rightarrow \rho \geq 1 \tag{77}$$

i.e., $x_{m'}$ never enters the support for any $\rho < 1$, and thus one may conclude that it is the noise which introduces spurious estimates to the solution, if λ is set too low and the coherence is too high.

The literature on sparse regression also contains some methods for hyper-parameter-selection. The classical approach, as discussed in, e.g., [17], is the statistical cross-validation tool. It selects the regularization level, λ, which has the best prediction ℓ_2-fit. The exhaustive approach is leave-one-out cross validation, in which one calculates the path solution $x(\Lambda)$ for all observations except one, then calculate how well the estimate may be used to predict the excluded observation. This is done for the entire solution path, and then iterated by leaving out all observations in turn, one by one. A cost function is then obtained for each $\lambda \in \Lambda$, which is minimized in order to find the optimal regularization. Needless to say, this is a computationally burdensome approach. A batch version called R-fold cross-validation is often used, significantly speeding up the process, and a path solution is obtained for a discrete grid of candidate regularization levels. Still, the LASSO needs to be solved a large number of times in order to select the regularization level. A faster method of computing the solution path was proposed in [18], which, for real-valued signals, solves the entire solution path of $\hat{\mathbf{x}}(\lambda)$ with the same computational complexity as if solving for a single λ. Another approach is to use an information criteria, such as the Bayesian Information Criteria (BIC). The cross-validation and BIC method do not, however, make any guarantees in terms of support recovery. There are also a number of heuristic approaches to setting the regularization level. These often depend on the purpose of the estimation, as examplified in Section 1.2. If, for instance, λ is set too low, the solution is not sufficiently sparse, but will also not have any false exclusions of the true parameters. If the purpose is to estimate a non-linear parameter, obtaining a solution which is too dense might not be problematic. If one searches for a number of components with strong contribution to the signal, but some very small noise components are also included into the support, the main contributors will still be discernible. Also, if the main contributors in $\mathbf{x}$ are sought after, selecting λ too high might falsely exclude some of the smaller components in the data without decreasing the model fit too much. To that end, one may think of the solution in

terms of dynamic range. Thus, one may decide upon a dynamic range of δ (dB), such that the regularization becomes

$$\lambda = \lambda_0 \sqrt{10^{-\delta/10}} \tag{78}$$

which implies that the maximal dynamic range, i.e., difference in signal power between two components in the support, is $|\delta|$ dB. For example, for $\delta = 20$ dB, this yields $\lambda = \lambda_0 0.1$.

When utilizing more than one regularizer, selecting the level of regularization becomes more complex. Not only does the total regularization level need to balance the model fit, but each of the regularizers also needs to be weighed against each other, as to find the sought sparsity pattern. With J regularizers, a path solution generalizes to a J-dimensional regularization path, making cross-validation and information criteria methods computational burdensome.

2.7 Scaled LASSO

To make selection of the regularization parameter simpler, an auxiliary variable $\sigma > 0$ may be included such that, using (26),

$$\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 \leq \frac{1}{\sigma} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + N\sigma + \mu \|\mathbf{x}\|_1 \tag{79}$$

for $\lambda = \mu\sigma$, and one may equivalently solve [19]

$$\underset{\mathbf{x},\sigma>0}{\text{minimize}} \quad \frac{1}{\sigma} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + N\sigma + \mu \|\mathbf{x}\|_1 \tag{80}$$

Using a coordinate descent approach where one solves (80) over $\mathbf{x}$ and σ, one sees how the mechanics of the regularization level changes. First, keeping $\mathbf{x}$ fixed, the unconstrained solution of (80) with respect to σ becomes

$$\hat{\sigma} = \frac{1}{\sqrt{N}} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \tag{81}$$

As this estimate is always non-negative, the constraint $\sigma > 0$ is never a hard constraint. Inserting (81) back into (80), the optimization problem becomes

$$\underset{\mathbf{x}}{\text{minimize}} \quad 2 \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 + \frac{\mu}{N} \|\mathbf{x}\|_1 \tag{82}$$

which is also known as the square root LASSO [20]. Returning to (80), and by keeping σ fixed at $\hat{\sigma}$, the optimization problem becomes

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \mu\hat{\sigma}\,\|\mathbf{x}\|_1 \tag{83}$$

which is the standard LASSO formulation, with closed-form solution

$$\hat{x}_m = \frac{\mathcal{S}(\mathbf{a}_m^H\mathbf{r}_m, \mu\hat{\sigma})}{\mathbf{a}_m^H\mathbf{a}_m} \tag{84}$$

allowing the regularization parameter to be scaled by the estimate of σ, modeling the standard deviation of the noise. By selecting μ instead of λ, one may do so independently of the noise power. Assume the noise distribution to have expectation and variance

$$\mathrm{E}(\mathbf{e}) = \mathbf{0}, \quad \mathrm{V}(\mathbf{e}) = \sigma^2\mathbf{I} \tag{85}$$

and consequently, for the linear combination $\mathbf{A}^H\mathbf{e}$,

$$\mathrm{E}(\mathbf{A}^H\mathbf{e}) = \mathbf{0}, \quad \mathrm{V}(\mathbf{A}^H\mathbf{e}) = \sigma^2\mathbf{A}^H\mathbf{A} \tag{86}$$

For a noise component $m \notin \mathcal{I}$, where the coherence with other atoms may be neglected, will becomes non-zero zero if

$$\mu\hat{\sigma} < |\mathbf{a}_m^H\mathbf{y}| = |\mathbf{a}_m^H\mathbf{A}\mathbf{x} + \mathbf{e}| = |\mathbf{a}_m^H\mathbf{e}| \quad \Rightarrow \tag{87}$$

$$\mu^2\hat{\sigma}^2 < \mathbf{a}^H\mathbf{e}\mathbf{e}^H\mathbf{a} \tag{88}$$

Taking the expected value on both sides of (88) yields

$$\mu^2\hat{\sigma}^2 < \mathbf{a}^H\sigma^2\mathbf{I}\mathbf{a} = \sigma^2 \tag{89}$$

assuming that the bias in the LASSO estimate makes the estimated standard deviation larger than σ. Therefore, in order to set the noise component to zero, one must at least select $\mu > \hat{\sigma}/\sigma$.

2.8 Reweighted LASSO

For the noiseless observation vector in (1), not previously considered for estimation herein, one may obtain a sparse parameter estimate by solving the Basis Pursuit (BP) problem [21], i.e.,

$$\begin{aligned} &\underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{x}\|_1 \\ &\text{subject to} && \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{90}$$

It is worth noting that this optimization does not contain any regularization parameter, as it is not a regression problem where the model fit must be weighed against sparsity. Merely, as the data is noiseless, it finds the smallest ℓ_1-norm which perfectly reconstructs the data. To promote even sparser estimates than obtained by BP, the reweighted ℓ_1-minimization method was introduced in [22], which iteratively solves a weighted BP problem, i.e., for the j:th iteration,

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \sum_{m=1}^{M} \frac{|x_m|}{|\hat{x}_m^{(j-1)}| + \varepsilon} \\ \text{subject to} \quad & \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{91}$$

where $\hat{x}_m^{(j-1)}$ denotes the previous estimate of the m:th parameter, and where ε is a small positive constant used to avoid numerical instability. The parameters in $\mathbf{x}$ are thus iteratively weighted using the previous estimate, with the effect that small $|\mathbf{x}_m|$ are successively given a higher optimization cost, whereas the cost is successively lessened for large $|x_m|$. The iterative approach falls within the class of majorization-minimization (MM) algorithms (see, e.g., [23] for an overview), where some given objective function is minimized by iteratively minimizing a surrogate function which majorizes the objective function. Thus, consider the (non-convex) optimization problem

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & g(\mathbf{x}) = \sum_{m=1}^{M} \log(|x_m| + \varepsilon) \\ \text{subject to} \quad & \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{92}$$

which one wishes to solve via the MM approach. In the first step of this MM-algorithm, one lets $g(\mathbf{x})$ be majorized by its first-order Taylor approximation around $\mathbf{x} = \hat{\mathbf{x}}^{(j-1)}$, i.e.,

$$g(\mathbf{x}) \leq g\left(\hat{\mathbf{x}}^{(j-1)}\right) + \nabla g\left(\hat{\mathbf{x}}^{(j-1)}\right)^H \left(\mathbf{x} - \hat{\mathbf{x}}^{(j-1)}\right) \tag{93}$$

where ∇g denotes the gradient of g. Then, in the second step of the MM-algorithm, the majorizer is minimized for $\mathbf{x}$ in lieu of g, which becomes precisely the optimization problem in (91). The reweighted ℓ_1-minimization method can thus be seen as solving a series of convex problems approximating the (non-convex) logarithmic objective function. The effects of the iterative approach is

twofold; a logarithmic minimizer is both more sparsifying and gives a smaller parameter bias than the ℓ_1-minimizer, as could be seen in Figure 1 above. Using a similar analysis, the adaptive LASSO was introduced in [24] to approximate the use of a logarithmic regularizer in sparse regression, i.e., by iteratively solving

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_{m=1}^{M} \frac{|x_m|}{|\hat{x}_m^{(j-1)}| + \varepsilon} \tag{94}$$

where it is worth noting that the regularization parameter is once more included, as the optimization problems needs to select a trade-off level between model fit and sparsity. Thus, the reweighted LASSO problem can be seen to have individual regularization parameters for each variable, iteratively updated $\forall j$ as

$$\lambda_m^{(j)} = \frac{\lambda}{|\hat{x}_m^{(j-1)}| + \varepsilon} \tag{95}$$

growing for small variables, and shrinking for large variables. Typically, the iterations converge quickly, and around 5-20 iterations often suffices for most applications. The logarithmic regularizer is concave, and as a consequence, one cannot always expect to find a global optimum. It is therefore important to choose a suitable starting point. For instance, one may select λ slightly lower than for the standard LASSO, as spurious components are likely to dissapear, meanwhile increasing the chance of keeping the signal components.

3 Brief overview of numerical solvers

For the convex optimization problems described in the previous section, there exists a large number of numerical solvers. One solver, which uses the methodology of disciplined convex programming described in [25], comes with a software package, CVX [26], which makes implementation very approachable. CVX is an excellent tool for prototyping new optimization problems, such as regularizers in sparse regression. CVX makes use of commonly available interior point methods such as SeDuMi [27] and SDPT3 [28] to find solutions which approximately fulfill the KKT conditions for the stated problems. The CVX framework is designed for experimentation and toy examples; it is generally too computationally burdensome for practical estimation of the optimization problems considered in this thesis. The problems and scope of applications presented in the thesis calls for more efficient solvers, three of which will be briefly described herein: Coordinate descent algorithms typically suffers from slow convergence, however, for sparse parameter estimation, they may be utilized to reach coarse (but sufficient) convergence very efficiently [29], as is observed Paper D and F. When combining different regularizers, the alternating direction method of multipliers (ADMM) [30] is shown to provide efficient estimation, which is utilized in Paper A - C . Also, for recursive estimation scenarios, when new observations enters the estimator continuously, a proximal gradient approach may be implemented to reuse old computations and to avoid storing large matrices, as examined in Paper E.

3.1 Cyclic coordinate descent

For many of the methods presented earlier, coordinate-wise updates have been used to illustrate the effects of the different regularizers. In this section, a brief outline is given for the algorithm, including speed-ups. The CCD may be used to solve the optimization problems one parameter at a time, i.e., for all indices $i = 1, \ldots, M$, solving

$$\underset{x_i}{\text{minimize}} \quad f(x_i|\mathbf{x}_{-i}) + \lambda g(x_i|\mathbf{x}_{-i}) \tag{96}$$

while keeping the other parameters, denoted $\mathbf{x}_{-i}$, fixed. Typically, the parameters are cycled through in randomized order in each pass of the parameters, such that no parameter may benefit from being consequently estimated before another [31]. Moreover, a significant speed-up utilized in [29] is to focus iterations on the active set parameters, i.e., the (non-zero) parameters making up the estimated support.

This can be done by first doing a complete pass on all parameters, and then only iterate over the non-zero parameters, until convergence, whereafter another complete pass is done. If the active set then changes, the process is repeated, otherwise the estimation process is complete. An algorithm outline for CCD thus becomes

1. Initialize the solution with $\hat{\mathbf{x}}^{(0)} = \mathbf{0}$ and set an iteration counter $j = 0$.

2. Draw a random permutation order of the M parameter indices. Using this ordering, minimize the objective (96) and estimate each $\hat{x}_m^{(j+1)}$ in turn, while the other parameters are fixed at their most recent value. Increase the iteration counter by one, i.e., $j \leftarrow j + 1$.

3. Let $\hat{\mathcal{I}}$ denote the set of most recent non-zero parameter estimates, i.e., the active set.

4. Draw a random permutation order of the parameters in the active set and minimize the objective function (96) and estimate each $\hat{x}_m^{(j+1)}$ in turn, while the other parameters are fixed at their most recent value. Update $j \leftarrow j + 1$. Iterate this step until the solution of the active set converges to some accuracy.

5. Perform Step 2 and check whether the active set changes. If changed, redo Step 3 - Step 5.

6. Set $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{(j)}$ to finalize estimation.

The benefit of using CCD for sparse regression lies in the resulting sparsity of the estimates; as most parameters become zero, given a reasonable choice of λ, these are also likely not to change between iterations. Instead, by only iterating over the active set, and updating the set of all parameters infrequently, computational complexity may be drastically reduced. The complexity becomes at most $\mathcal{O}(M^2)$, but is, using the active set updates, greatly reduced for small active sets. Furthermore, as the LASSO estimate is biased, convergence in parameters is not essential. As the LASSO's main purpose often is to estimate the parameter support[5], convergence can therefore be set quite low.

[5] After determining the support, a non-biased parameter estimation can be done for the active set separately.

Regardless of the order in which the CCD is updated, convergence guarantees has been shown for objective functions consisting of a smooth and convex loss function and a possibly non-smooth but convex regularizer which is separable in the parameters [32]. The LASSO formulation is one such loss function (see also [17]).

3.2 The alternating direction method of multipliers

The ADMM is a Lagrangian-based approach which has gained popularity with sparse estimation due to its favorable properties for large-scale systems, (see [30], for an eloquent analysis). In general, ADMM solves problems of the form

$$\underset{\mathbf{z}}{\text{minimize}} \quad f(\mathbf{z}) + g(\mathbf{G}\mathbf{z}) \tag{97}$$

where $f(\cdot)$ and $g(\cdot)$ are closed, proper, and convex functions, and $\mathbf{G}$ is a known matrix. By introducing the new variable $\mathbf{u} = \mathbf{G}\mathbf{z}$, and adding this condition to the optimization problem, the ADMM approach is to iterate between solving for $\mathbf{z}$, while keeping $\mathbf{u}$ constant, and vice versa. The problem (97) may thus be equivalently expressed as [30]

$$\begin{aligned} &\underset{\mathbf{z}}{\text{minimize}} \quad f(\mathbf{z}) + g(\mathbf{u}) + \frac{\mu}{2}\,||\mathbf{G}\mathbf{z} - \mathbf{u}||_2^2 \\ &\text{subject to} \quad \mathbf{G}\mathbf{z} - \mathbf{u} = \mathbf{0} \end{aligned} \tag{98}$$

for any smoothing parameter μ, as the penalty term disappears when the constraint is fulfilled. To solve this convex program, the augmented Lagrangian for the scaled form of the ADMM [30, p. 15] is formed as

$$L_\mu(\mathbf{z}, \mathbf{u}, \mathbf{d}) = f(\mathbf{z}) + g(\mathbf{u}) + \frac{\mu}{2}||\mathbf{G}\mathbf{z} - \mathbf{u} + \mathbf{d}||_2^2 \tag{99}$$

where $\mathbf{d}$ denotes the scaled dual variable. At iteration $(j+1)$, the parameters are obtained by solving

$$\mathbf{z}^{(j+1)} = \arg\min_{\mathbf{z}} L_\mu(\mathbf{z}, \mathbf{u}^{(j)}, \mathbf{d}^{(j)}) \tag{100}$$

$$\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}} L_\mu(\mathbf{z}^{(j+1)}, \mathbf{u}, \mathbf{d}^{(j)}) \tag{101}$$

and then updating the scaled version dual variable as

$$\mathbf{d}^{(j+1)} = \mathbf{d}^{(j+1)} - \mu(\mathbf{G}\mathbf{z}^{(j+1)} - \mathbf{u}^{(j+1)}) \tag{102}$$

Clearly, using the ADMM optimization scheme is worthwhile when (100) and (101) are such that they may be carried out much easier than the original problem in (97). For the LASSO, this is precisely the case, as will be shown in the next section.

3.3 Solving the LASSO problem using ADMM

To solve the LASSO using an ADMM approach, consider an augmented optimization problem equivalent to the one in (26), i.e.,

$$\begin{aligned} &\underset{\mathbf{z},\mathbf{u}}{\text{minimize}} \quad \|\mathbf{y}-\mathbf{A}\mathbf{z}\|_2^2 + \lambda||\mathbf{u}||_1 + \mu\,\|\mathbf{z}-\mathbf{u}\|_2^2 \\ &\text{subject to} \quad \mathbf{z}-\mathbf{u}=\mathbf{0} \end{aligned} \tag{103}$$

to which the augmented Lagrangian for the scaled form ADMM may be expressed as

$$L_\mu(\mathbf{z},\mathbf{u},\mathbf{d}) = \|\mathbf{y}-\mathbf{A}\mathbf{z}\|_2^2 + \lambda||\mathbf{u}||_1 + \mu\,\|\mathbf{z}-\mathbf{u}+\mathbf{d}\|_2^2 \tag{104}$$

such that $\mathbf{d}$ denotes the scaled dual variable. To find the expressions which minimize (104) with respect to $\mathbf{z}$ and $\mathbf{u}$, similar to (100) and (101), one must differantiate the Langrangian, set the derivative to zero, and solve for the current variable at iteration $k+1$. For $\mathbf{z}$, this yields an expression similar to the TR estimate in (24), i.e.,

$$\mathbf{z}^{(j+1)} = \left(\mathbf{A}^H\mathbf{A}+\mu\mathbf{I}\right)^{-1}\left(\mathbf{A}^H\mathbf{y}+\mu\left(\mathbf{u}^{(j)}-\mathbf{d}^{(j)}\right)\right) \tag{105}$$

while for $\mathbf{u}$, the Lagrangian is non-differentiable due to the ℓ_1 penalty. However, notice that that the two terms which depend on $\mathbf{u}$ resembles a simplified version of the LASSO, where the parameters $\mathbf{u}_m,\ m=1,\ldots,M$ uncouple from each other, and may thus be estimated exactly with one cycle over the parameter vector, where each estimate of u_m is obtained by a simple thresholding operation, i.e.,

$$u_m^{(j+1)} = \mathcal{S}(x_m+d_m,\lambda/\mu) \tag{106}$$

Finally, the dual variable is updated as in (102), with $\mathbf{G}=\mathbf{I}$. The main computational cost occurs in (105), where the inversion requires $\mathcal{O}(M^3)$ operations, although it should be noted that this step can be computed offline. Thus, at each iteration, the estimation process incurs a cost which is at most $\mathcal{O}(M^2)$ operations, which can be reduced further by utilizing efficient matrix-vector multiplication speed-ups [33].

3.4 Proximal Gradient

This section deals with a first-order optimization method, utilizing gradients to do local optimization of the objective function in combination with a quadratic smoothness term. The proximal gradient solver is a special case of the projected gradient methods, where the objective function is a combination of a smooth and convex function and a non-differentiable and convex function, $f(\mathbf{x}) + \lambda g(\mathbf{x})$, which is often the case for the regularized regression problems in this thesis. Recall that the regularizer can be seen as a constraint for the optimization problem, of which the objective function is the Lagrange form. The main idea is then to, at the j:th iteration,

1. Take a gradient step $\mathbf{z} = \mathbf{x}^{(j)} - s^{(j)} \nabla f(\mathbf{x}^{(j)})$
2. Project the gradient step onto the solution set obeying the optimization constraint, by calculating a proximal map, i.e., $\mathbf{x}^{(j+1)} = \text{prox}_g(\mathbf{z})$

where $s^{(j)}$ denotes step length, ∇ the gradient, and where the proximal map is the projection operator

$$\text{prox}_g(\mathbf{z}) = \arg\min_{\mathbf{u}} \ \|\mathbf{z} - \mathbf{u}\|_2^2 + \lambda g(\mathbf{u}). \tag{107}$$

For the LASSO problem, the gradient step becomes

$$\mathbf{z} = s^{(j)} \left(\mathbf{A}^H \mathbf{y} - \left(\mathbf{A}^H \mathbf{A} - \frac{1}{s^{(j)}} \mathbf{I} \right) \mathbf{x}^{(j)} \right) \tag{108}$$

and the proximal map becomes the threshold operator, i.e.,

$$\mathbf{x}_m^{(j+1)} = \mathcal{S}(z_m, s^{(j)} \lambda) \tag{109}$$

for all parameter indices $m = 1, \ldots, M$ (see, e.g., [17] for more details). The main computational cost of the proximal gradient method is incurred when taking the gradient step, where the matrix-vector multiplication in (108) has complexity $\mathcal{O}(M^3)$ operations, which can be reduced to $\mathcal{O}(M^2)$ by computing the matrix inner-product offline. For online-estimation, i.e., when new observations are aquired on a running basis, the dictionary also changes, for which the proximal gradient method enables cheap updating steps where the lower computational complexity is kept. This is shown in Paper E.

4 Introduction to selected applications

This section introduces some preliminaries for the applications discussed in the thesis; spectral estimation, audio processing, and array processing.

4.1 Spectral analysis

For many applications, a periodic signal of interest may often be well described by the sinusoidal model

$$y(t) = s(t) + e(t), \quad s(t) = \sum_{k=1}^{K} z_k e^{i2\pi f_k t} \tag{110}$$

where $s(t)$ denotes the noise-free super-positioning of K sinusoidal components, sampled in some form of additive noise, $e(t)$, for $t = 0, \ldots, N-1$. For the k:th component, z_k and $f_k \in [0, 1)$ denote the complex-valued amplitude and the frequency, respectively. By forming the sample vector

$$\mathbf{y} = \begin{bmatrix} y(0) & \ldots & y(N-1) \end{bmatrix}^T \tag{111}$$

the sinusoidal model (110) may be equivalently formulated as

$$\mathbf{y} = \mathbf{s} + \mathbf{e}, \quad \mathbf{s} = \sum_{k=1}^{K} \mathbf{w}_k z_k = \mathbf{W}\mathbf{z} \tag{112}$$

where the noise-free signal vector, $\mathbf{s}$, and the noise vector, $\mathbf{e}$, are defined similarly to $\mathbf{y}$. Thus, some simple algebraic manipulations allows the signal vector to be compactly expressed as a matrix-vector multiplication, given that

$$\mathbf{W} = \begin{bmatrix} \mathbf{w}_1 & \ldots & \mathbf{w}_K \end{bmatrix} \tag{113}$$

$$\mathbf{w}_k = \sqrt{N^{-1}} \begin{bmatrix} e^{i2\pi f_k 1} & \ldots & e^{i2\pi f_k (N-1)} \end{bmatrix}^T \tag{114}$$

$$\mathbf{z} = \begin{bmatrix} z_1 & \ldots & z_K \end{bmatrix}^T \tag{115}$$

The noise-free signal vector may therefore also be seen as a linear combination of the columns in $\mathbf{W}$, each column being a Fourier vector parametrizing the sinusoidal component, mixing them using the complex weights in $\mathbf{z}$.

4.1.1 Non-linear estimation of line spectra

If K is known *a priori*, it may be convenient to view (112) as a non-linear regression problem, where the spectral components at frequencies $\mathbf{\Psi} = \{f_k\}_{k=1}^K$ are multiplied by the linear amplitudes. Using the least squares criterion, given by

$$\left\{\hat{\mathbf{\Psi}}, \hat{\mathbf{z}}\right\} = \arg\min_{\mathbf{\Psi},\mathbf{z}} \|\mathbf{y} - \mathbf{W}\mathbf{z}\|_2^2 \tag{116}$$

i.e, as the arguments minimizing the sum of squared model residuals. As earlier shown, the closed form estimate of the amplitudes for a given selection of $\mathbf{\Psi}$ is

$$\hat{\mathbf{z}} = \mathbf{W}^\dagger \mathbf{y} \tag{117}$$

which, if inserted into (116), gives the non-linear least squares (NLS) criterion

$$\hat{\mathbf{\Psi}} = \arg\max_{\mathbf{\Psi}} \mathbf{y}^H \mathbf{W}\mathbf{W}^\dagger \mathbf{y}. \tag{118}$$

One may then, for instance, form the frequency estimates by maximizing the NLS criteria over a K-dimensional grid. Furthermore, as is shown in, e.g., [34], the NLS estimation errors of $\mathbf{\Psi}$ will have the asymptotic covariance matrix

$$\mathrm{Cov}(\hat{\mathbf{\Psi}}) = \frac{6\sigma^2}{N^3} \mathrm{diag}\left(\begin{bmatrix} \frac{1}{|z|_1^2} & \cdots & \frac{1}{|z|_K^2} \end{bmatrix}\right) \tag{119}$$

where diag($\mathbf{c}$) denotes a diagonal matrix the vector $\mathbf{c}$ along its diagonal. In the case of white Gaussian noise, i.e., $\mathbf{e} \sim \mathcal{N}(0, \sigma^2\mathbf{I})$, the covariances in (119) reach the Cramér-Rao Lower Bound (CRLB), as was shown in, e.g., [35], which gives the lower bound for the covariance matrix of any unbiased estimator of $\mathbf{\Psi}$. A similar analysis can be done for $\hat{\mathbf{z}}$ in (117), showing that the NLS method provides a statistically efficient estimate of the parametric line spectra. However, the NLS criterion works poorly in practice for this problem, and the reason is twofold. Firstly, (118) is non-convex, often highly multimodal, and the global maximum is typically very sharp, and therefore, to obtain the correct estimates, the maximization needs to be well initialized, as well as evaluated over a sufficiently fine grid. Secondly, any two frequencies must be sufficiently separated in order for the estimator to work properly. To see this, consider the square matrix $\mathbf{W}^H\mathbf{W}$ in the middle of the NLS criterion, which needs to be inverted. This matrix measures the linear dependence between the components in $\mathbf{W}$, wherein each element

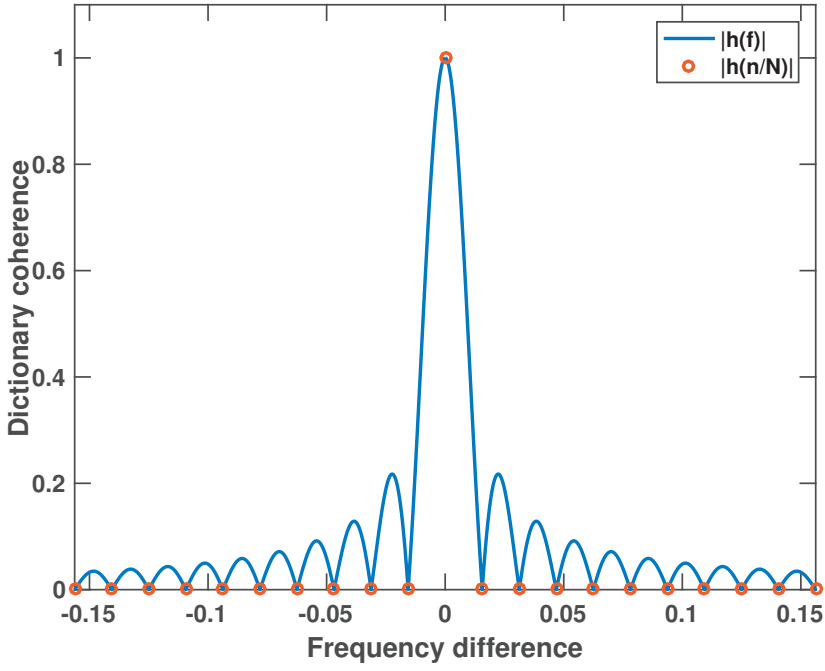


Figure 3: Absolute values of the complex function $h(\Delta f)$, measuring the amount of linear dependence between two Fourier vectors, spaced in frequency by Δf. The example illustrates the function for $N = 64$ samples, where orthogonality is found at every n/N, $n \neq 0$.

corresponds to the coherence measure used in (71), which for line spectral components can be shown to only depend n the difference in frequency [4, p. 160], i.e., for element k, k' is

$$h(f_k - f_{k'}) \triangleq \left\{\mathbf{W}^H\mathbf{W}\right\}_{k,k'} \tag{120}$$

$$= \mathbf{w}_k^H \mathbf{w}_{k'} \tag{121}$$

$$= \begin{cases} 1 & f_k = f_{k'} \\ e^{i2\pi(f_k - f_{k'})} \frac{e^{i2\pi N(f_k - f_{k'})} - 1}{N(e^{i2\pi(f_k - f_{k'})} - 1)} & f_k \neq f_{k'} \end{cases} \tag{122}$$

where a special case is $h(n/N) = 0$, for $n = \{n \in \mathbb{Z} : n \neq 0\}$. An example of this function can be seen in Figure 3, which shows the absolute values of the function for $N = 64$. Thus, if two frequencies are too closely spaced, the columns of

$\mathbf{W}$ become linearly dependent, making the inversion and the estimation problem ill-conditioned. In fact, under the quite restrictive assumption that all frequencies in $\boldsymbol{\Psi}$ are spaced by n/N, $n \in \mathbb{Z}$, then $\mathbf{W}^H\mathbf{W} = \mathbf{I}$ and (118) reduces to

$$\hat{\boldsymbol{\Psi}} = \arg\max_{\boldsymbol{\Psi}} \left\|\mathbf{W}^H\mathbf{y}\right\|_2^2, \quad \hat{\mathbf{z}} = \mathbf{W}^H\mathbf{y} \tag{123}$$

which is the (squared) ℓ_2-norm of the periodogram estimates. Given this, some remarks regarding the performance of the periodogram for estimation of line spectra may be noted.

Remark 1: For a single sinusoid in white Gaussian noise, i.e., $K = 1$, the periodogram is the ML estimator of the amplitude, as $\mathbf{W}^H\mathbf{W} = \mathbf{w}^H\mathbf{w} = 1$. To obtain the frequency estimate in (123), one usually evaluates the periodogram on an oversampled discrete Fourier transform (DFT) grid, i.e.,

$$\boldsymbol{\Psi} = \left\{\frac{m}{rN}\right\}_{m=0,\ldots,rN-1} \tag{124}$$

where r is the oversampling or super-resolution factor, such that $M = rN$, and picking the largest peak of the corresponding magnitude estimate

$$|\hat{\mathbf{z}}| = |\mathbf{W}^H\mathbf{y}| \tag{125}$$

yields the frequency and corresponding amplitude estimate.

Remark 2: For $K > 1$, one usually proceeds in the same manner as for one sinusoid. In the unlikely case that all frequencies are separated by at least $1/N$ and lie exactly on the standard DFT grid, where $r = 1$, the periodogram would be an efficient estimator, as $\mathbf{W}^H\mathbf{W} = \mathbf{I}$ and so (118) and (123) are equal. Otherwise, when the frequencies lie off-grid, the periodogram is typically a reasonable, but not an efficient, estimator [4, p. 161].

Remark 3: The resolution of the periodogram is limited, so that two sinusoids closely spaced in frequency are only likely to be resolved if that spacing is at least $1/N$. If spaced finer, they will appear to coincide in the resulting spectral estimate. However, if given the correct frequencies, (117) gives a very accurate amplitude estimate. Thus, the problem resides in finding the non-linear frequency parameters. Some commonly used parametric methods for frequency estimation with good statistical accuracy include the HOYW, MUSIC, and ESPRIT methods [4, ch. 4].

Remark 4: Throughout the analysis in this section, the model order, K, is assumed to be known, which is also a requirement for most parametric estimation

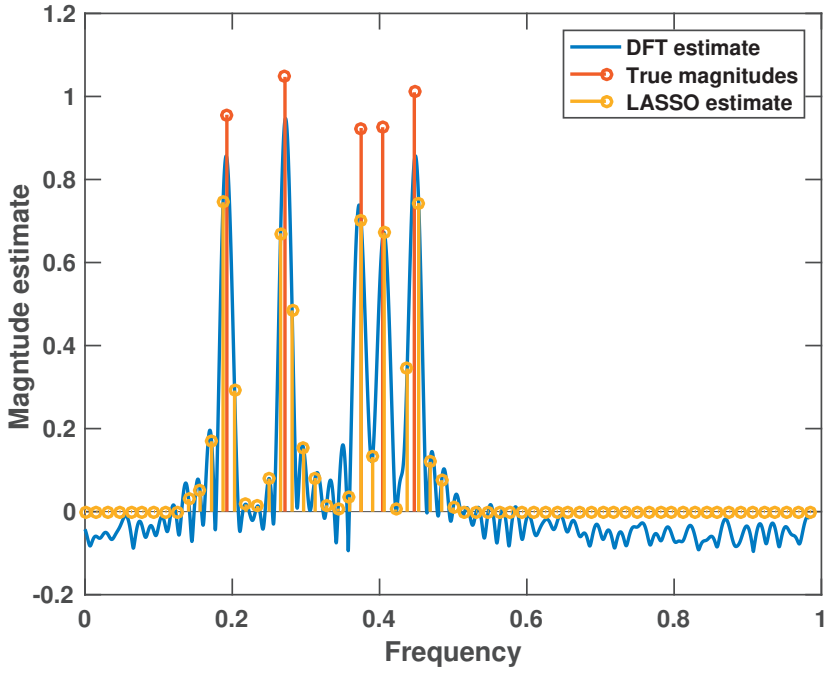


Figure 4: The LASSO amplitude estimates for $K = 5$ well separated sinusoids in white Gaussian noise. In comparison with a thresholded DFT estimate, $\hat{a}_{\text{DFT}} - \lambda$.

methods. However, in practice, the model order is typically unknown, which requires a model order estimation procedure.

4.1.2 LASSO for line spectra

As introduced in [7], the LASSO can be used for spectral estimation. Assuming that the spectral content of the data is narrowband, and using an oversampled DFT matrix as dictionary, sparsity in $\mathbf{x}$ follows. Due to the strong linear dependence between sinusoids separated by less than $1/N$, as described in (120), it is reasonable to assume that a grid of finely spaced sinusoids may well model the true spectral lines, even if the frequencies in the dictionary does not exactly match the true frequencies. Thus,

$$\mathbf{s} = \mathbf{W}\mathbf{z} \approx \mathbf{A}\mathbf{x} \tag{126}$$

where $\mathbf{A}$ and $\mathbf{x}$ is the dictionary and the sparse parameter vector, respectively. Assuming that the number of candidate spectral lines in $\mathbf{A}$ is much larger than the number of observations, $M \gg N$, estimating $\mathbf{x}$ by minimizing the ℓ_2-norm of the residual vector is an ill-posed problem, and $\hat{\mathbf{x}} = \mathbf{A}^\dagger \mathbf{y}$ has no closed-form solution, as was discussed above. Sparse regression facilitates a linear methodology for solving (116), which is non-linear with respect to the frequency parameters $\{f_k\}_k = 1, \ldots, K$. However, seeking a continuous parameter using a discrete grid of candidate parameter values, the LASSO has no true support, as defined in (64). Instead, the typical support sought using the LASSO is the peak of non-zero candidates nearest to the true frequency. For this reason, the number of spectral lines found when using a LASSO is not the number of non-zero elements in $\hat{\mathbf{x}}$, but in practice the number of peaks. When the dictionary is chosen as the standard DFT matrix, i.e., with $r = 1$, the LASSO problem becomes uncoupled, i.e., (26) being equivalent to

$$\underset{\mathbf{x}}{\text{minimize}} \quad \sum_{m=1}^{M} x_m^H \left(x_m - 2\mathbf{a}_m^H \mathbf{y} \right) + \lambda |x_m| \tag{127}$$

which has the (coordinate-wise) closed-form solution

$$\hat{x}_m = \mathcal{S}(\mathbf{a}_m^H \mathbf{y}, \lambda) \tag{128}$$

for $m = 1, \ldots, M$. Note how the elements in $\mathbf{x}$ are uncoupled from each other in (127), and thus the corresponding LASSO estimates may be formed for each element independently of the other variables, as compared to the general case in (34). The reason for this is that the DFT dictionary is an orthogonal base, i.e.,, $\mathbf{A}^H \mathbf{A} = \mathbf{I}$ implying that $\mathbf{a}_m^H \mathbf{r}_m = \mathbf{a}_m^H \mathbf{y} - \sum_{m' \neq m} \mathbf{a}_m^H \mathbf{a}_{m'}^H \hat{x}_{m'} = \mathbf{a}^H \mathbf{y}$. Figure 4 illustrates an example of this, where five sinusoids, well separated by more than $1/N$ from each other, are estimated using the LASSO with such an orthogonal dictionary, for $N = 64$. As a comparison, the DFT estimate thresholded with the bias, i.e., $|\hat{x}_m|_{\text{DFT}} - \lambda$, is shown, as to clearly illustrate the soft-thresholding in the LASSO estimate. Typically, the LASSO has better resolution capabilities than the periodogram. Thus, an oversampled (and thus coherent) DFT matrix is often used as dictionary, and the estimates become coupled. An example of this is seen in Figure 5, where the LASSO estimate for a dictionary with oversampling $r = 20$ is plotted. The two closely spaced sinusoids are resolved, but their respective magnitude is divided between several dictionary elements. As a

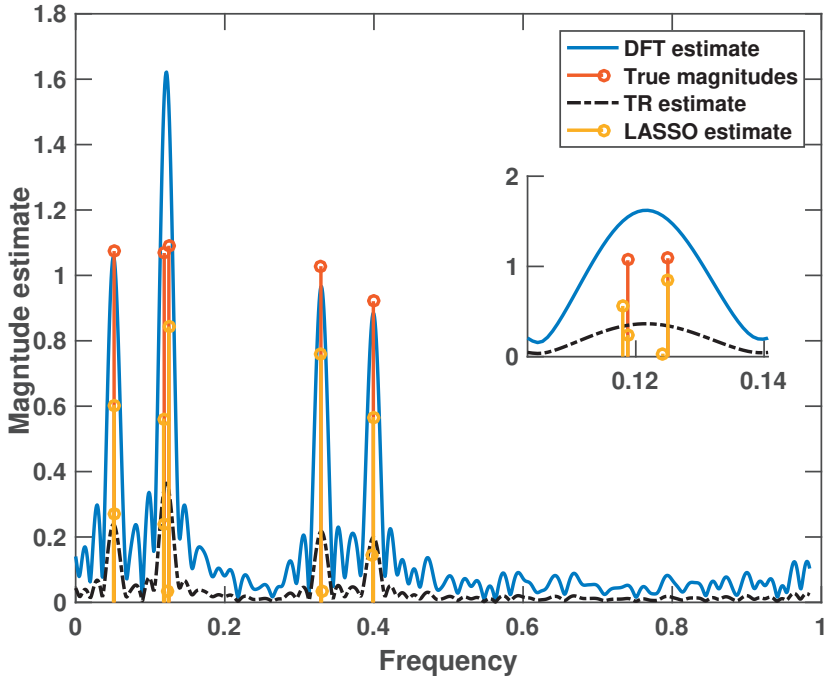


Figure 5: LASSO, Tikhonov regularization, and DFT amplitude estimates for $K = 5$ sinusoids in white Gaussian noise, where two of them are spaced by 2/5N, as compared to the true amplitudes.

comparison, the TR estimate is plotted. One may note that due it's regularizer, the TR estimate resembles a scaled version of the DFT estimate $\hat{\mathbf{x}} = \mathbf{A}^H\mathbf{y}$, which is also seen in the figure. As is also shown, the LASSO it will generally have good super-resolution performance, and will typically cope with resolutions upwards of $5 \leq r \leq 10$ [36]. In terms of theoretical estimation guarantees, the results are quite pessimistic. There are several different methods of assessing the suitability of a dictionary for sparse estimation, which include the exact recovery coefficient (ERC) [37], the spark [9], the two restricted isometry criteria (RIC)[6] [5], and the two coherence measures in [37] (cumulative coherence) and in [9] (mutual coherence). As noted in [38], only the latter two can be readily calculated for an

[6]The two RICs are the well-known restricted isometry property (RIP) and the restricted orthogonality property (ROP), respectively.

arbitrary dictionary. Focusing on the mutual coherence, it is defined as the maximum linear dependence present in the dictionary, which for line spectra becomes

$$\rho(\mathbf{W}) \triangleq \max_{f_p \neq f_q} \frac{1}{N} \left| h(f_p, f_q) \right| \tag{129}$$

for $f_p, f_q \in \mathbf{\Psi}$. The theoretical implications for mutual coherence in line spectra was examined in [39], where it is claimed that a sufficient condition for robust recovery is that $\mu \leq \sqrt{2} - 1$. In contrast to practical observations this would thus correspond to a minimal grid point spacing of approximately $|f_p - f_q| \geq 2N/3$, and so robust recovery in this sense is not possible for super-resolution dictionaries, i.e., for $s > 1$. Except for super-resolution, another issue affecting performance is off-grid effects. Re-examining Figure 4, it is apparent that the LASSO does not robustly recover the correct number of frequency components, even if an orthogonal dictionary is used. In spite of this, it has been found that if choosing the largest peaks of the LASSO estimate, rather than all non-zero parameters, sparse modeling works well for line spectra in practice [36]. In addition, the estimation may be further improved by using the LASSO estimates as an initial solution to the NLS method.

4.2 Audio Signal Processing

In modern audio processing, one primarily deals with the digital representation of sound waves, i.e., longitudinal waves where a medium[7] is compressed and decompressed. A substantial part of the research in audio signal processing during the last decades has focused on speech processing, as to fill the emerging need of solutions for digital communication (see, e.g., [40], and the references therein). However, in more recent years, much research in audio processing has also been devoted to musical signals, perhaps not very surprisingly given the large role of digital media in everyday life (for an overview, see, e.g., [41]). Combined, the two fields of speech and music processing are formidably vast, and they cannot possibly be given any form of justice in this introduction. Instead, some brief excerpts are given, as to give some context to the methods of which this thesis consist. For both fields, given the nature of sound, signals are periodic and, for our purposes, their spectral representations are highly relevant. Many audio signals are

[7] Sounds in air are typically recorded using microphones, but sounds in water are also often considered, for instance in the sonar applications, where the audio is recorded by hydrophones.

well described as narrowband, i.e., the spectral energy is largely limited to a few narrow intervals on the frequency axis. As a result, parametric estimation using the sinusoidal model is often a good approach to quantifying the properties of speech and music. A common model used for voiced speech and tonal music is the harmonic model, or pitch model, which is of the form [42]

$$y(t) = s(t) + e(t), \; s(t) = \sum_{\ell=1}^{L} z_\ell e^{i2\pi f \ell t} \tag{130}$$

for a single pitch. Typically, the non-tonal components are detailed as additive noise, $e(t)$, whereas the pitch signal, $s(t)$, is assumed to consist of a group of complex-valued[8] sinusoids, whose relation are described as

$$\psi(f, \ell) = f\ell, \quad \ell \in \mathcal{L} \tag{131}$$

where the frequency components are integer multiples of the fundamental frequency f, in the set $\mathcal{L}$. Typically, a pitch is defined by its fundamental frequency, i.e., $\psi(f, 1) = f$, and the individual sinusoids are referred to as its harmonics. A common misconception is that the fundamental is always the lowest frequency in the pitch, which is only true if $1 \in \mathcal{L}$. This is, however, not always the case, as some harmonics may be missing, including the fundamental. Instead, it is in most cases better to view the fundamental frequency as the smallest commonly occurring distance between two adjacent harmonics in a pitch group. Thus, if a certain pitch f has the following set of harmonics,

$$\mathcal{L} = \{2, 4, 6, 8, \ldots, 2L\} \tag{132}$$

it may preferably be seen as a pitch with fundamental frequency $f' = 2f$, and corrsponding set $\mathcal{L}' = \{1, 2, 3, 4, \ldots, L\}$ of harmonics. As there might be ambiguities as how to chose f and $\mathcal{L}$, such as, e.g., the example given above, the basic assumption, which is extensively used in this thesis, is that the spectral envelope of the pitch should be smooth [43], i.e., that adjacent harmonics should be of comparable magnitude. This is obviously not the case for the pitch described in (132), as all uneven harmonics have zero magnitude. Promoting such smoothness

[8] Naturally, recorded audio signals are not complex-valued. However, by using the analytic representation of the real-valued signals, both analysis and estimation may be greatly simplified. This is mainly because real-valued signals contain two spectral lines for every frequency f present in the signal, located at $\pm f$, where the negative component is removed in the analytic signal.

to avoid ambiguity is one of the objectives of paper B. Another common property for harmonic audio signals, in particular for some musical instruments, is a slight, but systematic, deviation from even distances between harmonics. This is referred to as inharmonicity, which for stringed instruments may be well described as

$$\psi(f,\ell) = f\ell\sqrt{1+\ell^2 B}, \quad \ell \in \mathcal{L} \tag{133}$$

where B is called the inharmonicity coefficient, specific to each string; typically $B \in [10^{-5}, 10^{-3}]$ [44]. Another feature of audio, highly related to pitch and especially used in musical contexts, is chroma. Mathematically, chroma is a change of variables, such that it represents fundamental frequency on a cyclical scale. To that end, consider the chroma parameter $c \in [0,1)$, to which the corresponding fundamental frequencies may be expressed as

$$f = f_{\text{base}} 2^{c+m}, \quad \forall m \in \mathbb{Z} \tag{134}$$

where m is referred to the octave, and where f_{base} is a tuning or offset frequency, defining the specific location of a chroma in frequency. This implies that the linear frequency scale collapses into a cyclic chroma scale, as all fundamental frequencies which fulfill (134), for some integer o, belong to the same chroma, i.e., if $f \in c$, then

$$f' \in c \Rightarrow f' \in \left\{ \ldots, \frac{f}{8}, \frac{f}{4}, \frac{1}{2}, f, 2f, 4f, 8f, 16f, \ldots \right\} \tag{135}$$

and all fundamentals in a chroma are thus related by some power of 2. One benefit of the chroma representation is that it groups together pitches that have largely overlapping frequency content, which makes chroma estimation much less ambiguous than pitch estimation. In music, the chroma representation is a common grouping criterion, as all pitches in a chroma are perceived as being similar by the human hearing [41]. In the Western musicological system, for instance, the chroma interval is discretized into twelve semitones, uniformly spaced on $[0,1)$, i.e.,

$$c \in \left\{ 0, \frac{1}{12}, \frac{2}{12}, \ldots, \frac{11}{12} \right\} \tag{136}$$

In paper C, the chroma model for Western music is used with sparse modeling to form estimates, cruder than pitch but more robust, of the spectral components of an audio signal.

4.3 Array Processing

In the field of array processing, a common objective is to locate signal emitting sources by measuring their emissions over an array of sensors. The emitted energy may be of various types, e.g., acoustic or electromagnetic, to which different corresponding types of sensors are used. In this section, some basic results for source localization is given as to facilitate a bit of background for the methods presented in Paper A. In general, the objective may be put as finding the distribution of energy in the spatial domain. If assuming that all sensors have the same gain, the signal model for the impinging source signal at the j:th sensor may be expressed as

$$y_j(t) = s(t - \tau_j) + e_j(t) \tag{137}$$

where τ_j is the source-sensor time-delay with respect to some reference point, such that the source signal $x(t)$ is at each sensor delayed with respect to the specific geometry of the array. Consider that $x(t)$ follows the sinusoidal signal model in (110). As such a signal is formed by a sum of narrowband components, the time-delay in (137) may typically be well modeled as a phase offset in each component, exponentially proportional to its frequency, i.e.,

$$y_j(t) = \sum_{k=1}^{K} z_k e^{i2\pi f_k (t-\tau_j)} + e_j(t) \tag{138}$$

which for the sample vector is equivalent to

$$\mathbf{y}_j = \sum_{k=1}^{K} \mathbf{w}_k z_k e^{-i2\pi f_k \tau_j} + \mathbf{e}_j, \tag{139}$$

where $(\cdot)_j$ denotes the j:th sensor. By column-wise stacking the sample vectors for all sensors, i.e.,

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 & \dots & \mathbf{y}_J \end{bmatrix} \tag{140}$$

the signal model for the entire array may be expressed as

$$\mathbf{Y} = \sum_{k=1}^{K} \mathbf{w}_k z_k \mathbf{u}^T + \mathbf{E} = \mathbf{W}\,\mathrm{diag}(\mathbf{z})\mathbf{U}^T + \mathbf{E} \tag{141}$$

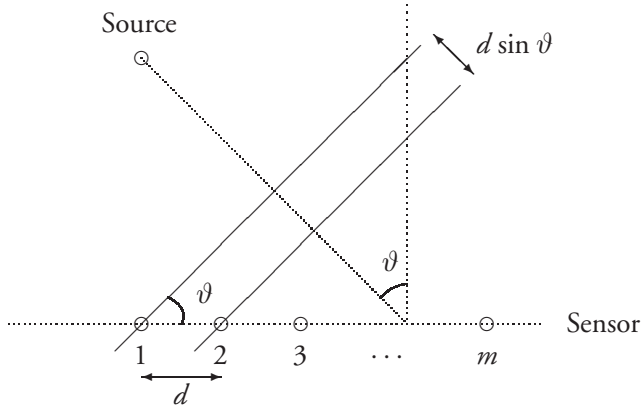


Figure 6: Principle sketch of a far-field point-source, which from ϑ emit planar wavefronts, that are impinging on a ULA, with equidistant sensor spacing d.

where, $\mathbf{z}$ denotes the amplitude parameters, and $\mathbf{W}$ a matrix of Fourier vectors. Furthermore, $\mathbf{E}$ denotes the observation noise, defined similarly to (140), and where

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \dots & \mathbf{u}_K \end{bmatrix} \tag{142}$$

$$\mathbf{u}_k = \begin{bmatrix} e^{-i2\pi f_k \tau_1} & \dots & e^{-i2\pi f_k \tau_J} \end{bmatrix}^T \tag{143}$$

denote the phase offset for each sinusoidal component in each sensor, which depend on both the frequency and the time-delay. The time-delays are inherently related to both the source position and the geometry of the array, whose relation may be modeled by imposing some assumptions on the source, and the array, respectively. Two assumptions, which are very common for localization in array processing, are

- The source is a point source in the far-field, i.e., the source is at an infinite distance from the sensor array. This implies that the impinging signal wavefronts are essentially planar, so that a source's location solely depends on its Direction-Of-Arrival (DOA).
- The sensors are positioned as a Uniform Linear Array (ULA), meaning that

they are equidistantly located on a line. This implies that the positions will be defined to a 2-D space of locations, described by DOA and distance.

Figure 6 illustrates these two assumptions, where the DOA is the 1-D angular deviation from the array's normal, denoted $\vartheta \in [-\pi, \pi]$. Note that the ULA will not discriminate between a source impinging from the front or from the back of the array. From these assumptions, time-delays may thus be expressed as a function of DOA, i.e.,

$$\tau_j = \frac{d \sin(\vartheta)}{c}(j-1) \tag{144}$$

where d and c is the sensor distance, and the wave propagation speed, respectively. Therefore, (143) may be equivalently expressed as

$$\mathbf{u}_k = \left[\begin{array}{cccc} 1 & e^{-i2\pi f_k \frac{d \sin(\vartheta)}{c}} & \dots & e^{-i2\pi f_k \frac{d \sin(\vartheta)}{c}(J-1)} \end{array} \right]^T \tag{145}$$

where

$$\left| f_k \frac{d \sin(\vartheta)}{c} \right| \leq \frac{1}{2} \Rightarrow d \leq \frac{c}{2 f_k} \tag{146}$$

should be fulfilled as to guarantee that aliasing effects are avoided. For the far-field source and ULA case, $\mathbf{u}_k$ may thus be seen as a uniformly sampled spatial DFT vector. In paper A, the preliminaries presented herein are extended, and a joint multi-pitch and location estimator is proposed, for sources which are near-field rather than far-field, and when the array's geometry is arbitrary rather than a ULA.

5 Outline of the papers in this thesis

This section briefly summarizes the papers of which this thesis consist, together with information of where they have been published or submitted.

Paper A: Sparse Localization of Harmonic Audio Sources

In paper A, a two-step procedure is used to form joint estimates of pitches and near- or far-field locations from measurements on an arbitrary, but calibrated, sensor array. In the first step, a sparse group-LASSO generalized for array signals is used to find the active pitches. Then, for estimated pitch, another variation on the sparse group-LASSO is used on the estimated parameters, which contain information of both TDOA and signal attenuation. This information is consequently exploited to form location estimates, which may be more than one for each pitch. The implications of using the sparse modeling approach is interesting, as it facilitates an opportunity to position sources despite of reverberation effects, which usually are detrimental to localization. The performance of the proposed method is validated using both synthetic and real recorded signals, showing promising results.

The work in paper A has been published/submitted in part as

> Stefan Ingi Adalbjörnsson, Ted Kronvall, Simon Burgess, Kalle Åström, and Andreas Jakobsson, "Sparse Localization of Harmonic Audio Sources". *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 117-129, November 2015.
>
> Ted Kronvall, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, "Joint DOA and Multi-pitch Estimation using Block Sparsity", *39th IEEE International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, May 4-9, 2014.

Paper B: An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization

In paper B, a novel a novel adaptive penalty approach to estimate the parameters in the multi-pitch model with the use of sparse modeling is proposed. It further examines the total variation (TV) regularizer examined in the PEBS-TV method [45], adressing the problem of suboctave errors, a common source of

misclassifying the fundamental frequency estimation. In PEBS-TV, an additional regularizer, which is a modification of the TV penalty, is introduced, which is shown to migitate such issues. However, this method requires tuning three regularization parameters, which we circumvent in this paper by using the adaptive approach, in which the TV regularizer is the key, enabling one to drop the ℓ_2-norm regularizer of the group-LASSO altogether. The method may thus be seen as solving a series of convex problems, where each is a sparse fused LASSO, having two tuning parameters. The strength of using TV compared to group sparsity is that the former promotes solutions with smooth parameter envelopes, discouraging suboctave errors. The method is shown to work well for highly coherent dictionaries, and even outperforms the method in [45].

The work in paper B has been published in part as

> Filip Elvander, Ted Kronvall, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, "An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization", *Elsevier Signal Processing*, vol. 127, pp. 56-70, October 2016.
>
> Ted Kronvall, Filip Elvander, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, "An Adaptive Penalty Approach to Multi-pitch Estimation". *23rd European Signal Processing Conference*, Nice, France, August 31 - September 4, 2015.

Paper C: Sparse Modeling of Chroma Features

In paper C, an alternative modeling approach for the pitch estimation problem is used. Instead of focusing on estimating the parameter group of a specific pitch, groups are formed consisting of all pitches that belong to the same chroma group, i.e., defined as all fundamental frequencies which on a log-scale are related by a multiple of 2. The chroma is a concept from musical theory, and transcribing a piece of audio with respect to its chroma content is a pre-processing step that is done for a variety of different MIR applications. In the paper, the formulation proposed uses a combination of group-sparsity and TV, such that the group-sparsity promotes solutions where few chroma blocks are active, and where TV discourages misclassification due to musical harmony, as chroma groups have partly overlapping frequency components. The method is numerically evaluated for both synthetic and recorded audio signals, and indicates a preferred performance for transcription purposes. In the paper, each amplitude of each component is also extended to possibly vary over time, which is modeled using a spline basis.

As the approach increases the number of parameters proportional to the number of spline knots, the method is especially suitable for longer sequences of data, where, for audio signals longer than 40 ms, the signal exhibits a large degree of non-stationarity. The approach may also be beneficial for sounds that are very transient, or for capturing the onset of a signal. It is shown that for a recorded violin signal, the proposed method estimates the signal envelope more accurately than for constant amplitudes.

The work in Paper C has been published in part as

> Ted Kronvall, Maria Juhlin, Johan Swärd, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, "Sparse Modeling of Chroma Features", *Elsevier Signal Processing*, vol. 30, pp. 106-117, January 2017.
>
> Maria Juhlin, Ted Kronvall, Johan Swärd, and Andreas Jakobsson, "Sparse Chroma Estimation for Harmonic Non-stationary Audio", *23rd European Signal Processing Conference*, Nice, France, August 31 - September 4, 2015.
>
> Ted Kronvall, Maria Juhlin, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson, "Sparse Chroma Estimation for Harmonic Audio", *40th International Conference on Acoustics, Speech, and Signal Processing*, Brisbane, Australia, April 19-24, 2015.
>
> Stefan Ingi Adalbjörnsson, Johan Swärd, Ted Kronvall, and Andreas Jakobsson, "A Sparse Approach for Estimation of Amplitude Modulated Sinusoids", *The Asilomar Conference on Signals, Systems, and Computers*, Asilomar, USA, November 2-5, 2014.

Paper D: Group-Sparse Regression using the Covariance Fitting Criterion

In paper D, the group-sparse regression problem is formulated using a covariance fitting criterion, a common metric used in array processing, where the objective function measures the ℓ_2-norm of the misfit between a parametric model for the covariance matrix and the observed covariance matrix. As shown in [46], the covariance fitting criteria, when the covariance matrix is modeled using a highly redundant combination of linear basis functions, i.e., a dictionary, promotes sparse solutions. Furthermore, the covariance fitting criterion is hyperparameter-free,

i.e., lacks any user-parameter, which is uncommon for regularized regression problems. In the paper, the covariance fitting criterion is generalized for groups of dictionary atoms, which is herein shown to promote group-sparse parameter estimates, for which an efficient CCD-based numerical solver is proposed. It is also shown in the paper that the proposed method is equivalent to a group-version of the square-root LASSO [47], where the regularization parameter has been pre-selected. It is shown using simulation studies that this regularization level is slightly too low, and thus includes some noise components into the solution, but is robust against false exclusion of the true parameters. It may also be noted that the regularization level is set at no additional computational cost, which, as discussed in Section 2.6, is typically not the case. Numerical results for synthetic signals shows the method to perform on par with an optimally regularized group-LASSO in terms of recovering the true signal components, and outperform both the method's non-grouped counterpart [46] and greedy group-sparse estimators.

The work in Paper D has been published in part as

> Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, "Group-Sparse Regression using the Covariance Fitting Criterion", *Elsevier Signal Processing*, vol. 139, pp. 116-130, October 2017.
>
> Ted Kronvall, Stefan Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, "Hyperparameter-free sparse linear regression of grouped variables", *Proceedings of the 50th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, November 6-9 2016.

Paper E: Online Group-Sparse Regression using the Covariance Fitting Criterion

In paper E, an extension of Paper D is proposed, where samples enters the optimization problem in small batches or one-by-one. Instead of repeating the estimation process for all observations when new data is added, the proposed method, in a class of so called online estimator, computes recursive update steps at a small computational cost. To that end, the group-version of the covariance fitting criterion is reformulated as a square-root LASSO problem with a pre-defined regularization level, which is optimized using a proximal gradient solver, which allows for low complexity updates small memory storage. A simulation study shows preferable performance in comparison with other group-sparse estimators.

The work in Paper E has been published in part as

> Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson, "Online Group-Sparse Regression using the Covariance Fitting Criterion", *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, Kos, Greece, August 28 - September 2, 2017.

Paper F: Hyperparameter-selection for sparse regression: a probablistic approach

In paper F, a probabilistic method for choosing the regularization level in the LASSO and group-LASSO methods is proposed. By analyzing how the noise term propagates into the parameter estimates at different levels of regularization, the hyperparameter may be calculated for some false probability threshold, thereby optimizing support recovery more directly than other methods, such as, e.g., cross-validation (CV), does. Support recovery, or sparsistency, is achieved when the hyperparameter is selected to be larger than the noise components, but still smaller than the unknown signal components. This tradeoff is often quantified in detection therory by selecting a false positive threshold, typically being a quantile from the appropriate noise distribution. For the LASSO and group-LASSO, the appropriate quantile follows an extremal distribution quantified by the maximal inner product between the dictionary and the noise, on which inference can be done using the Monte Carlo method. To select the regularization level independently of the unknown noise variance, the scaled LASSO method is used, wherein the noise variance is simultaneously estimated. As the proposed method is data-independent, its computational burden becomes much less than statistical approaches, such as CV or hyperparameter-selection using the Bayesian Inference Criterion (BIC). Numerical simulations illustrate how the proposed method outperforms the statistical approaches both in terms of sparsistency and computational time.

The work in Paper F has been published in part as

> Ted Kronvall, and Andreas Jakobsson, "Hyperparameter-Selection for Sparse Regression: A Probablistic Approach", *Proceedings of the 51st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, October 29 - November 2, 2017.

and has been submitted for possible publication as

> Ted Kronvall, and Andreas Jakobsson, ”Hyperparameter-Selection for Group-sparse Regression: A Probablistic Approach”.

References

[1] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*, Prentice-Hall, Englewood Cliffs, N.J., 1993.

[2] L. L. Scharf, *Statistical Signal Processing: Detection Estimation, and Time Series Analysis*, Addison-Wesley, New York, 1991.

[3] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[4] P. Stoica and R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, N.J., 2005.

[5] E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

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

[7] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, Seoul, July 2008, pp. 10225–10229.

[8] Y.H. Li, J. Scarlett, P. Ravikumar, and V. Cevher, "Sparsistency of l1-Regularized M-Estimators," *Journal of Machine Learning Research*, vol. 38, pp. 644–652, 2015.

[9] D.L. Donoho, M. Elad, and V.N. Temlyakov, "Stable Recovery of Sparse Overcomplete Representations in the Presence of Noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, Jan 2006.

[10] R.J. Tibshirani, "The Lasso Problem and Uniqueness," *Electronic Journal of Statistics*, vol. 7, no. 0, pp. 1456–1490, 2013.

[11] O. Arslan, "Weighted LAD-LASSO Method for Robust Parameter Estimation and Variable Selection in Regression," *Computational Statistics & Data Analysis*, vol. 56, no. 6, pp. 1952 – 1965, 2012.

[12] R.J. Tibshirani and J. Taylor, "The Solution Path of the Generalized Lasso," *The Annals of Statistics*, vol. 39, no. 3, pp. 1335–1371, June 2011.

[13] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, no. 1, pp. 91–108, January 2005.

[14] H. Zou and T. Hastie, "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[15] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[16] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A Sparse-Group Lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[17] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, Chapman and Hall/CRC, 2015.

[18] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, April 2004.

[19] T. Sun and C. H. Zhang, "Scaled sparse linear regression," *Biometrika*, vol. 99, no. 4, pp. 879, 2012.

[20] A. Belloni, V. Chernozhukov, and L. Wang, "Square-Root LASSO: Pivotal Recovery of Sparse Signals via Conic Programming," *Biometrika*, vol. 98, no. 4, pp. 791–806, 2011.

[21] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM Review*, vol. 43, pp. 129–159, 2001.

[22] E. J. Candès, M. B. Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted l_1 Minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec. 2008.

[23] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-Minimization Algorithms in Signal Processing, Communications, and Machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, February 2017.

[24] H. Zou, "The Adaptive Lasso And Its Oracle Properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[25] M. Grant, *Disciplined Convex Programming*, Ph.D. thesis, Information Systems Laboratory, Department of Electrical Engineering, Stanford University, 2004.

[26] Inc. CVX Research, "CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta," `http://cvxr.com/cvx`, Sept. 2012.

[27] J. F. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11-12, pp. 625–653, August 1999.

[28] R. H. Tutuncu, K. C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using SDPT3," *Mathematical Programming Ser. B*, vol. 95, pp. 189–217, 2003.

[29] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization Paths for Generalized Linear Models via Coordinate Descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, 2010.

[30] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[31] Y. Nesterov, "Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[32] P. Tseng, "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization," *Journal of Optimization Theory and Applications*, vol. 109, no. 3, pp. 475–494, 2001.

[33] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Estimating Multiple Pitches Using Block Sparsity," in *38th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vancouver, May 26–31, 2013.

[34] P. Stoica and A. Nehorai, "Statistical Analysis of Two Nonlinear Least-Squares Estimators of Sine-Wave Parameters in the Colored-Noise Case," *Circuits, Systems, and Signal Processing*, vol. 8, no. 1, pp. 3–15, 1989.

[35] P. Stoica, R. Moses, B. Friedlander, and T. Söderström, "Maximum Likelihood Estimation of the Parameters of Multiple Sinusoids from Noisy Measurements," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 378–392, March 1989.

[36] P. Stoica and P. Babu, "Sparse Estimation of Spectral Lines: Grid Selection Problems and Their Solutions," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 962–967, Feb. 2012.

[37] J.A. Tropp, "Just Relax: Convex Programming Methods for Identifying Sparse Signals in Noise," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1030–1051, March 2006.

[38] Z. Ben-Haim, Y.C. Eldar, and M. Elad, "Coherence-Based Performance Guarantees for Estimating a Sparse Vector Under Random Noise," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5030–5043, Oct 2010.

[39] J. Karlsson and L. Ning, "On Robustness of l1-Regularization Methods for Spectral Estimation," in *IEEE 53rd Annual Conference on Decision and Control*, Dec 2014, pp. 1767–1773.

[40] J. Benesty, M. Sondhi, M. Mohan, and Y. Huang, *Springer handbook of speech processing*, Springer, 2008.

[41] M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard, "Signal Processing for Music Analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1088–1110, 2011.

[42] M. G. Christensen, P. Stoica, A. Jakobsson, and S. H. Jensen, "Multi-pitch estimation," *Signal Processing*, vol. 88, no. 4, pp. 972–983, April 2008.

[43] A. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 804–816, 2003.

[44] H. Fletcher, "Normal vibration frequencies of stiff piano string," *Journal of the Acoustical Society of America*, vol. 36, no. 1, 1962.

[45] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[46] P. Stoica, P. Babu, and J. Li, "New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 35–47, Jan 2011.

[47] F. Bunea, J. Lederer, and Y. She, "The Group Square-Root Lasso: Theoretical Properties and Fast Algorithms," *IEEE Trans. Inf. Theor.*, vol. 60, no. 2, pp. 1313–1325, Feb. 2014.

A

Paper A

Sparse Localization of Harmonic Audio Sources

Stefan Ingi Adalbjörnsson, Ted Kronvall, Simon Burgess, Kalle Åström, and Andreas Jakobsson

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

In this paper, we propose a novel method for estimating the locations of near- and/or far-field harmonic audio sources impinging on an arbitrary, but calibrated, sensor array. Using a joint pitch and location estimation formed in two steps, we first estimate the fundamental frequencies and complex amplitudes under a sinusoidal model assumption, whereafter the location of each source is found by utilizing both the difference in phase and the relative attenuation of the magnitude estimates. As audio recordings often consist of multi-pitch signals exhibiting some degree of reverberation, where both the number of pitches and the source locations are unknown, we propose to use sparse heuristics to avoid the necessity of detailed a priori assumptions on the spectral and spatial model orders. The method's performance is evaluated using both simulated and measured audio data, with the former showing that the proposed method achieves near-optimal performance, whereas the latter confirms the method's feasibility when used with real recordings.

Key words: Multi-pitch estimation, near-field and far-field localization, TDOA, block sparsity, convex optimization, ADMM, non-convex sparsity

1 Introduction

Sound localization has been a topic of interest in a wide range of applications for centuries, and is well known to be a difficult problem, especially in a reverberating room environment (see, e.g., [1–7], and the references therein). Typically, a source is located in relation to an array of sensors by exploiting the time delay between sensors for when they receive its emitted signal. In the literature, this is referred to as either time of arrival (TOA) estimation, if the time of signal emission is known, or otherwise time difference of arrival (TDOA) estimation, where only the relative time delays are used. Common techniques for delay estimation include different variations on cross-correlation or canonical correlation analysis (CCA), which then allows the sources to be located in a second step using tri- och multi-lateration (see, e.g., [8]). Such estimates may also be further improved by matching the relative received signal gains to a model for signal attenuation. If the source is far from the sensor array, i.e., in the far-field, its range may not be determined due to the lack of curvature of the impinging sound pressure wavefront, which is then approximately planar, making the range estimation problem ill-posed. The scope is then restricted to determining the direction of arrival (DOA) of the source relative to the sensor array for the 2-D case, or determining azimuth and elevation angles for a 3-D scenario. Historically, such methods are not restricted to sound, but are commonly used, in e.g., military applications, with electromagnetic signals (see, e.g., [9–11]). Perhaps, partly due to differences in application for near-field and far-field techniques, these problems are often treated separately. In this work, and for our purposes with audio signals, the two problems may indifferently be treated together. A common issue with correlation-based techniques is that of reverberation. Although often described in a temporal sense as a filter for each sensor through which the signal is convoluted [12], it may also be analyzed using a spatial formulation. In principle, reverberation occurs when the original source signal is received together with a number of reflections of it, which are both time delayed and dislocated in space with respect to the original. Localization in reverberant environments is still very much an open topic, although several correlation-based approaches exist which shows some degree of robustness (see, e.g., [2]). By assuming a temporal and spectral parametric structure on the received signals, localization may be improved by jointly forming estimates of location together with the parameters of such structures. This is quite common for audio signals such as voiced speech [12], and many forms of harmonic audio sources, such as stringed, wind, and pitched percussion in-

struments [13], which typically have lots of structure. At a glance, the spectral distribution of energy for such signals is typically broadband, but further analysis shows that it is in fact dominantly multi-narrowband, and may be well described using the harmonic model, i.e., as a sum of harmonically related sinusoids [14]. Under this assumption, a source's difference in delay and attenuation when received at the different sensors translates into phase shifted and magnitude scaled versions of the original signal. Exploiting this, joint estimation of the DOA and the pitch frequency has been addressed, such as in [15–17], wherein the authors consider the estimation of the DOA of a single harmonic sound source using a uniform linear array (ULA) of receiver sensor, typically assuming oracle knowledge of the number of harmonic signals in the sound source. Here, we extend on these works, albeit with some generalizations. We are allowing for an unknown number of near- or far-field harmonic sources, each having an unknown number of harmonics, to impinge on an arbitrary, but calibrated, sensor array, in the presence of some degree of reverberation. This feat is attempted through the use of a sparse recovery framework, which avoids making explicit assumptions on the number of harmonic signals, i.e., the number of pitches, as well as for the number of source locations for each pitch. Instead, only an implicit constraint which controls a lower threshold for acceptable source power is needed, which may typically be set using some simple heuristics. Sparse recovery frameworks have in earlier works been found to allow high quality estimates for sinusoidal signals; typical examples include [18–21], wherein the sparse signal reconstruction from noisy observations were accomplished with the by now well-known sparse least squares (LS) technique. More recently, the technique has been extended to the case of harmonically related audio signals [22, 23]. Using the techniques introduced there, we propose a two-step procedure, first creating a dictionary of candidate pitches to model the harmonic components of the sources, without taking the locations of the sources into account, and then, in a second step, a dictionary of possible locations, including simultaneously near- and far-field locations, to model the observed phase differences, as well as the relative attenuations, of the magnitudes of each sinusoidal component. In terms of computational complexity, the estimation problem in each of the two steps is convex, which thus guarantees convergence, and may be solved using a second order cone (SOC) program. As this is typically quite costly, we introduce a computationally efficient implementation based on the alternating direction method of multipliers (ADMM), which makes the proposed method very managable in an off-line estimation procedure. The remainder

of this paper is organized as follows: in the next section, we present the assumed signal model and discuss the imposed restrictions on the sensor array. Then, in section 3, we present the proposed pitch and localization estimator. Section 4 accounts for the ADMM-based implementation, followed in section 5 with an evaluation of the presented technique using both simulated and measured audio signals. Finally, we conclude on our work in section 7.

2 Spatial pitch signal model

In this work, we restrict our attention to the localization of complex-valued[1] harmonically related audio signals, consisting of $\tilde{K}$ distinct sources, $x_k(t)$, for $k = 1, \ldots, \tilde{K}$. Each source is thus assume to consist of L_k harmonically related sinusoids, such that it may be detailed as (see also [14])

$$x_k(t) = \sum_{\ell=1}^{L_k} a_{k,\ell} e^{i\omega_k \ell t} \tag{1}$$

where $\omega_k = 2\pi f_k / f_s$ is the normalized fundamental frequency, with sampling frequency f_s, and with $a_{k,l}$ denoting the complex amplitude of the ℓ:th harmonic.

2.1 Multi-sensor characteristics in near-field environments

When a source signal impinges on a sensor array, it is both delayed and attenuated, such that at sensor m it may be expressed as

$$x_{k,m}(t) \triangleq \frac{d_{k,1}}{d_{k,m}} x_k(t - \tau_{k,m}) \tag{2}$$

where $d_{k,m}$ denotes the sensor-source distance, i.e.,

$$d_{k,m} = \|\mathbf{s}_k - \mathbf{r}_m\|_2 \tag{3}$$

with $\mathbf{s}_k$ and $\mathbf{r}_m$ denoting the location coordinates of the k:th source and the m:th sensor, respectively, and $\|\cdot\|_2$ the Euclidean norm. Thus, (2) accounts for the approximative attenuation of the signal when propogating in space, according to the free-space path loss model. Furthermore, $\tau_{k,m}$ denotes the propagation delay,

[1] Clearly, the measured audio sources will be real-valued, but to simplify notation and in order to reduce complexity, we will here initially compute the discrete-time analytic signal versions of the measured signals, whereafter all processing is done on these signals (see also [14, 24]).

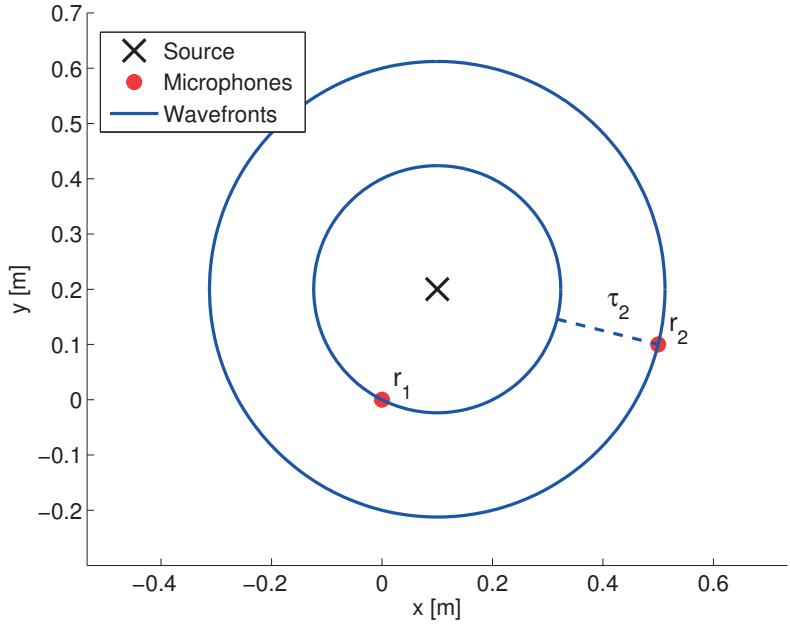


Figure 1: Illustration of a two sensor scenario, with spherical wavefronts propagating from the source. The dashed line shows the scaled TDOA of the second sensor with respect to the first sensor, i.e., τ_2.

i.e., the TDOA, relative to a selected reference sensor, say $m = 1$, so that

$$\tau_{k,m} = c^{-1}\left(d_{k,m} - d_{k,1}\right) \tag{4}$$

for $m = 1, \ldots, M$, where $\tau_{k,1} \triangleq 0$, with c denoting the propagation velocity. An illustration of this is shown in Figure 1, for the case of a single source and two sensors. When recording audio, we often obtain multi-pitch signals of the type

$$x(t) = \sum_{k=1}^{\tilde{K}} x_k(t) \tag{5}$$

which may be either a single source in the physical environment emitting multiple pitch signals, such as an instrument playing a chord, or multiple sources in the physical environment each emitting a single pitch, such as multiple speakers

talking at the same time from different locations. We may also receive a combination of these two types. Without loss of generality, we will hereafter term a source as a spatio-temporal object which has a unique combination of fundamental frequency and location. Two sources may thus have the same fundamental frequency or the same location in space, although not both. This has rather large implications when considering reverberation, where we, apart from the original source, also receive a large number of reflections of it, each reflection having highly similar spectral content, albeit differently attenuated and delayed, i.e., having different magnitudes and phases. All reflections will thus be modeled as separate sources, which implies that under such a model assumption $\tilde{K}$ generally becomes very large. If not seen as separate sources, however, the localization of the original source will become biased by the interference caused from its reflections. To see this, consider for example a sinusoid with frequency ω, magnitude a_1, and phase φ_1, measured in superimposition with its $S-1$ reverberating reflections, having magnitudes $a_2, \ldots, a_S$, and phases $\varphi_2, \ldots, \varphi_S$. For the mth sensor, the measured (noise-free) signal becomes

$$x_m(t) = \sum_{s=1}^{S} a_s e^{-i(\omega t+\varphi_s)} \triangleq b e^{-i(\omega_0 t+\psi)} \tag{6}$$

i.e., a single sinusoid with magnitude $b \in \mathbb{R}_+$ and phase $\psi \in [-\pi, \pi)$, generally being different from the original source. Thus, if trying to estimate the TDOA using phase estimates without taking all reflections into account, for instance by using a correlation-based measure, then only the biased phase, ψ, would be obtained. However, separation of all reflections for all fundamental frequencies is a quite difficult problem, and in this work, we propose to split the estimation procedure into two subproblems. In the first, we find the present fundamental frequencies, and then for each of these we separate the original source(s) from its reflections. To that end, consider $K \leq \tilde{K}$ as the number of unique fundamentals. The noisy signal measured at sensor m may thus be expressed as

$$y_m(t) = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} b_{k,\ell,m} e^{i\omega_k \ell t} + e_m(t) \tag{7}$$

where the TDOA and attenuation of all S_k reflections of the k:th pitch, for overtone ℓ and sensor m, is gathered in the complex amplitude of the signal, $b_{k,\ell,m}$

using (2) in the same manner as in (6), i.e.,

$$b_{k,\ell,m} = \sum_{s=1}^{S_k} a_{k,\ell,s} \frac{d_{k,1,s}}{d_{k,m,s}} e^{-i\omega_k \ell \tau_{k,m,s}} \tag{8}$$

where $a_{k,\ell,s}$, $d_{k,m,s}$, and $\tau_{k,m,s}$ denote the amplitude, the distance to the mth sensor, and the TDOA for the sth reflection, respectively. Thus, as $\tilde{K} = \sum_{k=1}^{K} S_k$, the estimation procedure first finds the K active fundamentals, whereafter for each one, the original source is separated from its reflections. This approach offers great simplification in contrast to decoupling all $\tilde{K}$ sources simultaneously. To simplify presentation, and without loss of generality, we will here restrict our attention to the case when all sources and signals are restricted to a 2-D plane, i.e., $\mathbf{s} \in \mathbb{R}^2$ and $\mathbf{r} \in \mathbb{R}^2$.

2.2 Avoiding spatial aliasing in arbitrary array geometries

In the literature, keeping below half wavelength sensor spacing is generally preferred to avoid spatial aliasing, although some methods of circumventing this have been published, see e.g. [25]. In this work, we assume a calibrated, although arbitrary, sensor array, without requiring it to satisfy the pairwise half wavelength spacing. We will therefore briefly examine the spatial aliasing effect in the near-field environment, which is the phase difference ambiguity between sensors, resulting when the solution may map to several feasible source locations. To that end, consider a reverberation-free, delayed, and attenuated complex amplitude from a single sinusoidal signal, b. Naturally,

$$b_m = \frac{d_1}{d_m} a e^{-i\omega\tau_m} = \frac{d_1}{d_m} a e^{-i(\omega\tau_m + k2\pi)} \tag{9}$$

and thus the mapping between phase and TDOA is ambiguous for any $k \in \mathbb{Z}$. Considering a given TDOA, and by combining (3) and (4), one will note that any source $\mathbf{s}$ located on a half-space of an hyperbolic curve, i.e.,

$$\tau_m c = \|\mathbf{s} - \mathbf{r}_m\|_2 - \|\mathbf{s} - \mathbf{r}_1\|_2 \tag{10}$$

is a feasible location. To obtain a unique solution, we add additional sensors, and we may thus form new sensor pairs yielding new hyperbolas, where the feasible solution set will be restricted by the intersection of these curves. Ambiguity may

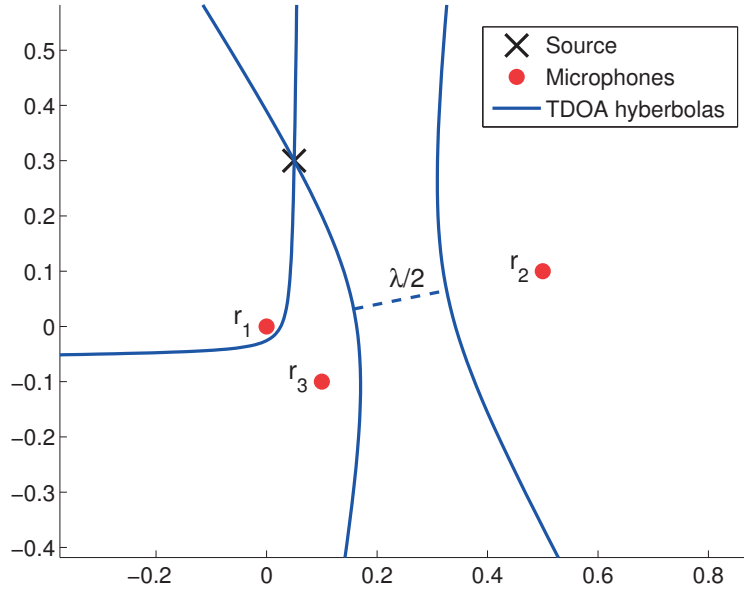


Figure 2: TDOA hyperbolas representing all feasible locations of a single source received by three sensors. As $||\mathbf{r}_2 - \mathbf{r}_1|| > \lambda/2$, spatial aliasing yields another hyperbola of feasible locations. And yet, in this case, there exists only one intersection between the hyperbolas and so the estimate may still be obtained unambiguously.

arise when, for each sensor pair, there exist another TDOA (and thus another k) which fulfills (9), giving rise to an additional hyperbolic curve of feasible points, also intersecting the hyperbolas for other sensor pairs. To identify such ambiguous cases, we first show that a feasible TDOA is restricted to an interval. Using the triangle inequality,

$$|\tau_m c| = \left| \|\mathbf{s} - \mathbf{r}_m\|_2 - \|\mathbf{s} - \mathbf{r}_1\|_2 \right| \leq \|\mathbf{r}_m - \mathbf{r}_1\|_2 \tag{11}$$

it is directly implied that the TDOA must satisfy

$$\tau_m c \in \left[-\|\mathbf{r}_m - \mathbf{r}_1\|_2 , \|\mathbf{r}_m - \mathbf{r}_1\|_2 \right] \tag{12}$$

i.e., is restricted by the sensor-sensor distance. And so, using (9), an estimate of $\arg b \in [-\pi, \pi]$ will map to any TDOA

$$\tau_m c = \frac{\lambda \arg b}{2\pi} + \lambda k \in \left[-\left\|\mathbf{r}_m - \mathbf{r}_1\right\|_2, \left\|\mathbf{r}_m - \mathbf{r}_1\right\|_2 \right] \tag{13}$$

where $k \in \mathbb{Z}$, and $\lambda = 2\pi c/\omega$ is the wavelength of the signal. Therefore, if the sensors are spaced by less than $\lambda/2$, the feasible τ_m is unique, and there is no ambiguity in the resulting estimates. If instead some sensors are spaced further apart than $\lambda/2$, then, for all such sensor pairs, there will be more than one feasible TDOA, thereby yielding as many hyperbolas indicating feasible source locations, with a minimum distance of $\lambda/2$ apart. Our main argument to relax the halv wavelength spacing limit is that, when using sufficiently many sensors, the feasible source locations are restricted to the intersection of many hyperbolas, which will, with a high probability, yield a unique solution. Consider an example illustrated in Figure 2, where a single source emits a 1000 Hz signal, which is recorded by three sensors. As shown in the figure, between sensors one and three, which are less than $\lambda/2$ apart, the source gives a single TDOA and a corresponding hyperbola, where the source may be located. Between sensors one and two, which are spaced by more than $\lambda/2$ apart, a second TDOA is feasible, λ/c apart from the true one, also fulfilling (13). However, as shown in the figure, the combined hyperbolas coincide in only a single feasible location, thus still allowing for an unambiguous estimate of the source location. Furthermore, for pitch signals, each overtone will yield a separate set of hyperbolas, which all must intersect to the same location, which further helps to avoid ambiguity. Modeling the attenuation between sensors also helps to avoid ambiguity. Examining the magnitude of the the complex amplitude in (9), we find that

$$|b_m| = \frac{d_1}{d_m}|a| \tag{14}$$

for each pair, consisting of the first and the m:th microphone, which limits $\mathbf{s}$ to lie on a circle. Using the same arguments as above, a feasible source location in terms of attenuation is thus the intersection of circles for all microphone pairs, and will further contribute to avoid spatial aliasing. Even if, despite of intersecting the feasible solutions for all harmonics in terms of both delay and attenuation, ambiguities still remain, then as more sensors are added to the array the set of possible locations quickly becomes small, and a unique solution generally exists,

even if not guaranteed. We thus deem that the imposed restriction on the array's geometry is mild.

3 Joint estimation of pitch and location

We proceed to detail the proposed two-step procedure to form reliable estimates of both the pitches and locations of the sources impinging on the array, without assuming detailed model knowledge of either the number of sources, K, the number of overtones for each source, L_k, the number of reflections experienced due to a possibly reverberant environment, S_k, or requiring knowledge about if sources are far- or near-field. In the first step, the magnitudes, phases, fundamental frequencies, and model orders of the present pitches are estimated, whereas, in the second step, the phase estimates are used to find the locations of these sources. Let

$$\mathbf{\Phi} = \left\{ \left\{ b_{k,\ell,m} \right\}_{\substack{\ell=1,\ldots,L_k \\ m=1,\ldots,M}}, \omega_k, L_k \right\}_{k=1,\ldots,K} \tag{15}$$

denote the set of unknown parameters to be determined in the first step. Minimizing the squared model residual in (7), an estimate of $\mathbf{\Phi}$ may thus be formed as

$$\hat{\mathbf{\Phi}} = \arg\min_{\mathbf{\Phi}} \sum_{t=1}^{N} \sum_{m=1}^{M} \left| y_m(t) - \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} b_{k,\ell,m} e^{i\omega_k \ell t} \right|^2 \tag{16}$$

Clearly, given the dimensionality of the problem, and the required model order estimation steps in order to determine K and L_k, this is a non-trivial problem, and needs to be modified to allow for an efficient solution, as is detailed below. Moving over to the second step, where the found magnitude and phase estimates, $\hat{b}_{k,l,m}$, are exploited to form estimates of the source locations, let

$$\mathbf{\Psi}_k = \left\{ \left\{ a_{k,\ell,s} \right\}_{\ell=1,\ldots,L_k}, \mathbf{s}_s \right\}_{s=1,\ldots,S_k} \tag{17}$$

be the amplitudes and coordinates for a present fundamental frequency k. The locations may be determined by minimizing the squared model residual in (8), i.e.,

$$\hat{\mathbf{\Psi}}_k = \arg\min_{\mathbf{\Psi}_k} \sum_{\ell=1}^{\hat{L}_k} \sum_{m=1}^{M} \left| \hat{b}_{k,\ell,m} - \sum_{s=1}^{S_k} a_{k,\ell,s} d_{k,m,s}^{-1} e^{-i\omega_k \ell \tau_{k,m,s}} \right|^2 \tag{18}$$

where $\tau_{k,m,s}$ and $d_{k,m,s}$ are functions of the location $\mathbf{s}_s$, as defined in (3) and (4). As before, this minimization is also non-trivial, requiring an estimate of S_k, and also needs to be modified to allow for a reasonably efficient solution. In the following, we will elaborate on the proposed modifications of the above minimizations. In order to do so, we first extend the sparse pitch estimation algorithm presented in [22, 23] to allow for multiple measurement vectors. In the second minimization, we then introduce a similar sparsity pattern to solve the localization problem. We begin by examining the extended pitch estimation algorithm.

3.1 Step 1: Sparse pitch estimation

Define the measurement matrix

$$\mathbf{Y} = \left[\begin{array}{ccc} \mathbf{y}(1) & \dots & \mathbf{y}(N) \end{array} \right]^T \tag{19}$$

where

$$\mathbf{y}(t) = \left[\begin{array}{ccc} y_0(t) & \dots & y_{M-1}(t) \end{array} \right]^T \tag{20}$$

denotes a sensor snapshot for each time point $t = 1, \dots, N$, with $(\cdot)^T$ being the transpose. The measurements may then be concisely expressed as

$$\mathbf{Y} = \sum_{k=1}^{K} \mathbf{W}_k \mathbf{B}_k + \mathbf{E} \tag{21}$$

where $\mathbf{E}$ denotes the combined noise term constructed similar to $\mathbf{Y}$, and

$$\mathbf{W}_k = \left[\begin{array}{ccc} \mathbf{w}_k^1 & \dots & \mathbf{w}_k^{L_k} \end{array} \right] \tag{22}$$

$$\mathbf{w}_k = \left[\begin{array}{ccc} e^{i\omega_k} & \dots & e^{i\omega_k N} \end{array} \right]^T \tag{23}$$

$$\mathbf{B}_k = \left[\begin{array}{ccc} \mathbf{b}_{k,1} & \dots & \mathbf{b}_{k,L_k} \end{array} \right]^T \tag{24}$$

$$\mathbf{b}_{k,\ell} = \left[\begin{array}{ccc} b_{k,\ell,1} & \dots & b_{k,\ell,M} \end{array} \right]^T \tag{25}$$

Reminiscent to the sparse estimation framework proposed in [18], we form an extended dictionary of feasible fundamental frequencies, $\omega_1, \dots, \omega_P$, where $P \gg K$, being chosen so large that K of these will reasonably well coincide with the true pitches in the signal. In the same manner, the number of harmonics of each

pitch is extended to an arbitrary upper level, say $L_{\max}$, for all dictionary elements. The signal model may thus be expressed as

$$\mathbf{Y} = \sum_{p=1}^{P} \mathbf{W}_p \mathbf{B}_k + \mathbf{E} = \mathcal{W}\mathcal{B} + \mathbf{E} \tag{26}$$

where the block dictionary matrices are formed by stacking the matrices such that

$$\mathcal{W} = \begin{bmatrix} \mathbf{W}_1 & \dots & \mathbf{W}_P \end{bmatrix} \tag{27}$$

$$\mathcal{B} = \begin{bmatrix} \mathbf{B}_1^T & \dots & \mathbf{B}_P^T \end{bmatrix}^T \tag{28}$$

Note from (26) that if the element (ℓ, r) of the matrix $\mathbf{B}_k$ is non-zero, the frequency $\ell\omega_k$ is present in the signal at sensor r. Furthermore, since we assume all sensors to receive essentially the same signal, although time-delayed, one may assume that for a harmonic signal, the rows off a non-zero $\mathbf{B}_k$ will either be non-zero, implying that the harmonic ℓ is present in the pitch, or zero, if the harmonic is missing. An appropriate criterion, that promotes a combination of model to data fit and the sparsity pattern just described, may thus be formed as

$$\begin{aligned} \underset{\mathcal{B}}{\text{minimize}} \Bigg\{ \frac{1}{2} \|\mathbf{Y} - \mathcal{W}\mathcal{B}\|_{\mathcal{F}}^2 + \lambda \sum_{p=1}^{P} \sum_{\ell=1}^{L_p} \left\|\mathbf{b}_{p,\ell}\right\|_2 \\ + \sum_{p=1}^{P} \gamma_p \left\|\mathbf{B}_p\right\|_{\mathcal{F}} \Bigg\} \end{aligned} \tag{29}$$

where two different kinds of group sparsities are imposed, and with $\|\cdot\|_{\mathcal{F}}$ denoting the Frobenius norm. This can be seen to be a generalization of the sparse group lasso to the multiple measurement case (see also [23, 26]). Here, the double sum of 2-norms in the second entry of the minimization should enforce sparsity in the solution in the rows of $\mathcal{B}$, and ideally only have as many non-zero rows as there are sinusoids in the signal. The third entry makes the solution (matrix) block sparse over the candidate pitches, penalizing the number of pitches with non-zero magnitude in the signal, ideally making them as many as there are pitches in the signal, i.e., K. Given an optimal point, $\hat{\mathcal{B}}$, the number of pitches is thus estimated as the number of non-zero matrices $\hat{\mathbf{B}}_k$, and, for each pitch, the number of harmonics, L_k, is estimated as the number of non-zero rows. The user parameters

$\lambda, \gamma_p \in \mathbb{R}_+$ weighs the fit of the solution to its vector and matrix sparsity, respectively. It is well known (see, e.g., [27]) that the amplitudes in the sparse estimate will be increasingly biased towards zero as sparse regularizers are increased. As we here intend to use both the estimated phases and the magnitudes, we propose to refine the amplitude estimates using a reweighting scheme similar to the one presented in [28]. This is accomplished by iteratively solving (29), such that at iteration $j+1$, one updates

$$\gamma_p^{(j+1)} = \frac{\gamma_p^{(0)}}{\left|\left|\hat{\mathbf{B}}_p^{(j)}\right|\right|_{\mathcal{F}} + \varepsilon} \tag{30}$$

where $\hat{\mathbf{B}}_p^{(j)}$ is block p of the optimal point for iteration j, and all $\gamma_p^{(0)}$ are set to be equal in the first iteration. As a result, the block matrices, $\hat{\mathbf{B}}_p^{(j)}$, which have a small Frobenius norm at iteration j will be penalized harder in the next step, whereas the ones that have a larger Frobenius norm will be penalized less, and as a result reducing the bias. The resulting algorithm can be seen as a sequence of iterative convex programs to approximate the concave $\log(\sum_{p=1}^{P} \gamma_p^{(0)} \left|\left|\mathbf{B}_p\right|\right|_{\mathcal{F}} + \varepsilon)$ penalty function [29], where ε is chosen as a small number to avoid numerical difficulties. The introduction of the reweighting yields sparser estimates due to the introduction of the log penalty [28, 30], and the resulting technique may be viewed as an alternative to using an information criterion (as was done in [23], to avoid spurious peaks caused by the signal model and data miss-match).

It is worth noting that as we are here focusing on localization, we have selected to use a somewhat simplistic audio model that ignores several important features in harmonic audio signals, such as issues of inharmonicities, pitch halvings and doublings, and of the commonly occurring forms of amplitude modulation exhibited by most audio sources (see also [14]). Clearly, the used model could be refined reminiscent to models such as the one used in [23,31], introducing a total variation penalty to each column of $\mathcal{B}$, and/or using an uncertainty volume to allow for inharmonicity. However, for localization purposes, these issues are of less concern, as halvings/doublings and/or amplitude modulations will not affect the below localization procedure more than marginally. Inharmonicity is more pressing, but we have in our numerical studies found that given the size of the calibration errors, the inharmonicity is not affecting the solution significantly, and in the interest of reducing the complexity, we have here opted to exclude this aspect from the estimator.

As for the selection of the tuning parameters, one may use, for example, cross validation techniques, although it may be noted that, in high SNR cases, one can often get good results by simply inspecting the periodogram and by then setting the tuning parameters appropriately (see also [23] for a further discussion on this issue). Furthermore, we note that in the case of different noise variances at each sensor in the array, the Frobenius norm in the first entry of the minimization criterion may be replaced with a weighed Frobenius norm. Finally, we note that non-Gaussian noise distributions can also be used as long as the negative log-likelihood is convex.

3.2 Step 2: Sparse localization

According to the signal model (7), $\hat{\mathcal{B}}$ will inherently contain the TDOA and attenuation for all reflections of any fundamental frequency present in the signal, which enables a range of post-processing steps to, for instance, estimate position, track, and/or calibrate the sensors. Here, we limit our attention to estimating the source positions. Let $\hat{\mathcal{B}}$ denote the solution obtained from minimizing (29), and consider a scenario where the sources are well separated in their pitch frequencies, and, initially, suffering from negligible reverberation, implying that $S_1 = \ldots = S_P = 1$. Then, the minimization in (18) may be seen as a generalization of the time-varying amplitude modulation problem examined in [32] (see also [11]) to the case of several realizations of the same signal, sampled at irregular time points, and with a different initial phase for each realization. Reminiscent to the solution presented in [11, p. 186], one may thus find the source locations, for far-field signals, for every pitch p with non-zero amplitudes in $\mathbf{B}_p$, as

$$\hat{\mathbf{s}}_p = \arg\max_{\mathbf{s}_p} \sum_{\ell=1}^{L_p} \left| \sum_{m=1}^{M} \hat{b}^2_{p,\ell,m} e^{-i2\omega_p \ell \tau_{p,\ell,m}} \right|^2 \tag{31}$$

where the TDOAs $\tau_{p,\ell,m}$ are found as a function of the source location $\mathbf{s}_p$, using (4). This minimization may be well approximated by 1-D searches over range and DOA (or over range, azimuth, and elevation in the 3-D case). Considering also reverberating room environments, wherein each of the pitches may appear as originating from many different locations, the minimization needs to be extended to allow for varying number of reflections, S_k. To allow for such reflections, we proceed to model every non-zero amplitude block from the pitch estimation step

as

$$\mathbf{B}_k = \sum_{s=1}^{S_k} \operatorname{diag}\left(\mathbf{a}_{k,s}\right) \mathbf{U}_{k,s} + \boldsymbol{\mathcal{E}}_k \tag{32}$$

with diag($\mathbf{x}$) denoting a diagonal matrix with the vector $\mathbf{x}$ along its diagonal, $\boldsymbol{\mathcal{E}}_k$ the combined noise term constructed in the same manner as $\mathbf{B}_k$, and

$$\mathbf{U}_{k,s} = \begin{bmatrix} \mathbf{u}_{k,s}^1 & \dots & \mathbf{u}_{k,s}^{\hat{L}_k} \end{bmatrix} \tag{33}$$

$$\mathbf{u}_{k,s} = \begin{bmatrix} \frac{e^{i\omega_k \tau_{k,1,s}}}{1} & \dots & \frac{e^{i\omega_k \tau_{k,M,s}}}{d_{k,M,s}/d_{k,m,s}} \end{bmatrix}^T \tag{34}$$

$$\mathbf{a}_{k,s} = \begin{bmatrix} \mathbf{a}_{k,1,s} & \dots & \mathbf{a}_{k,\hat{L}_k,s} \end{bmatrix}^T \tag{35}$$

where $\tau_{k,m,s}$ and $d_{k,m,s}$ are related to the source location as given by (3) and (4), respecively. Analogously to the above procedure for the pitch estimation, we then extend the dictionary of feasible source locations for the kth source, $\mathbf{s}_1, \dots, \mathbf{s}_{S_k}$, onto a grid of $Q \gg S_k$ candidate locations $\mathbf{s}_q$, for $q = 1, \dots, Q$, with Q chosen large enough to allow some of the introduced dictionary elements to coincide, or closely so, with the true source locations in the signal. Clearly, this may force Q to be very large. Striving to keep the size of the dictionary as small as possible, we consider grid points in polar coordinates, with equal resolution for all considered DOAs, and linearly spaced grid points over the distance in each DOA. Thus, we get a denser grid in the close proximity to the sensor array, where the resolution capacity is highest, and then a less and less dense grid for sources further away from the array. Finally, to also allow for far-field sources, one may include one dictionary element for each direction at an infinite range, for which, naturally, the attenuation effect may be disregarded, i.e., $d_{k,m,s} \triangleq 1$ for all sensors. Thus, we may estimate the source locations for the k:th pitch using a sparse modelling framework as

$$\begin{aligned} \underset{\mathbf{a}_{k,1},\dots,\mathbf{a}_{k,Q}}{\text{minimize}} \Bigg\{ & \frac{1}{2} \left\| \mathbf{B}_k - \sum_{q=1}^{Q} \operatorname{diag} \mathbf{a}_{k,q} \mathbf{U}_{k,q} \right\|_{\mathcal{F}}^2 \\ & + \sum_{q=1}^{Q} \varkappa_q \left\| \mathbf{a}_{k,q} \right\|_2 + \rho \sum_{q=1}^{Q} \left\| \mathbf{a}_{k,q} \right\|_1 \Bigg\} \end{aligned} \tag{36}$$

where, again, two types of sparsity is imposed on the solution. The 2-norm penalty term imposes sparsity to the blocks $\mathbf{a}_{k,q}$, i.e., penalizing the number of source locations present in the signal. Furthermore, the 1-norm term penalizes the number of harmonics, to allow for cases when some sources may have missing harmonics. Thus, here the number of sources is estimated as the number of nonzero blocks in an optimal point and any zero elements within a block corresponding to a missing harmonic. Here, $\varkappa_q, \rho \in \mathbb{R}_+$ are tuning parameters, controlling the amount of sparsity and the weight between sparsity in pitches and in harmonics, respectively, whereas the factor ρ is only used if two sources share the same fundamental frequency but differ in which harmonics are present. Finally, $\varkappa_q$ may be updated in the same manner as described in section III.A. As shown in the following section, the optimization problem in (29) and (36) are equivalent, so these tuning parameters may be set in a similar fashion.

4 Efficient implementation

It is worth noting that both the minimization in (29) and (36) are convex, as the tuning parameters are non-negative and all the functions are convex. Their solutions may thus be found using standard convex minimization techniques, e.g., using CVX [33, 34], SeDuMi [35], or SDPT3 [36]. Regrettably, such solvers will scale poorly both with increasing data length, the use of a finer grid for the fundamental frequencies, and with the number of sensors. Furthermore, such implementations are unable to utilize the full structure of the minimization, and may, as a result, be computationally cumbersome in practical situations. To alleviate this, we proceed to formulate a novel ADMM re-formulation of the minimizations, offering efficient and fast implementations of both minimizations. For completeness and to introduce our notation, we briefly review the main steps involved in an ADMM (we refer the reader to [37, 38] for further details on the ADMM). Considering the convex optimization problem

$$\underset{\mathbf{z}}{\text{minimize}}\, f(\mathbf{z}) + g(\mathbf{z}) \tag{37}$$

where $\mathbf{z} \in \mathbb{R}^p$ is the optimization variable, with $f(\cdot)$ and $g(\cdot)$ being convex functions. Introducing the auxiliary variable, $\mathbf{u}$ (37) may be equivalently be expressed as

$$\underset{\mathbf{z},\mathbf{u}}{\text{minimize}}\, f(\mathbf{z}) + g(\mathbf{u}) \quad \text{subject to } \mathbf{z} - \mathbf{u} = \mathbf{0} \tag{38}$$

Algorithm 1 The ADMM algorithm

1: Initiate $\mathbf{z} = \mathbf{z}_0$, $\mathbf{u} = \mathbf{u}_0$, and $k = 0$
2: **repeat**
3: $\quad \mathbf{z}_{k+1} = \underset{\mathbf{z}}{\operatorname{argmin}} f(\mathbf{z}) + \frac{\mu}{2}||\mathbf{z} - \mathbf{u}_k - \mathbf{d}_k||_2^2$
4: $\quad \mathbf{u}_{k+1} = \underset{\mathbf{u}}{\operatorname{argmin}}\, g(\mathbf{u}) + \frac{\mu}{2}||\mathbf{z}_{\mathbf{k+1}} - \mathbf{u} - \mathbf{d}_k||_2^2$
5: $\quad \mathbf{d}_{k+1} = \mathbf{d}_k - (\mathbf{z}_{\mathbf{k+1}} - \mathbf{u}_{\mathbf{k+1}})$
6: $\quad k \leftarrow k + 1$
7: **until** convergence

since at any feasible point $\mathbf{z} = \mathbf{u}$. Under the assumption that there is no duality gap, which is true for the here considered minimizations, one may solve the optimization problem via the dual function defined as the infimum of the augmented Lagrangian, with respect to $\mathbf{x}$ and $\mathbf{z}$, i.e., (see also [37])

$$L_\mu(\mathbf{z}, \mathbf{u}, \mathbf{d}) = f(\mathbf{z}) + g(\mathbf{u}) + \mathbf{d}^T(\mathbf{z} - \mathbf{u}) + \frac{\mu}{2}||\mathbf{z} - \mathbf{u}||_2^2$$

The ADMM does this by iteratively maximizing the dual function such that at step $k + 1$, one minimizes the Lagrangian for one of the variables, while holding the other fixed at its most recent value, i.e.,

$$\mathbf{z}_{k+1} = \arg\min_{\mathbf{z}} L_\mu(\mathbf{z}, \mathbf{u}_k, \mathbf{d}_k) \tag{39}$$

$$\mathbf{u}_{k+1} = \arg\min_{\mathbf{u}} L_\mu\left(\mathbf{z}_{k+1}, \mathbf{u}_k, \mathbf{d}_k\right) \tag{40}$$

Finally, one updates the dual variable by taking a gradient ascent step to maximize the dual function, resulting in

$$\tilde{\mathbf{d}}_{k+1} = \tilde{\mathbf{d}}_k - \mu\left(\mathbf{z}_{k+1} - \tilde{\mathbf{d}}_{k+1}\right) \tag{41}$$

where μ is the dual variable step size. The general ADMM steps are summarized in Algorithm 1, using the scaled version of the dual variable $\mathbf{d}_k = \tilde{\mathbf{d}}/\mu$, which is more convenient for implementation. Thus, in cases when steps 3 and 4 of Algorithm 1 may be carried out more efficiently than for the original problem, the ADMM may be useful to form an efficient implementation of the considered minimization.

It may be noted that the minimizations in (29) and (36) are rather similar, both containing an affine function in a Frobenius norm, as well as a sum of the norm of different subset of the variable. In fact, by using the vec operation, i.e., vectorization, both minimizations may be shown to be equivalent with the problem

$$\underset{\mathbf{z}}{\text{minimize}}\Bigg\{\frac{1}{2}\left\|\mathbf{y}-\mathbf{A}\mathbf{z}\right\|_2^2+\gamma\sum_{k=1}^{P}\left\|\mathbf{z}_k\right\|_2 + \delta\sum_{k=1}^{P}\sum_{g=1}^{G_k}\left\|\mathbf{z}_{k,g}\right\|_2\Bigg\} \tag{42}$$

where the complex variable $\mathbf{z}$ is given as

$$\mathbf{z}=\begin{bmatrix}\mathbf{z}_1^T & \dots & \mathbf{z}_P^T\end{bmatrix}^T \tag{43}$$

$$\mathbf{z}_k=\begin{bmatrix}\mathbf{z}_{k,1}^T & \dots & \mathbf{z}_{k,G_k}^T\end{bmatrix}^T \tag{44}$$

where each $\mathbf{z}_k$ and $\mathbf{z}_{k,g}$ denote complex vectors with G_k and O elements, respectively. For the minimization in (29), this implies that

$$\mathbf{y}=\text{vec}(\mathbf{Y}) \tag{45}$$

$$\mathbf{z}=\text{vec}(\mathcal{B}) \tag{46}$$

$$\mathbf{A}=\mathbf{I}\otimes\mathcal{W} \tag{47}$$

where $\otimes$ and $\mathbf{I}$ denote the Kronecker product and an M-dimensional identity matrix, respectively, with G_k being equal to the number of harmonics, L_k, and O equals the number of sensors, M. Similarly, for the minimization in (36),

$$\mathbf{y}=\text{vec}(\mathbf{B}_p) \tag{48}$$

$$\mathbf{z}=\mathbf{a}_k \tag{49}$$

$$\mathbf{A}=\tilde{\mathbf{V}}_k \tag{50}$$

where

$$\mathbf{a}_k=\begin{bmatrix}\mathbf{a}_{k,1}^T & \dots & \mathbf{a}_{k,Q}^T\end{bmatrix}^T \tag{51}$$

$$\tilde{\mathbf{V}}_k=\begin{bmatrix}\tilde{\mathbf{V}}_{k,1} & \dots & \tilde{\mathbf{V}}_{k,Q}\end{bmatrix} \tag{52}$$

and $\mathbf{V}_{k,q} = \mathbf{U}_{k,q} \otimes \mathbf{I}$, with $\tilde{\mathbf{V}}_{k,q}$ being formed by removing all columns from $\mathbf{V}_{k,q}$ that correspond to zeros in the vector $\text{vec}(\text{diag}(\mathbf{a}_{k,q}))$, and G_k being equal to L_k and O equals 1. Thus, we can formulate an ADMM solution for (42) that solves both problem (29) and (36). To that end, defining

$$f(\mathbf{z}) = \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2^2 \tag{53}$$

$$g(\mathbf{u}) = \gamma \sum_{k=1}^{P} \|\mathbf{u}_k\|_2 + \delta \sum_{k=1}^{P} \sum_{g=1}^{Q_k} \left\|\mathbf{u}_{k,g}\right\|_2 \tag{54}$$

yields a quadratic problem in step 3 in Algorithm 1, with a closed form solution given by

$$\mathbf{z}_{k+1} = \left(\mu I + \mathbf{A}^H \mathbf{A}\right)^{-1} \left(\mu \left(\mathbf{u}_k - \mathbf{d}_k\right) + \mathbf{A}^H \mathbf{y}\right)$$

with $(\cdot)^H$ denoting the Hermitian transpose, whereas in step 4, by solving the sub-differential equations (see [23] for further details), one obtains

$$\mathbf{u}_{k+1} = \mathcal{S}^o \left(\mathcal{S}^i \left(\mathbf{z}_k - \mathbf{d}_k, \varkappa/\mu\right), \delta/\mu\right) \tag{55}$$

where the shrinkage operators $\mathcal{S}^o$ and $\mathcal{S}^i$ are defined using the vector shrinkage operator $\mathcal{S}$, defined for any vector $\mathbf{v}$ and positive scalar ξ such that

$$\mathcal{S}(\mathbf{v}, \xi) = \mathbf{v} \left(1 - \xi/||\mathbf{v}||_2\right)^+ \tag{56}$$

where $(\cdot)^+$ is the positive part of the scalar, and

$$\mathcal{S}(\mathbf{z}, \xi)^o = \left[\begin{array}{cccc} \mathcal{S}^T(\mathbf{z}_1, \xi) & \ldots & \mathcal{S}^T(\mathbf{z}_P, \xi) \end{array} \right]^T \tag{57}$$

$$\begin{aligned} \mathcal{S}(\mathbf{z}, \xi)^i = \Big[& \mathcal{S}^T(\mathbf{z}_{1,1}, \xi) \quad \ldots \quad \mathcal{S}^T(\mathbf{z}_{1,G_1}, \xi) \quad \ldots \\ & \mathcal{S}^T(\mathbf{z}_{P,1}, \xi) \quad \ldots \quad \mathcal{S}^T(\mathbf{z}_{P,G_P}, \xi) \Big]^T \end{aligned} \tag{58}$$

The resulting algorithm is here termed the Harmonic Audio LOcalization using block sparsity (HALO) estimator.

5 Numerical comparisons

We proceed to examine the performance of the proposed estimator using both synthetic and measured audio signals, initially examining the performance using

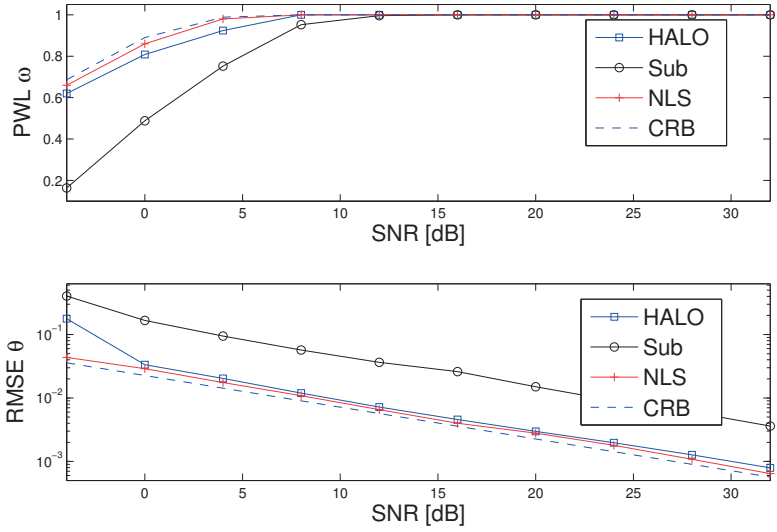


Figure 3: The PWL and RMSE for a single-pitch signal as compared with the optimal performance of an estimator reaching the CRB.

simulated audio signals. In the first examples, we limit ourselves to the case of letting a far-field signal impinge on a ULA. Figure 3 shows the percentage within limits (PWL), defined as the ratio of pitch estimates within a limit of ± 0.1 Hz from the true pitch, and the root mean square error (RMSE) of the DOA, defined as

$$\mathrm{RMSE}_{\vartheta} = \sqrt{\frac{1}{nK} \sum_{k=1}^{K} \sum_{i=1}^{n} \left(\hat{\vartheta}_{k,i} - \vartheta_k \right)^2} \tag{59}$$

where n denotes the number of Monte Carlo (MC) simulation estimates, and K the number of pitches in the signal, for the resulting estimates. For comparison, we use the Cramér-Rao lower bound (CRB), the NLS estimator, and the Sub approach (see [15] for further details on these methods and for the corresponding CRB). These results have been obtained using $n = 250$ MC simulations of a single pitch signal, with $\omega_1 = 220$ Hz and $L_1 = 7$ harmonics, impinging from $\vartheta_1 = -30^\circ$, where both the NLS and the Sub estimators have been al-

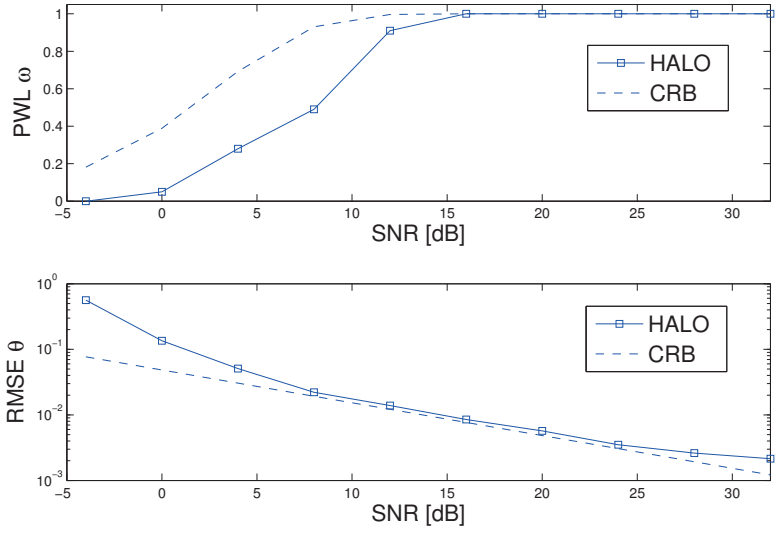


Figure 4: The PWL and RMSE for a multi-pitch signal with two pitches, as compared to the corresponding CRB.

lowed perfect a priori knowledge of both the number of sources and their number of harmonics, whereas the proposed method is allowed no such knowledge. As is clear from the figures, the HALO method offers a preferable performance as compared to the Sub estimator, and only marginally worse than the NLS estimator, in spite of both the latter being allowed perfect model orders information. Here, the number of sensors in the array was $M = 5$ and we used 20 ms of data sampled at $f_s = 8820$ Hz, i.e., $N = 176$ samples. Furthermore, $c = 343$ m/s and $d = c/f_s \approx 0.0389$ m. We proceed to consider the case of multi-pitch signals impinging on the array. Measuring as in the single-pitch case, we now form a multi-pitch signal with two pitches and fundamental frequencies $\{150,\ 220\}$ Hz containing $\{6,\ 7\}$ harmonics, coming from $\vartheta_1 = -30°$. Figure 4 shows the RMSE and PWL estimates, as obtained using 250 MC simulations, clearly showing that the HALO estimator is able to reach close to optimal performance also in this case. Here, no comparison is made with the NLS and Sub estimators of [15] as these are restricted to the single-pitch case. Throughout these evaluations, we have used $L_{\max} = 15$. Also, as the resulting estimates were found to be appropri-

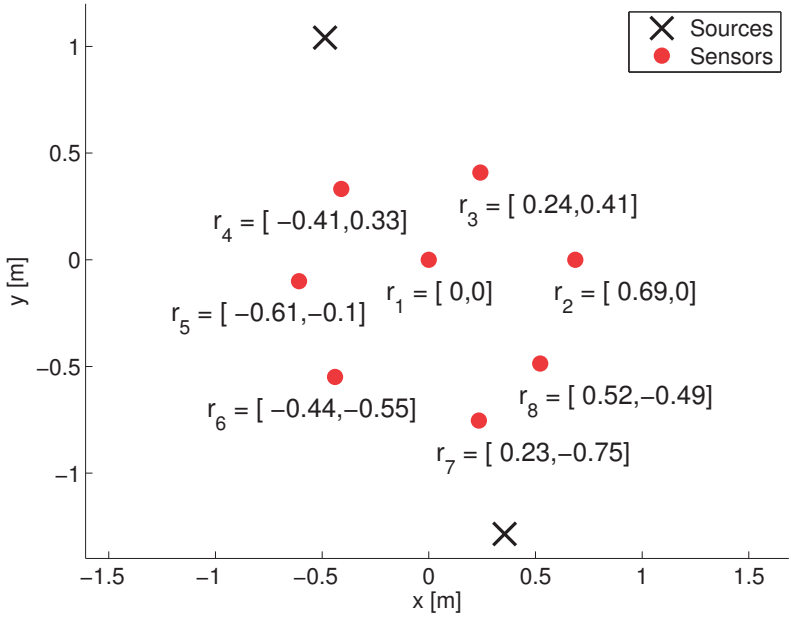


Figure 5: The two-source and eight-sensor layout in 2-D. The position of each sensor, shown in the plot with carthesian coordinates as $r_m = [x, y]$, was obtained in an *a priori* calibration step.

ately sparse when using only the convex penalties, and no reweighing steps were used. We next proceed to examine real measured signals. The measurements were made in an anechoic chamber, approximately $4 \times 4 \times 3$ meters in size, with the sensors and speakers located as shown in Figures 5 and 7. Two speakers were placed at locations (in polar coordinates) $\mathbf{s}_1 = [\vartheta_1, R_1] = [115.03^\circ, 1.15\,\text{m}]$ and $\mathbf{s}_2 = [\vartheta_2, R_2] = [-74.53^\circ, 1.33\,\text{m}]$, with respect to the central microphone, respectively. The positions of the sensors were determined by placing them together with the sources, using the acoustic method detailed in [39]. This is done by calibrating the sensors with a single moving source, using a correlation-based methodology. The positions were also confirmed via a computer vision approach were the positions were found by taking several photos and reconstructing the environment. The maximum deviation in position between these methods was

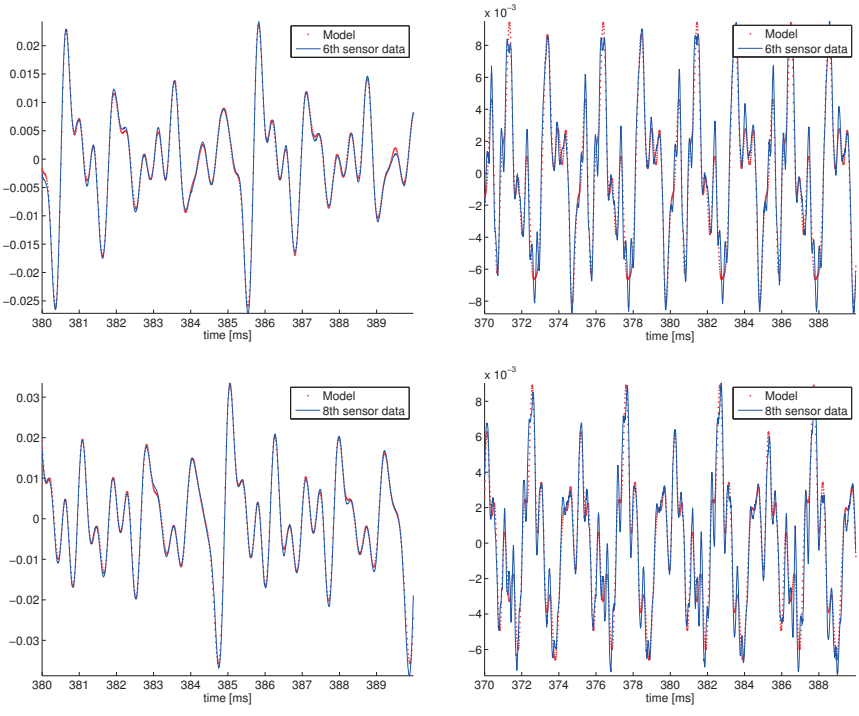


Figure 6: Time-domain data (lined) and estimated signal reconstruction (dotted) for the 6:th sensor (top two) and 8:th sensor (bottom two), for two different signals. The left two subfigures display a voice signal saying the phonetic 'a' in 'why', while the right two subfigures display a violin signal.

less than 1 cm. As the spatial impulse responses of the microphones were deemed to be reasonably omni-directional, as well as roughly the same for all the microphones, no further calibration of the sensor gains were performed. The positions were then projected onto a 2-D plane using principal component analysis. In order to illustrate the HALO estimator's ability to handle an environment with the same pitch signal originating from different sources, as a much simplified proof of concept for a reverberating room environment, we examine a case with two sources playing the same signal content. Both sources plays a (TIMIT) recording of a female voice saying 'Why were you away a year, Roy?', timing the source's playback so that the recording at each microphone sounds slightly echoic. The

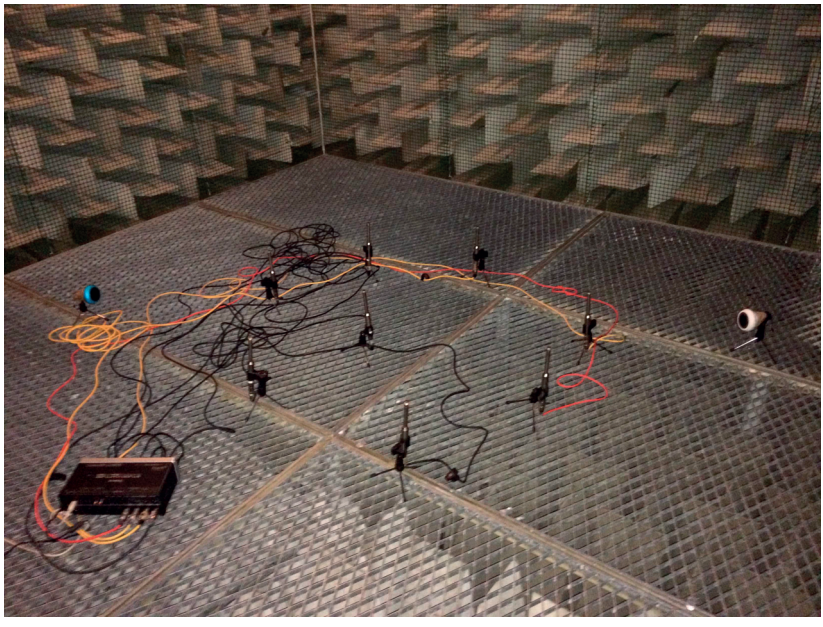


Figure 7: A photo showing the experimental setup in the anechoic chamber, where eight sensors are used to record two coherent sources.

eight microphones all record at a sample rate of $f_s = 96$ kHz. The data is then divided into time frames of 10 ms, i.e., $N = 960$ samples, which allow each frame to be well modelled as being stationary. Examining a part of the speech that is voiced, arbitrarily selected as the frame starting 380 ms into the recording, about when the voice is saying the voiced phonetic sound 'a' in 'why', Figure 4 show the signal measured at the 6th and 8th microphone, respectively, together with the reconstructed signal obtained from the pitch estimation step in HALO, obtained as

$$\hat{\mathbf{Y}} = \mathcal{W}\hat{\mathcal{B}} \tag{60}$$

using the resulting model orders and estimates. The estimator indicate that the signal contains a single pitch at $\hat{\omega}/2\pi = 193.5$ Hz, having $\hat{L} = 12$ overtones. As is clear from the figures, the estimator is well able to model the measured signal in spite of the presence of the reverberation. Comparing the figures, one may also note the time shift between the sensors, due to the additional time-delay for the

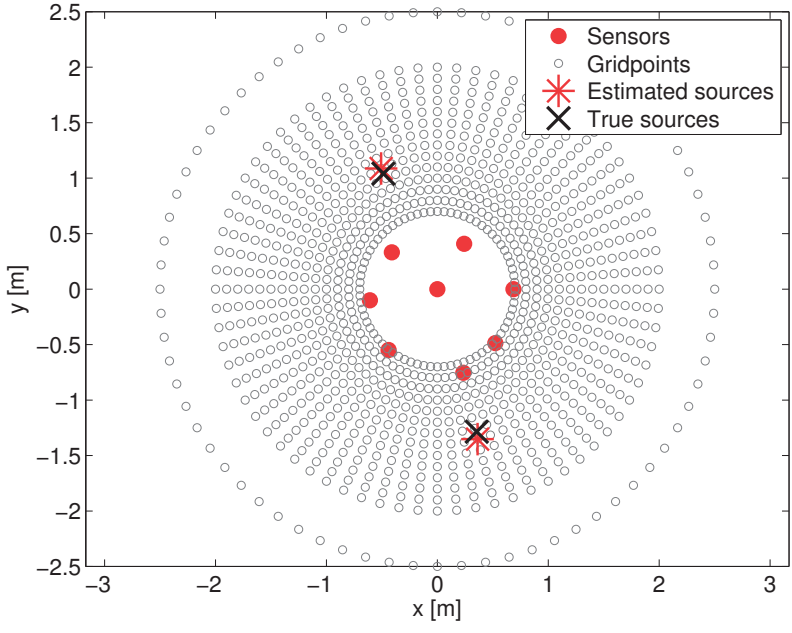


Figure 8: The experimental setup in the anechoic chamber, showing the sensor and loudspeaker locations, the considered dictionary grid, as well as the resulting estimated as obtained by the proposed algorithm.

wavefront traveling between them, corresponding to a linear combination of the two sources, each with their particular TDOA and attenuation. It should also be noted that the signals are not simply time-shifted versions of each other due to the room environment and the attenuation of the signal when propagating in space (which would thus create problems for an estimator based on the cross-correlation between the sensors). The same situation is illustrated in left two subfigures in Figure 4, showing the results when the signal source is replaced with that of a part of a (SQAM) violin signal. Again, the estimator can be seen to be able to well model the impinging signals, which is estimated as being a single pitch with the fundamental frequency $\hat{\omega}/2\pi = 198.0$ Hz, containing $\hat{L} = 14$ harmonics. In order to examine the location estimation, we construct a 2-D grid of feasible locations, chosen such that the space is discretized into 1008 points, consisting

of 72 directions between $[-180°, 180°)$, spaced every 5°, where each direction allows for ranges $R \in [0.7, 2]$ m, spaced 10 cm apart. The resulting grid is shown in Figure 8, which is roughly covering the entirety of the anechoic chamber. To also allow for far-field sources, a range of $R = \infty$ is also added to the grid for each direction, which we have chosen to illustrate by the outer circle in Figure 8. For these far-field grid points, the time-delays are instead computed as (see also [9])

$$\tau_m = \frac{\min_{\mathbf{z}} \|\mathbf{r}_m - \boldsymbol{\ell}(\mathbf{z})\|_2}{c} \tag{61}$$

for a location $\mathbf{z}$ on the line $\boldsymbol{\ell}(\cdot)$, which is perpendicular to the DOA and goes through $\mathbf{r}_1$. The figure also shows the locations for the sensors and the sound sources, as well as the estimated locations, as obtained by the second step of the HALO estimator (the estimated locations were identical for both audio recordings). The errors in position were 5 cm in range for each source, where a bias, overestimating the range, accounts for almost all of the error. On the other hand, as shown in the figure, the angles of the sources ϑ were accurately estimated. The overestimation of the range may to a large extent likely be explained by poor scaling when calibrating the array. One may note that, for localization in 3-D, the size of the dictionary will increase significantly as compared to the 2-D case used for numerical illustration in this paper. For the case above, if also the elevation angle is to be considered, having the same resolution as for the azimuth, this would yield a dictionary of 72 576 atoms. Although much larger, a sparse modeling systems of this size is by no mean impractical to work with. Also, our investigations show that a less dense location grid may be used, whereafter a zooming step can be taken. Finally, we illustrate the algorithm's performance using MC simulations, using simulated sources, one near- and one far-field source, detailed with $\omega = [200, 270]$ Hz, $L = [15, 14]$ harmonics, impinging from $\vartheta = [110°, -70°]$ at $R = [1.3, \infty]$ m, respectively. The sensors are placed as a uniform circular array, with 7 sensor placed evenly at a 0.5 m radius, together with a sensor being placed in the center of the array. First, we examine the position estimates using a coarse spacing for the possible sources, spaced by 11 cm in angle for all angles $\vartheta \in [-180°, 180°)$, and spaced by 10 cm in range, at $R \in [0.7, 3]$ m. In each MC simulation, the true location of each source was offset by a (uniformly distributed) range offset of plus minus one half the grid spacing. In all simulations, we ensured that neither of the sources were placed on a dictionary grid point. Figure 9 shows the PWL for the angle and range estimates, where the limit is chosen to

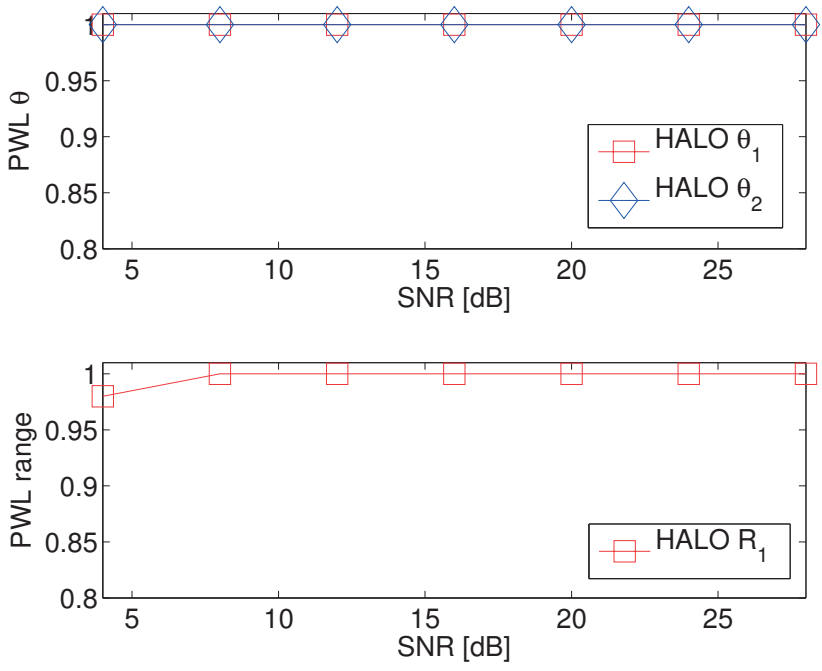


Figure 9: The PWL ratio for the angle and range estimates when using a coarsely spaced grid, indicating the ratio of estimates that are within ± 10 cm in range, and $\pm 5^\circ$ in angle.

be the same as the grid spacing, i.e., the ratio of estimates that are within ± 10 cm in range, and $\pm 5^\circ$ in angle. As seen from the figure, the both the range and the DOA of the sources are well determined, indicating that even with the use of a coarse grid, one is able to obtain reliable estimates. Proceeding to instead using a fine grid, the coarse estimates may then be refined by zooming in the grid over the found locations. Using a dictionary of the same size as the coarse grid, although centered around the found estimates, yields a resolution of ± 5 mm in range and $\pm 0.25^\circ$ in angle. Figure 10 shows the resulting RMSE for the angle and pitch estimates on the finer grid, as compared to the CRB (given in the Appendix). As can be seen from the figure, the RMSE (and the corresponding CRB) of the far-field source is somewhat lower than the near-field source, although both sources

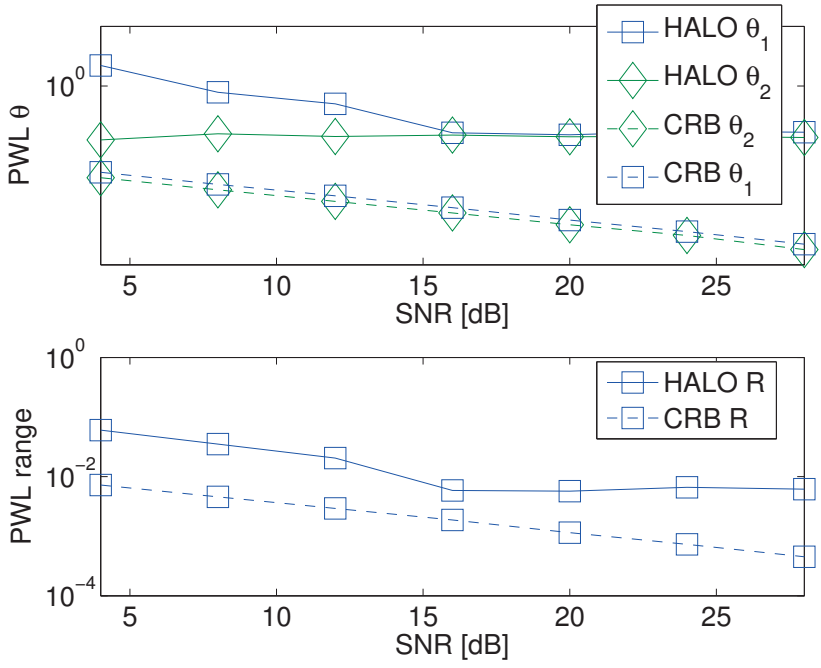


Figure 10: The RMSE for the angle and range estimates when using a finely spaced grid, indicating the ratio of estimates that are within ± 5 mm in range, and $\pm 0.25°$ in angle.

are well estimated, yielding a performance close to being optimal. The slight offset from the CRB is deemed to be largely due to a small bias in the final estimates, resulting from the smoothness of the approximative cost function resulting from the additive convex constraints. As is clear from the above presentation, the HALO estimate exploits the harmonic structure in the received audio signals to position the sources, using the pitch estimates to form a sparse estimate over a wide range of feasible positions. Obviously, most audio signals are not harmonic at all times, and the estimator should thus be used in combination with a tracking technique, possibly using a methodology reminiscent to the one presented in [40, 41]. In such a tracking scheme, the estimated pitch amplitudes should be used as an indicator for the reliability of the obtained positioning, yielding poor or maybe even erroneous positioning for unvoiced or non-harmonic audio sig-

nals, whereas reasonably accurate positions may be expected for more harmonic signals.

6 Conclusions

In this paper, we have presented an efficient sparse modeling approach for localizing harmonic audio sources using a calibrated sensor array. Assuming that each harmonic components in each pitch can only come from one source, the localization estimate is based on the phase and attenuation information for all of the harmonics jointly. The resulting model phases and attenuation will then depend on the source location. By using sparse modeling, the method inherently estimates both the number of sources, the number of harmonics in each source, as well as the extent of a possibly occurring reverberation. The effectiveness of the resulting algorithm is shown using both simulated and measured audio sources.

7 Acknowledgements

The authors wish to express their gratitude to the Signal Processing Group at Electrical and Information Technology, Lund University, for allowing use of their experimental facilities, as well as to the authors of [15] for sharing their Matlab implementations.

8 Appendix: The Cramér-Rao lower bound

In this appendix, we briefly summarize the Cramér-Rao lower bound (CRB) for the examined localization problem. As is well known, under the assumption of complex circularly symmetric Gaussian distributed noise, the Slepian-Bangs formula yields [11, p. 382]

$$\left[P_{cr}^{-1}\right]_{ij} = \text{trace}\left[\mathbf{\Gamma}^{-1}\mathbf{\Gamma}_i'\mathbf{\Gamma}^{-1}\mathbf{\Gamma}_j'\right] + 2\mathcal{R}\left[\boldsymbol{\mu}_i'^H\mathbf{\Gamma}^{-1}\boldsymbol{\mu}_j'\right] \tag{62}$$

where $\mathcal{R}$ denotes the real part of a complex scalar, $\mathbf{\Gamma}$ the covariance matrix of the noise process, and $\boldsymbol{\mu}$ is the deterministic signal component, with $\mathbf{\Gamma}_i'$ and $\boldsymbol{\mu}_i'$ denoting the derivative of $\mathbf{\Gamma}$ and $\boldsymbol{\mu}$ with respect to element i of the parameter vector, respectively. For the case of uncorrelated noise with a known variance σ^2,

this simplifies to

$$\left[P_{cr}^{-1}\right]_{ij} = 2\mathcal{R}\left[\boldsymbol{\mu}_i'^H \boldsymbol{\mu}_j'\right]/\sigma^2 \tag{63}$$

Using the assumed signal model as measured at sensor m, stacking the the observations as in (19), and then using the vec operator on the resulting matrix results, one obtains the $\boldsymbol{\mu}$ function needed for the CRB calculations. Here, the parameters to be estimated are

$$\boldsymbol{\Delta} = \left\{ \left\{ a_{k,\ell}, \varphi_{k,\ell} \right\}_{\ell=1,\ldots,L_k}, \omega_k, \vartheta_{s,k}, R_{s,k} \right\}_{\substack{s=1,\ldots,S \\ k=1,\ldots,K}} \tag{64}$$

Clearly, the resulting function may easily be derivated with respect to the magnitude, frequency and phase parameters. However, since the location parameter, $\vartheta_{s,k}$ and $R_{s,k}$, enter into the expression in a complicated manner depending on the sensor geometry, the corresponding derivatives are not straight forward for an arbitrary array. For this reason, for the considered array geometries, we here simply approximate the resulting expressions using numerically differentiated expressions.

References

[1] B. Champagne, S. Bedard, and A. Stephenne, "Performance of time-delay estimation in the presence of room reverberation," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 2, pp. 148–152, Mar 1996.

[2] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberent rooms," in *Microphone Arrays: Techniques and Applications*, M. Brandstein and D. Ward, Eds., pp. 157–180. Springer-Verlag, New York, 2001.

[3] T. Gustafsson, B. D. Rao, and M. Trivedi, "Source localization in reverberant environments: modeling and statistical analysis," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 791–803, Nov 2003.

[4] E. Kidron, Y. Y. Schechner, and M. Elad, "Cross-modal localization via sparsity," *IEEE Trans. Signal Process.*, vol. 55, no. 4, pp. 1390–1404, April 2007.

[5] M. D. Gillette and H. F. Silverman, "A linear closed-form algorithm for source localization from time-differences of arrival," *IEEE Signal Processing Letters*, vol. 15, pp. 1–4, 2008.

[6] K. C. Ho and M. Sun, "Passive source localization using time differences of arrival and gain ratios of arrival," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 464–477, Feb 2008.

[7] X. Alameda-Pineda and R. Horaud, "A geometric approach to sound source localization from time-delay estimates," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 6, pp. 1082–1095, June 2014.

[8] H. F. Silverman and S. E. Kirtman, "A two-stage algorithm for determining talker location from linear microphone array data," *Computer Speech & Language*, vol. 6, no. 2, pp. 129 – 152, 1992.

[9] H. Krim and M. Viberg, "Two Decades of Array Signal Processing Research," *IEEE Signal Process. Mag.*, pp. 67–94, July 1996.

[10] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part IV, Optimum Array Processing*, John Wiley and Sons, Inc., 2002.

[11] P. Stoica and R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, N.J., 2005.

[12] J. Benesty, M. Sondhi, M. Mohan, and Y. Huang, *Springer handbook of speech processing*, Springer, 2008.

[13] N. H. Fletcher and T. D. Rossing, *The Physics of Musical Instruments*, Springer-Verlag, New York, NY, 1988.

[14] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, 2009.

[15] J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Nonlinear Least Squares Methods for Joint DOA and Pitch Estimation," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 21, no. 5, pp. 923–933, 2013.

[16] S. Gerlach, S. Goetze, J. Bitzer, and S. Doclo, "Evaluation of joint position-pitch estimation algorithm for localising multiple speakers in adverse acoustical environments," in *Proc. German Annual Conference on Acoustics (DAGA)*, Düsseldorf, Germany, 2011, vol. Mar. 2011, pp. 633–634.

[17] J. X. Zhang, M. G. Christensen, S. H. Jensen, and M. Moonen, "Joint DOA and Multi-pitch Estimation Based on Subspace Techniques," *EURASIP J. on Advances in Signal Processing*, vol. 2012, no. 1, pp. 1–11, 2012.

[18] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, Seoul, jul 2008, pp. 10225–10229.

[19] 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, March 1997.

[20] M. D. Plumbley, S. A. Abdallah, T. Blumensath, and M. E. Davies, "Sparse representations of polyphonic music," *Signal Processing*, vol. 86, no. 3, pp. 417–431, March 2006.

[21] M. Genussov and I. Cohen, "Multiple fundamental frequency estimation based on sparse representations in a structured dictionary," *Digit. Signal Process.*, vol. 23, no. 1, pp. 390–400, Jan. 2013.

[22] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Estimating Multiple Pitches Using Block Sparsity," in *38th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vancouver, May 26–31, 2013.

[23] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[24] S. L. Marple, "Computing the discrete-time "analytic" signal via FFT," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2600–2603, September 1999.

[25] T. Ballal and C.J. Bleakley, "DOA Estimation of Multiple Sparse Sources Using Three Widely-Spaced Sensors," in *Proceedings of the 17th European Signal Processing Conference*, 2009.

[26] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A Sparse-Group Lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[27] M. Elad, *Sparse and Redundant Representations*, Springer, 2010.

[28] E. J. Candes, M. B. Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted l_1 Minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec. 2008.

[29] L. Qing, Z. Wen, and W. Yin, "Decentralized jointly sparse optimization by reweighted ell-q minimization," *Signal Processing, IEEE Transactions on*, vol. 61, no. 5, pp. 1165–1170, March 2013.

[30] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Comm. Pure Appl. Math.*, vol. 63, 2010.

[31] N. R. Butt, S. I. Adalbjörnsson, S. D. Somasundaram, and A. Jakobsson, "Robust Fundamental Frequency Estimation in the Presence of Inharmonicities," in *38th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vancouver, May 26–31, 2013.

[32] O. Besson and P. Stoica, "Exponential signals with time-varying amplitude: parameter estimation via polar decomposition," *Signal Processing*, vol. 66, pp. 27–43, 1998.

[33] Inc. CVX Research, "CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta," `http://cvxr.com/cvx`, Sept. 2012.

[34] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pp. 95–110. Springer-Verlag Limited, 2008, http://stanford.edu/∼boyd/graph_dcp.html.

[35] J. F. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11-12, pp. 625–653, August 1999.

[36] R. H. Tutuncu, K. C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using SDPT3," *Mathematical Programming Ser. B*, vol. 95, pp. 189–217, 2003.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[38] N. Parikh and S. Boyd, "Proximal Algorithms," *Found. Trends Optim.*, vol. 1, pp. 127–239, 2014.

[39] Z. Simayijiang, F. Andersson, Y. Kuang, and K. Åström, "An automatic system for microphone self-localization using ambient sound," in *European Signal Processing Conference (Eusipco 2014)*, 2014.

[40] I. Potamitis, H. Chen, and G. Tremoulis, "Tracking of multiple moving speakers with multiple microphone arrays," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 520–529, Sept 2004.

[41] D. Gatica-Perez, G. Lathoud, J. Odobez, and I. McCowan, "Audiovisual Probabilistic Tracking of Multiple Speakers in Meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 2, pp. 601–616, Feb 2007.

B

Paper B

An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization

Filip Elvander, Ted Kronvall, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

This work treats multi-pitch estimation, and in particular the common misclassification issue wherein the pitch at half the true fundamental frequency, the sub-octave, is chosen instead of the true pitch. Extending on current group LASSO-based methods for pitch estimation, this work introduces an adaptive total variation penalty, which enforces both group- and block sparsity, as well as deals with errors due to sub-octaves. Also presented is a scheme for signal adaptive dictionary construction and automatic selection of the regularization parameters. Used together with this scheme, the proposed method is shown to yield accurate pitch estimates when evaluated on synthetic speech data. The method is shown to perform as good as, or better than, current state-of-the-art sparse methods while requiring fewer tuning parameters than these, as well as several conventional pitch estimation methods, even when these are given oracle model orders. When evaluated on a set of ten musical pieces, the method shows promising results for separating multi-pitch signals.

Key words: Multi-pitch estimation, block sparsity, adaptive sparse penalty, self-regularization, ADMM

1 Introduction

Pitch estimation is a problem arising in a variety of fields, not least in audio processing. It is a fundamental building block in several music information retrieval applications, such as automatic music transcription, i.e., automatic sheet music generation from audio (see, e.g., [1], [2]). Pitch estimation could also be used as a component in methods for cover song detection and music querying, possibly improving currently available services. For example, the popular query service Shazam [3] operates by matching hashed portions of spectrograms of user-provided samples against a large music database. As a change of instrumentation would alter the spectrogram of a song, such algorithms can only identify recordings of a song that are very similar to the actual recording present in the database. Thus, services such as Shazam might fail to identify, e.g., acoustic alternate versions of rock songs. A query algorithm based on pitch estimation could on the other hand correctly match the acoustic version to the original electrified one as it would recognize, e.g., the main melody.

The applicability of pitch estimation to music is due to the fact that the notes produced by many instruments used in Western tonal music, e.g., woodwind instruments such as the clarinet, exhibit a structure that is well modeled using a harmonic sinusoidal structure [4]. However, for some plucked stringed instruments, such as the guitar and the piano, the tension of the string results in the harmonics deviating from perfect integer multiples of the fundamental frequency, a phenomenon called inharmonicity. For some instruments, such as the piano, there are models describing the structure of the inharmonicity based on physical properties of the instrument [5]. Such signals require agile pitch estimation algorithms allowing for this form of deviations (see, e.g., [6–8]). In this work, we will assume such deviations to be small, although noting that one may extend the here presented work along the lines in [6–8].

Estimating the fundamental frequencies of multi-pitch signals is generally a difficult problem. There are many methods available, see, e.g., [9], but most of them require *a priori* model order knowledge, i.e., they require knowledge of the number of pitches present in the signal, as well as the number of active harmonics for each pitch.[1] Three such methods will be used in this work as reference estimators. The first method, here referred to as ORTH, exploits orthogonality

[1] It may be noted that, generally, obtaining correct model order information is a most challenging problem, with the model order estimates strongly affecting the resulting performance of the estimator.

between the signal and noise subspaces to form pitch frequency estimates. The second method is an optimal filtering method based on the Capon estimator, and is therefore here referred to as Capon. The third method is an approximate non-linear least squares method, here referred to as ANLS [10–12] (see also [9] for an overview of these methods). Methods not requiring *a priori* model order knowledge have also been proposed. For example, Adalbjörnsson et al. [13] use a sparse dictionary representation of the signal and regularization penalties to implicitly choose the model order. A similar, but less general, method was introduced in [14], which used a dictionary specifically tailored to piano notes for estimating pitch frequencies generated by pianos. Other source specific methods include [15], [16]. In [17], the author proposes a sparsity-exploiting method, where the dictionary atoms are learned from databases of short-time Fourier transforms of musical notes. A similar idea is used in [18] for pitch-tracking in music. In [16], [19], pitch estimation is based on the assumption of spectral smoothness, i.e., the amplitudes of the harmonics within a pitch are assumed to be of comparable magnitude.

Another field of research is performing multi-pitch estimation, often in the context of automatic music transcription, by decomposing the spectrogram of the signal into two matrices, one that describes the frequency content of the signal and one that describes the time activation of the frequency components. This method makes use of the non-negative matrix factorization, first introduced in this context in [20] and since then widely used, such as in, e.g., [21]. There are also more statistical approaches to multi-pitch estimation, posing the estimation as a Bayesian inference problem (see, e.g., [22]).

The approach to multi-pitch estimation presented in this work is to solve the problem in a group sparse modeling framework, which allows us to avoid making explicit assumptions on the number of pitches, or on the number of harmonics in each pitch. Instead, the number of components in the signal is chosen implicitly, by the setting of some tuning parameters. These tuning parameters determine how appropriate a given pitch candidate is to be present in the signal and may be set using cross-validation, or by using some simple heuristics. The sparse modeling approach has earlier been used for audio (see, e.g., [23]), and specifically for sinusoidal components in [24]. We extend on these works by exploiting the harmonic structure of the signals in a block sparse framework, where each block represents a candidate pitch. A similar method was introduced in [13], where block sparsity was enforced using block-norms, penalizing the number of active pitches.

As the block-norm penalty, under some circumstances, cannot distinguish a true pitch from its sub-octave, i.e., the pitch with half the true fundamental frequency, the method is also complemented by a total variation penalty, which is shown to solve such issues. Total variation penalties are often applied in image analysis to obtain block-wise smooth image reconstructions (see, e.g., [25]). For audio data, one can similarly assume that signals often are block-wise smooth, as the harmonics of a pitch are expected to be of comparable magnitude [19]. Enforcing this feature will specifically deal with octave errors, i.e., the choosing of the sub-octave instead of the true pitch, as, in the noise free case, only every other harmonic of the sub-octave will have non-zero power. In this paper, we show that a total variation penalty, in itself, is enough to enforce a block sparse solution, if utilized efficiently. More specifically, by making the penalty function adaptive, we may improve upon the convex approximation used in [13], allowing us to drop the block-norm penalty altogether, and so reduce the number of tuning parameters. In some estimation scenarios, e.g., when estimating chroma using the approach in [26], this would simplify the tuning procedure significantly.

Furthermore, we show that the proposed method performs comparably to that of [13], albeit with the notable improvement of requiring fewer tuning parameters. The method operates by solving a series of convex optimization problems, and to solve these we present an efficient algorithm based on the alternating direction method of multipliers (ADMM) (see, e.g., [27] for an overview of ADMM in the context of convex optimization). As the proposed method requires two tuning parameters to operate, we also present a scheme for automatic selection of appropriate model orders, thereby avoiding the need of user-supplied parameters.

The remainder of this work is organized as follows; in the following section, we introduce the signal model, followed in Section 3 by the proposed estimation algorithm. Section 4 summarizes the efficient ADMM implementation whereas Section 5 examines how to adaptively choose the regularization parameters. Numerical results illustrating the achieved performance are presented in Section 6. Finally, Section 7 concludes upon the work.

2 Signal model

Consider a complex-valued[2] signal consisting of K pitches, where the kth pitch is constituted by a set of L_k harmonically related sinusoids, defined by the component having the lowest frequency, ω_k, such that

$$x(t) = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} a_{k,\ell} \mathrm{e}^{i\omega_k \ell t} \tag{1}$$

for $t = 1, \ldots, N$, where $\omega_k \ell$ is the frequency of the ℓth harmonic in the kth pitch, and with the complex number $a_{k,\ell}$ denoting its magnitude and phase. The occurrence of such harmonic signals is often in combination with non-sinusoidal components, such as, for instance, colored broadband noise or non-stationary impulses. In this work, only the narrowband components of the signal are part of the signal model, such that all other signal structures, including the signal's timbre and the background noise, are treated as part of an additive noise process, $e(t)$.

In general, selecting model orders in (8) may be a daunting task, with both the number of sources, K, and the number of harmonics in each of these sources, L_k, being unknown, as well as often being structured such that different sources may have spectrally overlapping overtones. In order to remedy this, this work proposes a relaxation of the model onto a predefined grid of $P \gg K$ candidate fundamentals, each having $L_{\max} \geq \max_k L_k$ harmonics. Here, $L_{\max}$ should be selected to ensure that the corresponding highest frequency harmonic is limited by the Nyquist frequency, and could thus vary depending on the considered candidate frequency (see also [13]). For notational simplicity, we will hereafter, without loss of generality, use the same $L_{\max}$ for all candidate frequencies. Assume that the candidate fundamentals are chosen so numerous and so closely spaced that the approximation

$$x(t) \approx \sum_{p=1}^{P} \sum_{\ell=1}^{L_{\max}} a_{p,\ell} \mathrm{e}^{i\omega_p \ell t} \tag{2}$$

holds reasonably well. As only K pitches are present in the actual signal, we want to derive an estimator of the amplitudes $a_{p,\ell}$ such that only few, ideally $\sum_{k=1}^{K} L_k$,

[2] For notational simplicity and computational efficiency, we here use the discrete-time analytical signal formed from the measured (real-valued) signal (see, e.g., [9], [28]).

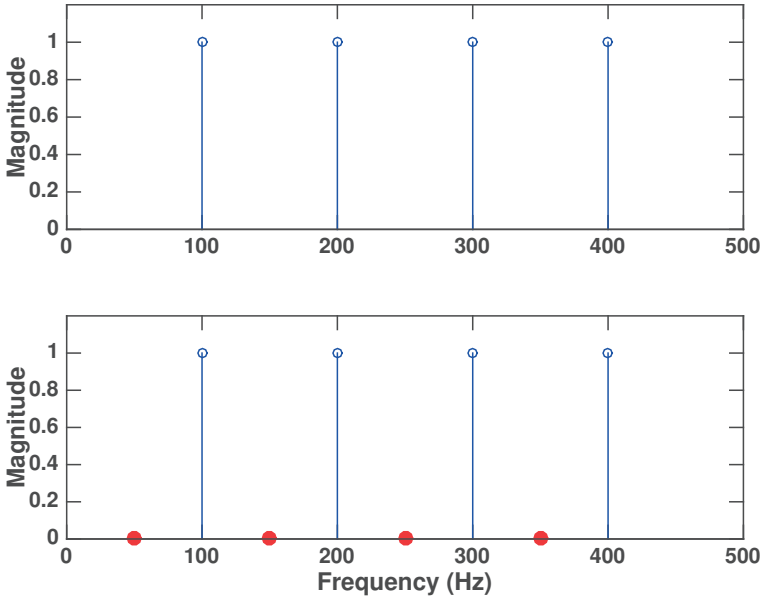


Figure 1: The upper picture depicts a pitch with fundamental frequency 100 Hz and four harmonics. The lower picture depicts a pitch with fundamental frequency 50 Hz and eight harmonics where all odd-numbered harmonics are zero (marked red dots).

of the amplitudes in (2) are non-zero. This approach may be seen as a sparse linear regression problem reminiscent of the one in [24] and has been thoroughly examined in the context of pitch estimation in, e.g., [13, 29, 30]. For notational convenience, define the set of all amplitude parameters to be estimated as

$$\mathbf{\Psi} = \{\mathbf{\Psi}_{\omega_1}, \ldots, \mathbf{\Psi}_{\omega_P}\} \tag{3}$$

$$\mathbf{\Psi}_{\omega_p} = \{a_{p,1}, \ldots, a_{p,L_{\max}}\} \tag{4}$$

where, as described above, most of the $a_{p,\ell}$ in $\mathbf{\Psi}$ will be zero. Note that $\mathbf{\Psi}$ will be sparse, i.e., having few non-zero elements. Also, the pattern of this sparsity will be group wise, meaning that if a pitch with fundamental frequency ω_p is not present, then neither will any of its harmonics, i.e., $\mathbf{\Psi}_{\omega_p} = \mathbf{0}$.

Due to the harmonic structure of the signal, candidate pitches having fundamental frequencies at fractions of the present pitches' fundamentals will have a partial fit of their harmonics. This may cause misclassification, i.e., erroneously identifying a present pitch as one or more non-present candidate pitches. This is the cause of the so-called sub-octave problem, which is mistaking the true pitch with fundamental frequency ω_p for the candidate pitch with fundamental frequency $\omega_p/2$. This may occur if the candidate set $\mathbf{\Psi}$ is structured such that the sub-octave pitch may perfectly model the true pitch, which is when $L_{\max} \geq 2L_p$. This is illustrated in Figure 1, displaying an extreme case with a pitch with fundamental frequency 100 Hz and four harmonics as well as its sub-octave, i.e., a pitch with fundamental frequency 50 Hz and eight harmonics where only the even-numbered harmonics are non-zero. Relating to music signals, this is the same as mistaking a pitch for the pitch an octave below it. Thus, when estimating the elements of $\mathbf{\Psi}$, one also has to take into account the structure of the block sparsity, in order to avoid erroneously selecting sub-octaves.

3 Proposed estimation algorithm

Consider N samples of a noise-corrupted measurement of the signal in (8), $y(t)$, such that it may be well modeled as $y(t) = x(t) + e(t)$, where $e(t)$ is a broadband noise signal. A straightforward approach to estimate $\mathbf{\Psi}$ would then be to minimize the residual cost function

$$g_1(\mathbf{\Psi}) = \frac{1}{2}\sum_{t=1}^{N}\left|y(t) - \sum_{p=1}^{P}\sum_{\ell=1}^{L_{\max}} a_{p,\ell}\mathrm{e}^{i\omega_p \ell t}\right|^2 \tag{5}$$

However, setting

$$\hat{\mathbf{\Psi}} = \arg\min_{\mathbf{\Psi}} g_1(\mathbf{\Psi}) \tag{6}$$

will not yield the desired sparsity structure of $\mathbf{\Psi}$ and will be prone to also model the noise, $e(t)$. Also, solutions (6) will not be unique due to the over-completeness of the approximation (2). A remedy for this would be to add terms penalizing solutions $\hat{\mathbf{\Psi}}$ that are not sparse, for example as

$$\hat{\mathbf{\Psi}} = \arg\min_{\mathbf{\Psi}} g_1(\mathbf{\Psi}) + \lambda ||\mathbf{\Psi}||_0 \tag{7}$$

where $||\mathbf{\Psi}||_0$ is the pseudo-norm counting the number of non-zero elements in $\mathbf{\Psi}$, and λ is a regularization parameter. However, this in general leads to a combinatorial problem whose complexity grows exponentially with the dimension of $\mathbf{\Psi}$. To avoid this, one can approximate the ℓ_0 penalty by the convex function

$$g_2(\mathbf{\Psi}) = \sum_{p=1}^{P} \sum_{\ell=1}^{L_{\max}} |a_{p,\ell}| \tag{8}$$

The resulting problem

$$\underset{\mathbf{\Psi}}{\text{minimize}}\ g_1(\mathbf{\Psi}) + \lambda g_2(\mathbf{\Psi}) \tag{9}$$

is known as the LASSO [31]. In fact, it can be shown that under some restrictions on the set of frequencies ω (see also [32]), the LASSO is guaranteed to retrieve the non-zero indices of $\mathbf{\Psi}$ with high probability, although these conditions are not assumed to be met here. To encourage the group-sparse behavior of $\hat{\mathbf{\Psi}}$, one can further introduce

$$g_3(\mathbf{\Psi}) = \sum_{p=1}^{P} \sqrt{\sum_{\ell=1}^{L_{\max}} |a_{p,\ell}|^2} \tag{10}$$

which is also a convex function. The inner sum corresponds to the ℓ_2-norm, and does not enforce sparsity within each pitch, whereas instead the outer sum, corresponding to the ℓ_1-norm, enforces sparsity between pitches. Thereby, adding the $g_3(\mathbf{\Psi})$ constraint will penalize the number of non-zero pitches. The resulting estimator was in [13] termed the Pitch Estimation using Block Sparsity (PEBS) estimator. However, if we for some p have $2L_p \leq L_{\max}$, the above penalties have no way of discriminating between the correct pitch candidate ω_p and the spurious sub-octave candidate $\omega_p/2$. However, as the candidates will differ in that the sub-octave will only contribute to the harmonic signal at every other frequency in the block, as was seen in Figure 1, one may reduce the risk of such a misclassification by further adding the penalty

$$\breve{g}_4(\mathbf{\Psi}) = \sum_{p=1}^{P} \sum_{\ell=0}^{L_{\max}} \Big| |a_{p,\ell+1}| - |a_{p,\ell}| \Big| \tag{11}$$

where we define

$$a_{p,0} = a_{p,L_{\max}+1} = 0 \quad , \forall p \tag{12}$$

which would add a cost to blocks where there are notable magnitude variations between neighboring harmonics. Unfortunately, (11) is not convex, but a simple convex approximation would be

$$\tilde{g}_4(\mathbf{\Psi}) = \sum_{p=1}^{P} \sum_{\ell=0}^{L_{\max}} |a_{p,\ell+1} - a_{p,\ell}| \tag{13}$$

which would be a good approximation of (11) if all the harmonics had similar phases. This estimator was in [13] termed the PEBS-TV estimator. Clearly, this may not be the case, resulting in that the penalty in (13) would also penalize the correct candidate. An illustration of this is found by considering the worst-case scenario, when all the adjacent harmonics are completely out of phase and have the same magnitudes, i.e., $a_{p,\ell+1} = a_{p,\ell} e^{i\pi}$ with magnitude $|a_{p,\ell}| = r$, for $\ell = 1, \ldots, L_p - 1$. Then, the penalty in (13) will yield a cost of $\tilde{g}_4(\mathbf{\Psi}_{\omega_p}) = 2rL_p$ rather than the desired $\breve{g}_4(\mathbf{\Psi}_{\omega_p}) = 2r$. The cost may also be compared with that of (8), which is $g_2(\mathbf{\Psi}_{\omega_p}) = rL_p$, suggesting that this would add a relatively large penalty. More interestingly, for the sub-octave candidate pitch, the cost will be just as large, i.e., if $\omega_{p'} = \omega_p/2$, then $\tilde{g}_4(\mathbf{\Psi}_{\omega_{p'}}) = 2rL_p$ provided that $L_{\max} \geq 2L_p$, thereby offering no possibility of discriminating between the true pitch and its sub-octave. Such a worst case scenario is just as unlikely as all harmonics having the same phase, if assuming that the phases are uniformly distributed on $[0, 2\pi)$. Instead, the $\tilde{g}_4$ penalty of the true pitch will be slightly smaller than its sub-octave counterpart, on average, and together with (10), the scales tip in favor of the true pitch, as shown in [13]. One may thus conclude that the combination of g_3 and $\tilde{g}_4$ provides a block sparse solution where sub-octaves are usually discouraged. However, it should be noted that such a solution requires the tuning of two functions to control the block sparsity.

This work proposes to simplify the PEBS-TV estimator by improving the approximation in (13), by using an adaptive penalty approach. In order to do so, let $\varphi_{p,\ell}$ denote the phase of the component with frequency $\omega_{p,\ell}$, and collect all the phases in the parameter set

$$\mathbf{\Phi} = \{\mathbf{\Phi}_{\omega_1}, \ldots, \mathbf{\Phi}_{\omega_P}\} \tag{14}$$

$$\mathbf{\Phi}_{\omega_p} = \{\varphi_{p,1}, \ldots, \varphi_{p,L_{\max}}\}\,. \tag{15}$$

The penalty function in (11) may then instead be approximated as

$$g_4(\mathbf{\Psi}, \mathbf{\Phi}) = \sum_{p=1}^{P} \sum_{\ell=0}^{L_{\max}} |a_{p,\ell+1} \mathrm{e}^{-i\varphi_{p,\ell+1}} - a_{p,\ell} \mathrm{e}^{-i\varphi_{p,\ell}}| \tag{16}$$

thus penalizing only differences in magnitude, given that the phases $\varphi_{p,\ell+1}$ have been chosen as to offset phase differences between the harmonics. In order to do so, the phases $\varphi_{p,\ell}$ need to be estimated as the arguments of the latest available amplitude estimates $a_{p,\ell}$. As a result, (16) yields an improved approximation of (11), avoiding the issues of (13) described above, and also promotes a block sparse solution. The block sparsity is promoted due to the introduction of zero amplitudes in (12). In effect, this introduces a penalty for activating a pitch block. As a result, the block-norm penalty function g_3 may be omitted, which simplifies the algorithm noticeably. Thus, we form the parameter estimates by solving

$$\hat{\mathbf{\Psi}} = \arg\min_{\mathbf{\Psi}} \; g_1(\mathbf{\Psi}) + \lambda_2 g_2(\mathbf{\Psi}) + \lambda_4 g_4(\mathbf{\Psi}, \mathbf{\Phi}) \tag{17}$$

where λ_2 and λ_4 are user-defined regularization parameters that weigh the importance of each penalty function with that of the residual cost. To form the convex criteria and to facilitate the implementation, consider the signal expressed in matrix notation as

$$\mathbf{y} = \begin{bmatrix} y(1) & \ldots & y(N) \end{bmatrix}^T = \sum_{p=1}^{P} \mathbf{W}_p \, \mathbf{a}_p + \mathbf{e} \triangleq \mathbf{W}\mathbf{a} + \mathbf{e} \tag{18}$$

where

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_1 & \ldots & \mathbf{W}_P \end{bmatrix} \tag{19}$$

$$\mathbf{W}_p = \begin{bmatrix} \mathbf{z}_p^1 & \ldots & \mathbf{z}_p^{L_{\max}} \end{bmatrix} \tag{20}$$

$$\mathbf{z}_p = \begin{bmatrix} \mathrm{e}^{i\omega_p 1} & \dots & \mathrm{e}^{i\omega_p N} \end{bmatrix}^T \tag{21}$$

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_1^T & \dots & \mathbf{a}_P^T \end{bmatrix}^T \tag{22}$$

$$\mathbf{a}_p = \begin{bmatrix} a_{p,1} & \dots & a_{p,L_{\max}} \end{bmatrix}^T \tag{23}$$

where the powers in the vectors $\mathbf{z}_p^k$ are taken element-wise. The dictionary matrix $\mathbf{W}$ is constructed by P horizontally stacked blocks, or dictionary atoms $\mathbf{W}_p$, where each is a matrix with $L_{\max}$ columns and N rows. In order to obtain an acceptable approximation of (11), the problem must be solved iteratively, where the last solution is used to improve the next. To pursue an even sparser solution, a re-weighting procedure is simultaneously used for $g_2(\mathbf{\Psi})$, similar to the one used in [33]. Redefining the functions g_j to operate on matrices, the solution is thus found at the kth iteration as

$$\hat{\mathbf{a}}^{(k)} = \arg\min_{\mathbf{a}} \frac{1}{2} \left\| \mathbf{y} - \mathbf{H}_1^{(k)} \mathbf{a} \right\|_2^2 + \lambda_2 \left\| \mathbf{H}_2^{(k)} \mathbf{a} \right\|_1 + \lambda_4 \left\| \mathbf{H}_4^{(k)} \mathbf{a} \right\|_1 \tag{24}$$

where

$$\mathbf{H}_1^{(k)} = \mathbf{W} \tag{25}$$

$$\mathbf{H}_2^{(k)} = \operatorname{diag}\left(1 / \left(\left| \hat{\mathbf{a}}^{(k-1)} \right| + \varepsilon \right) \right) \tag{26}$$

$$\mathbf{H}_4^{(k)} = \mathbf{F} \operatorname{diag}\left(\arg\left(\hat{\mathbf{a}}^{(k-1)} \right) \right)^{-1} \tag{27}$$

where $\operatorname{diag}(\cdot)$ denotes a diagonal matrix formed with the given vector along its diagonal, $|\cdot|$ is element-wise absolute value, $\arg(\cdot)$ is the element-wise complex argument, and $\varepsilon \ll 1$. If the magnitude of a certain component of $\hat{\mathbf{a}}^{(k-1)}$ is small, the construction of $\mathbf{H}_2^{(k)}$ will ensure that the magnitude of the corresponding component of $\hat{\mathbf{a}}^{(k)}$ will be penalized harder. This iterative re-weighting procedure will then be a sequence of convex approximations of a non-convex logarithmic penalty on the ℓ_1 norm of $\mathbf{a}$. The inclusion of ε is made to ensure that a division by zero is avoided. Also, $\mathbf{I}$ denotes the identity matrix, and $\mathbf{F}$ is a $P(L_{\max}+1) \times PL_{\max}$ matrix $\mathbf{F} = \operatorname{diag}(\mathbf{F}_1, \dots, \mathbf{F}_P)$, where each block $\mathbf{F}_p$ is a $(L_{\max}+1) \times L_{\max}$ matrix with elements

$$f_{k,\ell} = \begin{cases} 1 & \text{if } k = \ell = 1 \\ -1 & \text{if } k = \ell, \ell \neq 1 \\ 1 & \text{if } k = \ell + 1 \\ 0 & \text{otherwise} \end{cases} \tag{28}$$

As intended, the minimization in (24) is convex, and may be solved using one of many publicly available convex solvers, such as, for instance, the interior point methods SeDuMi [34] or SDPT3 [27]. However, these methods are quite computationally burdensome and will scale poorly with increased data length and larger grids. Instead, we here propose an efficient implementation using ADMM. The problem in (24) may be implemented in a similar manner as was done in [25], requiring only two tuning parameters, λ_2 and λ_4. The proposed method compares to the PEBS and PEBS-TV algorithms as improving upon the former, and requiring fewer tuning parameters than the latter. The proposed method is therefore termed a light and improved version of PEBS, here denoted the PEBSI-Lite algorithm.

4 ADMM implementation

In order to solve (24), we proceed to introduce an efficient ADMM implementation. To this end, let $\mathbf{z} \in \mathbb{C}^{PL_{\max}}$ be the primal optimization variable and introduce the auxiliary variables $\mathbf{u}_1 \in \mathbb{C}^N$, $\mathbf{u}_2 \in \mathbb{C}^{PL_{\max}}$, and $\mathbf{u}_4 \in \mathbb{C}^{P(L_{\max}+1)}$ and let

$$\mathbf{G}^{(k)} = \begin{bmatrix} \mathbf{H}_1^{(k)T} & \mathbf{H}_2^{(k)T} & \mathbf{H}_4^{(k)T} \end{bmatrix}^T \tag{29}$$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1^T & \mathbf{u}_2^T & \mathbf{u}_4^T \end{bmatrix}^T . \tag{30}$$

Thus, we want to solve

$$\underset{\mathbf{z}}{\text{minimize}}\, f\left(\mathbf{G}^{(k)}\mathbf{z}\right) \tag{31}$$

where

$$f\left(\mathbf{G}^{(k)}\mathbf{z}\right) = \frac{1}{2}\left\|\mathbf{y} - \mathbf{H}_1^{(k)}\mathbf{z}\right\|_2^2 + \lambda_2\left\|\mathbf{H}_2^{(k)}\mathbf{z}\right\|_1 + \lambda_4\left\|\mathbf{H}_4^{(k)}\mathbf{z}\right\|_1 . \tag{32}$$

Using the auxiliary variabel $\mathbf{u}$, one may equivalently solve

$$\begin{aligned} &\underset{\mathbf{z},\mathbf{u}}{\text{minimize}}\, f(\mathbf{u}) + \frac{\mu}{2}\left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u}\right\|_2^2 \\ &\text{subject to } \mathbf{G}^{(k)}\mathbf{z} - \mathbf{u} = \mathbf{0} \end{aligned} \tag{33}$$

where μ is a positive scalar, as the added term is zero for any feasible point. The Lagrangian can be succinctly expressed using the (scaled) dual variable

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_1^T & \mathbf{d}_2^T & \mathbf{d}_4^T \end{bmatrix}^T \tag{34}$$

where $\mathbf{d}_1 \in \mathbb{C}^N$, $\mathbf{d}_2 \in \mathbb{C}^{PL_{\max}}$, and $\mathbf{d}_4 \in \mathbb{C}^{P(L_{\max}+1)}$. By completing the square, the Lagrangian of the problem can be equivalently expressed as

$$L_\mu(\mathbf{z},\mathbf{u},\mathbf{d}) = f(\mathbf{u}) + \frac{\mu}{2}\left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u} - \mathbf{d}\right\|_2^2 - \frac{\mu}{2}\|\mathbf{d}\|_2^2 \ . \tag{35}$$

Also, define

$$\boldsymbol{\zeta}(j) = \left[\ \boldsymbol{\zeta}_1^T(j) \quad \boldsymbol{\zeta}_2^T(j) \quad \boldsymbol{\zeta}_4^T(j)\ \right]^T \tag{36}$$

where

$$\boldsymbol{\zeta}_\ell(j) = \mathbf{H}_\ell^{(k)}\mathbf{z}(j+1) - \mathbf{d}_\ell(j)\ ,\ \ell = 1,2,4\ . \tag{37}$$

The Lagrangian (35) is separable in the variables $\mathbf{z}$, $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_4$, and one may thus form an updating scheme similar to that in [25], as

$$\mathbf{z}(j+1) = \arg\min_{\mathbf{z}} \left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u}(j) - \mathbf{d}(j)\right\|_2^2 \tag{38}$$

$$\mathbf{u}_1(j+1) = \arg\min_{\mathbf{u}_1} \frac{1}{2}\|\mathbf{y} - \mathbf{u}_1\|_2^2 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_1(j) - \mathbf{u}_1\right\|_2^2 \tag{39}$$

$$\mathbf{u}_2(j+1) = \arg\min_{\mathbf{u}_2} \lambda_2\|\mathbf{u}_2\|_1 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_2(j) - \mathbf{u}_2\right\|_2^2 \tag{40}$$

$$\mathbf{u}_4(j+1) = \arg\min_{\mathbf{u}_4} \lambda_4\|\mathbf{u}_4\|_1 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_4(j) - \mathbf{u}_4\right\|_2^2 \tag{41}$$

$$\mathbf{d}(j+1) = \mathbf{u}(j+1) - \boldsymbol{\zeta}(j)\ . \tag{42}$$

The updates of $\mathbf{z}$ and $\mathbf{u}_1$ are given by

$$\mathbf{z}(j+1) = \left(\mathbf{G}^{(k)H}\mathbf{G}^{(k)}\right)^{-1}\mathbf{G}^{(k)H}\left(\mathbf{u}(j) + \mathbf{d}(j)\right) \tag{43}$$

and

$$\mathbf{u}_1(j+1) = \frac{\mathbf{y} + \mu\boldsymbol{\zeta}_1(j)}{1+\mu} \tag{44}$$

respectively.

Algorithm 1 The proposed PEBSI-Lite algorithm

initiate $k := 0$, $\mathbf{H}_1^{(0)} = \mathbf{I}$, $\mathbf{H}_4^{(0)} = \mathbf{F}$, and
  $\hat{\mathbf{a}}^{(0)} = \mathbf{z}_{\text{save}} = \mathbf{d}_{\text{save}} = \mathbf{0}^{PL_{\max} \times 1}$
repeat {adaptive penalty scheme}
  initiate $j := 0$, $\mathbf{u}_2(0) = \hat{\mathbf{a}}^{(k)}$,
    $\mathbf{z}(0) = \mathbf{z}_{\text{save}}$, and $\mathbf{d}(0) = \mathbf{d}_{\text{save}}$
  repeat {ADMM scheme}
    $\mathbf{z}(j) = \left(\mathbf{G}^{(k)H}\mathbf{G}^{(k)}\right)^{-1}\mathbf{G}^{(k)H}\left(\mathbf{u}(j) + \mathbf{d}(j)\right)$
    $\mathbf{u}_1(j+1) = \frac{\mathbf{y}+\mu\boldsymbol{\zeta}_1(j)}{1+\mu}$
    $\mathbf{u}_2(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_2(j), \frac{\lambda_2}{\mu}\right)$
    $\mathbf{u}_4(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_4(j), \frac{\lambda_4}{\mu}\right)$
    $\mathbf{d}(j+1) = \mathbf{u}(j+1) - \boldsymbol{\zeta}(j)$
    $j \leftarrow j+1$
  until convergence
  store $\hat{\mathbf{a}}^{(k)} = \mathbf{u}_2(\text{end})$, $\mathbf{z}_{\text{save}} = \mathbf{z}(\text{end})$, and $\mathbf{d}_{\text{save}} = \mathbf{d}(\text{end})$
  update $\mathbf{H}_2^{(k+1)} = \text{diag}\left(1/|\hat{\mathbf{a}}^{(k)}| + \varepsilon\right)$, $\mathbf{H}_4^{(k+1)} = \mathbf{F}\,\text{diag}\left(\arg\left(\hat{\mathbf{a}}^{(k)}\right)\right)^{-1}$
  $k \leftarrow k+1$
until convergence

Using the element-wise shrinkage function,

$$\mathbf{T}\left(\mathbf{x}, \xi\right) = \frac{\max(|\mathbf{x}| - \xi, 0)}{\max(|\mathbf{x}| - \xi, 0) + \xi} \odot \mathbf{x} \tag{45}$$

where the max function operates on each element in the vector $\mathbf{x}$ separately and $\odot$ denotes element-wise multiplication, one may update $\mathbf{u}_2$ and $\mathbf{u}_4$ as

$$\mathbf{u}_2(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_2(j), \frac{\lambda_2}{\mu}\right) \tag{46}$$

and

$$\mathbf{u}_4(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_4(j), \frac{\lambda_4}{\mu}\right) \tag{47}$$

respectively. The resulting PEBSI-Lite algorithm is summarized in Algorithm 1, where the solution is given as $\hat{\mathbf{a}} = \hat{\mathbf{a}}^{(k_{\text{end}})}$ with k_{end} denoting the last iteration index

of the outer loop. The complexity of the resulting algorithm will be dominated by the computation of step 5 in Algorithm 1. This system of equations can be solved efficiently by storing the Cholesky factorization of the matrix to be inverted, with a one-time cost of $\mathcal{O}\left(p^3\right)$ operations, where p denotes the number of variables (here, assumed to be larger than the number of data points). Furthermore, at each iteration, one needs to perform a back solve costing $\mathcal{O}\left(p^2\right)$ operations.

5 Self-regularization

The quality of the pitch estimates produced by the PEBSI-Lite algorithm depends on the values of the regularization parameters λ_2 and λ_4. In general, large values of λ_2 encourage sparse solutions, whereas large values of λ_4 encourage solutions that are smooth within blocks. As the model order is unknown, it is generally hard to determine how sparse the solution should be in order to be considered the desired one. Therefore, one often determines the values of the regularization parameters using cross-validation schemes, making the performance of the methods user dependent. Instead, one would like to have a systematic and preferable automatic method for choosing λ_2 and λ_4, and thereby the model order.

A common approach to solving model order problems is to use information criteria such as AIC or BIC [35], which measure the fit of the model to the data, while penalizing high model orders, resulting in a trade-off criterion that should take its optimal value for the correct model order. For the LASSO problem, there have been suggestions of appropriate model order criteria [36], [37]. In [13], the authors suggest a BIC-style criterion for multi-pitch estimation for given regularization parameters. However, this criterion can only be used to determine which of the found pitches are true and which are spurious, and not to determine the appropriate regularization parameters. Thus, even if one has an efficient criterion for choosing between different models, one first has to form a set of candidate models, in effect running Algorithm 1 for different values of λ_2 and λ_4. For the simpler case of the LASSO, the analog is to solve (9) for all $\lambda \in \mathbb{R}_+$, for which there are algorithms such as LARS [38]. There have also been methods suggested for solving the LASSO for only a finite number of values λ, i.e., only values of the regularization parameter where the number of active components of the solution change (see, e.g., [37]). For our problem, the analog is to find solutions for the set of parameter values

$$\{(\lambda_2, \lambda_4) | (\lambda_2, \lambda_4) \in \mathbb{R}_+ \times \mathbb{R}_+\} . \tag{48}$$

For the real-variable counterpart of the here considered pitch estimation problem, known as the Sparse Fused LASSO [39], there have been algorithms suggested for computing the whole solution surface. In [40], the authors present an elegant way of finding a solution path for the case of the dictionary $\mathbf{W}$ being the identity matrix, meaning that the estimated amplitude vector is just a smoothed version of the signal $\mathbf{y}$. The algorithm can be used for general matrices $\mathbf{W}$, under the condition that $\mathbf{W}$ has full column rank, something that is not true for dictionaries in high-resolution spectral estimation applications such as the one considered here. In [41], the authors present an approach to find the solution path of

$$\underset{\boldsymbol{\beta}}{\text{minimize}}\ \frac{1}{2}\left\|\mathbf{y}-\mathbf{W}\boldsymbol{\beta}\right\|_2^2+\lambda\left\|\mathbf{D}\boldsymbol{\beta}\right\|_1 \tag{49}$$

for the real-variable case with a general penalty matrix $\mathbf{D}$ by considering the solution paths of the dual variable. Unfortunately, this is only for the one-dimensional case, i.e., for the case when the minimization has only a single regularization parameter.

Despite the above efficient ADMM implementation, it is computationally cumbersome to conduct a search on (48) in order to find an appropriate model order, with the computation complexity increasing both in the case of longer signals, and when using more elements in the dictionary. Instead of constructing a fully general path algorithm for PEBSI-Lite, we therefore proceed to propose a scheme for constructing a reduced size signal adapted dictionary that combined with a parametrization of the regularization parameters (λ_2, λ_4) will allow us to form good pitch estimates without having to predefine values of the regularization parameters, by means of a simple line search instead of searching through (48). The proposed dictionary construction begins by estimating the frequency content of the signal without imposing any harmonic structure. This estimation may be performed by any standard method, such as ESPRIT (see, e.g., [42]). As the number of sinusoidal components is unknown, estimates corresponding to different model orders can be evaluated using, for instance, the BIC criterion (see, e.g., [35])

$$\mathrm{BIC}_k = 2N\log\hat{\sigma}_k^2+(5k+1)\log N \tag{50}$$

where $\hat{\sigma}_k^2$ is the maximum likelihood estimate of the residual variance corresponding to the model constituted by k estimated sinusoids, in order to choose a suitable model order. The accuracy of the frequency estimates produced by ESPRIT will

suffer if a too low model order is determined, whereas it is less sensitive to cases when the model order is moderately overestimated. Thus, we propose to increase the robustness of the frequency estimates by using $k + \delta$, $\delta \geq 1$, estimated sinusoids for the case when order k is determined optimal by the BIC. As the only interesting pitch candidates are those having at least one harmonic corresponding to a present sinusoidal component, we can then design a considerably reduced dictionary, containing only pitches with such matching harmonics. If one has some prior knowledge of the nature of the signal, one could impose stronger assumptions on the candidate pitches in order to reduce the dictionary further, e.g., by allowing only pitches whose first harmonic is found in the set of estimated sinusoids. Using the obtained dictionary, one could then proceed to conduct a search for λ_2 and λ_4.

Although considerably cheaper as compared to when performed using a full dictionary, a complete evaluation of the $\lambda_2\lambda_4$-plane is still somewhat expensive. To avoid a full grid search, the following heuristic concerning the connection between λ_2 and λ_4 can be used. Assume that we have a single-pitch signal where all L_k harmonics have equal magnitude r. Further, assume that when setting $\lambda_4 = 0$, λ' is the largest value of λ_2 resulting in a nonzero solution, where each harmonic amplitude is estimated to r_0. If we would instead set $\lambda_2 = 0$, and consider which value of λ_4 that should result in the same solution, this value should be

$$\lambda_4 = \frac{L_k}{2}\lambda' \tag{51}$$

as this would result in precisely the same penalty as with $\lambda_4 = 0$, $\lambda_2 = \lambda'$. More compactly, we have that

$$\lambda_2 = \alpha\lambda' \ , \ \lambda_4 = \big(1 - \alpha\big)\frac{L_k}{2}\lambda' \tag{52}$$

yields the penalty $\lambda' \, L_k r_0$ for all $\alpha \in [0, 1]$. If we assume (52) to be true, we should, for spectrally smooth signals, expect to see ridges in the solution surface where the number of pitches present in the solution changes, and the shapes of the ridges in the $\lambda_2\lambda_4$-plane should be described by lines similar to (52).

This is illustrated in Figure 2, presenting a plot of the number of pitches present in the solution for different values (λ_2, λ_4) for a signal consisting of three pitches with fundamental frequencies 400, 550 and 700 Hz, and with 4, 8, and 12 harmonics, respectively. The magnitude of each harmonic amplitude has

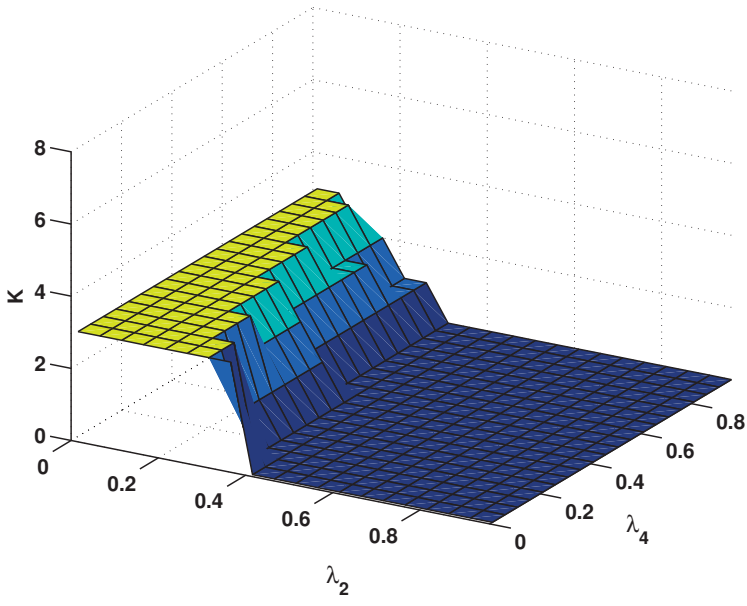


Figure 2: Number of pitches, K, present in the solution of PEBSI-Lite for different values (λ_2, λ_4) when applied to a three pitch signal with 4, 8, and 12 harmonics, respectively.

been drawn uniformly on $(0.9, 1.1)$ and each phase has been drawn uniformly on $(0, 2\pi)$. The signal was sampled at frequency 20 kHz in a time frame of length 40 ms, generating 800 samples of the signal. The signal-to-noise ratio (SNR), as defined in (55), was 20 dB. On the plateau with two pitches, the pitch with four harmonics have been forced to zero, whereas on the plateau with one pitch present, only the pitch with 12 harmonics is present. Note the shape of the different plateaus: seen in the $\lambda_2\lambda_4$-plane, the slopes of the ridges seem to be well described by (52) where $L_k = 4, 8$, and 12, for the three ridges corresponding to changes from three to two, from two to one, and from one to zero pitches, respectively. The signal corresponding to Figure 2 has a relatively low level of noise. Increasing the noise level, the least regularized solutions, i.e., with λ_2 and λ_4 close to zero, results in more than three non-zero pitches. Guided by this observation, one could reduce the search for (λ_2, λ_4) from a 2-D to a 1-D search by using a

Algorithm 2 Self-Regularized PEBSI-Lite

initiate $\ell = 1$
repeat {sinusoidal component estimation}
  $\hat{\boldsymbol{\omega}}_\ell \leftarrow \ell$ sinusoidal components from ESPRIT
  $\mathrm{BIC}_\ell \leftarrow 2N \log \hat{\sigma}^2(\hat{\boldsymbol{\omega}}_\ell) + (5\ell + 1) \log N$
until $\mathrm{BIC}_\ell > \mathrm{BIC}_{\ell-1}$
$\hat{\boldsymbol{\omega}}_{\ell+\delta} \leftarrow \ell + \delta$ sinusoidal components from ESPRIT, where $\delta \geq 1$ is a safety margin
construct dictionary $\mathbf{W}$ from $\hat{\boldsymbol{\omega}}_{\ell+\delta}$
$L \leftarrow$ largest number of active harmonics among candidate pitches in $\mathbf{W}$
initiate $\lambda = \varepsilon$, $k = 1$
$\hat{\sigma}_y^2 \leftarrow \mathrm{Var}\big(y\big)$
$\hat{\sigma}^2_{\mathrm{MLE}} \leftarrow$ maximum likelihood (least squares) estimate of noise power
repeat {regularization parameter line search}
  $\lambda_2 \leftarrow \lambda$, $\lambda_4 \leftarrow \frac{L}{2}\lambda$
  form amplitude estimate $\hat{\mathbf{a}}^{(k)}$ from Algorithm 1
  estimate the power of the model residual $\hat{\sigma}^2(\lambda_2, \lambda_4)$
  $\lambda \leftarrow \lambda + \varepsilon$
  $k \leftarrow k + 1$
until $\big(\hat{\sigma}^2(\lambda_2, \lambda_4) - \hat{\sigma}^2_{\mathrm{MLE}}\big) > \tau\hat{\sigma}_y^2$
$\hat{\mathbf{a}} \leftarrow \hat{\mathbf{a}}^{(k-1)}$

re-parametrization. Keeping the plateaus in Figure 2 and our assumption of spectral smoothness in mind, we should expect a desirable solution to correspond to a (λ_2, λ_4)-pair with $\lambda_2 \leq \lambda_4$. In order to get solutions regularized with respect to spectral smoothness, while keeping the risk of getting only zero solutions low, the following parametrization can be used. Let λ denote the only free parameter and set

$$\lambda_2 = \lambda \tag{53}$$

$$\lambda_4 = \frac{L}{2}\lambda \tag{54}$$

where L is the largest number of harmonics among the pitches present in the signal. Although L is unknown, it can be estimated during the dictionary construction phase using the BIC and ESPRIT estimates, permitting us to conduct a

Estimator	SNR (dB)	-5	0	5	10	15	20
PEBS-TV	λ_2	0.2	0.2	0.2	0.15	0.1	0.1
	λ_3	0.3	0.3	0.3	0.2	0.2	0.15
	λ_4	0.1	0.1	0.1	0.75	0.75	0.05
PEBS	λ_2	0.2	0.2	0.2	0.15	0.15	0.1
	λ_3	0.4	0.4	0.4	0.3	0.3	0.2

Table 1: Regularization parameter values for PEBS-TV and PEBS.

line search for the value of λ. Having obtained a solution with PEBSI-Lite using the regularization parameter λ, the residual power σ_λ^2 can be estimated by least squares. It is worth noting that in low noise environments, it can be expected that false pitches modeling noise will not contribute much to the signal power. Thus, the first significant rise in residual power is expected to occur when one of the true pitches are set to zero. Therefore, we propose keeping only models that correspond to lower values of σ_λ^2 and then choosing the optimal model as the one having the least number of active pitches. The complete algorithm for the dictionary construction, line search, and pitch estimation is outlined in Algorithm 2, where ε denotes the step size of the line search and $\tau \in (0, 1)$ is a threshold for detecting an increase in model residual power. The step size ε can be chosen based on afforded estimation time, as small values of ε will result in more steps for the line search. τ can be chosen based on estimates of the noise power, if available.

6 Numerical results

We proceed to examine the performance of the proposed algorithm using signals simulated from the pitch model (8) as well as synthetic audio signals generated from MIDI, and measured audio signals.

6.1 Two-pitch signal

We initially examine a simulated dual-pitch signal, measured in white Gaussian noise at different SNRs ranging from -5 dB to 20 dB in steps of 5 dB. The SNR is here defined as

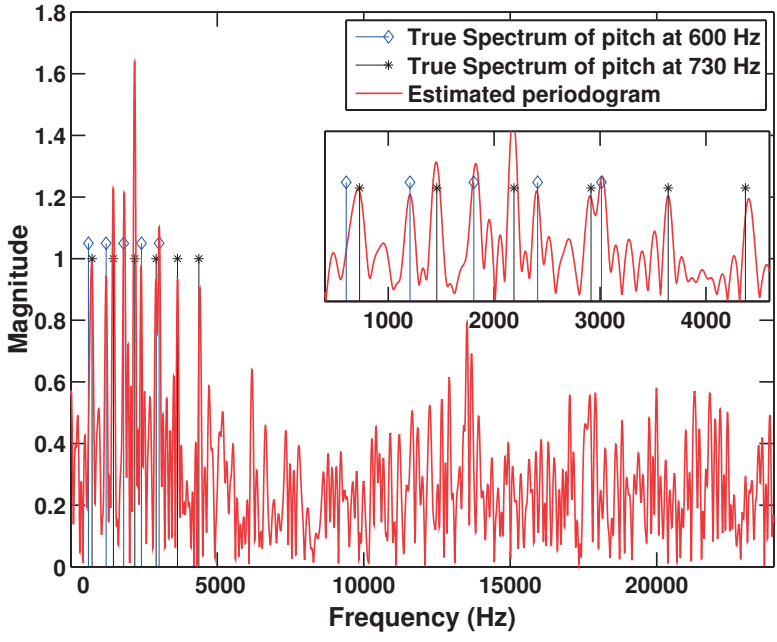


Figure 3: The periodogram estimate and the true signal studied in Figure 4.

$$\mathrm{SNR} = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2} \tag{55}$$

where σ_x^2 and σ_e^2 are the powers of the signal and the noise, respectively. For a pitch signal generated by (8), under the simplifying assumption of distinct sinusoidal components, the power of the signal is given by

$$\sigma_x^2 = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} \frac{|a_{k,\ell}|^2}{2} \,. \tag{56}$$

At each SNR, 200 Monte Carlo simulations were performed, each simulation generating a signal with fundamental frequencies of 600 and 730 Hz. As PEBS and PEBS-TV rely on a predefined frequency grid, the fundamental frequencies were randomly chosen at each simulation uniformly on $600 \pm d/2$ and $730 \pm d/2$, where d is the grid point spacing, to reflect performance in present of off-grid

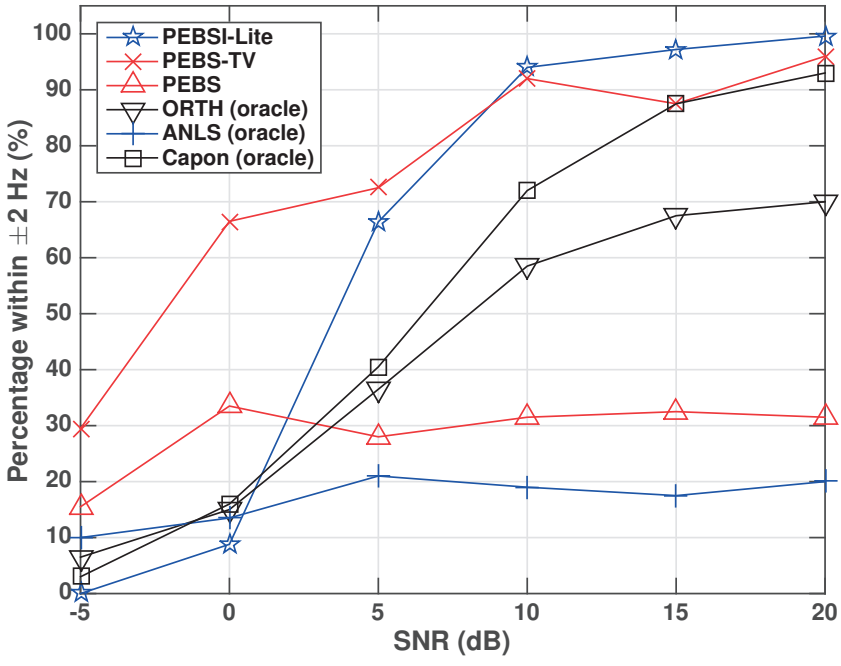


Figure 4: Percentage of estimated pitches where both fundamental frequencies lie at most 2 Hz, or $d/5 = 1/50N$, from the ground truth, plotted as a function of SNR. Here, the pitches have $[5, 6]$ harmonics, respectively, and $L_{\max} = 10$.

effects. The phases of the harmonics in each pitch were chosen uniformly on $[0, 2\pi)$, whereas all had unit magnitude. The signal was sampled at $f_s = 48$ kHz on a time frame of 10 ms, yielding $N = 480$ samples per frame. As a result, the pitches were spaced by approximately f_s/N Hz, which is the resolution limit of the periodogram. This is also seen in Figure 3, illustrating the resolution of the periodogram as well as the frequencies of the harmonics, at SNR $= -5$ dB. From the figure, it may be concluded that the signal contains more than one harmonic source, as the observed peaks are not harmonically related. Furthermore, it is clear that the fundamental frequencies are not separated by the periodogram, indicating that any pitch estimation algorithm based on the periodogram would suffer notable difficulties. For PEBSI-Lite, the estimates are formed using Algorithm 2 with $\tau = 0.1$ and $\varepsilon = 0.05$. The safety margin for the sinusoidal model order is $\delta = 1$. For PEBS and PEBS-TV, the estimation procedure is initiated using

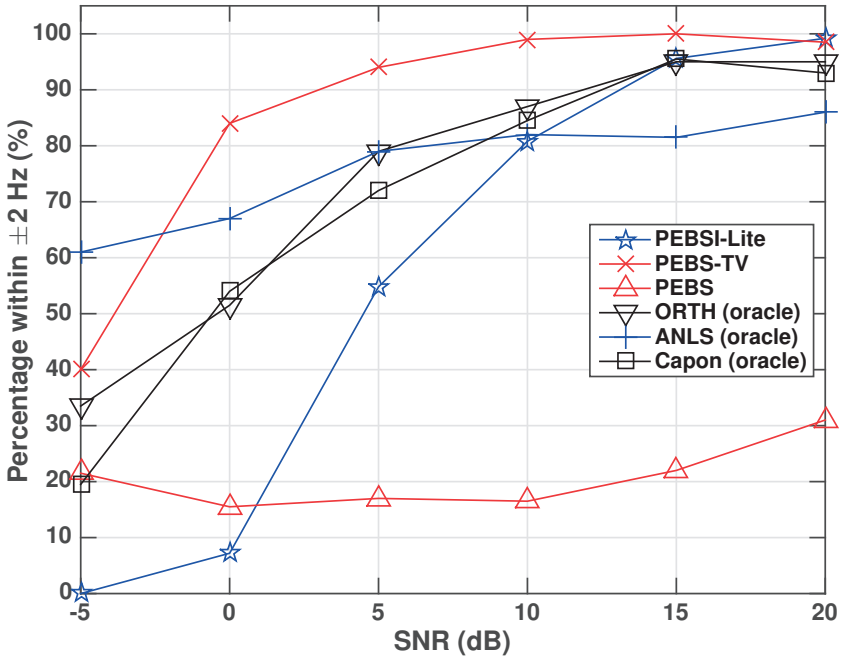


Figure 5: Percentage of estimated pitches where both fundamental frequencies lie at most 2 Hz, or $d/5 = 1/50N$, from the ground truth, plotted as a function of SNR. Here, the pitches have $[10, 11]$ harmonics, respectively, and $L_{\max} = 20$.

a coarse dictionary, with candidate pitches uniformly distributed on the interval $[280, 1500]$ Hz, thus also including $\omega_p/2$ and $2\omega_p$ for both pitches. The coarse resolution was $d = 10$ Hz, i.e., still a super-resolution of $f_s/10N$. After estimation on this grid, a zooming step was taken where a new grid with spacing $d/10$ was laid $\pm 2d$ around each pitch having non-zero power. The regularization parameter values used for PEBS-TV and PEBS are presented in Table 1. The values where selected using *manual cross-validation* for similar signals. Comparisons were also made with the ANLS, ORTH, and the harmonic Capon estimators, which had been given the *oracle* model orders (see [9] for more details on these methods). The simulation and estimation procedure was performed for two cases; one where the number of harmonics L_k were set to 5 and 6, and one where L_k were set to 10 and 11. In the former case, $L_{\max} = 10$ and in the latter, $L_{\max} = 20$, i.e., well above the true number of harmonics.

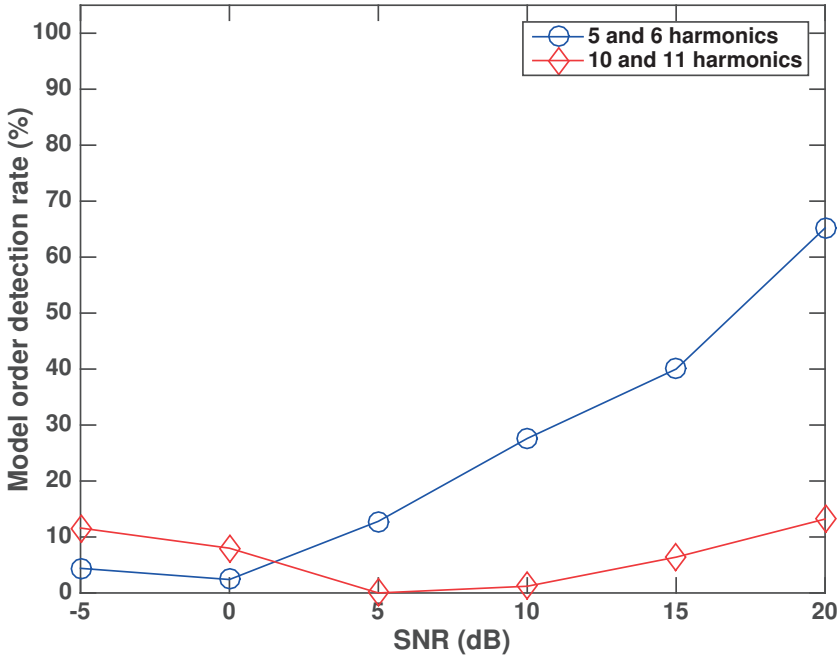


Figure 6: The percentage of the estimates in which the model order choice criterion (50) correctly determines the number of sinusoidal components in the two-pitch signal, for the case of 5 and 6 harmonics, and 10 and 11 harmonics, respectively.

Figures 4 and 5 show the percentage of pitch estimates where both lie within ± 2 Hz from the true values for the six compared methods, for the case of 5 and 6 as well as 10 and 11 harmonics, respectively. In this setting, PEBS performs poorly, as the generous choices of $L_{\max}$ allow it to pick the sub-octave, as predicted. As can be seen in Figure 4, PEBSI-Lite performs better than all reference methods for SNRs above and including 10 dB despite not having the model order information given to ORTH, ANLS, and Capon, nor having the supervised regularization parameter choices of PEBS and PEBS-TV. Though, in higher noise settings, the performance of PEBSI-Lite degrades and its pitch frequency estimates are worse than those produced by the reference methods for SNRs below 10 dB.

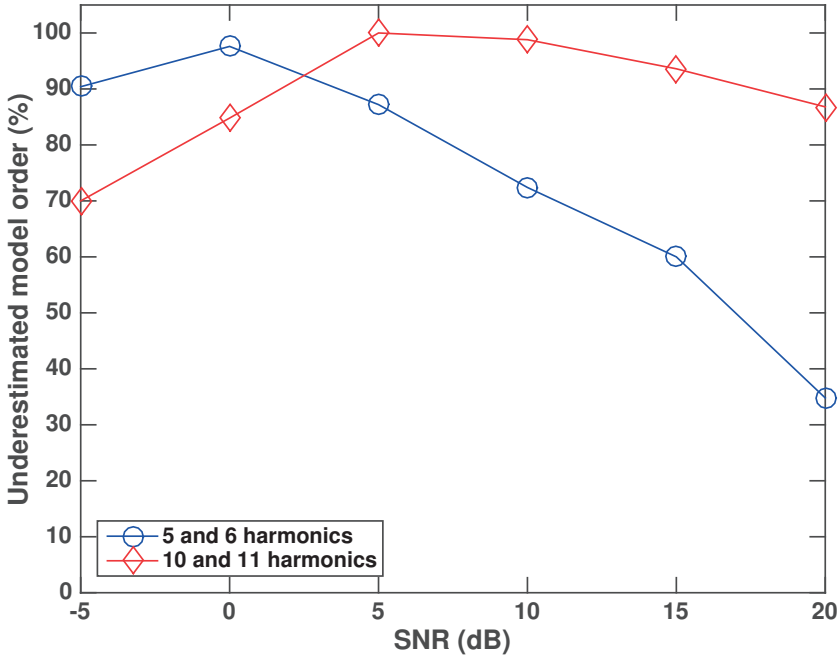


Figure 7: The percentage of the estimates in which the model order choice criterion (50) selects a model with too few sinusoidal components for the two-pitch signal, for the case of 5 and 6 harmonics, and 10 and 11 harmonics, respectively.

For the case of 10 and 11 harmonics, PEBSI-Lite performs on par with the reference methods for SNRs above and including 15 dB, while performing worse in higher noise settings. As shown in Figures 6 and 7, the drop in performance for lower SNRs results from the difficulty of accurately estimating the total number of sinusoids, as used by the ESPRIT step, for such signals. In Figure 6, the percentage of the estimates in which the the BIC criterion (50) correctly determines the number of sinusoidal components in the signal is presented, whereas Figure 7 shows the percentage of the estimates in which the BIC criterion (50) determines a too low model order. As is clear from the figures, the model order estimates strongly degrade for lower SNRs, thus causing the PEBSI-Lite dictionary to be

inaccurate. Clearly, all the other methods here shown using oracle model order information would suffer drastically from such inaccuracies, although it should be stressed that one may expect these methods to suffer further, as they also need to perform an exhaustive combinatorial search to determine the number of pitches given the found number of sinusoids.

6.2 Three-pitch signal

To further examine the performance of Algorithm 2, it was evaluated using a simulated triple-pitch signal, measured in white Gaussian noise at different SNR levels, ranging from 0 dB to 25 dB, in steps of 5 dB. Instead of using unit magnitudes of the harmonics, as was the case for the above presented two-pitch setting, the spectral envelopes of the three pitch components were constructed from periodograms of three different speech recordings. The formants of the three pitches are displayed in Figure 8. The pitches had fundamental frequencies 200, 350, and 530 Hz, and 7, 8, and 11 harmonics, respectively. At each level of SNR, 1000 Monte Carlo simulations were performed, where the fundamental frequencies were chosen uniformly on 200 ± 2.5, 350 ± 2.5, and 530 ± 2.5 Hz, respectively, and the phase of each harmonic was chosen uniformly on $[0, 2\pi)$. The signal was sampled in a 40 ms window at a sampling frequency of 20 kHz, generating 800 samples of the signal. The algorithm settings were $\tau = 0.1$, $\varepsilon = 0.05$, and $\delta = 1$. Here, Algorithm 2 was compared to the ANLS, ORTH, harmonic Capon, as well as PEBS-TV estimators. The three first comparison methods were given the oracle model orders.

To illustrate the fact that the choice of regularization parameter values is not universal, the values found using cross-validation for the two-pitch case (see Table 1) were used for PEBS-TV initially. However, this resulted in such poor performance that the parameter values had to be slightly altered in order to make PEBS-TV an interesting reference method. As a compromise, the parameter values corresponding to SNR 20 dB in Table 1 were used for all SNRs in this simulation setting. For the dictionaries of PEBSI-Lite and PEBS-TV, $L_{\max} = 16$ was used, well above the true model orders. Figure 9 shows the percentage of the pitch estimates where all three pitch estimates lie within ± 2 Hz of the true values for the five different methods. As can be seen, the performance of PEBSI-Lite is again poor for low SNRs while improving considerably for lower noise levels. The low scoring for PEBSI-Lite for low SNRs is mainly due to the selection of wrong model orders. This is illustrated in Figure 10, which shows the percentage

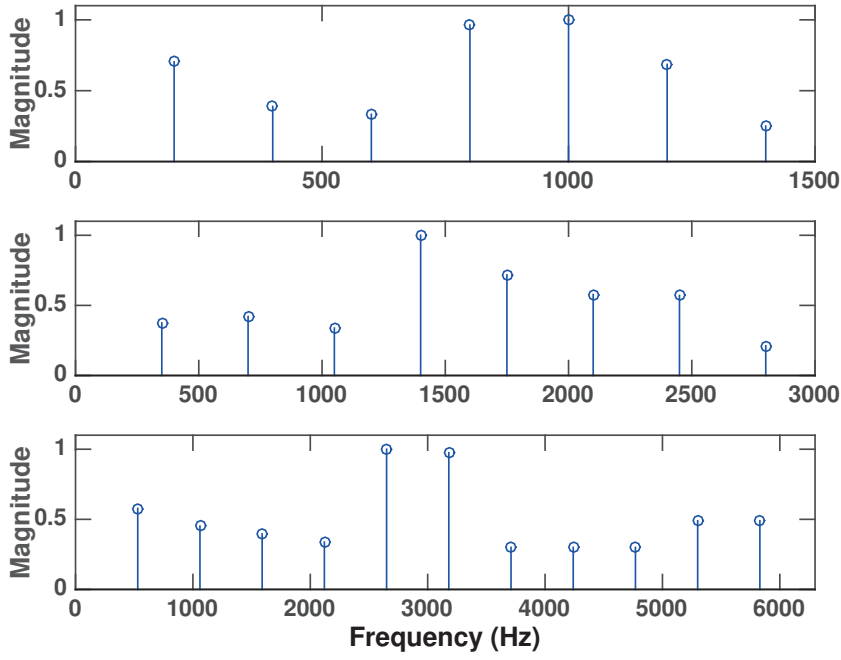


Figure 8: Magnitudes for the harmonics of the three pitches constituting the test signal for the Monte Carlo simulations.

of the estimates in which PEBSI-Lite and PEBS-TV select the correct number of pitches. As can be seen, for an SNR of 0 dB, PEBSI-Lite selects the true model order in less than 10% of the simulations. Mostly, a too high model order is selected, which is to be expected as the model order choice is based on the power of the model residual and that the pitch estimates depend on the accuracy of the initial ESPRIT estimates. Arguably, one could improve on these results by either using prior knowledge of the noise level or by estimating it, and based on this make the model order selection scheme more robust. Figure 11 shows the root mean squared error (RMSE) for the estimated fundamental frequencies. Instead of presenting three separate RMSE plots, Figure 11 shows an aggregate version where the MSE for the three pitches have been summed. In order to compute relevant RMSE values for PEBSI-Lite and PEBS-TV, estimates where the model order has not been correctly determined have been discarded. Thus, for an SNR

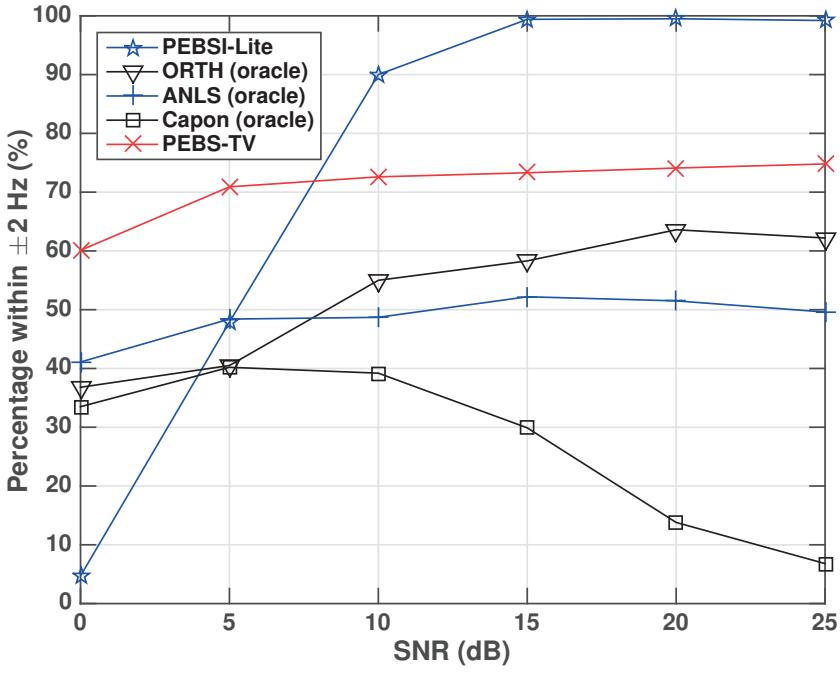


Figure 9: Percentage of estimated pitches where all three fundamental frequencies lie at most 2 Hz from the ground truth.

level of 0 dB, the RMSE values for PEBSI-Lite are based on quite few samples. However, as PEBSI-Lite finds the correct model order for high SNR levels with high probability, the corresponding RMSE values are more trustworthy in these regions. For the reference methods ORTH, ANLS, Capon, and PEBS-TV, some of the estimates deviate from the true pitch frequencies with as much as 100 Hz, resulting in very large RMSE values should all estimates be used in their computation. Thus, in order to obtain RMSE values comparable to that of the PEBSI-Lite estimates, only estimates found within 2 Hz of the true pitch frequencies are used when computing RMSE for the reference methods. With this, as can be seen in Figure 11, PEBSI-Lite performs worse than the reference methods for SNRs below and including 10 dB, while outperforming all reference methods except Capon for SNRs above and including 20 dB. Though, one should bear in mind that the RMSE values for Capon for these SNRs are based on only 15% respectively 8% of the available pitch estimates, as can be seen in Figure 9, and that the

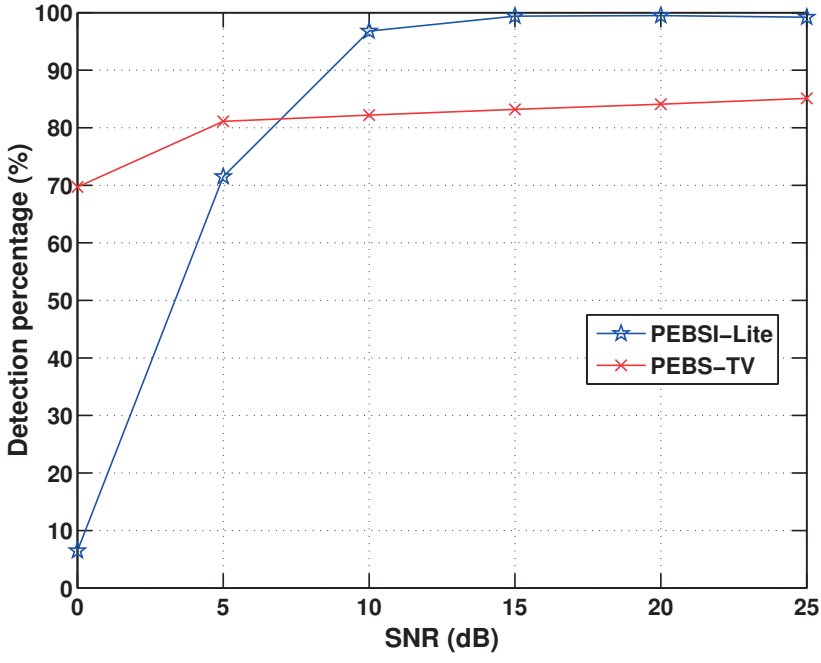


Figure 10: Estimated probability of PEBSI-Lite determining the correct number of pitches for the triple pitch test signal.

Capon method has been allowed oracle model order knowledge. Also presented in Figure 11 is the root Cramér-Rao lower bound (CRLB) for the estimates of the pitch frequencies. As the frequencies of the harmonics in this case are distinct and the additive noise is white Gaussian, the lower limit for the variance of an unbiased pitch frequency estimate $\hat{f}_k$ is given by [9]

$$\mathrm{Var}\left(\hat{f}_k\right) \geq \frac{6\sigma^2\left(f_s/2\pi\right)^2}{N(N^2-1)\sum_{\ell=1}^{L_k}|a_{k,\ell}|^2\ell^2} \tag{57}$$

where σ^2 is the power of the additive noise, $a_{k,\ell}$ is the amplitude of harmonic ℓ of pitch k, N is the number of data samples, and f_s is the sampling frequency. In analog with the summed MSE values for the pitch estimates, the root CRLB

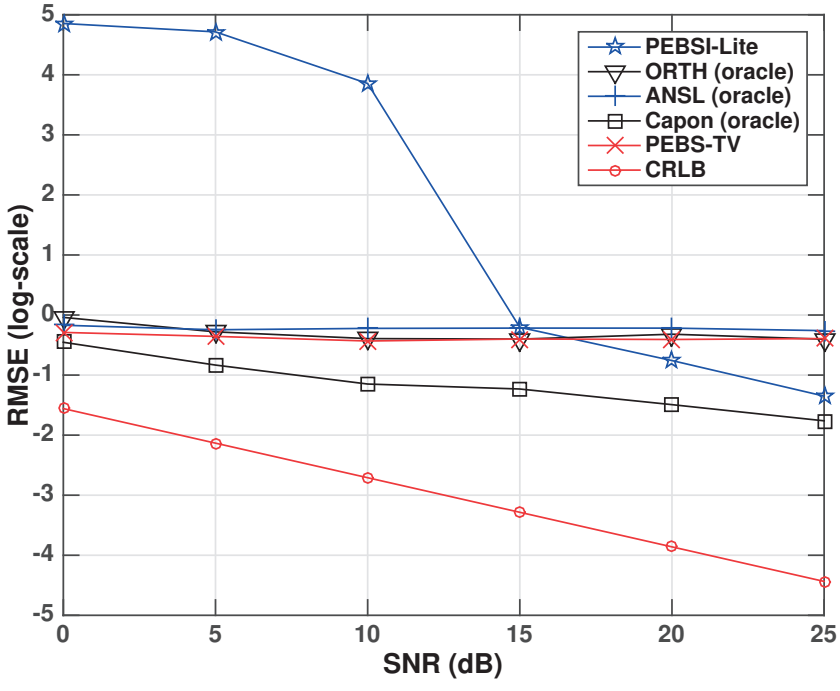


Figure 11: The RMSE for the fundamental frequency estimates for the triple pitch test signal, as compared to the (root) CRLB. For PEBSI-Lite and PEBS-TV, only estimates where the number of pitches is found are considered. For the reference methods ORTH, ANLS, Capon, and PEBS-TV only estimates where all estimated pitch frequencies lie within 2 Hz of the true pitch frequencies are considered.

curve presented here is the sum of the three separate limits, i.e.,

$$\mathrm{CRLB} = \sum_{k=1}^{3} \frac{6\sigma^2 \left(f_s/2\pi\right)^2}{N(N^2-1)\sum_{\ell=1}^{L_k} |a_{k,\ell}|^2 \ell^2} \,. \tag{58}$$

As can bee seen in Figure 11, PEBSI-Lite, as well as the other methods, fails to reach the CRLB. In an attempt to improve the PEBSI-Lite estimates for SNR levels above and including 15 dB, a non-linear least squares (NLS) search was performed, using the presented algorithm estimate as an initial estimate of all the unknown parameters, including the model orders. This means that we obtain refined

estimates of the pitch frequencies f_k contained in the vector $\mathbf{f}$ as (see, e.g, [42])

$$\mathbf{f} = \arg\max_{\mathbf{f}} \mathbf{y}^{\mathbf{H}}\mathbf{B}\left(\mathbf{B}^{\mathbf{H}}\mathbf{B}\right)^{-\mathbf{1}}\mathbf{B}^{\mathbf{H}}\mathbf{y} \tag{59}$$

where $\mathbf{B}$ is a block matrix consisting of K blocks,

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & \dots & \mathbf{B}_K \end{bmatrix} \tag{60}$$

where each block $\mathbf{B}_j$ corresponds to a separate pitch and is constructed as

$$\mathbf{B}_j = \begin{bmatrix} e^{i2\pi f_j/f_s t_1} & \dots & e^{i2\pi L_j f_j/f_s t_1} \\ \vdots & & \vdots \\ e^{i2\pi f_j/f_s t_N} & \dots & e^{i2\pi L_j f_j/f_s t_N} \end{bmatrix}. \tag{61}$$

Given that the PEBSI-Lite estimates are fairly close to the true pitch frequencies, we expect the NLS scheme to converge if we solve (59) using routines like MATLAB's *fminsearch* initialized with the PEBSI-Lite estimates. However, the success of such a scheme is not only dependent on good initial frequency estimates, we also need the true number of harmonics L_j for each pitch.

Figure 12 presents a plot of the average absolute error in the number of detected harmonics for each pitch for the test signal when using PEBSI-Lite. As can be seen, the number of detected harmonics is only correct for the third pitch even for the largest SNRs. The errors in number of harmonics for the first and second pitches are due to the relatively small amplitudes of both pitches highest order harmonics, as shown in Figure 8, making these harmonics prone to occasionally being cancelled out by the PEBSI-Lite regularization penalties. Using erroneous harmonic orders as input to the NLS search, we expect the resulting pitch frequency estimates to be somewhat biased. Indeed, this is what happens. Figure 13 presents a plot of the RMSE of the pitch frequency estimates when the PEBSI-Lite estimates for SNRs above and including 15 dB have been post-processed using NLS. As can be seen, the estimator still fails to reach the CRLB, although the estimation errors have become smaller. Note also that the slopes of the RMSE curve for PEBSI-Lite and CRLB are now somewhat different, which is due to that the erroneous harmonic orders induces varying degrees of bias in the estimates. Considering computational complexity, ANLS and ORTH are by far the fastest methods, with average running times of 0.03 and 1.6 seconds per estimation cycle on a regular PC, respectively. For Capon and PEBS-TV, the corresponding running times are 6.1 and 6.4 seconds for the considered example, respectively, while

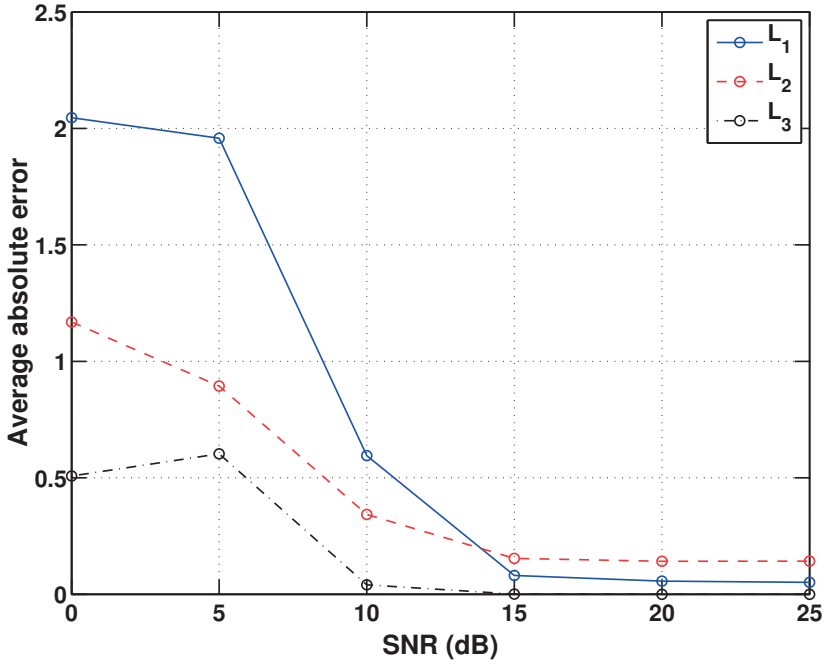


Figure 12: The average absolute error in the number of detected harmonics (L_1, L_2, L_3) for the three pitches of the test signal when using PEBSI-Lite. Only estimates where the correct number of pitches is found are considered.

running PEBSI-Lite using Algorithm 2 requires on average 40.1 seconds per estimation cycle. As a comparison, it may be noted that if one replaces Algorithm 1 in Algorithm 2 to instead use SeDuMi or SDPT3, the computation time for this step of Algorithm 2 increases almost tenfold[3]. Although Algorithm 2 is considerably more expensive to run than the reference methods, it should be noted that the method does not require any user input in terms of regularization parameter values. PEBS-TV could arguably be tuned to perform on par with PEBSI-Lite if one is allowed to change the values of its regularization parameters. However, PEBS-TV needs the setting of three parameter values and after trying only seven such triplets, the computational time is the same as running Algorithm 2 in its

[3]For all algorithms, the given execution times are those of direct implementations of the corresponding methods. Clearly, these methods can be more efficiently implemented by fully exploiting their inherent structures.

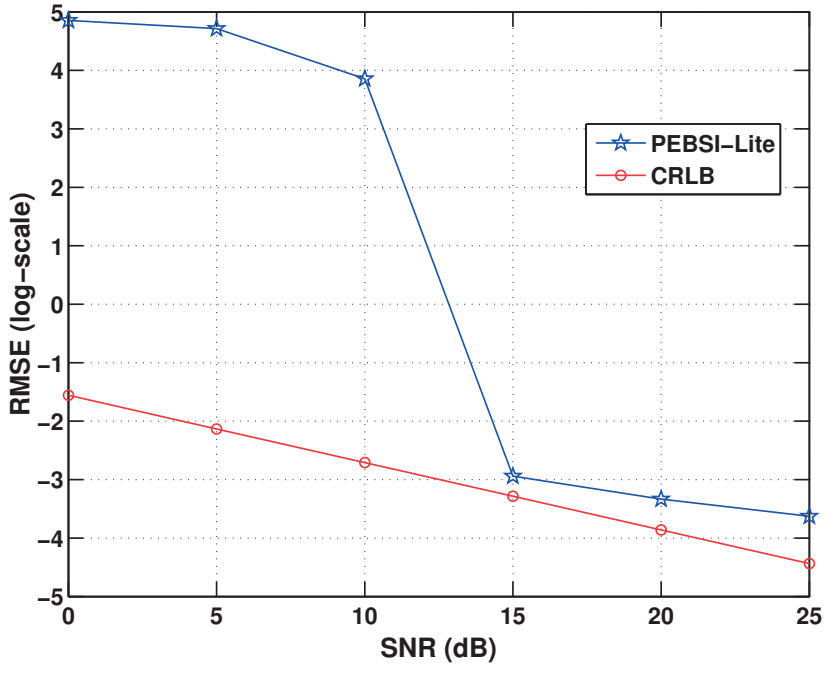


Figure 13: The RMSE for the fundamental frequency estimates where the estimates obtained using PEBSI-Lite have been improved using NLS for SNR levels 15, 20, and 25 dB, as compared to the (root) CRLB. Only estimates where the number of pitches is found are considered.

entirety.

6.3 MIDI and measured audio signals

Figure 14 shows a plot of the spectrogram of a signal consisting of three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz. The signal was sampled initially at 44 kHz and then down sampled to 20 kHz. The 311 Hz saxophone starts out alone and is after 0.45 seconds joined by the 277 Hz saxophone and after 0.95 seconds by the 440 Hz saxophone. The image is quite blurred for the later parts of the signal, but for the first half second, one can clearly see the harmonic structure of the saxophone pitch. It is worth noting that a large number of harmonics is present. Figure 15 shows pitch estimates pro-

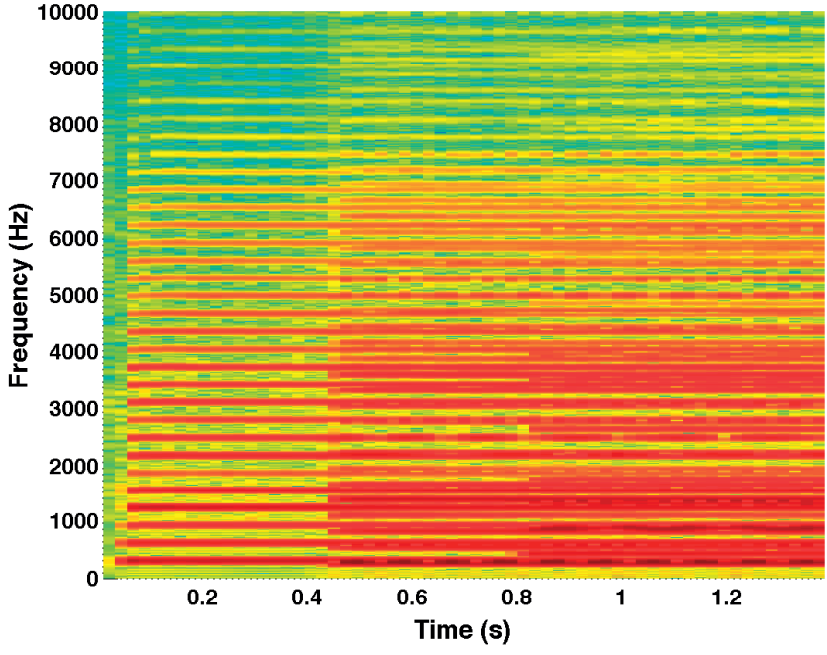


Figure 14: Spectrogram for a signal consisting of one, two and lastly three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz, respectively.

duced by Algorithm 2, using $\tau = 0.1$ and $L_{\max} = 15$, when applied to the same signal, using windows of lengths 40 ms. As can be seen, the estimates are quite accurate, with the exception of the beginning of the first tone and for a single frame where the 440 Hz pitch is mistaken for a 220 Hz pitch. It is worth noting that such errors may be avoided using the information resulting from earlier frames, for instance using an approach similar to [22]. The figure also shows the estimated pitch tracks obtained using the ESACF estimator [43]; this estimator requires *a priori* knowledge of the number of sources in the signal, but is, given this information, able to estimate the number of harmonics of each source. Here, ESACF has thus been provided oracle knowledge of the number of sources, with each source given the same maximum harmonic order as used by PEBSI-Lite (as before, the latter also has to estimate the number of sources). As can be seen from

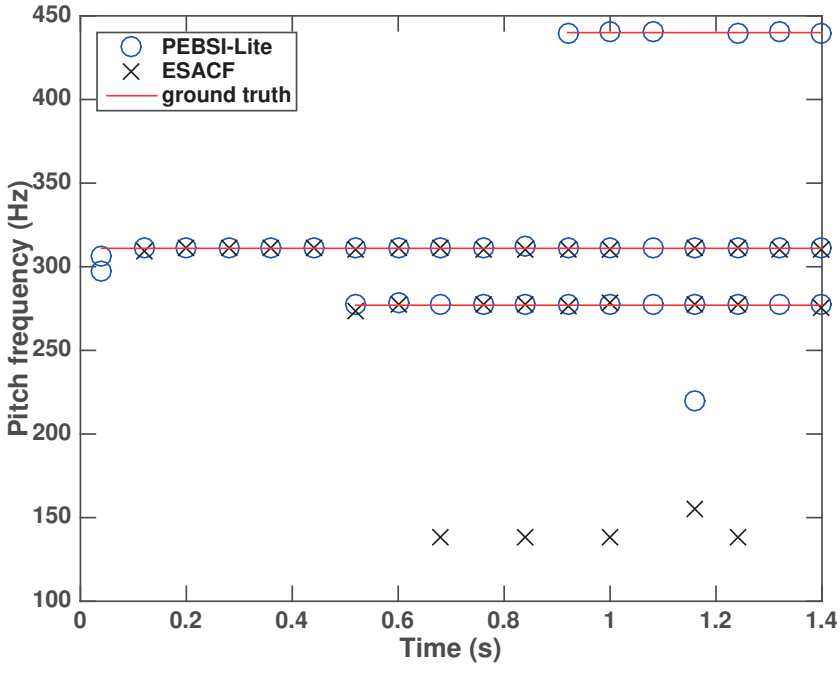


Figure 15: Pitch tracks for a signal consisting of one, two, and lastly three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz, respectively.

the figure, the ESACF estimator fails to track the pitches in several of the frames. In particular, it fails to estimate the pitch with fundamental frequency 440 Hz altogether. Furthermore, Figure 16 examines the performance of the PEBSI-Lite estimator when applied to a measured audio signal. The considered signal consists of three trumpets playing the notes A4, B4, and C♯4, which, using concert tuning, corresponds to the fundamental frequencies 440, 493.883, and 554.365 Hz, respectively. However, it should be noted, that as the musicians play with vibrato, the fundamental frequencies are not constant across the frames, which may also be seen in the resulting estimates. To facilitate for a comparison, the ground truth estimates of the fundamental frequencies have been obtained using the joint order and (single) pitch estimation algorithm ANLS, presented in [11], when applied to each individual trumpet separately. As a comparison, the figure also shows the three fundamental frequencies obtained using the ESACF estimator (which

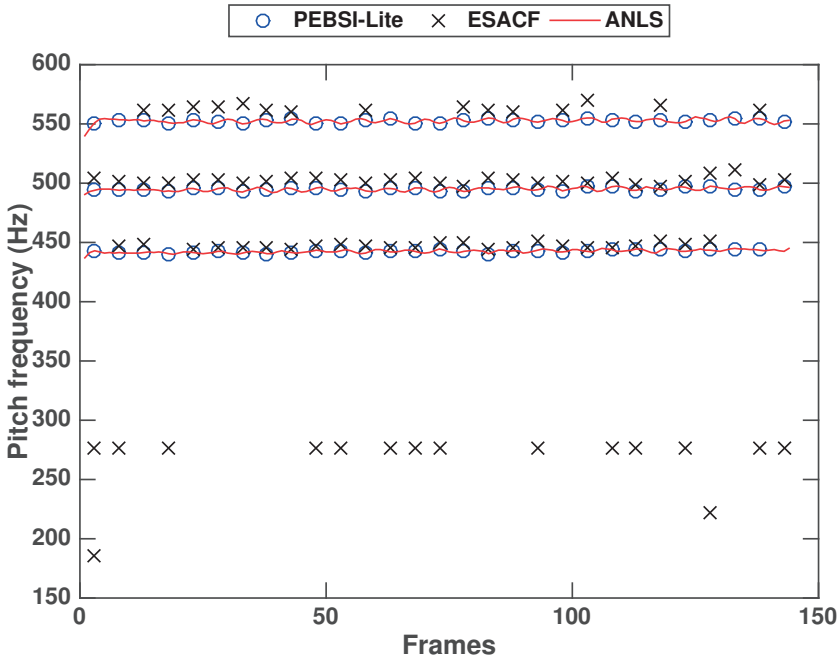


Figure 16: Pitch tracks produced by PEBSI-Lite as well as ESACF when applied to a triple-pitch signal consisting of three trumpets. The ground truth has been obtained using ANLS applied to the single source signals.

has here, again, been allowed oracle knowledge of the number of sources, but using the same maximum number of harmonics as used by PEBSI-Lite). As can be seen, PEBSI-Lite accurately tracks each of the three pitches, even catching the pitch variations caused by the vibrato. As before, it may be noted that the estimates produced by ESACF have lower accuracy as compared to PEBSI-Lite, with the ESACF estimator here erroneously picking one of the sub-octaves in some of the frames. The trumpet signal was sampled at 8 kHz. The pitch estimates where formed in non-overlapping frames of length 30ms.

The performance of PEBSI-Lite and ESACF on real audio was also evaluated on the Bach10 dataset [44]. This dataset consists of ten chorales composed by Johann Sebastian Bach. The parts are performed by a violin, a clarinet, a saxophone, and a bassoon, with each piece being approximately 30 seconds long. Each piece was sampled at 44.1 kHz, then downsampled to 22.05 kHz, and divided

Performance measure	PEBSI-Lite	ESACF
Accuracy	0.499	0.269
Precision	0.631	0.471
Recall	0.609	0.386

Table 2: Performance measures for PEBSI-Lite and ESACF when evaluated on the Bach10 dataset.

into non-overlapping frames of length 30 ms. Estimates of the ground truth fundamental frequencies in each frame were obtained by applying YIN [45] to each individual channel. Obvious errors in the YIN estimates were then corrected manually.

As before, to yield its best possible performance, ESACF was given oracle knowledge of the number of present pitches and both methods were given a maximum harmonic order of 15. For PEBSI-Lite, $\tau = 0.1$ was used. Table 2 presents the resulting measures of the *accuracy*, *precision*, and *recall* for the dataset, defined as

$$\text{Accuracy} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)+\text{FP}(t,i)+\text{FN}(t,i)} \tag{62}$$

$$\text{Precision} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)+\text{FP}(t,i)} \tag{63}$$

$$\text{Recall} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i}\text{TP}(t,i)+\text{FN}(t,i)} \tag{64}$$

where $\text{TP}(t,i)$, $\text{FP}(t,i)$, and $\text{FN}(t,i)$ denote the number of true positive, false positive, and false negative pitch estimates, respectively, for frame t in music piece i. Furthermore, T_i is the number of frames for music piece i, whereas I is the number of music pieces. Here, an estimated pitch is associated with a ground truth pitch only if its fundamental frequency lies within a quarter tone, or 3%, of the ground truth pitch (see also, e.g., [46]). To avoid the most non-stationary frames, where we cannot expect the estimates produced by PEBSI-Lite and ESACF, nor the ground truth, to be reliable, frames containing note onsets, defined as frames where one of the ground truth pitches change with more than a semitone, have been excluded when computing the measures. As can be seen from

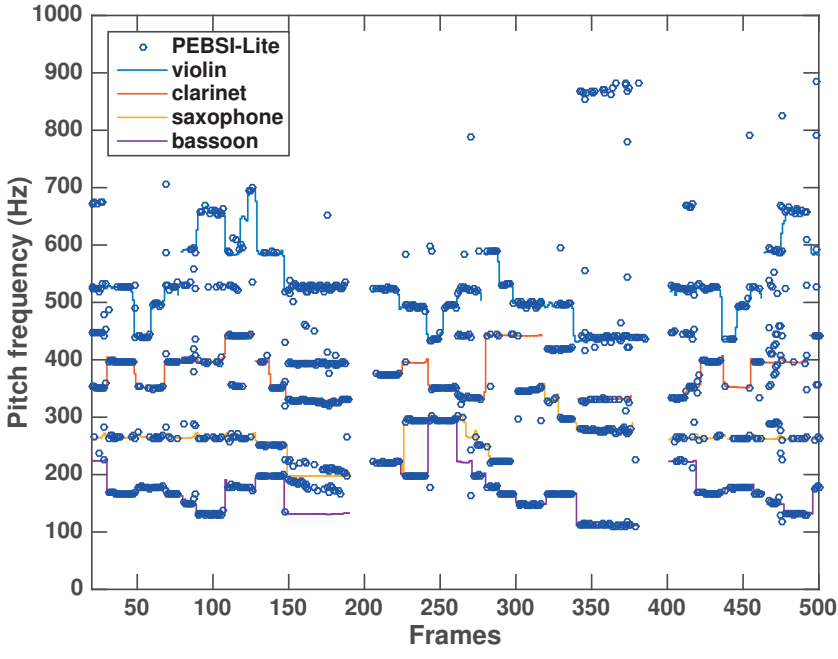


Figure 17: Pitch tracks produced by PEBSI-Lite when applied to first 15 seconds of J. S. Bach's *Ach, lieben Christen*, performed by a violin, a clarinet, a saxophone, and a bassoon. The ground truth has been obtained using YIN applied to the single source signals.

the table, PEBSI-Lite performs better than ESACF for all of the three considered measures *accuracy*, *precision*, and *recall*. As PEBSI-Lite does, for now, not incorporate information between adjacent frames, these results are most promising for what might be achievable when extended to include such information.

As an illustration of the performance, Figures 17 and 18 present pitch tracks produced by PEBSI-Lite and ESACF when applied to the first 15 seconds of one of the pieces in the dataset, namely *Ach, lieben Christen*. As can be seen from the figures, PEBSI-Lite tracks the fundamental frequencies of the violin, the saxophone, and the bassoon fairly well, while having trouble with the clarinet. This problem is caused by the shape of the spectral envelope of the clarinet, as it is dominated by a large peak at the fundamental frequency, with very weak overtones, and thus deviates from the here used model assumption of spectral

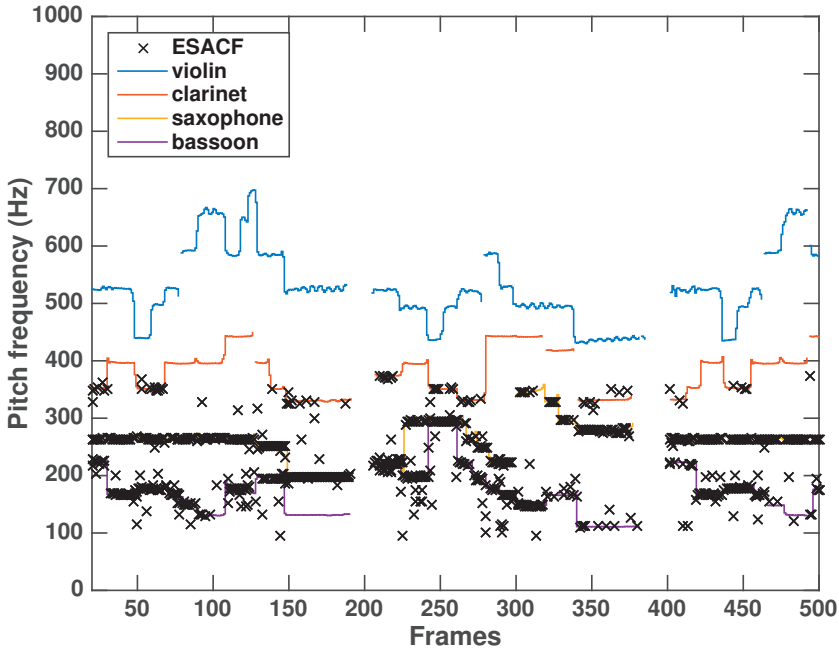


Figure 18: Pitch tracks produced by ESACF when applied to the first 15 seconds of J. S. Bach's *Ach, lieben Christen*, performed by a violin, a clarinet, a saxophone, and a bassoon. The ground truth has been obtained using YIN applied to the single source signals.

smoothness. It may also be noted that PEBSI-Lite has better performance at the stationary parts of the signal, while producing more erroneous estimates at note on- and offsets due to quickly changing spectral content. The ESACF estimator on the other hand has serious problems tracking the violin and clarinet, often picking sub-octaves estimates instead of the correct pitch, although being able to track the saxophone and bassoon fairly well.

7 Conclusions

The proposed algorithm PEBSI-Lite has been shown to be an accurate method for multi-pitch estimation. The method was shown to perform as good as, or better than, state-of-the-art methods. As compared to related methods, the presented algorithm requires fewer regularization parameters, simplifying the calibration of the method. Furthermore, the work introduces an adaptive dictionary scheme for determining suitable regularization parameters. Combined with this scheme, PEBSI-Lite was shown to outperform other multi-pitch estimation methods for high levels of SNR, while breaking down in too noisy settings. However, even if this scheme would fail to select the correct model order, the obtained efficient dictionary facilitates a more rigorous grid search in terms of computational complexity. Such a grid search could also exploit information about the solution surface obtained from the line search. Using an additional refinement step, the proposed algorithm is found to yield estimates reasonably close to being efficient, if considering that the method has not been allowed any knowledge of the model order of the signal.

References

[1] M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard, "Signal Processing for Music Analysis," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 6, pp. 1088–1110, 2011.

[2] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic Music Transcription: Challenges and Future Directions," *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 407–434, December 2013.

[3] A. Wang, "An Industrial Strength Audio Search Algorithm," in *4th International Conference on Music Information Retrieval*, Baltimore, Maryland USA, Oct. 26-30 2003.

[4] N. H. Fletcher and T. D. Rossing, *The Physics of Musical Instruments*, Springer-Verlag, New York, NY, 1988.

[5] H. Fletcher, "Normal vibration frequencies of stiff piano string," *Journal of the Acoustical Society of America*, vol. 36, no. 1, 1962.

[6] N. R. Butt, S. I. Adalbjörnsson, S. D. Somasundaram, and A. Jakobsson, "Robust Fundamental Frequency Estimation in the Presence of Inharmonicities," in *38th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vancouver, May 26–31, 2013.

[7] S. M. Nørholm, J. R. Jensen, and M. G. Christensen, "On the Influence of Inharmonicities in Model-Based Speech Enhancement," in *European Signal Processing Conference*, Marrakesh, Sept. 10-13 2013.

[8] T. Nilsson, S. I. Adalbjörnsson, N. R. Butt, and A. Jakobsson, "Multi-Pitch Estimation of Inharmonic Signals," in *European Signal Processing Conference*, Marrakech, Sept. 9-13, 2013.

[9] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[10] M. G. Christensen, S. H. Jensen, S. V. Andersen, and A. Jakobsson, "Subspace-based Fundamental Frequency Estimation," in *European Signal Processing Conference*, Vienna, September 7-10 2004.

[11] M. G. Christensen, A. Jakobsson, and S. H. Jensen, "Joint High-Resolution Fundamental Frequency and Order Estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 15, no. 5, pp. 1635–1644, July 2007.

[12] M. G. Christensen, P. Stoica, A. Jakobsson, and S. H. Jensen, "Multi-pitch estimation," *Signal Processing*, vol. 88, no. 4, pp. 972–983, April 2008.

[13] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[14] M. Genussov and I. Cohen, "Multiple fundamental frequency estimation based on sparse representations in a structured dictionary," *Digit. Signal Process.*, vol. 23, no. 1, pp. 390–400, Jan. 2013.

[15] C. Kim, W. Chang, S-H. Oh, and S-Y. Lee, "Joint Estimation of Multiple Notes and Inharmoncity Coefficient Based on f0-Triplet for Automatic Piano Transcription," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1536–1540, December 2014.

[16] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 6, pp. 1643–1654, Aug. 2010.

[17] K. O'Hanlon, "Structured Sparsity for Automatic Music Transcription," in *37th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Kyoto, 25-30 March 2012.

[18] M. Bay, A.F. Ehmann, J.W. Beauchamp, P. Smaragdis, and J.S. Downie, "Second Fiddle is Important Too: Pitch Tracking Individual Voices in Polyphonic music," in *13th Annual Conference of the International Speech Communication Association*, Portland, September 2012, pp. 319–324.

[19] A. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 804–816, 2003.

[20] P. Smaragdis and J.C. Brown, "Non-Negative Matrix Factorization for Polyphonic Music Transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 177–180.

[21] N. Bertin, R. Badeau, and E. Vincent, "Enforcing Harmonicity and Smoothness in Bayesian Non-Negative Matrix Factorization Applied to Polyphonic Music Transcription," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 18, no. 3, pp. 538–549, 2010.

[22] S. Karimian-Azari, A. Jakobsson, J. R. Jensen, and M. G. Christensen, "Multi-Pitch Estimation and Tracking using Bayesian Inference in Block Sparsity," in *23rd European Signal Processing Conference*, Nice, Aug. 31-Sept. 4 2015.

[23] R. Gribonval and E. Bacry, "Harmonic decomposition of audio signals with matching pursuit," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 101–111, jan. 2003.

[24] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, Seoul, July 2008, pp. 10225–10229.

[25] M. A. T. Figueiredo and J. M. Bioucas-Dias, "Algorithms for imaging inverse problems under sparsity regularization," in *Proc. 3rd Int. Workshop on Cognitive Information Processing*, May 2012, pp. 1–6.

[26] T. Kronvall, M. Juhlin, S. I. Adalbjörnsson, and A. Jakobsson, "Sparse Chroma Estimation for Harmonic Audio," in *40th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Brisbane, Apr. 19-24 2015.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[28] S. L. Marple, "Computing the discrete-time "analytic" signal via FFT," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2600–2603, September 1999.

[29] T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "Joint DOA and Multi-Pitch Estimation Using Block Sparsity," in *39th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Florence, May 4-9 2014.

[30] T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "Joint DOA and Multi-pitch estimation via Block Sparse Dictionary Learning," in *22nd European Signal Processing Conference (EUSIPCO)*, Lisbon, Sept. 1-5 2014.

[31] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[32] E. J. Candès, J. Romberg, and 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.

[33] E. J. Candès, M. B. Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted l_1 Minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec. 2008.

[34] R. H. Tutuncu, K. C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using SDPT3," *Mathematical Programming Ser. B*, vol. 95, pp. 189–217, 2003.

[35] P. Stoica and Y. Selén, "Model-order Selection — A Review of Information Criterion Rules," *IEEE Signal Process. Mag.*, vol. 21, no. 4, pp. 36–47, July 2004.

[36] C. D. Austin, R. L. Moses, J. N. Ash, and E. Ertin, "On the Relation Between Sparse Reconstruction and Parameter Estimation With Model Order Selection," *IEEE J. Sel. Topics Signal Process.*, vol. 4, pp. 560–570, 2010.

[37] A. Panahi and M. Viberg, "Fast Candidate Point Selection in the LASSO Path," *IEEE Signal Processing Letters*, vol. 19, no. 2, pp. 79–82, Feb 2012.

[38] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, April 2004.

[39] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, no. 1, pp. 91–108, January 2005.

[40] H. Hoefling, "A Path Algorithm for the Fused Lasso Signal Approximator," *Journal of Computational and Graphical Statistics*, vol. 19, no. 4, pp. 984–1006, December 2010.

[41] R.J. Tibshirani and J. Taylor, "The Solution Path of the Generalized Lasso," *The Annals of Statistics*, vol. 39, no. 3, pp. 1335–1371, June 2011.

[42] P. Stoica and R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, N.J., 2005.

[43] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 708–716, 2000.

[44] Z. Duan and B. Pardo, "Bach10 dataset," http://music.cs.northwestern.edu/data/Bach10.html, Accessed December 2015.

[45] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Am.*, vol. 111, no. 4, pp. 1917–1930, 2002.

[46] M. Bay, A. F. Ehmann, and J. S. Downie, "Evaluation of Multiple-F0 Estimation and Tracking Systems," in *International Society for Music Information Retrieval Conference*, Kobe, Japan, October 2009.

C

Paper C

Sparse Modeling of Chroma Features

Ted Kronvall, Maria Juhlin, Johan Swärd,
Stefan Ingi Adalbjörnsson, and Andreas Jakobsson

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

This work treats the estimation of chroma features for harmonic audio signals using a sparse reconstruction framework. Chroma has been used for decades as a key tool in audio analysis, and is typically formed using a periodogram-based approach that maps the fundamental frequency of a musical tone to its corresponding chroma. Such an approach often leads to problems with tone ambiguity, which we adress via sparse modeling, allowing us to appropriately penalize ambiguous estimates while taking the harmonic structure of tonal audio into account. Furthermore, we also allow for signals to have time-varying envelopes. Using a spline-based amplitude modulation of the chroma dictionary, the presented estimator is able to model longer frames than what is conventional for audio, as well as to model highly time-localized signals, and signals containing sudden bursts, such as trumpet or trombone signals. Thus, we may retain more signal information as compared to alternative methods. The performance of the proposed methods is evaluated by analyzing average estimation errors for synthetic signals, as compared to the Cramér-Rao lower bound, and by visual inspection for estimates of real instrument signals, showing strong visual clarity, as compared to other commonly used methods.

Keywords: Chroma, multi-pitch estimation, sparse modeling, amplitude modulation, block sparsity, ADMM

1 Introduction

Music is an art-form that people have enjoyed for millennia. Perhaps music is even enjoyed more today, as the development of personalized computers and smart telephones have enabled ubiquitous music listening, automatic identification of songs, or even the chance for anyone to be a self-made DJ. When listening, learning, composing, mixing, and identifying music, there are a number of musical features one may utilize (see, e.g. [1]). One of the fundamental building blocks in music, the musical note, is a periodic sound, typically characterized by its pitch, timbre, intensity, and duration. For transcription purposes, i.e., to separate one tone from another, pitch serves as the common descriptor. From a conventional perspective, pitch is measured on an ordinal scale, at which a pitch is humanly perceived as either higher, lower, or the same as another pitch. However, from a perspective of scientific audio analysis, pitches are quantified using an interval scale, in which its spectral distribution of energy is modeled. A single pitch may be seen as a superposition of several narrowband spectral peaks, which are approximately integer multiples of a fundamental frequency. Thus, we refer to the group of frequencies as the pitch, and to each frequency component as the harmonic, or, alternatively, as the partial harmonic. As to the fundamental frequency, it is either the lowest partial, or, if that partial is missing, the smallest spectral distance between adjacent partials. The number of harmonics in a certain pitch, as well as the relative power between these, varies greatly between different sounds, as well as over time. Identifying pitches in a way similar to our human perception has proved to be a difficult estimation problem. Partly, this difficulty is due to coinciding frequency components between certain pitches. For instance, two pitches, where one has exactly twice the fundamental frequency of the other, are referred to as being octave equivalent, as the relative distance by a factor of two is called an octave. These will typically share a large number of partials, often making an estimation procedure ambiguous between octaves. To further complicate matters, other pairs of pitches may also have many coinciding partials, and these are typically found together in audio, an aspect which is referred to as harmony, since they are perceptually pleasant to hear [2]. Jointly estimating several pitches in a signal, i.e., multi-pitch estimation, has been thoroughly examined in the literature (see e.g., [3–5], and the references therein). However, separating intricate combinations of frequency components into multiple pitches often proves difficult, even if the harmonic structure of each musical tone is taken into account. Typically, issues arise when the complexity of the audio signal increases, such that

there are simultaneously two or more pitches with overlapping spectral content present, for instance played by two or more instruments. In the Western musicological system, the frequency interval corresponding to an octave is discretized into twelve intervals, called semi-tones. By gathering all pitches with octave equivalence to their respective semi-tone, these form twelve groups of pitches, called chroma. As octave equivalent pitches share a large number of harmonics, the notion of chroma is thus a method for grouping together those pitches which are perceived as most similar. Therefore, chroma features are widely used in applications such as cover song detection, transcription, and recommender systems (see, e.g. [6–8]). Most methods for chroma estimation begin by obtaining estimates of the pitches in a signal, which are then mapped into their respective chroma. Some of these take the harmonic structure into account, and others do not. The commonly used method by Ellis [9] is formed via a time-smoothed version of the Short-Time Fourier Transform (STFT), whereas the CP and CENS methods by Müller and Ewert [10] use a filterbank approach. The method in [11] uses a sparse methodology, and the method in [12] uses a non-negative least squares approach. Neither of these take the harmonic structure of pitches into account. Other approaches instead allow for the harmonic structure, such as the method presented in [13], which uses a comb filtering technique, and the method in [14], in which post-processing on the periodogram is performed. Most existing methods have in common that their estimates are not directly formed from the actual data, but rather on a representation of these measurements, such as, for instance, using the STFT or the magnitude of the periodogram. Herein, we propose to estimate the chroma using a sparse model reconstruction framework, where explicit model orders are not required. The estimate is found as the solution to a convex optimization problem, where the solution is obtained as a linear combination of an over-complete chroma-based set of Fourier basis functions. Overfitting is avoided by introducing convex penalties promoting solutions having the sought chroma structure. The model orders are thus set implicitly, using tuning parameters which may be obtained using cross-validation, or by utilizing some simple heuristics. In this paper, we generalize upon the work in [5], taking into account the chroma structure, as well as allowing the frequency components to have time-varying amplitudes. The proposed extension increases robustness, as it allows for highly non-stationary signals, or signals with sudden bursts, like trumpets, whose nature may easily be misinterpreted when using ordinary chroma selection techniques. As in [15], the extended model uses a spline basis to detail the time-

varying envelope of the signal, thereby enabling the amplitudes to evolve smoothly with time. The theoretical performance of the proposed estimator is verified using synthetic signals, which are compared to the Cramér-Rao Lower Bound (CRLB), which we here present for the chroma signal model. The practical use of the proposed estimator is illustrated using some excerpts from a recorded trumpet signal, showing an increased visual performance, as compared to some typical reference methods.

2 The chroma signal model

A sound signal typically contains a broad band of frequency content. However, for tonal audio, it is well-known that a predominant part of the spectral energy is confined to a small number of frequency locations. Let $\psi(f,\ell)$ denote the function which describes the frequency of the ℓ:th component. If this function is known, the entire group of components, or partials, representing a musical tone may be described by their fundamental frequency, f. Many oscillating sources, such as, for instance, the human vocal tract and stringed, or wind, instruments, emit tonal audio where the partials are integer multiples of the fundamental, i.e.,

$$\psi(f,\ell) = f\ell, \quad \ell \in \mathcal{L} \subseteq \mathbb{N} \tag{1}$$

where $\mathcal{L}$ denotes the index set of partials present in the signal. However, for an arbitrary $\mathcal{L}$, the definition in (1) is not sufficient to uniquely describe a pitch, as the set of frequencies may map to infinitely many combinations of f and $\mathcal{L}$. For example, for any $n \in \mathbb{N}$, the two pitches

$$\Psi = \{\psi(f,\ell) : f \in \mathbb{R}, \forall \ell \in \mathcal{L} \subseteq \mathbb{N}\} \tag{2}$$

$$\Psi' = \left\{ \psi(f',\ell') : f' = \frac{f}{n}, \forall \ell' \in \mathcal{L}' = \{n\ell : \ell \in \mathcal{L}\} \right\} \tag{3}$$

have identical frequency components. Therefore, some constraints need to be imposed on $\mathcal{L}$. A common assumption for pitches is spectral smoothness of the harmonics, i.e., that adjacent harmonics should be of comparable magnitude [16]. This implies that $\mathcal{L}$ typically has few missing harmonics, and that n is as small as possible. However, in some signals, the first harmonic might be missing, so rather than defining the pitch as the signal's smallest frequency component, we define the fundamental frequency more rigorously. If the set of frequencies in a pitch

may be described by (2), then for any $n \in \mathbb{Q}$, the fundamental frequency is the largest $f' = f/n$ which fulfill (3), i.e., which ensures that $\mathcal{L}' = \{n\ell, \ell \in \mathcal{L}\} \subseteq \mathbb{N}$. The index set therefore plays a vital role in the definition of the pitch frequency. Furthermore, because of the harmonic structure, many different pitches will have coinciding partials. To illustrate this, consider two pitches

$$\psi = \{\psi(f, \ell) : f \in \mathbb{R}, \forall \ell \in \{1, 2, \ldots, L\}\} \tag{4}$$

$$\psi' = \left\{ \psi(f', \ell') : f' = \frac{f}{n}, \forall \ell' \in \{1, 2, \ldots, nL\} \right\} \tag{5}$$

which consist of all harmonics from $\ell = 1$ up to L and nL, respectively. Here, n may be a rational number, as long as (5) is fulfilled. Indeed, both pitches are unique according to our definition. Still, they will share a large number of harmonics, in fact L of them, as ψ forms a perfect subset of ψ', i.e., $\psi \in \psi'$, and they will also, as sounds, be perceived as being similar, especially if n is small. This motivates the introduction of chromas, which are also referred to as pitch classes. The chroma, which means 'color' in greek, is the collection of pitches which are an integer number of octaves apart, meaning that n in (5) fulfills

$$n = 2^{-m}, m \in \mathbb{Z} \tag{6}$$

with $m \in \mathbb{N}$ denoting the octave, which implies that $n \in \mathbb{Q}$. The fundamental frequency may thus be modeled in terms of its chroma, $\tilde{c}$, and its octave, m, as (see also, e.g., [1])

$$f = f_b \, 2^{\tilde{c}+m} \tag{7}$$

where $\tilde{c} \in [0, 1)$ and f_b denote the chroma class and a base (tuning) frequency, respectively. Using this formulation, the parametric pitch model presented in [17] may be extended into a parametric chroma model. Thus, the frequency peaks in a complex-valued[1] noise-free musical tone may be modeled as

$$x(t) = \sum_{\ell=1}^{L} a_\ell(t) e^{i2\pi f_b 2^{\tilde{c}+m} \ell t} \tag{8}$$

for a time-frame $t = 1, ..., N$, where $a_\ell(t)$ denotes the complex-valued amplitude of the ℓ:th harmonic, which may be either constant over the time-frame, or may

[1] In order to simplify notation, we here examine the discrete-time analytic signal version (see, e.g., [3, 18]) of the measured audio signal.

vary slowly. Here, $\tilde{c}$, m, and L denote the chroma, octave, and the number of sinusoids of the tone, respectively. It may be noted that the data is thus modeled in the time domain, as this is shown to render more efficient estimates than using the magnitude STFT [3]. In most Western music, there are twelve chroma classes, defined as the twelve semitones

$$C, C\#, D, D\#, E, F, F\#, G, G\#, A, A\#, \text{ and } B \tag{9}$$

and the concatenation of a chroma with its octave number, e.g., A4, denotes a musical tone. Here, two adjacent semitones are relatively spaced by $2^{1/12}$. Thus, the chroma parameter $\tilde{c}$ is discretized into twelve values, uniformly spaced on $[0, 1)$, i.e.,

$$\tilde{c} \in \left\{0, \frac{1}{12}, \frac{2}{12} \ldots, \frac{11}{12}\right\} \tag{10}$$

The tuning parameter f_b often varies somewhat amongst musicians, but a common standard sets 'A4' to 440 Hz [19]. This corresponds to $\tilde{c} = 9/12$, and $m = 4$, yielding the (normalized) tuning frequency

$$f_b = \frac{440}{f_s} 2^{-(9/12+4)} \tag{11}$$

where f_s denotes the sampling frequency. Our auditory system does not only perceive tones with these chroma as being distinctly different from each other, but also as equally spaced, which gives credit to the idea that our hearing is log-tempered. Furthermore, coinciding harmonics are not restricted to pitches within the same chroma, as pitches in different chromas may yield coinciding harmonics. For instance, for $n = 3/2 \approx 2^{7/12}$, the two pitches in our example will have many coinciding partials; two such tones are referred to as fifths. Fifths are thus spaced by approximately seven semitones and are commonly used together in musical compositions, as the overlapping spectral content is often deemed perceptually pleasant. Thus, if assuming that a polyphonic audio signal consists of K superimposed musical tones, the signal may be well modeled as

$$y(t) = x(t) + e(t) \tag{12}$$

where

$$x(t) = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} a_{k,\ell}(t) e^{i2\pi f_b 2^{c_k} + m_k \ell t} \tag{13}$$

with the subscript k denoting the parameter of the k:th tone, and where e(t) is some form of additive noise. As (13) only models the sinusoidal part of the signal $y(t)$, any other features, such as, e.g., the timbre, will, without any loss of generality, be modeled as a part of the noise. In this work, the amplitude is allowed to be either constant, i.e., $a_{k,\ell}(t) = a_{k,l}, \forall(k,\ell)$, or slowly varying within each considered time-frame of N samples. Reminiscent to the approach in [15], we model the amplitude's time-varying nature using a spline basis with uniformly spaced knots (see, e.g., [20, p. 151]), i.e., such that the amplitudes in the time-frame follow a superposition of R B-spline bases,

$$a_{k,\ell}(t) = \sum_{r=1}^{R} \gamma_r(t) s_{k,\ell,r} \tag{14}$$

where the r:th spline base is weighted by an unknown complex amplitude, $s_{k,\ell,r}$.

3 Sparse chroma modeling and estimation

One way of estimating the unknown parameters in (13) may be to form the estimate as the one minimizing the (possibly weighted) squared estimation residuals, e.g., by using the non-linear least squares (NLS) algorithm. However, such an estimate requires precise knowledge about the model orders, something which generally is unknown. Such model orders are typically difficult to estimate for multi-pitch signals, as both the number of pitches and the number of harmonics in each pitch must then be determined. Furthermore, even if the true model orders are known, the NLS estimate will still require solving a multidimensional minimization over a typically multimodal cost function, thus necessitating an accurate search initialization [21]. On the other hand, if one tries to estimate the tonal content using, for instance, a periodogram-based approach, where the spectral peaks are estimated without taking the chroma structure into account, and thereafter grouping together the resulting estimates, this yields an involved combinatorial problem, as a number of frequency components typically belong in several tones, due to harmony. Instead, in this work, we construct an estimator based on the assumption that any given frequency component will be part of an ordered group of harmonic frequencies, i.e., a pitch. To achieve this, we propose to use a sparse modeling approach, reminiscent of the one presented in [5], where the non-linear model in (13) is replaced by a linear approximation of it, consisting of a highly overdetermined linear system, where the number of non-zero parameters

in the sought solution should be few, i.e., the solution should be sparse. Thereby, one may take the spectral structure of musical tones into account, while circumventing the need for explicitly estimating the model orders. Thus, consider the linear approximation

$$x(t) \approx \tilde{x}(t) = \sum_{c=0}^{11} \sum_{m=M_{\min}}^{M_{\max}} \sum_{\ell=1}^{L_{\max}} a_{c,m,\ell} e^{i2\pi f_b \ell t 2^{(c/12+m)}} \tag{15}$$

where $\tilde{x}(t)$ denotes the signal model representing the chromas in the Western musicological system, as described in (9)-(10). By denoting the twelve semitones using $c = 12\tilde{c}$, ordered as in (9), (15) includes all candidate tones within a range of octaves, from $M_{\min}$ to $M_{\max}$. Furthermore, $L_{\max}$ denotes the maximal number of harmonics considered, and $a_{c,m,\ell}$ the (complex-valued) amplitude for the ℓ:th harmonic in the m:th octave of pitch class c. From this approximation, it is clear that the spectral content is discretized into $Q = 12(M_{\max} - M_{\min})L_{\max}$ feasible frequencies, grouped into pitches of the same chroma. Also, as noted above, many of the harmonics between tones typically coincide, and it is therefore insufficient to simply map individual frequencies to a chroma, as they will likely map to several other chromas as well. To illustrate the sought sparsity structure of the solution, let

$$\mathbf{\Psi} = \left\{ \{a_{c,m,1}, \ldots, a_{c,m,L_{\max}}\}_{m=M_{\min},\ldots,M_{\max}} \right\}_{c=0,\ldots,11} \tag{16}$$

be the set of linear amplitude parameters for all possible frequencies in the overcomplete model. As the set $\mathbf{\Psi}$ is much larger than the actual solution set, most amplitudes, $a_{c,m,\ell}$, in (16) should be equal to zero, i.e., $\mathbf{\Psi}$ should be sparse. If, for instance, only the key C#5 is played, then all amplitudes, except $a_{1,5,\ell}$, for those ℓ present in this tone, should be zero. To measure the fit of the selected and estimated non-zero parameters, one may examine the minimum of the squared model residuals, by solving

$$\underset{\mathbf{\Psi}}{\text{minimize}} \sum_{t=1}^{N} \left| y(t) - \tilde{x}_{\mathbf{\Psi}}(t) \right|^2 \tag{17}$$

However, such a minimization will not promote the sought sparsity structure, and we therefore impose constraints to ensure a more desirable sparsity structure. In principle, we will do so by adding penalties to (17), reminiscent to the ones

used in [22–24], which add cost to non-desirable solutions that violate the sought sparsity pattern. The use of these will be somewhat different depending on if the amplitudes are allowed to vary or not; in the next two sections, we will deal with the two approaches separately.

3.1 Promoting sparsity when the amplitudes are constant

We proceed by first detailing the proposed chroma estimation procedure for the case without amplitude modulation. To simplify the exposition, consider the signal model in (15) for the entire time-frame expression on vector form as

$$\mathbf{y} = \left[\begin{array}{cccc} y(1) & \ldots & y(N) \end{array} \right]^T \tag{18}$$

$$= \sum_{c=0}^{11} \mathbf{W}_c \, \mathbf{a}_c + \mathbf{e} \triangleq \mathbf{W}\mathbf{a} + \mathbf{e} \tag{19}$$

where $(\cdot)^T$ denotes the transpose, and where

$$\mathbf{W} = \left[\begin{array}{ccc} \mathbf{W}_0 & \ldots & \mathbf{W}_{11} \end{array} \right]^T \tag{20}$$

$$\mathbf{W}_c = \left[\begin{array}{ccc} \mathbf{W}_{c,M_0} & \ldots & \mathbf{W}_{c,M} \end{array} \right]^T \tag{21}$$

$$\mathbf{W}_{c,m} = \left[\begin{array}{ccc} \mathbf{w}_{c,m}^1 & \ldots & \mathbf{w}_{c,m}^{L_{\max}} \end{array} \right]^T \tag{22}$$

$$\mathbf{w}_{c,m} = \left[\begin{array}{ccc} e^{i2\pi 2^{(c/12+m)}} & \ldots & e^{i2\pi N 2^{(c/12+m)}} \end{array} \right]^T \tag{23}$$

denote the dictionary of candidate tones and their partials, respectively. Also, let

$$\mathbf{a} = \left[\begin{array}{ccc} \mathbf{a}_c^T & \ldots & \mathbf{a}_c^T \end{array} \right]^T \tag{24}$$

$$\mathbf{a}_c = \left[\begin{array}{ccc} \mathbf{a}_{c,M_0}^T & \ldots & \mathbf{a}_{c,M}^T \end{array} \right]^T \tag{25}$$

$$\mathbf{a}_{c,m} = \left[\begin{array}{ccc} a_{c,m,1} & \ldots & a_{c,m,L_{\max}} \end{array} \right]^T \tag{26}$$

denote the linear amplitude parameters, $\mathbf{\Psi}$, of the over-complete dictionary on vector form. Thus, the blocks-within-blocks dictionary, $\mathbf{W} \in \mathbb{C}^{N \times Q}$, consists of twelve blocks of candidate chroma, such that each chroma is a block of $(M_{\max} - M_{\min})$ octave equivalent pitches, where each of these, in turn, consists of a block of $L_{\max}$ Fourier vectors. Our proposed method obtains the sought sparsity structure

by minimizing

$$||\mathbf{y} - \mathbf{W}\mathbf{a}||_2^2 + \lambda_2||\mathbf{a}||_1 + \lambda_3 \sum_{c=0}^{11} ||\mathbf{a}_c||_2 + \lambda_4||\mathbf{F}\mathbf{a}||_1 \tag{27}$$

where λ_i, for $i = 2, 3, 4$, denotes the user-defined sparse regularizers which weigh the importance between the different terms in (27), and where $\mathbf{F} \in \mathbb{C}^{(Q-1)\times Q}$ denotes the first order difference matrix, having elements $\mathbf{F}_{i,i} = 1$ and $\mathbf{F}_{i,i+1} = -1$ for $i = 1, \ldots, Q-1$, and zeros elsewhere. The first term in (27) penalizes the distance between the model and the measured signal, whereas the second term governs the overall sparsity of the amplitudes, thus forcing small values of $\mathbf{a}$ to be zero, affecting all indices equally. The third term is a group sparsity penalty, promoting sparsity between chromas, thereby countering the contributions from other chromas with partially overlapping spectral content. The last term in (27) is a total variation penalty which will penalize non-zero amplitudes at wrong octaves within the chroma, so that they will be efficiently clustered.

3.2 Promoting sparsity while allowing for time-varying amplitudes

To also allow for time-varying amplitudes, one has to consider some additions as well as some alterations to the earlier described method. Firstly, to allow for amplitude modulation, one has to extend the original problem with an additional parameter dimension. Using (14), the amplitudes' time-varying nature may be expressed on vector form as

$$\mathbf{a}_{k,l} = \sum_{r=1}^{R} \boldsymbol{\gamma}_r s_{r,k,l} = \Gamma \mathbf{s}_{k,l} \tag{28}$$

so that the amplitude vector, $\mathbf{a}_{k,l}$, is a linear combination of the $\boldsymbol{\gamma}_r \in \mathbb{R}^{N\times 1}$, for $r = 1, \ldots, R$, spline basis vectors, and where $s_{r,k,l}$ denotes the corresponding complex amplitude at spline point r of the l:th harmonic for the k:th pitch, and with

$$\mathbf{a}_{k,l} = \begin{bmatrix} a_{k,l}(1) & a_{k,l}(2) & \cdots & a_{k,l}(N) \end{bmatrix}^T \tag{29}$$

$$\mathbf{s}_{k,l} = \begin{bmatrix} s_{1,k,l} & s_{2,k,l} & \cdots & s_{R,k,l} \end{bmatrix}^T \tag{30}$$

$$\Gamma = \begin{bmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_R \end{bmatrix} \tag{31}$$

Using this formulation, the signal model for the time dependent amplitude becomes

$$\mathbf{y} = \sum_{m=M_0}^{M} \sum_{c=0}^{11} \mathrm{diag}(\mathbf{\Gamma} \mathbf{S}_{c,m} \mathbf{W}_{c,m}^T), \tag{32}$$

where

$$\mathbf{S}_{c,m} = \begin{bmatrix} \mathbf{s}_{c,m,1} & \cdots & \mathbf{s}_{c,m,L_{max}} \end{bmatrix} \tag{33}$$

$$\mathbf{s}_{c,m,l} = \begin{bmatrix} s_{1,c,m,l} & \cdots & s_{R,c,m,l} \end{bmatrix}^T \tag{34}$$

As a result, the sought chroma features of the considered signal frame may be found as the parameters minimizing

$$\underset{S_{0,M_0} \cdots S_{11,M}}{\text{minimize}} \frac{1}{2} \left|\left| \mathbf{y} - \sum_{c=0}^{11} \sum_{m=M_0}^{M} \mathrm{diag}(\mathbf{\Gamma} \mathbf{S}_{c,m} \mathbf{W}_{c,m}^T) \right|\right|_2^2 \tag{35}$$

where $\mathbf{y}$ denotes the vector containing the measured signal. To promote a sparse solution, one may rewrite and extend (35) as

$$\underset{S_P}{\text{minimize}} \frac{1}{2} \left|\left| \mathbf{y} - \sum_{p=0}^{P} \mathrm{diag}(\mathbf{\Gamma} \mathbf{S}_p \mathbf{W}_p^T) \right|\right|_2^2 \tag{36}$$

$$+ \lambda_2 \sum_{p=0}^{P} \sum_{l=1}^{L_{max}} \left|\left| \mathbf{s}_{p,l} \right|\right|_2 + \lambda_3 \sum_{c=0}^{11} \left|\left| \tilde{\mathbf{S}}_c \right|\right|_F \tag{37}$$

where the reparametrization from c, m to p is $p = 12(m - M_0) + c$, with P denoting the total number of chroma-octave pairs in the dictionary, and with

$$\tilde{\mathbf{S}}_c = \begin{bmatrix} \mathbf{S}_{c,M_0} & \cdots & \mathbf{S}_{c,M} \end{bmatrix} \tag{38}$$

The first term in (37) measures the distance between the signal model and the measured data, the second term in (37) has the effect of setting columns in $\mathbf{s}_{p,l}$ with small l_2-norm to zero, whereas the third term promotes the sparsity of the resulting chroma estimate.

4 Efficient implementations

The optimization problems in (27) and (37) are convex, and may thus be solved using one of the many freely available interior point methods, such as, e.g., SeDuMi [25] and SDPT3 [26]. However, these methods typically scale poorly with increasing data lengths or with increasing dictionary sizes. To allow for a more efficient implementation, we here propose an implementation based on the Alternating Direction Method of Multipliers (ADMM), splitting the optimization into two or more simpler optimizations, which are then solved iteratively. Depending on the complexity of these sub-problems, the ADMM in general reaches a good approximate solution very fast, while thereafter converging more slowly to a really accurate solution [27]. For sparse modeling, this becomes evident as the ADMM converges quickly to the correct set of non-zero variables, while convergence to the correct relative amplitudes requires some further iterations. For the constant amplitude case in (27), the generalized ADMM (for more than two functions) is used, reminiscent to the approach proposed in [28]; this case is detailed in the following.

4.1 Chroma estimation with constant amplitudes via ADMM

The ADMM considers convex optimization problem which can be expressed as the sum of two convex functions by separating the variable into two parts

$$\underset{\mathbf{z},\mathbf{u}}{\text{minimize}} \quad f(\mathbf{z}) + g(\mathbf{u}) \quad \text{subject to} \quad \mathbf{u} - \mathbf{G}\mathbf{z} = \mathbf{0} \tag{39}$$

whereafter the augmented Lagrangian, i.e.,

$$L_\rho(\mathbf{z}, \mathbf{u}, \mathbf{d}) = f(\mathbf{z}) + g(\mathbf{u}) + \frac{\rho}{2} ||\mathbf{G}\mathbf{z} - \mathbf{u} + \mathbf{d}||_2^2 \tag{40}$$

can be used to find a solution to the original problem by iteratively solving

$$\mathbf{z}(\ell + 1) = \arg\min_{\mathbf{z}} L_\rho(\mathbf{z}, \mathbf{u}(\ell), \mathbf{d}(\ell)) \tag{41}$$

$$\mathbf{u}(\ell + 1) = \arg\min_{\mathbf{u}} L_\rho(\mathbf{z}(\ell + 1), \mathbf{u}, \mathbf{d}(\ell)) \tag{42}$$

$$\mathbf{d}(\ell + 1) = \mathbf{G}\mathbf{z}(\ell + 1) - \mathbf{u}(\ell + 1) + \mathbf{d}(\ell) \tag{43}$$

To cast (27) in this framework we use the generalization idea proposed in [27] to extend the ADMM to problems with more than two convex function. This is

done by assuming that $f = 0$, and defining g as the sum of the functions in the original problem, i.e.,

$$\underset{\mathbf{u}}{\text{minimize}} \quad \sum_{i=1}^{3} g_i(\mathbf{H}_i\mathbf{u}) \tag{44}$$

with $\mathbf{H}_1 = \mathbf{W}$, $\mathbf{H}_2 = \mathbf{I}$, $\mathbf{H}_3 = \mathbf{F}$, and

$$g_1(\mathbf{W}\mathbf{u}) = ||\mathbf{y} - \mathbf{W}\mathbf{u}||_2^2 \tag{45}$$

$$g_2(\mathbf{u}) = \lambda_2||\mathbf{u}||_1 + \lambda_3 \sum_{c=0}^{11} ||\mathbf{u}_c||_2 \tag{46}$$

$$g_3(\mathbf{F}\mathbf{u}) = \lambda_4||\mathbf{F}\mathbf{u}||_1 \tag{47}$$

The augmented Lagrangian of (27) is

$$\begin{aligned} L(\mathbf{z}, \mathbf{u}, \mathbf{d}) = g_1(\mathbf{u}_1) + g_2(\mathbf{u}_2) + g_3(\mathbf{u}_3) + \frac{\mu}{2}||\mathbf{W}\mathbf{z} - \mathbf{u}_1 - \mathbf{d}_1||_2^2 \\ + \frac{\mu}{2}||\mathbf{z} - \mathbf{u}_2 - \mathbf{d}_2||_2^2 + \frac{\mu}{2}||\mathbf{F}\mathbf{z} - \mathbf{u}_3 - \mathbf{d}_3||_2^2 \end{aligned} \tag{48}$$

where

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1^T & \mathbf{u}_2^T & \mathbf{u}_3^T \end{bmatrix}^T \tag{49}$$

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_1^T & \mathbf{d}_2^T & \mathbf{d}_3^T \end{bmatrix}^T \tag{50}$$

denote the additional variables used to rewrite the optimization problem, and the dual variables, respectively. Thus, for the ℓ:th iteration,

$$\mathbf{z}(\ell + 1) = \arg\min_{\mathbf{z}} L(\mathbf{z}, \mathbf{u}(\ell), \mathbf{d}(\ell)) \tag{51}$$

which has the solution

$$\mathbf{z}(\ell + 1) = \left(\mathbf{G}^H\mathbf{G}\right)^{-1} \mathbf{G}^H \left(\mathbf{u}(\ell) + \mathbf{d}(\ell)\right) \tag{52}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{W}^T & \mathbf{I} & \mathbf{F}^T \end{bmatrix}^T \tag{53}$$

For $\mathbf{u}_1$,

$$\mathbf{u}_1(\ell+1) = \arg\min_{\mathbf{u}_1} L(\mathbf{z}(\ell+1), \mathbf{u}_1, \mathbf{d}_1(\ell)) \tag{54}$$

which may be solved as

$$\mathbf{u}_1(\ell+1) = \frac{\mathbf{y} + \mu\left(\mathbf{W}\mathbf{z}(\ell+1) - \mathbf{d}_1(\ell)\right)}{1+\mu} \tag{55}$$

For the remaining variables,

$$\mathbf{u}_2(\ell+1) = \arg\min_{\mathbf{u}_2} L\left(\mathbf{z}(\ell+1), \mathbf{u}_2, \mathbf{d}_2(\ell)\right) \tag{56}$$

$$\mathbf{u}_3(\ell+1) = \arg\min_{\mathbf{u}_3} L(\mathbf{z}(\ell+1), \mathbf{u}_3, \mathbf{d}_3(\ell)) \tag{57}$$

which have the solutions (see, e.g., [29])

$$\mathbf{u}_2(\ell+1) = \mathbf{T}\left(\mathbf{t}\left(\mathbf{z}(\ell+1) - \mathbf{d}_2(\ell), \frac{\lambda_2}{\mu}\right), \frac{\lambda_3\sqrt{(M)}}{\mu\sqrt{(12)}}\right) \tag{58}$$

$$\mathbf{u}_3(\ell+1) = \mathbf{t}\left(\mathbf{F}\mathbf{z}(\ell+1) - \mathbf{d}_3(\ell), \frac{\lambda_3\sqrt{(M)}}{\mu\sqrt{(12)}}\right) \tag{59}$$

where the shrinkage mappings $\mathbf{T}(\cdot)$ and $\mathbf{t}(\cdot)$ are defined as

$$\mathbf{t}(\mathbf{x}, \varkappa) = \frac{\mathbf{x}_k}{|\mathbf{x}_k|}\max\left(|\mathbf{x}_k| - \varkappa, 0\right), \text{ for all elements in } \mathbf{x} \tag{60}$$

$$\mathbf{T}(\mathbf{x}, \varkappa) = \frac{\mathbf{x}}{||\mathbf{x}||_2}\max\left(||\mathbf{x}||_2 - \varkappa, 0\right) \tag{61}$$

The augmented dual variable is updated as

$$\mathbf{d}(\ell+1) = \mathbf{d}(\ell) - (\mathbf{G}\mathbf{z}(\ell+1) - \mathbf{u}(\ell+1)) \tag{62}$$

The final chroma estimate is then found as setting $\hat{\mathbf{a}} = \mathbf{z}(\ell_{\text{final}})$. The resulting estimator is termed Chroma Estimation using Block Sparsity (CEBS). A summary of CEBS is shown in Algorithm 1.

Algorithm 1 The proposed CEBS algorithm

1: Initiate $\mathbf{z} = \mathbf{z}(0), \mathbf{u} = \mathbf{u}(0), \mathbf{d} = \mathbf{d}(0)$, and $\ell = 0$
2: **repeat**
3: $\quad \mathbf{z}(\ell+1)$ is updated as (52)
4: $\quad \mathbf{u}_1(\ell+1)$ is updated as (55)
5: $\quad \mathbf{u}_2(\ell+1)$ is updated as (58)
6: $\quad \mathbf{u}_3(\ell+1)$ is updated as (59)
7: $\quad \mathbf{d}(\ell+1)$ is updated as (62)
8: **until** convergence

4.2 Chroma estimation with amplitude modulation via ADMM

After the addition of amplitude modulation to the signal model, the problem is still convex, and we make use, once again, of the ADMM formulation, reminiscent to the approach proposed in [27]. The derivation becomes some what different to that in the previous section, since the amplitude modulated chroma model is more intricate. Denoting $\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 & \cdots & \mathbf{S}_P \end{bmatrix}$, (37) may be rewritten as

$$\underset{X,Z}{\text{minimize}} \quad f(\mathbf{X}) + g(\mathbf{Z}) \quad \text{subject to} \quad \mathbf{X} - \mathbf{Z} = 0 \tag{63}$$

where

$$\begin{aligned} f(\mathbf{X}) &= \frac{1}{2} \left|\left| \mathbf{y} - \sum_{p=1}^{P} \text{diag}(\mathbf{\Gamma} \mathbf{X}_p \mathbf{W}_p) \right|\right|_2^2 \\ g(\mathbf{Z}) &= \lambda \sum_{p=1}^{P} \sum_{l=1}^{L_{max}} \|\mathbf{z}_{p,l}\|_2 + \gamma \sum_{c=0}^{11} \|\mathbf{Z}_c\|_F \end{aligned} \tag{64}$$

with $\mathbf{X}$ and $\mathbf{Z}$ having the same structure as $\mathbf{S}$. It is worth noting that the ADMM separates the sought variable into two unknown variables, here denoted $\mathbf{X}$ and $\mathbf{Z}$, enabling the original problem to be decomposed into easier sub-problems. These are in turn solved iteratively until convergence. The augmented Lagrangian of (63) becomes

$$L_\rho(\mathbf{X}, \mathbf{Z}, \mathbf{D}) = f(\mathbf{X}) + g(\mathbf{Z}) + \frac{\rho}{2} ||\mathbf{X} - \mathbf{Z} + \mathbf{D}||_2^2 \tag{65}$$

where $\mathbf{D}$ represents the scaled dual variable (see also [27]), which allows (65) to be solved iteratively as

$$\mathbf{X}(\ell+1) = \arg\min_{\mathbf{X}} L_\rho(\mathbf{X}, \mathbf{Z}(\ell), \mathbf{D}(\ell)) \tag{66}$$

$$\mathbf{Z}(\ell+1) = \arg\min_{\mathbf{Z}} L_\rho(\mathbf{X}(\ell+1), \mathbf{Z}, \mathbf{D}(\ell)) \tag{67}$$

$$\mathbf{D}(\ell+1) = \mathbf{X}(\ell+1) - \mathbf{Z}(\ell+1) + \mathbf{D}(\ell) \tag{68}$$

at the ℓ:th iteration. To solve (66), one differentiates $f(\mathbf{X}) + \frac{\rho}{2}||\mathbf{X} - \mathbf{Z} + \mathbf{D}||_2^2$ with respect to $\mathbf{X}_p$ and sets the result equal to zero, which yields

$$-\sum_{n=1}^{N} y(n)\mathbf{\Gamma}(n,\cdot)^H \mathbf{W}_p(\cdot,n)^H + \frac{\rho}{2}(\mathbf{X}_p - \mathbf{Z}_p + \mathbf{D}_p)$$
$$+\sum_{u=1}^{P}\sum_{n=1}^{N} \mathbf{\Gamma}(n,\cdot)^H \mathbf{\Gamma}(n,\cdot)\mathbf{X}_u \mathbf{W}_u(\cdot,n)\mathbf{W}_p(\cdot,n)^H = 0$$

By stacking all columns in $\mathbf{X}$ on top of each other, this may be represented as

$$\sum_{n=1}^{N} \mathbf{a}(p,n)^H y(n) + \frac{\rho}{2}(\mathbf{z}_p - \mathbf{d}_p) = \sum_{n=1}^{N}\sum_{u=1}^{P} \mathbf{a}(p,n)^H \mathbf{a}(u,n)\mathbf{x}_u + \frac{\rho}{2}\mathbf{x}_p \tag{69}$$

where

$$\mathbf{a}(u,n) = \mathbf{W}_u(\cdot,n)^T \otimes \mathbf{\Gamma}(n,\cdot) \tag{70}$$

$$\mathbf{x}_u = \mathrm{vec}(\mathbf{X}_u) \tag{71}$$

$$\mathbf{z}_u = \mathrm{vec}(\mathbf{Z}_u) \tag{72}$$

$$\mathbf{d}_u = \mathrm{vec}(\mathbf{D}_u) \tag{73}$$

with $\otimes$ denoting the Kronecker product, and $\mathbf{W}_u(\cdot,n)$ and $\mathbf{\Gamma}(n,\cdot)$ denoting the n:th column in $\mathbf{W}_u$ and the n:th row $\mathbf{\Gamma}$, respectively. Let

$$\mathbf{A}(p,u) = \sum_{n=1}^{N} \mathbf{a}(p,n)^H \mathbf{a}(u,n) \tag{74}$$

$$\tilde{\mathbf{y}}(p) = \sum_{n=1}^{N} \mathbf{a}(p,n)^H y(n) \tag{75}$$

Algorithm 2 The proposed CEAMS algorithm

1: Initiate $\mathbf{X} = \mathbf{X}(0), \mathbf{Z} = \mathbf{Z}(0), \mathbf{D} = \mathbf{D}(0)$, and $\ell = 0$
2: **repeat**
3: $\quad \mathbf{X}(\ell+1) = (\mathbf{A}^H\mathbf{A} + \frac{\rho}{2}\mathbf{I})^{-1}\mathbf{A}^H\tilde{\mathbf{Y}}$
4: $\quad \mathbf{Z}(\ell+1) = \boldsymbol{\mathcal{T}}(\mathbf{T}(\mathbf{X}_p(\ell+1) + \mathbf{D}_p(\ell), \beta/\rho), \alpha/\rho), \forall p$
5: $\quad \mathbf{D}(\ell+1) = \mathbf{X}(\ell+1) - \mathbf{Z}(\ell+1) + \mathbf{D}(\ell)$
6: $\quad \ell \leftarrow \ell + 1$
7: **until** convergence

$$\tilde{\mathbf{Y}} = \begin{bmatrix} \tilde{\mathbf{y}}(1) & \cdots & \tilde{\mathbf{y}}(P) \end{bmatrix}^T \tag{76}$$

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}(1,1) & \cdots & \mathbf{A}(1,P) \\ \vdots & \ddots & \vdots \\ \mathbf{A}(P,1) & \cdots & \mathbf{A}(P,P) \end{pmatrix} \tag{77}$$

This yields the proposed algorithm, which is summarized in Algorithm 2, where $\mathbf{T}(\cdot)$ is defined as in (60), and $\boldsymbol{\mathcal{T}}(\cdot)$ is defined as

$$\boldsymbol{\mathcal{T}}(\mathbf{X}, \varkappa) = \frac{\mathbf{X}}{||\mathbf{X}||_F} \max\left(||\mathbf{X}||_F - \varkappa, 0\right) \tag{78}$$

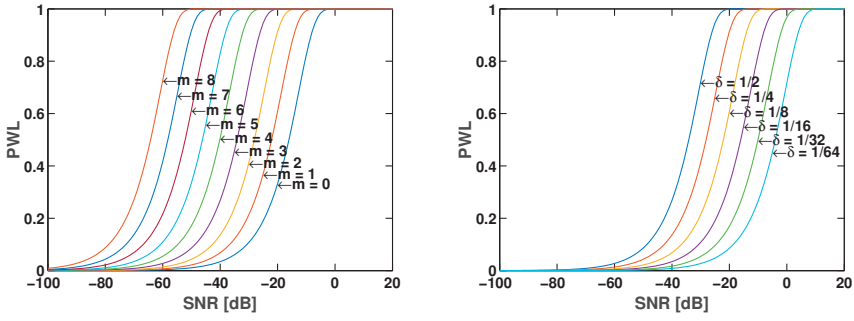
and is interpreted column wise, with $\boldsymbol{\mathcal{T}}(\cdot)$ operating over each part of $\mathbf{X}_p + \mathbf{D}_p$ that corresponds to $\tilde{\mathbf{S}}_{\tilde{c}}$. We term the resulting algorithm the Chroma Estimation of Amplitude Modulated Signals (CEAMS) method.

5 Numerical results

We proceed to examine the performance of the proposed estimators as a function of the Signal-to-Noise Ratio (SNR), measured in dB, defined as

$$\text{SNR} = 20\log_{10}\frac{\sigma_x}{\sigma_e} \tag{79}$$

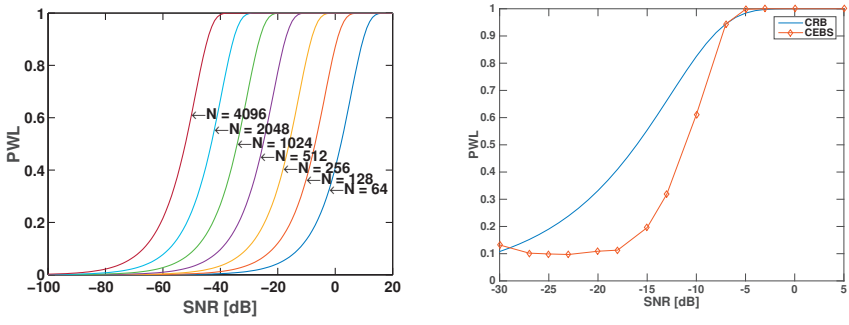
where σ_x and σ_e denote the power of the noise-free signal and the noise, respectively. As noted, the noise signal is here considered to consist of both the actual background noise and of any non-harmonic components in the recording. Therefore, in the case of strong formants, inharmonicity, or other musical features not



(a) (b)

Figure 1: Percentage of estimates within $c \pm 1/2$ from the true tone, when using twelve chromas, corresponding to the twelve semi-tones. Here, (a) is evaluated for the note C at different octaves, m, whereas (b) is evaluated for the note C3 when $c \in [0, 12)$ is discretized into $6/\delta$ points. For both, $N = 1024$ and $f_s = 20$ KHz (which equals a signal of approximately 51 ms).

modeled in this work, this signal might be quite strong. To examine the statistical limitations of chroma estimates, we initially examine the estimation limits, as obtained by the CRLB, which is derived in the appendix. As chroma is conventionally not considered a continuous variable, but rather as a number of grid points corresponding to some musicological system, we examine the achievable performance using the percentage-within-limits (PWL). This measures the number of estimates which are expected to fall within some pre-defined limit from the true value, i.e., $c \pm \delta$. For $\delta = 1/2$, this corresponds to the probability of obtaining estimates within the correct semi-tone, as $c = 0, \ldots, 11$. For $\delta = 1/4$, the PWL instead determines the likelihood of correctly estimating each quarter tone, and so forth. Figure 1(a) illustrates the performance of C notes at octaves $m = 0$ through $m = 8$, illustrating how the estimation problem becomes more difficult as the frequencies move closer to zero. The note is here formed from $N = 1024$ samples of a three-harmonic single pitch signal, measured at $f_s = 20$ KHz, which corresponds to a signal of approximately 51 ms. As can be seen from the figure, the PWL will reach 100% for the lowest note, i.e., being the most difficult estimation problem, at an SNR of approximately 0 dB. Figure 1(b) further illustrates the estimation limit for half tones up to the 64th tones, for a C3 tone, again reaching a perfect PWL at an SNR of approximately 0 dB, even for the



(a) (b)

Figure 2: Percentage of estimates within $c \pm 1/2$ from the true tone, when using twelve chromas. Here, (a) is evaluated for the note C3, for different data lengths, using $f_s = 20$ KHz, which implies a signal of N/f_s seconds, i.e., being (from the left) approximately 205 ms, 102 ms, 51 ms, 26 ms, 13 ms, 6 ms, and 3 ms. In (b), the estimated PWL for the CEBS estimator is compared to the CRLB for the C0 note, using $N = 1024$.

64th tone. Figure 2(a) similarly illustrates the estimation bounds as a function of the data length for the C3 note, using $\delta = 1/2$. All three figures thus indicate that one may expect a statistically efficient estimator to have no problems in correctly estimating the chromas, even in cases of SNR being significantly lower than expected for most audio recordings. However, due to the introduced penalties in the proposed estimators, one cannot expect these to be statistically efficient, even if the noise signal was a white sequence. This as the penalties will introduce an estimation bias, that although minor for most cases, will prevent the estimators to reach the CRLB. This is illustrated in Figure 2(b), showing the estimated PWL for the CEBS estimator, as obtained using 1000 Monte Carlo simulations, as compared to the corresponding CRLB. As may be seen in the figure, the actually achieved performance is, as expected, somewhat worse than predicted by the CRLB, although the latter gives a good indication of the achievable performance. Next, we proceed to examine the clarity of the proposed estimates, as compared to the (publicly available) estimators in [9, 10], using two audio signals from [30], namely a two channel FM-violin playing a middle C scale (all tones from C4 to C5), and a C-major chord, both in equal temperament, sampled at $f_s = 22$

KHz, mixed to a single channel using the method detailed in [10]. Figure 3 illustrate the resulting log-chromagrams for the Ellis, the Müller and Ewert, and the CEBS estimators. We have here divided the signal in segments of length $N = 1024$ samples (about 46 ms), having an overlap of 50%. For CEBS, we set $\lambda_2 = 0.05$, $\lambda_3 = 2.3$, and $\lambda_4 = 0.1$, for the chord, and $\lambda_2 = 0.05$, $\lambda_3 = 4$, and $\lambda_4 = 0.1$ for the scale, which are chosen using a simple heuristics from the FFT (see, e.g., [5]). The tuning frequency is here set to $f_{\text{base}} = 440$, and results remain quite unchanged at ± 3 Hz. As can be seen in the figures, the CEBS estimator yields a preferable estimate, suffering from noticeably less leakage and spurious estimates. Continuing, we examine the performance of the proposed estimators using a concert C-scale played by a trumpet acquired from [31], i.e., a highly non-stationary signal. Figure illustrates the resulting chromagrams, as obtained using the estimators in [10], [9], the CEBS estimator and the CEAMS estimator, respectively. For the CEAMS, we use $\lambda = 0.3$ and $\gamma = 193$, a window length of 1024 samples, a sampling frequency of 22050 Hz, $L_{max} = 9$ overtones, and 9 spline points. As is clear from the figure, both the estimators in [9, 10] suffer from apparent problems in choosing the correct chroma-bin for the scale. The CEBS estimate is notably cleaner, but still suffers from some spurious chroma features due to the inharmonicity of the signal. These spurious peaks have almost completely vanished in the CEAMS estimate. Here, we have used the same basic settings for CEBS as for CEAMS, and with $\lambda_2 = 0.05$, $\lambda_3 = 3$ and $\lambda_4 = 0.1$ (in setting these parameters, we have taken care to find the best possible setting for CEBS). It may be noted that the G in the scale is not detected by any method. This is because the fundamental frequency found in those time frames is 808 Hz, which is slightly closer to $G\#5$ than to $G5$, using concert tuning. To illustrate the difference in time-localization between CEBS and CEAMS, Figure show the 3-D chromagrams, where it once again can be noted that CEBS fails to identify the chroma-bin at $G\#$. Moreover, one may note the spurious peaks produced in CEBS, compared to the rest of the chromagram. This is in contrast to CEAMS, where none of the above mentioned behavior is present. Finally, we examine how well the proposed estimators capture the actual signal dynamics, by studying the envelopes of the reconstructed signals, formed from the respective estimates. Figure 6 illustrates how the amplitude modulation introduced in the CEAMS estimator has an advantage over the CEBS estimator. The CEAMS estimator captures both the shape and magnitude of the true signal envelope, whereas the CEBS estimator captures the shape reasonably, but fails to capture the amplitude.

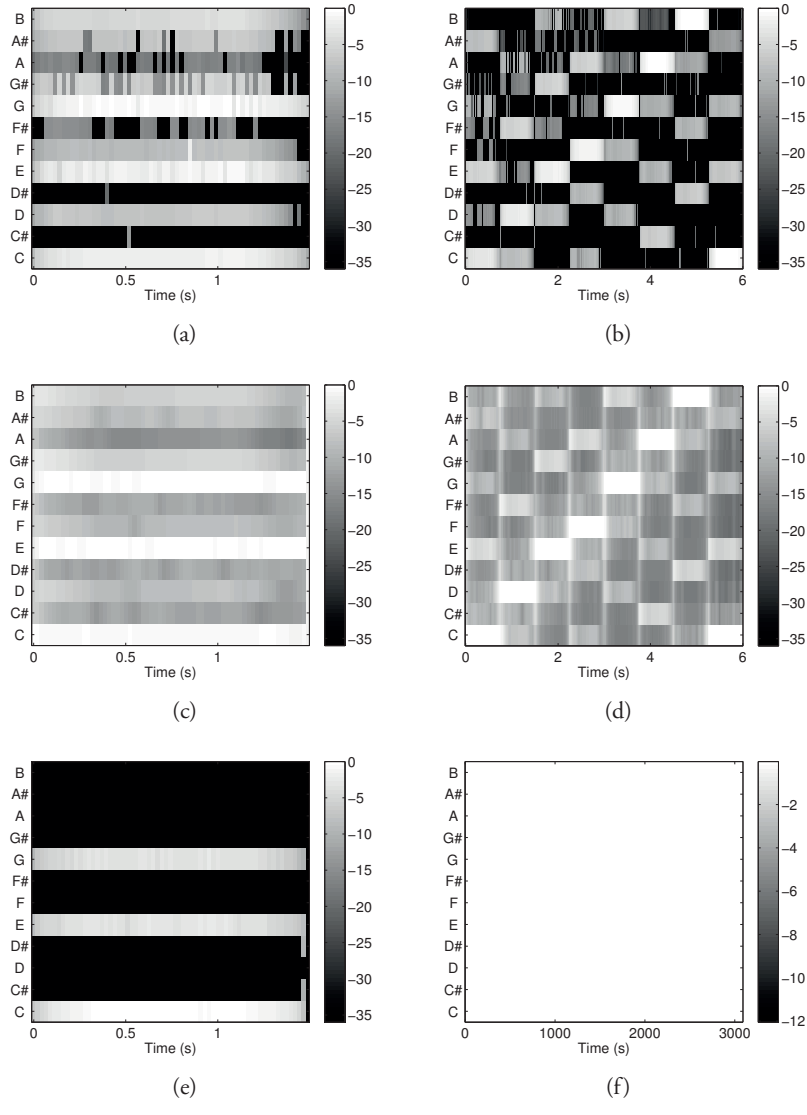


Figure 3: The performance for the (a,b) Ellis's method, (c,d) the Müller and Ewert method, and (e,f) the proposed CEBS algorithm, when evaluated on a C-chord (left), and a C-scale (right).

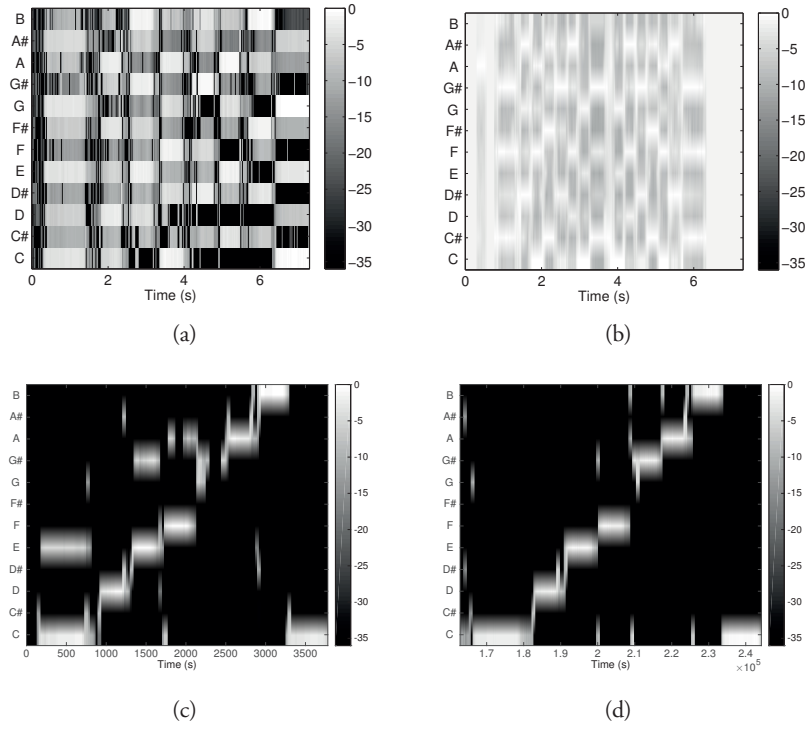


Figure 4: The figures above display the chromagrams for the trumpet scale, obtained using (a) Ellis's method, (b) the Müller and Ewert method, (c) CEBS, and (d) CEAMS.

6 Conclusions

In this article, we have presented two new methods for chroma estimation based on a sparse modeling reconstruction framework. The first method, CEBS, is designed to handle stationary time signals, and uses a fixed amplitude dictionary to model the measured signal. The method was further extended to also allow for time-varying signals, using a a spline-base model to capture the time-localization of the signal; the resulting estimator was termed the CEAMS method. The performance of the proposed estimators are compared both to the CRLB, presented herein for the problem at hand, as well as to two well-known chroma estimators using both real audio signals. It was found that the proposed estimators offer

a notable performance gain as compared to the comparable methods, with the CEAMS method being the better at capturing both the time-varying nature of the signal and the overall signal envelope.

7 Appendix: The Cramér-Rao lower bound

In this appendix, we present the Cramér-Rao Lower Bound (CRLB) for the chroma estimation problem. The signal in (15) may be equivalently be expressed as

$$x(t) = \sum_{k=1}^{K} \sum_{m=1}^{M_k} \sum_{l=1}^{L_k} a_{c_k,m,l} e^{j(2\pi f_b l t 2^m e^{(\ln(2)c_k/12)} + \varphi_{c_k,m,l})} \tag{80}$$

where M_k and L_k denote the highest octave and the highest harmonic for chroma class k, respectively. The the unknown parameters of the model are

$$\vartheta = [c_k, a_{c_k,1,1}, \varphi_{c_k,1,1} \cdots a_{c_k,m,l}, \varphi_{c_k,m,l}, c_{k+1}, a_{c_{k+1},1,1}, \varphi_{c_{k+1},1,1} \cdots] \tag{81}$$

The variance of the k:th parameter, ϑ_k, will thus be bound as

$$\mathrm{var}(\vartheta_k) \geq [\mathbf{B}(\vartheta)]_{k,k} \tag{82}$$

where $\mathbf{B}(\vartheta)$ denotes the CRLB matrix. Let

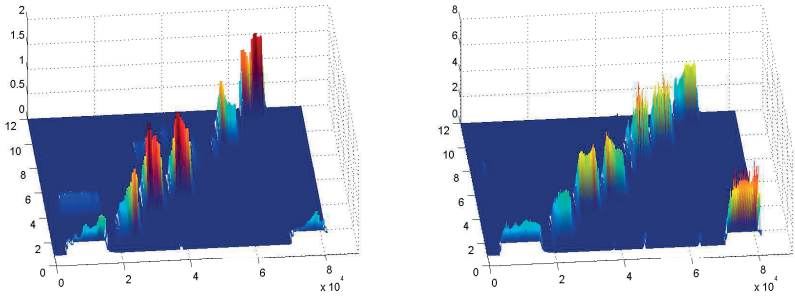
$$\hat{x}(\vartheta) = \begin{bmatrix} \hat{x}(0,\vartheta) & \cdots & \hat{x}(N-1,\vartheta) \end{bmatrix}^T \tag{83}$$

Assuming that the noise is independent of the parameters to be estimated, as well as having a Gaussian distribution with covariance matrix $\mathbf{Q}$, the Slepian-Bangs formula yields (see, e.g., [32, 33])

$$\mathbf{B}^{-1}(\vartheta) = 2\,\mathrm{Re}\left\{ \frac{\partial \hat{\mathbf{x}}^H(\vartheta)}{\partial \vartheta} \mathbf{Q}^{-1} \frac{\partial \hat{\mathbf{x}}(\vartheta)}{\partial \vartheta^T} \right\} \tag{84}$$

Introduce

$$\nu_{c_k,m,l} = 2\pi f_b l t 2^m \frac{\ln(2)}{12} e^{\ln(2)c_k/12} a_{c_k,m,l} \tag{85}$$



(a) (b)

Figure 5: The chromagrams with time localization for the (a) CEBS and (b) CEAMS methods.

and form the partial derivatives with respect to the parameters as

$$\frac{\partial x(t,\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} = \begin{bmatrix} \sum_{m=1}^{M_{\mathrm{k}}} \sum_{l=1}^{L_{\mathrm{k}}} j\nu_{c_k,m,l} e^{j(2\pi f_b l t 2^m e^{(\ln(2)c_k/12)} + \varphi_{c_k,m,l})} \\ e^{j(2\pi f_b l t 2^m e^{(\ln(2)c_k/12)} + \varphi_{c_k,m,l})} \\ j a_{c_k,m,l} e^{j(2\pi f_b l t 2^m e^{(\ln(2)c_k/12)} + \varphi_{c_k,m,l})} \\ \vdots \end{bmatrix} \tag{86}$$

Making the further assumption that the noise is white, i.e., $\mathbf{Q} = \sigma^2 \mathbf{I}$, the CRLB matrix may be written as

$$\mathbf{B}^{-1}(\boldsymbol{\vartheta}) = \frac{2}{\sigma^2} C \tag{87}$$

where $\mathbf{C}$ is defined as

$$\mathbf{C} = \mathrm{Re}\left\{ \frac{\partial \hat{\mathbf{x}}^H(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} \frac{\partial \hat{\mathbf{x}}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}^T} \right\} \tag{88}$$

Next, define

$$\chi_k = \left[\frac{\partial \hat{\mathbf{x}}(0,\boldsymbol{\vartheta})}{\partial c_k} \cdots \frac{\partial \hat{\mathbf{x}}(N-1,\boldsymbol{\vartheta})}{\partial c_k} \right]^T \tag{89}$$

$$\Psi_{c_k,m,l} = \begin{bmatrix} \frac{\partial \hat{\mathbf{x}}(0,\boldsymbol{\vartheta})}{\partial a_{c_k,m,l}} & \cdots & \frac{\partial \hat{\mathbf{x}}(N-1,\boldsymbol{\vartheta})}{\partial a_{c_k,m,l}} \\ \frac{\partial \hat{\mathbf{x}}(0,\boldsymbol{\vartheta})}{\partial \varphi_{c_k,m,l}} & \cdots & \frac{\partial \hat{\mathbf{x}}(0,\boldsymbol{\vartheta})}{\partial \varphi_{c_k,m,l}} \end{bmatrix} \tag{90}$$

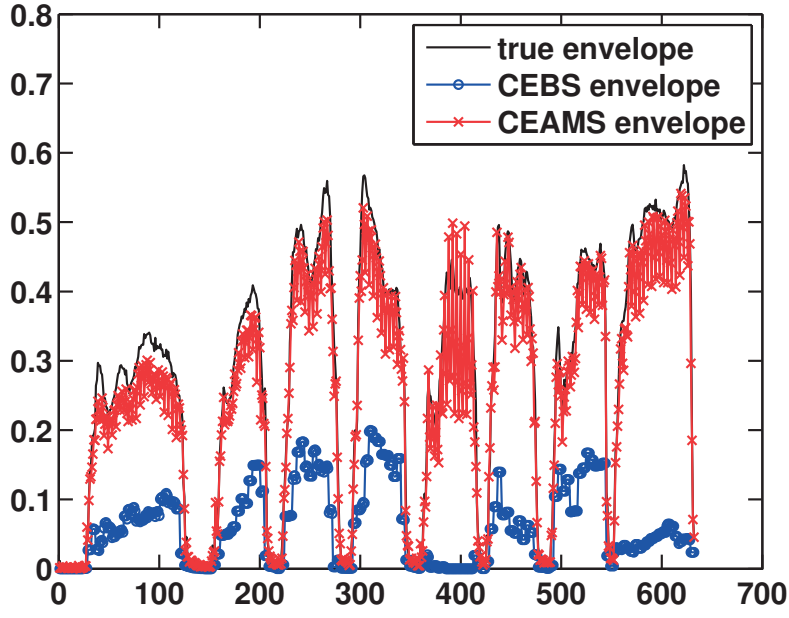


Figure 6: The figure above displays the time envelopes for the original signal (black) and the reconstructed signals.

Then, using $\sum_{p=1}^{P} = \sum_{m=1}^{M_k} \sum_{l=1}^{L_k}$,

$$\mathbf{C}_{1,1} = \begin{bmatrix} \chi_1^H \chi_1 & \chi_1^H \Psi_{1,1} & \chi_1^H \Psi_{1,2} & \cdots & \chi_1^H \Psi_{1,P} \\ \Psi_{1,1}^H \chi_1 & \Psi_{1,1}^H \Psi_{1,1} & \Psi_{1,1}^H \Psi_{1,2} & \cdots & \Psi_{1,1}^H \Psi_{1,P} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \Psi_{1,P}^H \chi_1 & \Psi_{1,P}^H \Psi_{1,1} & \Psi_{1,P}^H \Psi_{1,2} & \cdots & \Psi_{1,P}^H \Psi_{1,P} \end{bmatrix} \tag{91}$$

and, analogously,

$$\mathbf{C}_{2,1} = \begin{bmatrix} \chi_2^H \chi_1 & \chi_2^H \Psi_{1,1} & \chi_2^H \Psi_{1,2} & \cdots & \chi_2^H \Psi_{1,P} \\ \Psi_{2,1}^H \chi_1 & \Psi_{2,1}^H \Psi_{1,1} & \Psi_{2,1}^H \Psi_{1,2} & \cdots & \Psi_{2,1}^H \Psi_{1,P} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \Psi_{2,P}^H \chi_1 & \Psi_{2,P}^H \Psi_{2,1} & \Psi_{2,P}^H \Psi_{2,2} & \cdots & \Psi_{2,P}^H \Psi_{1,P} \end{bmatrix} \tag{92}$$

Thus,

$$\mathbf{C} = \mathrm{Re}\left\{ \begin{bmatrix} \mathbf{C}_{1,1} & \mathbf{C}_{1,2} & \mathbf{C}_{1,3} & \cdots & \mathbf{C}_{1,k} \\ \mathbf{C}_{2,1} & \mathbf{C}_{2,2} & \mathbf{C}_{2,3} & \cdots & \mathbf{C}_{2,k} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{C}_{k,1} & \mathbf{C}_{k,2} & \mathbf{C}_{k,3} & \cdots & \mathbf{C}_{k,k} \end{bmatrix} \right\} \tag{93}$$

with

$$\mathrm{Re}\{\chi_k^H \chi_k\} = \sum_{m=1}^{M_k} \sum_{l=1}^{L_k} \frac{a_{c_k,m,l}^2 (2\pi 2^m f_b l \frac{\ln(2)}{12} e^{\ln(2) c_k/12})^2}{6/\left(N(N+1)(2N+1)\right)}$$

$$\mathrm{Re}\{\Psi_{c_k,m,l}^H \Psi_{c_k,m,l}\} = \begin{bmatrix} N & 0 \\ 0 & N a_{c_k,m,l}^2 \end{bmatrix} \tag{94}$$

$$\mathrm{Re}\{\Psi_{c_k,m,l}, \chi_k\} = \begin{bmatrix} 0 \\ a_{c_k,m,l}^2 2\pi f_b l 2^m \frac{\ln(2)}{12} e^{\ln(2) c_k/12} \frac{N(N-1)}{2} \end{bmatrix} \tag{95}$$

$$\mathrm{Re}\{\Psi_{k,m,l}, \Psi_{k,m,r}\} = 0 \text{ for } \mathrm{l} \neq \mathrm{r} \tag{96}$$

If there is a spectral overlap between the chroma groups, and/or when the octaves considered have overlapping harmonics, the matrices $\mathbf{C}_{k,r}$, with $\mathrm{k} \neq \mathrm{r}$ will have non-zero entries. However, for the case considered herein, using 12 distinct chroma classes and only one tone, the following simplifications may be made:

$$\mathrm{Re}\{\chi_k \chi_r\} = 0 \text{ for } \mathrm{k} \neq \mathrm{r} \tag{97}$$

$$\mathrm{Re}\{\Psi_{k,p}, \Psi_{k,q}\} \approx 0 \tag{98}$$

$$\mathrm{Re}\{\Psi_k, \chi_r\} \approx 0, \tag{99}$$

implying that $\mathbf{C}$ will be a block-diagonal matrix, with all off diagonal blocks being zero, such that

$$\mathbf{C}^{-1} = \mathrm{Re}\left\{ \begin{bmatrix} \mathbf{C}_{1,1}^{-1} & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{C}_{2,2}^{-1} & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \mathbf{C}_{k,k}^{-1} \end{bmatrix} \right\} \tag{100}$$

Partitioning the matrix $\mathbf{C}_{k,k}$ as

$$\mathbf{C}_{k,k} = \begin{bmatrix} c & \mathbf{d}^H \\ \mathbf{d} & \mathbf{E} \end{bmatrix} \tag{101}$$

where c is a constant, $\mathbf{d}$ is a vector, and $\mathbf{E}$ is a diagonal matrix, one may use the matrix inversion lemma to form the inverse matrix $[\mathbf{C}_{k,k}^{-1}]_{1,1}$ as

$$[\mathbf{C}_{k,k}^{-1}]_{1,1} = (c - \mathbf{d}^H \mathbf{E}^{-1} \mathbf{d})^{-1} \tag{102}$$

yielding the bound

$$\mathrm{var}(c_k) \geq \frac{6\sigma^2}{\sum_{m=1}^{M_\mathrm{k}} \sum_{l=1}^{L_\mathrm{k}} (a_{c_k,m,l} 2\pi f_b l 2^m \frac{\ln(2)}{12} e^{\ln(2)c_k/12})^2 N(N-1)^2} \tag{103}$$

References

[1] M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard, "Signal Processing for Music Analysis," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 6, pp. 1088–1110, 2011.

[2] R. Shepard, "Circularity in Judgements of Relative Pitch," *Journal of Acoustical Society of America*, vol. 36, no. 12, pp. 2346–2353, Dec. 1964.

[3] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[4] A. Klapuri and M. Davy, *Signal Processing Methods for Music Transcription*, Springer, 2006.

[5] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[6] M. A. Bartsch and G. H. Wakefield, "Audio Thumbnailing of Popular Music Using Chroma-based Representations," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 96–104, Feb. 2005.

[7] S. Kim and S. Narayanan, "Dynamic Chroma Feature Vectors with Applications to Cover Song Identification.," in *10th IEEE Workshop on Multimedia Signal Processing*, 2008, pp. 984–987.

[8] T.-M. Chang, E.-T. Chen, C.-B. Hsieh, and P.-C. Chang, "Cover Song Identification with Direct Chroma Feature Extraction from AAC Files," in *IEEE 2nd Global Conference on Consumer Electronics*, Oct. 2013, pp. 55–56.

[9] D. P. W. Ellis, "Chroma Feature Analysis and Synthesis," `http://www.ee.columbia.edu/~dpwe/resources/matlab/chroma-ansyn/`, accessed Sept. 2014.

[10] M. Müller and S. Ewert, "Chroma Toolbox: MATLAB Implementations for Extracting Variants of Chroma-based Audio Features," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, 2011.

[11] J.S. Jacobson, *L1 Minimization for Sparse Audio Processing*, Ph.D. thesis, University of California, 2012.

[12] M. Mauch and S. Dixon, "Approximate Note Transcription for the Improved Identification of Difficult Chords," in *11th Int. Soc. Music Inf. Retrieval Conf.*, 2010, pp. 135–140.

[13] E. Gómez, *Tonal Description of Music Audio Signals*, Ph.D. thesis, Universitat Pompeu Fabra, 2006.

[14] M. Varewyck, J. Pauwels, and J.-P. Martens, "A Novel Chroma Representation of Polyphonic Music Based on Multiple Pitch Tracking Techniques," in *16th ACM International Conference on Multimedia*, New York, NY, USA, 2008, pp. 667–670, ACM.

[15] S. I. Adalbjörnsson, J. Swärd, T. Kronvall, and A. Jakobsson, "A Sparse Approach for Estimation of Amplitude Modulated Sinusoids," in *Proceedings of the* 48th *Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2-5 2014.

[16] A. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 804–816, 2003.

[17] M. G. Christensen, P. Stoica, A. Jakobsson, and S. H. Jensen, "Multi-pitch estimation," *Signal Processing*, vol. 88, no. 4, pp. 972–983, April 2008.

[18] S. L. Marple, "Computing the discrete-time "analytic" signal via FFT," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2600–2603, September 1999.

[19] ISO, "Acoustics - Standard Tuning Frequency (Standard Musical Pitch)," Standard ISO 16:1975, International Organization for Standardization, Geneva, CH, 1975, ISO/TC 43, stage 90.93 (2011-12-22), ICS: 17.140.01.

[20] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2 edition, 2009.

[21] P. Stoica, R. Moses, B. Friedlander, and T. Söderström, "Maximum Likelihood Estimation of the Parameters of Multiple Sinusoids from Noisy Measurements," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 3, pp. 378–392, March 1989.

[22] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[23] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[24] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, no. 1, pp. 91–108, January 2005.

[25] J. F. Sturm, "Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11-12, pp. 625–653, August 1999.

[26] R. H. Tutuncu, K. C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using SDPT3," *Mathematical Programming Ser. B*, vol. 95, pp. 189–217, 2003.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[28] M. A. T. Figueiredo and J. M. Bioucas-Dias, "Algorithms for imaging inverse problems under sparsity regularization," in *Proc. 3rd Int. Workshop on Cognitive Information Processing*, May 2012, pp. 1–6.

[29] R. Chartrand and B. Wohlberg, "A Nonconvex ADMM Algorithm for Group Sparsity with Sparse Groups," in *38th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 26-31 2013.

[30] M. Romain, "Sound Examples," https://ccrma.stanford.edu/~mromaine/220a/fp/sound- examples.html, accessed Sept. 2014.

[31] Mrs. Thomas, "Sound Examples," http://www.hffmcsd.org/webpages/ arushkoski/nyssma.cfm, accessed Feb. 2015.

[32] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*, Prentice-Hall, Englewood Cliffs, N.J., 1993.

[33] P. Stoica and R. Moses, *Spectral Analysis of Signals*, Prentice Hall, Upper Saddle River, N.J., 2005.

D

Paper D

Group-Sparse Regression Using the Covariance Fitting Criterion

Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson

Abstract

In this work, we present a novel formulation for efficient estimation of group-sparse regression problems. By relaxing a covariance fitting criteria commonly used in array signal processing, we derive a generalization of the recent SPICE method for grouped variables. Such a formulation circumvents cumbersome model order estimation, while being inherently hyperparameter-free. We derive an implementation which iteratively decomposes into a series of convex optimization problems, each being solvable in closed-form. Furthermore, we show the connection between the proposed estimator and the class of LASSO-type estimators, where a dictionary-dependent regularization level is inherently set by the covariance fitting criteria. We also show how the proposed estimator may be used to form group-sparse estimates for sparse groups, as well as validating its robustness against coherency in the dictionary, i.e., the case of overlapping dictionary groups. Numerical results show preferable estimation performance, on par with a group-LASSO bestowed with oracle regularization, and well exceeding comparable greedy estimation methods.

Keywords: covariance fitting, SPICE, group sparsity, group-LASSO, hyperparameter-free, convex optimization

1 Introduction

The last decades' development in compressive sensing, sparse estimation (which includes sparse modeling, sparse subset selection, and sparse regression), and related fields has resulted in a convenient toolbox of methods, allowing practitioners to relatively easily tackle a wide range of problems in areas such as, e.g., audio, video, and image analysis, spectroscopy, seismology, and genome sequencing (see, e.g., [1–3], for an overview). Commonly, such problems contains data sets which can be either transformed into, or be well approximated by, overdetermined linear systems, where only a small subset of the explanatory variables are necessary to represent the data. Commonly, the candidate regressors are denoted atoms, and the collection of all atoms is referred to as the dictionary, which is typically designed specifically for the application. The main idea of sparse estimation is to infer means of restricting, or regularizing, the parameter space to have few active (or non-zero) elements, e.g., by a shrinkage operator, as was used for wavelets in the early work [4]. In statistical modeling, the sparse estimation problem is referred to as sparse regression, for which the seminal least absolute selection and shrinkage operator (LASSO) was proposed in [5]. The LASSO solves a minimization problem which contains two positive terms; a fitting term which goes to zero when the model fits the data, which is offset by a penalty term which grows when the explanatory variables grow. In signal processing, the problem is referred to as basis pursuit [6], or basis pursuit de-noising (BPDN) [7] in the noisy case. In fact, the LASSO and BPDN have equivalent problem formulations, and in the remainder of this paper, we will, out of convenience, simply refer to the methodology as the LASSO. Often, sparse estimation is pursued in a greedy manner, by including non-zero components into the solution one-by-one; referred to as stepwise regression in statistical modeling, and matching pursuit in signal processing, for which in the latter an orthonormalization step is often included [8]. Early on, there were also alternatives to the LASSO for sparse estimation, such as the estimator proposed in [9], which formulates a penalized (or regularized) likelihood problem.

One reason for the wide-spread praise of the LASSO is its ability to produce robust and accurate estimates, supported by several theoretical results for recovery guarantees, described by, e.g., mutual coherence [10], the restricted isometry property [11], or via the so-called spark of the dictionary [1]. These results, however, directly or indirectly, assume low correlation in the parameter space, for which the elastic net was proposed in [12] to better avoid mismatch in correlated

dictionary designs. Another reason of the success for the LASSO is that it is a convex ℓ_1-relaxation of a ℓ_0-regularized estimation problem, for which user-friendly scientific software exists, e.g. [13].

In spectral analysis, as well as in array processing, two important purposes of sparse estimation are to linearize the non-linear problem formulation related to parametric estimation of frequencies or locations, and to circumvent the difficult model order estimation problem. This is achieved by discretizing the parameter space into a large grid of possible frequencies or locations, from which sparse modeling selects the best (sparse) subset of atoms to parametrize the data [14].

In array processing, a common estimation approach is to perform matching between the estimate of a covariance matrix, and the covariance matrix parametrized by a certain model. Overviewed in [15], the covariance matching estimation technique (COMET) may, for instance, be used to find the direction-of-arrival (DOA) for a signal impinging on a sensor array. In a recent effort, the sparse iterative covariance-based estimation (SPICE) method proposes to model the covariance matrix using a highly underdetermined system of candidate parameters, and shows how the covariance matching formulation thereby promotes sparse estimates, both for line spectra [16], and for DOA estimation [17]. Other methods for sparse estimation in array processing include [18] and [19] for DOA estimation, and [20] for MIMO radar imaging. Related to spectral analysis, sparse estimation is applied to music analysis in [21] and [22]. Sparse estimation using SPICE has been extensively studied in the works of [23], [24], and [25], showing how SPICE is equivalent to the least absolute deviation (LAD) LASSO [26] under the assumption of the signal being corrupted by heteroscedastic noise, and to the square-root (SR) LASSO [27], under the assumption of homoscedastic noise. The difference between the standard LASSO and these variants lies in the fitting term; for LASSO this is the ℓ_2-norm squared, whereas for the LAD-LASSO and the SR-LASSO it is the ℓ_1- and ℓ_2-norms, respectively. The SR-LASSO is essentially equivalent to the LASSO, whereas the LAD-LASSO offers more robustness against data outliers.

In this paper, we are mainly interested in the group-sparse estimation problem, i.e., when the dictionary atoms each hold a group of regressors rather than just one, and the aim is to find a small subset of such atoms to model the data. The LASSO may still recover the true support for such models, but it cannot recognize which component belongs to which group. Hence, for correlated dictionary designs, the LASSO will typically overfit the data by introducing spuri-

ous non-zero variable estimates for regressors having non-zero linear dependence with some of the true regressors. To form clustered estimates, the probing absolute least squares modeling [28] was developed for data mining applications, and the group-LASSO [29] to improve performance in analysis-of-variance (ANOVA) problems with multi-factor variables, with subsequent theoretical results being presented in, e.g., [30, 31]. In the group-LASSO, the fitting term is regularized by an ℓ_1/ℓ_2-term in lieu of the ℓ_1-norm penalty, where the ℓ_2-norm is used within a group and the ℓ_1-norm between groups. In [32], different approaches of modifying the group-LASSO penalty was examined, and in [33] and [34], the group-LASSO was applied to logistic regression and multivariate regression, respectively. In some cases, it may be reasonable to assume that not all components within a group are present in the data. This case was examined in [35], wherein a group-LASSO for sparse groups was proposed, extending the ℓ_1/ℓ_2 penalty by an additional ℓ_1 penalty. Another penalty for sparse estimation was introduced in [36], where a total-variation penalty, commonly used in image analysis, was used to group estimates by fusing together adjoining variables of similar size.

Thus, depending on the sparsity structure sought for the specific application, one may model one's own combination of penalties to promote sparse estimates with such structure. Recently, this idea was applied to multi-pitch estimation [37,38], a problem formulation commonly used in audio analysis where the signal consists of a small number of groups of spectral lines, which for each group are located at integer multiples of some fundamental frequency. Similarly, sparse estimation was also used for joint multi-pitch estimation and source location [39], as well as for estimation of chroma features in music processing [40].

Although there exists theoretical recovery guarantees for group-sparse estimation, these often assume the dictionary is sufficiently incoherent, i.e., that there is low co-linearity between the dictionary atoms. In several applications, for instance the group-sparse multi-pitch estimation problem, the dictionary design in inherently highly correlated. For these problems, the real benefit of group-sparse estimation is that the estimator selects among candidates which all partly fit the data, but where one does so while also fitting the sought sparsity structure. To further improve sparse estimation for correlated dictionary designs, some methods have been proposed; e.g., the overlap and graph group-LASSO [41], and the trace LASSO [42].

So far, we have not mentioned the inherent caveat in the sparse estimation framework. i.e., choosing the regularization level. Certainly, by imposing sparsity

on some over-complete data parametrization, the amount of sparsity inferred on the solution needs to be selected, and for most methods, there is one or more hyperparameters that need to be set. To that end, a homotopy method was presented in [43], and later the least angle regression (LARS) algorithm in [44], which compute a solution path, i.e., all solutions over an interval of the hyperparameter. Notably, LARS operates at the same cost as typical solvers for the LASSO, although requiring a relatively low degree of dictionary coherence, especially for grouped variables. By utilizing warm-starts, the solution path may also be computed relatively fast by the other methods mentioned herein. Still, an appropriate level of regularization needs to be selected, i.e., a point on the solution path. This may be done in different ways, e.g., using some heuristics, or using cross-validation (as was done in [45] for the multi-pitch estimation problem), or using some information or model order criteria (see, e.g., [46, 47]), which may be difficult- and or time-consuming depending on the problem. By contrast, SPICE is promoted as a hyperparameter-free sparse estimation method [48]. However, given that SPICE may be formulated as a particular LASSO problem, this hyperparameter is rather pre-selected than nonexistent. The published works on SPICE, cited herein, furthermore indicate that the method works very well for a wide array of problems, not being limited to array processing, for which the covariance fitting criteria was originally intended.

In this work, we propose a generalization of the SPICE formulation for promoting group-sparse estimates, by a relaxation of the covariance fitting criteria. Still being convex, we introduce an efficient implementation of the proposed group-SPICE which is inspired by, like SPICE, an approach from optimal experimental design [49]. Thus, an auxiliary variable is introduced, and we formulate an estimator solving a sequence of simple optimization problems, computable in closed form via the Karush-Kuhn-Tucker conditions [50]. Similar to the connection between SPICE and the LASSO, we establish the connection between group-SPICE and the group variants of the LAD-LASSO and the SR-LASSO [51], illustrating how the covariance fitting criteria implicitly will set the hyperparameter(s) for these estimators. We also show that group-SPICE yields group-LASSO formulations where an ℓ_1/ℓ_q penalty, for $1 \leq q \leq 2$, can be used to improve performance when sparsity exists within groups. Furthermore, we illustrate the performance of group-SPICE for Gaussian dictionaries, as well as for multi-pitch dictionaries used for audio recordings, illustrating how it optimally regularizes the group-sparse estimation problem, while clearly outperforming the corresponding

greedy estimators, especially for highly coherent regressor matrices.

2 Promoting group sparsity by covariance fitting

Consider a length N complex-valued measurement, constituting a mix of C sources, each parametrized by a group of L_c components, such that (see also, e.g., [39])

$$\mathbf{y} = \sum_{c=1}^{C} \mathbf{s}_c + \mathbf{e}' \tag{1}$$

where $\mathbf{e}' \in \mathbb{C}^N$ denotes an additive noise component, and

$$\mathbf{s}_c = \sum_{\ell=1}^{L_c} \mathbf{a}(\vartheta_c, \ell) x_{\vartheta_c,\ell} \tag{2}$$

with $\mathbf{s}_c \in \mathbb{C}^N$ is the parametrization of the c:th source, with $\mathbf{a}(\vartheta_c, n) \in \mathbb{C}^N$ denoting the regressor vector, and $x_{\vartheta_c,n}$ the corresponding complex-valued amplitude (or regressand). Thus, the c:th source is fully parametrized by the (unknown) parameters

$$\{\vartheta_c, x_{\vartheta_c,1}, \ldots, x_{\vartheta_c,L_c}\}_{c=1,\ldots C} \tag{3}$$

which are subject to estimation. However, typically the number of sources, C, is unknown, and in some applications also the group sizes, L_c, necessitating an estimate of the different model orders before these parameters can be determined. Using a sparse reconstruction framework, the proposed method selects the appropriate model orders as a part of the estimation procedure, avoiding the need of explicit (and difficult) model order selection. To do so, we proceed to formulate a sparse regression model, introducing a predefined dictionary of possible candidates over the parameter space ϑ, i.e., consisting of potential candidates ϑ_k, for $k = 1, \ldots, K$, with $K \gg C$ selected large enough to ensure that some of the K candidates well coincides with the true parameters (see also, e.g., [52]). The size of each group, L_k, may be either known or unknown. If known, then those L_k in the true support will be equal to the corresponding L_c, while if unknown, if, e.g., only a subset of the group's regressors are present in the data, an upper bound, L,

is selected such that $L \geq \max_c L_c$. To simplify the notation, we hereafter simply use $(\cdot)_k$ in place of $(\cdot)_{\vartheta_k}$, allowing (1) to be expressed compactly as

$$\mathbf{y} = \sum_{k=1}^{K} \mathbf{A}_k \mathbf{x}_k + \mathbf{e} = \tilde{\mathbf{A}}\mathbf{x} + \mathbf{e} \tag{4}$$

where $\mathbf{e}$ is a noise component analogous to $\mathbf{e}'$, and

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_1 & \dots & \mathbf{A}_K \end{bmatrix} \in \mathbb{C}^{N \times M} \tag{5}$$

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{a}_{k,1} & \dots & \mathbf{a}_{k,L_k} \end{bmatrix} \in \mathbb{C}^{N \times L_k} \tag{6}$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1^\top & \dots & \mathbf{x}_K^\top \end{bmatrix}^\top \in \mathbb{C}^M \tag{7}$$

$$\mathbf{x}_k = \begin{bmatrix} x_{k,1} & \dots & x_{k,L_k} \end{bmatrix}^\top \tag{8}$$

with $M = \sum_{k=1}^{K} L_k$ denoting the number of columns in the dictionary, and $(\cdot)^\top$ the matrix transpose. Furthermore, define the covariance matrix of the noise component as

$$\mathbf{\Sigma} = E\{\mathbf{e}\mathbf{e}^H\} = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \sigma_N \end{bmatrix} \tag{9}$$

where $E\{\cdot\}$ denotes the expectation and $(\cdot)^H$ the conjugate transpose. Furthermore, we adopt the common assumption that the phases of $x_{k,l}$ are independent and uniformly distributed on $[0,\, 2\pi]$, see, e.g., [53, p. 176], implying that the covariance matrix of the measurement vector, $\mathbf{R} = E(\mathbf{y}\mathbf{y}^H)$, may be modeled as

$$\mathbf{R} = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} |x_{k,\ell}|^2 \mathbf{a}_{k,\ell} \mathbf{a}_{k,\ell}^H + \mathbf{\Sigma} \triangleq \mathbf{A}\mathbf{P}\mathbf{A}^H \in \mathbb{R}^{N \times N} \tag{10}$$

where the dictionary has been augmented such that $\mathbf{A} = \begin{bmatrix} \tilde{\mathbf{A}} & \mathbf{I} \end{bmatrix} \in \mathbb{C}^{N \times (M+N)}$, with $\mathbf{I}$ denoting the $n \times n$ identity matrix, and where similarly

$$\mathbf{P} = \begin{bmatrix} \operatorname{diag}\left(\begin{bmatrix} \mathbf{p}_1^\top & \cdots & \mathbf{p}_K^\top \end{bmatrix}^\top \right) & \mathbf{0}^\top \\ \mathbf{0} & \mathbf{\Sigma} \end{bmatrix} \in \mathbb{R}^{(M+N) \times (M+N)} \tag{11}$$

$$\mathbf{p}_k = \begin{bmatrix} |x_{k,1}|^2 & \cdots & |x_{k,L_k}|^2 \end{bmatrix}^\top \triangleq \begin{bmatrix} p_{k,1} & \cdots & p_{k,L_k} \end{bmatrix}^\top \tag{12}$$

with diag($\mathbf{c}$) denoting a diagonal matrix with vector $\mathbf{c}$ along its diagonal and where $\mathbf{0}$ is an $N \times K$ zero matrix. The matrix $\mathbf{P}$ is thus diagonal, and, for notational convenience, we let $\mathbf{P} \triangleq \text{diag}(\mathbf{p})$, where $\mathbf{p} \in \mathbb{C}^{(M+N)}$ denotes the (unknown) grouped vector of diagonal loadings in the covariance model, i.e.,

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_1^\top & \cdots & \mathbf{p}_K^\top & p_{K+1} & \cdots & p_{K+N} \end{bmatrix}^\top \tag{13}$$

with $p_{K+n} = \sigma_n$, for $n = 1, \ldots, N$, implying that the noise powers represent independent groups of size one. The covariance structure is thereby completely described by the diagonal loadings, $\mathbf{p}$, and the dictionary matrix, $\mathbf{A}$.

In order to form an estimator which yields a group sparse solution using the weighted covariance fitting criterion, we here generalize upon the works presented in [17], where $\mathbf{p}$ is estimated in lieu of $\mathbf{x}$. To that end, consider minimizing a function describing the mismatch between the theoretical and sample covariance matrices, i.e.,

$$f = ||\mathbf{R}^{-1/2}(\hat{\mathbf{R}} - \mathbf{R})||_{\mathcal{F}}^2 \tag{14}$$

$$= ||\mathbf{y}||^2 \text{tr}(\mathbf{R}^{-1}\hat{\mathbf{R}}) + \text{tr}(\mathbf{R}) + D \tag{15}$$

$$= \underbrace{||\mathbf{y}||^2 \mathbf{y}^H \left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)^{-1} \mathbf{y}}_{\triangleq f_1} + \underbrace{\text{tr}\left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)}_{\triangleq f_2} + D \tag{16}$$

with $||\cdot||_{\mathcal{F}}$ denoting the Frobenius norm, tr($\cdot$) the trace, $\hat{\mathbf{R}} = \mathbf{y}\mathbf{y}^H$, D is a constant, and where (15) is formed by completing the square. Analyzing (16), it holds two terms, f_1 and f_2, that balance each other. The first describes how well the observations follows the combined signal and noise model, with $f_1 \to 0_+$ for some $p_{k,\ell} \to \infty$. However, the second term adds a cost to that variable being non-zero, with $f_2 \to \infty$ as $p_{k,\ell} \to \infty$. In this work, we propose that f_2 is modified such that the cost of having a solution clustered in few source groups becomes smaller as compared to having it spread out across many groups. Applying Hölder's inequality to each group yields

$$f_2 = \sum_{k=1}^{K+N} \sum_{\ell=1}^{L_k} p_{k,\ell} \overbrace{||\mathbf{a}_{k,\ell}||_2^2}^{\triangleq w_{k,\ell}} \tag{17}$$

$$= \sum_{k=1}^{K+N} \mathbf{p}_k^\top \overbrace{\begin{bmatrix} w_{k,1} & \cdots & w_{k,L_k} \end{bmatrix}^\top}^{\triangleq \mathbf{w}_k} \tag{18}$$

$$= \sum_{k=1}^{K+N} \langle \mathbf{p}_k, \mathbf{w}_k \rangle \leq \sum_{k=1}^{K+N} \|\mathbf{p}_k\|_r \, ||\mathbf{w}_k||_s \triangleq g_2 \tag{19}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, and where $r, s \in [1, \infty]$, with $r^{-1} + s^{-1} = 1$. Minimizing $f_1 + g_2$ in lieu of f should thus shift the optimum point such that a group sparse solution is preferable to one that is not. Consider $\mathbf{p}^*$ to be the optimal point for f, such that $f(\mathbf{p}^*) \leq f(\mathbf{p}), \forall \mathbf{p}$. Let g be the relaxed covariance fitting criteria, $g = f_1 + g_2$, i.e.,

$$g(\mathbf{p}) = ||\mathbf{y}||^2 \cdot \mathbf{y}^H \left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)^{-1} \mathbf{y} + \sum_{k=1}^{K+N} \|\mathbf{p}_k\|_r \, ||\mathbf{w}_k||_s \tag{20}$$

and let $\bar{\mathbf{p}}^*$ be its optimal point. We then have that

$$f(\mathbf{p}^*) \leq f(\bar{\mathbf{p}}^*) \leq g(\bar{\mathbf{p}}^*) \leq g(\mathbf{p}^*) \tag{21}$$

to which one may conclude that the optimum of the proposed criteria g will be a relaxation of the optimal value of f, while still being bounded above by the optimal point of f, i.e., $\mathbf{p}^*$, evaluated in g. In cases when the optimal point for g is the same as that for f, the bound is tight. Instead of minimizing (16), the sought solution is found by solving the relaxed covariance fitting problem

$$\begin{aligned} \underset{\mathbf{p}}{\text{minimize}} \quad & g(\mathbf{p}) = \mathbf{y}^H \mathbf{R}^{-1} \mathbf{y} + \sum_{k=1}^{K+N} v_k \|\mathbf{p}_k\|_r \\ \text{subject to} \quad & \mathbf{R} = \mathbf{A}\mathbf{P}\mathbf{A}^H \quad p_{k,\ell} \geq 0, \ \forall (p, \ell) \end{aligned} \tag{22}$$

where $v_k \triangleq ||\mathbf{w}_k||_s$. Here, the factor $||\mathbf{y}||^2$ has been dropped from the minimization, as it can be incorporated into the v_k:s. Also, as is shown in the following, the minimization is actually invariant to such scaling. In comparison with the SPICE method which minimizes f, in this paper, for $r > 1$, the minimization in (22) further increases the cost of activating a component in a new candidate group, k', as compared to activating a component within an already active group, k, given that these components model the same data characteristics. Thus, (22) will promote a group sparse solution, as is also further discussed in the following. In the next section, we proceed to derive an algorithm for solving (22).

3 A group-sparse iterative covariance-based estimator

The optimization problem in (22) is convex, as it is being formed from an appropriate combination of convex functions [50, p. 84]. The first term is a positive weighted sum of the inverse of the terms in $\mathbf{p}$, which is convex for $p_{k,\ell} > 0, \forall(k,\ell)$. The second term is a positive weighted sum of norms, which is convex for any r-norm. Being convex, the minimization in (22) enjoys favorable properties such that any local optima is also the global optimum, that there exists a well defined theory stating necessary and sufficient conditions for optimality, as well as the opportunity for efficient computational methods. In the following, we will derive one such estimator. As the first part of the cost function in (22) is non-separable in the estimation parameters, $p_{k,\ell}$, we here propose, reminiscent to [17], to split the optimization problem into two simpler convex subproblems, each of which has a closed-form solution, and then to solve these iteratively. This is done by introducing an auxiliary variable, making the original variable separable in the parameters. For clarity of presentation, this step is shown via an intermediate auxiliary variable. In this first step, let this intermediate variable, $\mathbf{Q} \in \mathbb{C}^{N\times(M+N)}$, be a matrix which fulfills

$$\mathbf{Q}^H\mathbf{P}^{-1}\mathbf{Q} = \left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)^{-1} \quad \Longleftrightarrow \quad \mathbf{A}\mathbf{Q} = \mathbf{I} \tag{23}$$

allowing for the formulation of the equivalent optimization problem

$$\begin{aligned} \underset{\mathbf{p},\mathbf{Q}}{\text{minimize}} \quad & g(\mathbf{p},\mathbf{Q}) = \mathbf{y}^H\mathbf{Q}^H\mathbf{P}^{-1}\mathbf{Q}\mathbf{y} + \sum_{k=1}^{K+N} v_k \left\|\mathbf{p}_k\right\|_r \\ \text{subject to} \quad & \mathbf{A}\mathbf{Q} = \mathbf{I} \quad p_{k,\ell} \geq 0, \ \forall(p,\ell) \end{aligned} \tag{24}$$

over $\mathbf{p}$ and $\mathbf{Q}$. Next, let the main auxiliary variable be defined as $\boldsymbol{\beta} \triangleq \mathbf{Q}\mathbf{y}$, $\boldsymbol{\beta} \in \mathbb{C}^{(M+N)}$, such that

$$\mathbf{A}\mathbf{Q} = \mathbf{I} \quad \Longrightarrow \quad \mathbf{A}\boldsymbol{\beta} = \mathbf{y} \tag{25}$$

This change of variables yields the equivalent optimization problem

$$\begin{aligned} \underset{\mathbf{p},\boldsymbol{\beta}}{\text{minimize}} \quad & g(\mathbf{p},\boldsymbol{\beta}) = \boldsymbol{\beta}^H\mathbf{P}^{-1}\boldsymbol{\beta} + \sum_{k=1}^{K+N} v_k \left\|\mathbf{p}_k\right\|_r \\ \text{subject to} \quad & \mathbf{A}\boldsymbol{\beta} = \mathbf{y}, \quad p_{k,\ell} \geq 0, \ \forall(p,\ell) \end{aligned} \tag{26}$$

which may be identified as a (convex) quadratic-over-linear program [50, p. 76], and is the central optimization problem for this paper. To see that (22) and (26) are equivalent, one may fix $\mathbf{p}$ and solve (26) for $\boldsymbol{\beta}$. As this is a constrained optimization problem, the Lagrangian becomes

$$\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\mu}) = \boldsymbol{\beta}^H \mathbf{P}^{-1} \boldsymbol{\beta} + \sum_{k=1}^{K+N} v_k \left\| \mathbf{p}_k \right\|_r + \boldsymbol{\mu}^H (\mathbf{A}\boldsymbol{\beta} - \mathbf{y}) \tag{27}$$

where $\boldsymbol{\mu} \in \mathbb{C}^N$ is the Lagrange dual variable. Next, the saddle point of the Lagrangian is obtained by minimizing over $\boldsymbol{\beta}$ and maximizing over $\boldsymbol{\mu}$. Using Wirtinger calculus for complex-valued variables [54], one may form the derivative of the Lagrangian with respect to $\boldsymbol{\beta}$ and set it equal to zero, obtaining

$$\mathbf{P}^{-1}\boldsymbol{\beta} + \mathbf{A}^H \boldsymbol{\mu} = \mathbf{0} \quad \Longrightarrow \quad \hat{\boldsymbol{\beta}} = -\mathbf{P}\mathbf{A}^H \boldsymbol{\mu} \tag{28}$$

which inserted into (27) yields the dual problem, and its solution

$$\underset{\boldsymbol{\mu}}{\text{maximize}} \; -\boldsymbol{\mu}^\top \mathbf{A}\mathbf{P}\mathbf{A}^H \boldsymbol{\mu} - \boldsymbol{\mu}^\top \mathbf{y} \quad \Longrightarrow \quad \hat{\boldsymbol{\mu}} = -\left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)^{-1} \mathbf{y} \tag{29}$$

which inserted into (28) then yields the solution to (26) as

$$\hat{\boldsymbol{\beta}} = \mathbf{P}\mathbf{A}^H \left(\mathbf{A}\mathbf{P}\mathbf{A}^H\right)^{-1} \mathbf{y} \tag{30}$$

By inserting (30) into (26), one obtains the minimization problem in (22), which is thus equivalent to (26). Here, we propose to solve (26) using a block coordinate descent approach, where we iteratively alternate between solving for $\mathbf{p}$, with $\boldsymbol{\beta}$ fixed at its most recent value, and then solving for $\boldsymbol{\beta}$, with $\mathbf{p}$ fixed at its most recent value. The latter problem has the closed-form solution given by (30), as shown in (27)-(30). We proceed to examine the former problem for two separate cases.

3.1 The general case of heteroscedastic noise

We initially consider the case of heteroscedastic noise, i.e., when the noise variances are formed as in (9), allowing different samples to have different noise variances. In the next subsection, we will then proceed to examine the equivariance

case. Using $\boldsymbol{\beta}$, g may be expressed as a fully separable function in the $K+N$ groups, such that

$$g(\mathbf{p}) = \sum_{k=1}^{K+N} \left(\sum_{\ell=1}^{L_k} \frac{|\beta_{k,\ell}|^2}{p_{k,\ell}} + v_k \|\mathbf{p}_k\|_r \right) \triangleq \sum_{k=1}^{K+N} g_k(\mathbf{p}_k) \tag{31}$$

where $\beta_{k,\ell}$ is the ℓ:th component in the k:th candidate group. Being separable, one may therefore optimize each function g_k over the variables $p_{k,1}, \ldots, p_{k,L_k}$ independently of the other groups. The Lagrangian for the k:th subproblem becomes

$$\mathcal{L}(\mathbf{p}_k, \boldsymbol{\mu}_k) = \sum_{\ell=1}^{L_k} \frac{|\beta_{k,\ell}|^2}{p_{k,\ell}} + v_k \|\mathbf{p}_k\|_r - \boldsymbol{\mu}_k^\top \mathbf{p}_k \tag{32}$$

where $\boldsymbol{\mu}_k \in \mathbb{R}^{L_k}$ is the Lagrange dual variable. Regrettably, one may not form a closed-form solution of $\boldsymbol{\beta}_k$ for this constrained problem for a general r-norm; however, we may instead exploit a property of the optimality conditions. The Karush-Kuhn-Tucker conditions [50] state that a local solution yields the global minima of g_k if it (i) is a point where zero is in the sub-differential of the Langrangian, (ii) it is primal and dual feasible, i.e., $p_{k,\ell} \geq 0, \forall \ell$ and $\mu_{k,\ell} \geq 0, \forall \ell$, and (iii) that complementary slackness holds, i.e., $\mu_{k,\ell} p_{k,\ell} = 0, \forall \ell$. Considering the third condition, it states that if the solution will lie within the interior of the feasible set, such that $p_{k,\ell} > 0, \forall \ell$, then $\mu_{k,\ell} = 0, \forall \ell$, implying that the last term in (32) vanishes. It is also worth noting that $\mathbf{p}$ is constrained to non-negative solutions in (26), whereas the second term in g is non-differentiable on the boundary $p_{k,\ell} = 0$, implying that the use of subdifferentials are needed in solving the optimization problem. However, when $p_{k,\ell} \to 0_+$, then $g_k \to \infty$, and so g_k implicitly has its own barrier to prevent a zero solution. This allows the solution to be found as follows; assuming that $p_{k,\ell} > 0, \forall \ell$, which implies that $\boldsymbol{\mu}_k = \mathbf{0}$, and that (32) is differentiable, we solve for $\boldsymbol{\beta}_k$ and analyze whether the solution strays from the interior of the feasible set. Setting the derivative with respect to the ℓ:th variable to zero, one obtains

$$\frac{\partial \mathcal{L}(\mathbf{p}_k, \mathbf{0})}{\partial p_{k,\ell}} = -\frac{|\beta_{k,\ell}|^2}{p_{k,\ell}^2} + \frac{v_k p_{k,\ell}^{r-1}}{\|\mathbf{p}_k\|_r^{r-1}} = 0 \tag{33}$$

yielding

$$p_{k,\ell} = \frac{|\beta_{k,\ell}|^{2/r+1} \ \|\mathbf{p}_k\|_r^{r-1/r+1}}{(\sqrt{v_k})^{2/r+1}} \tag{34}$$

where $\|\cdot\|_m^{p/q}$ denotes the m-norm to the p/q:th power. Next, one may obtain an expression for $\|\mathbf{p}_k\|_r$ by first taking the r:th power on both sides of (34), and then summing these terms over ℓ, yielding

$$\|\mathbf{p}_k\|_r^r = \|\mathbf{b}_k\|_{2r/r+1}^{2r/r+1} \left(\frac{\|\mathbf{p}_k\|_r^{r-1/r+1}}{\left(\sqrt{v_k}\right)^{2/r+1}} \right)^r \tag{35}$$

where

$$\boldsymbol{\beta}_k = \left[\ \beta_{k,1} \quad \cdots \quad \beta_{k,L_k} \ \right]^\top, \quad \text{for } k = 1, \ldots, K+N \tag{36}$$

Solving for $\|\mathbf{p}_k\|_r$ yields

$$\|\mathbf{p}_k\|_r = \frac{\left\|\boldsymbol{\beta}_k\right\|_{2r/r+1}}{\sqrt{v_k}} \tag{37}$$

which, if inserted in (34), yields the estimate

$$\hat{p}_{k,\ell} = \frac{|\beta_{k,\ell}|^{2/r+1} \ \left\|\boldsymbol{\beta}_k\right\|_{2r/r+1}^{r-1/r+1}}{\sqrt{v_k}} \tag{38}$$

$\forall(k,\ell)$ in the parameter set. Thus, whenever $\beta_{k,\ell} \neq 0$, the solution is guaranteed to lie in the interior of the feasible set, and so (38) is the solution to (26). In turn, from (30) it holds that $\beta_{k,\ell} = 0$ only if either $p_{k,l} = 0$, or if there is exactly[1] zero linear dependence between the regressor $\mathbf{a}_{k,\ell}$ and the residual $\mathbf{R}^{-1}\mathbf{y}$. Thus, as long as neither $\mathbf{p}$ or $\boldsymbol{\beta}$ are initiated with the zero solution, an iterative scheme of solving (26) using (30) and (38) will stay within the feasible set, and so $\boldsymbol{\mu}_k = \mathbf{0}$. This in turn implies that the optimization scheme for estimating $\mathbf{p}$ is valid. Note that when $L_1 = \cdots = L_K = 1$,

$$\hat{p}_{k,\ell} = \frac{|\beta_{k,\ell}|}{\|\mathbf{a}_k\|_2} \tag{39}$$

[1] In practice, for noisy data, this is highly unlikely. But in the event of the unexpected, we set that particular $p_{k,l} = 0$ and exclude it from further estimation.

which thus coincides with the SPICE solution (see, e.g., [16, eq. (34)]). It is worth noting that $\boldsymbol{\beta}$ has here been introduced for the sake of implementational convenience. However, by combining (4) with (25), one obtains

$$\begin{aligned} &\mathbf{A}\boldsymbol{\beta} = \tilde{\mathbf{A}}\mathbf{x} + \mathbf{e} \quad \Longleftrightarrow \quad \boldsymbol{\beta} = \begin{bmatrix} \mathbf{x} \\ \mathbf{e} \end{bmatrix} \quad \Longleftrightarrow \\ &\boldsymbol{\beta}_k = \begin{cases} \mathbf{x}_k, & k = 1, \ldots, K \\ e_{k-K}, & k = K+1, \ldots, K+N \end{cases} \end{aligned} \tag{40}$$

where e_n denotes the estimated noise component at sample point n. Thus, one may note that the first K groups in $\boldsymbol{\beta}$ correspond to the response vectors $\mathbf{x}_k$, for $k = 1, \ldots, K$, whereas the last N elements in $\boldsymbol{\beta}$ correspond to the noise component $\mathbf{e}$. Using (30), an estimate of the response vector may be formed as

$$\hat{\mathbf{x}}_k = \operatorname{diag}(\hat{\mathbf{p}}_k)\, \mathbf{A}_k^H \left(\mathbf{A}\hat{\mathbf{P}}\mathbf{A}^H \right)^{-1} \mathbf{y}, \quad k = 1, \ldots, K \tag{41}$$

for $k = 1, \ldots, K$, which incidentally also corresponds to the linear minimum mean square error estimator formula in [25]. Algorithm 1 summarizes the proposed method, here termed group-SPICE.[2] The main computational cost, computing the covariance matrix, $\mathbf{R}$, occurs on line 7, such that the overall complexity of group-SPICE becomes $\mathcal{O}(NM^2)$. This may be compared to interior-point methods, such as, e.g., [13], which typically require $\mathcal{O}(M^3)$ evaluations. The algorithm is initialized in the same manner as SPICE, but the results are rather insensitive to this choice, and one may also use $\boldsymbol{\beta}^{(0)} = \mathbf{A}^\dagger \mathbf{y}$ inserted into (38), where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse. We deem the solution as converged when the change in variable is small, i.e., when $\left\| \mathbf{p}^{(j)} - \mathbf{p}^{(j-1)} \right\|_2 < \delta$, for some $\delta > 0$, or, for convenience sake, when some maximum number of iterations has been reached. Empirically, we find that sparse estimation solvers typically converge in support quite fast, i.e., determining which elements are zero (or near-zero), whereas convergence in magnitude is slower. Thus, if support recovery is the main objective, the convergence precision can be set rather low without loosing performance.

3.2 The special case of homoscedastic noise

We proceed to consider the case when the noise in homoscedastic, i.e., where $\boldsymbol{\Sigma} = \sigma\mathbf{I}$, i.e, when the signal is corrupted by an equivariance noise. In this case,

[2] An implementation will be provided online upon publication.

Algorithm 1 The heteroscedastic group-SPICE algorithm

initialize $j \leftarrow 0$,
for all (k, ℓ) **do**
  $p_{k,\ell}{}^{(0)} = \frac{|\mathbf{a}_{k,\ell}^H \mathbf{y}|^2}{||\mathbf{a}_{k,\ell}||^4}$
end for
repeat
  covariance update:
  $\mathbf{R} = \mathbf{APA}^H,\ \mathbf{z} = \mathbf{R}^{-1}\mathbf{y}$
  power update:
  for all (k, ℓ) **do**
    $r_{k,\ell} = |\mathbf{a}_{k,\ell}^H \mathbf{z}|$
    $p_{k,\ell}{}^{(j+1)} = \frac{(p_{k,\ell}{}^{(j)} r_{k,\ell})^{\frac{2}{r+1}} \left(\sum_{\ell=1}^{L_k} (p_{k,\ell}{}^{(j)} r_{k,\ell})^{\frac{2r}{r+1}} \right)^{\frac{r-1}{2r}}}{\sqrt{v_k}}$
  end for
  $j \leftarrow j + 1$
until convergence
for $k = 1, \ldots, K$ and $\forall \ell$ **do**
  $\hat{x}_{k,\ell} = p_{k,\ell}{}^{(\text{end})}\, \mathbf{a}_{k,\ell}^H \mathbf{z}$
end for

instead of (26) and (31), one obtains

$$\begin{aligned} \underset{\mathbf{p},\boldsymbol{\beta}}{\text{minimize}} \quad & g = \sum_{k=1}^{K} \left(\sum_{\ell=1}^{L_k} \frac{|\beta_{k,\ell}|^2}{p_{k,\ell}} + v_k \left\| \mathbf{p}_k \right\|_r \right) \\ & \quad + \left(\frac{1}{\sigma} \sum_{k=K+1}^{K+N} |\beta_k|^2 + N\sigma \right) \triangleq \sum_{k=1}^{K+1} g_k \\ \text{subject to} \quad & p_{k,\ell} \geq 0 \ \forall (k,\ell),\ \sigma \geq 0,\ \mathbf{A}\boldsymbol{\beta} = \mathbf{y} \end{aligned} \tag{42}$$

such that the noise in g is still separable from the signal components. Thus, for the K first groups, one obtains K optimization problems identical to those in the heteroscedastic case, with identical closed-form solutions. For the noise component, one may, using an argument similar to the one above and assuming that $\sigma > 0$, take the derivative of the corresponding Lagrangian of (42) with

respect to σ, with $\mu_{K+1} = 0$, and setting it to zero, obtaining

$$\frac{\partial \mathcal{L}(\sigma, 0)}{\partial \sigma} = -\frac{1}{\sigma^2} \sum_{k=K+1}^{K+N} |\beta_k|^2 + N = 0 \quad \Longrightarrow \tag{43}$$

$$\hat{\sigma} = \sqrt{\frac{1}{N} \sum_{k=K+1}^{K+N} |\beta_k|^2} = \frac{\|\mathbf{e}\|_2}{\sqrt{N}} \tag{44}$$

where (40) was used in the last step. As before, an iterative optimization scheme will not reach the feasibility boundary, i.e., $\sigma = 0$, as long as the noise component does not become zero. Similarly to the argument made for the heteroscedastic case, (10) is accepted as the solution to the constrained $(K + 1)$:th optimization problem described in (42). Algorithm 2 outlines the proposed group-SPICE algorithm for homoscedastic noise.

4 A connection to the group-LASSO

The optimization problem outlined in (26), together with the solution scheme where the closed-form estimation steps (30) and (38) are carried out iteratively, can be seen as a generalization of the SPICE algorithm, solving the extended problem for the case where each candidate dictionary atom may consist of a group of components, instead of just one component. For the SPICE algorithm, there is a connection between the covariance fitting criterion in (14) and the class of LASSO-type estimators (see [23–25] for details). However, for the LASSO, there typically exists at least one hyperparameter allowing the user to prioritize between the fit of a solution and its sparsity. By constrast, SPICE is designed to be hyperparameter-free and to require no tuning, or more precisely, the required hyperparameter has been selected to be optimal in the covariance fitting sense. Thus, for the covariance model and the corresponding covariance fitting criterion, SPICE yields both an, in some sense, optimal strategy for choosing the LASSO regularization parameter, and suggests an efficient implementation of the equivalent estimator. For the proposed group-SPICE, we will similarly establish its connection with the group-LASSO, as to show how the group sparse estimator may be optimally regularized in the context of covariance fitting. First, it is shown that group-SPICE formulation is scaling-invariant in same sense as SPICE is, i.e., that the addition of a multiplicative user parameter for either term in (20) will

Algorithm 2 The homoscedastic group-SPICE algorithm

initialize $j \leftarrow 0$, $\sigma^{(0)} = \sqrt{\frac{\mathbf{y}^H\mathbf{y}}{N}}$
for $k = 1, \ldots, K$, and $\forall \ell$ **do**
  $p_{k,\ell}{}^{(0)} = \frac{|\mathbf{a}_{k,\ell}^H \mathbf{y}|^2}{||\mathbf{a}_{k,\ell}||^4}$
end for
repeat
  covariance update:
  $\mathbf{R} = \mathbf{A}\mathbf{P}\mathbf{A}^H$, $\mathbf{z} = \mathbf{R}^{-1}\mathbf{y}$
  power update:
  $\sigma^{(j+1)} = \sigma^{(j)} \sqrt{\frac{\mathbf{z}^H\mathbf{z}}{N}}$, and
  for $k = 1, \ldots, K$, and $\forall \ell$ **do**
    $r_{k,\ell} = |\mathbf{a}_{k,\ell}^H \mathbf{z}|$, and
    $p_{k,\ell}{}^{(j+1)} = \frac{(p_{k,\ell}{}^{(j)} r_{k,\ell})^{\frac{2}{r+1}} \left(\sum_{\ell=1}^{L_k} (p_{k,\ell}{}^{(j)} r_{k,\ell})^{\frac{2r}{r+1}} \right)^{\frac{r-1}{2r}}}{\sqrt{v_k}}$
  end for
  $j \leftarrow j + 1$
until convergence
for $k = 1, \ldots, K$, and $\forall \ell$ **do**
  $\hat{x}_{k,\ell} = p_{k,\ell}{}^{(\text{end})} \, \mathbf{a}_{k,\ell}^H \mathbf{z}$
end for

not affect the estimate of the regressor variables, $\mathbf{x}_k$, for $k = 1, \ldots, K$. To that end, consider the two optimization problems

$$\hat{\mathbf{p}} = \arg\min_{\mathbf{p}} \quad g = \mathbf{y}^H \left(\mathbf{A}\mathbf{P}\mathbf{A}^H \right)^{-1} \mathbf{y} + \sum_{k=1}^{K+N} v_k \left\| \mathbf{p}_k \right\|_r \tag{45}$$

$$\hat{\bar{\mathbf{p}}} = \arg\min_{\bar{\mathbf{p}}} \quad g' = \mathbf{y}^H \left(\mathbf{A}\bar{\mathbf{P}}\mathbf{A}^H \right)^{-1} \mathbf{y} + \gamma^2 \sum_{k=1}^{K+N} v_k \left\| \bar{\mathbf{p}}_k \right\|_r \tag{46}$$

where the second problem has been scaled by an arbitrary $\gamma > 0$. For (45), the group-SPICE solution for fixed β is given in (38). For (46), by incorporating γ^2 into the v_k:s, i.e.,

$$v_k' \triangleq \gamma^2 v_k, \forall k \tag{47}$$

its solution follows analogously, i.e.,

$$\hat{\tilde{p}}_{k,\ell} = \frac{|\beta_{k,\ell}|^{2/r+1} \, \|\mathbf{b}_k\|_{2r/r+1}^{r-1/r+1}}{\sqrt{\gamma^2 v_k}} = \frac{1}{\gamma}\hat{p}_{k,\ell} \quad \Longleftrightarrow \quad \hat{\tilde{\mathbf{p}}} = \frac{1}{\gamma}\hat{\mathbf{p}} \tag{48}$$

and using (41), the response vector estimate for (46) becomes

$$\begin{aligned} &\hat{\tilde{\mathbf{x}}}_k = \frac{1}{\gamma}\hat{\mathbf{P}}\,\mathbf{A}_k^H \left(\mathbf{A}\frac{1}{\gamma}\hat{\mathbf{P}}\mathbf{A}^H\right)^{-1}\mathbf{y}, \quad k = 1,\ldots,K \quad \Longrightarrow \\ &\hat{\tilde{\mathbf{x}}}_k = \hat{\mathbf{x}}_k, \quad k = 1,\ldots,K \end{aligned} \tag{49}$$

implying that the optimization problem in (46) yields an identical estimate to that obtained from (46). We may therefore conclude that group-SPICE exhibits a scaling invariance similar to the original SPICE formulation. This observation offers justification for removing $\|\mathbf{y}\|_2^2$ from the first term in g in (22). Next, we show how the estimate relates to the LASSO.

4.1 The connection with the LASSO for heteroscedastic noise

In deriving the proposed group-SPICE estimator, a change of variables was made, introducing the auxiliary variable $\boldsymbol{\beta}$, thereby transforming the relaxed covariance fitting problem in (22) into the group-SPICE problem in (26). In this section, we rewrite the group-SPICE problem into a LASSO-type problem by finding an equivalent optimization problem through a change of variables. By using (38), one may reformulate (31) to be expressed in $\boldsymbol{\beta}$ exclusively, such that

$$g(\hat{\mathbf{p}},\boldsymbol{\beta}) = \sum_{k=1}^{K+N}\left(\sum_{\ell=1}^{L_k}\frac{|\beta_{k,\ell}|^2}{\hat{p}_{k,\ell}} + v_k\,\|\hat{\mathbf{p}}_k\|_r\right) \tag{50}$$

$$= \sum_{k=1}^{K+N}\left(\sum_{\ell=1}^{L_k}\frac{|\beta_{k,\ell}|^2}{|\beta_{k,\ell}|^{2/r+1}}\sqrt{v_k}\,\|\boldsymbol{\beta}_k\|_{2r/r+1}^{-\frac{r-1}{r+1}} + v_k\frac{\|\boldsymbol{\beta}_k\|_{\frac{2r}{r+1}}}{\sqrt{v_k}}\right) \tag{51}$$

$$= \sum_{k=1}^{K+N}\left(\|\boldsymbol{\beta}\|_{2r/r+1}^{2r/r+1}\sqrt{v_k}\,\|\boldsymbol{\beta}_k\|_{2r/r+1}^{-\frac{r-1}{r+1}} + \sqrt{v_k}\,\|\boldsymbol{\beta}_k\|_{\frac{2r}{r+1}}\right) \tag{52}$$

$$= 2\sum_{k=1}^{K+N}\sqrt{v_k}\,\|\boldsymbol{\beta}_k\|_{\frac{2r}{r+1}} \tag{53}$$

Then, one may formulate an optimization problem equivalent to (26) as

$$\begin{aligned} \underset{\boldsymbol{\beta}}{\text{minimize}} \quad & g = \sum_{k=1}^{K+N} \sqrt{v_k} \left\| \boldsymbol{\beta}_k \right\|_{\frac{2r}{r+1}} \\ \text{subject to} \quad & \mathbf{A}\boldsymbol{\beta} = \mathbf{y} \end{aligned} \tag{54}$$

Changing the variables from $\boldsymbol{\beta}$ to $\mathbf{x}$ and $\mathbf{e}$, as per (40), yields

$$g(\mathbf{x}, \mathbf{e}) = \sum_{k=1}^{K} \sqrt{v_k} \left\| \mathbf{x}_k \right\|_{\frac{2r}{r+1}} + \sum_{k=K+1}^{K+N} \sqrt{v_k} |e_{k-K}| \tag{55}$$

$$= \sum_{k=1}^{K} \sqrt{v_k} \left\| \mathbf{x}_k \right\|_{\frac{2r}{r+1}} + \left\| \mathbf{e} \right\|_1 \tag{56}$$

and the equivalent optimization problem

$$\begin{aligned} \underset{\mathbf{x},\mathbf{e}}{\text{minimize}} \quad & g = \left\| \mathbf{e} \right\|_1 + \sum_{k=1}^{K} \sqrt{v_k} \left\| \mathbf{x}_k \right\|_{\frac{2r}{r+1}} \\ \text{subject to} \quad & \tilde{\mathbf{A}}\mathbf{x} + \mathbf{e} = \mathbf{y} \end{aligned} \tag{57}$$

Incorporating the constraint into the cost function, by expressing $\mathbf{e}$ using $\mathbf{x}$, we obtain the equivalent LASSO-type optimization problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad g = \left\| \mathbf{y} - \tilde{\mathbf{A}}\mathbf{x} \right\|_1 + \sum_{k=1}^{K} \sqrt{v_k} \left\| \mathbf{x}_k \right\|_{\frac{2r}{r+1}} \tag{58}$$

which is the group-version of the weighted LAD-LASSO [26]. For this LASSO-type estimator, there typically exists a hyperparameter, λ_k, for each group, which is, by construction, chosen as $\lambda_k = \sqrt{v_k}$ for the proposed heteroscedastic group-SPICE.

4.2 The connection with the LASSO for homoscedastic noise

In the case of homoscedastic noise, i.e., when $\Sigma = \sigma\mathbf{I}$, one may use g as expressed in (42) and perform a change of variables similar as is done above, which yields

$$g(\hat{\mathbf{p}}, \boldsymbol{\beta}) = \sum_{k=1}^{K} \left(\sum_{\ell=1}^{L_k} \frac{|\beta_{k,\ell}|^2}{\hat{p}_{k,\ell}} + v_k \left\| \hat{\mathbf{p}}_k \right\|_r \right) + \frac{1}{\hat{\sigma}} \sum_{k=K+1}^{K+N} |\beta_k|^2 + N\hat{\sigma} \tag{59}$$

Variable representation	**p**	**p** and β	β
Optimization problem	(22)	(26)	(54)
Methodology	covariance fitting	group-SPICE	group-LASSO

Table 1: Interpretation of the relaxed covariance fitting criterion for grouped variables under different choices of variable representations

$$= 2\sum_{k=1}^{K}\sqrt{v_k}\left\|\beta_k\right\|_{\frac{2r}{r+1}} + \frac{\sqrt{N}}{\|\mathbf{e}\|_2}\|\mathbf{e}\|_2^2 + N\frac{\|\mathbf{e}\|_2}{\sqrt{N}} \tag{60}$$

$$= 2\left(\sqrt{N}\|\mathbf{e}\|_2 + \sum_{k=1}^{K}\sqrt{v_k}\left\|\beta_k\right\|_{\frac{2r}{r+1}}\right) \tag{61}$$

Thus, one may formulate an optimization problem equivalent to (42) as

$$\begin{aligned} \underset{\mathbf{x},\mathbf{e}}{\text{minimize}} \quad & g = \sqrt{N}\|\mathbf{e}\|_2 + \sum_{k=1}^{K+N}\sqrt{v_k}\|\mathbf{x}_k\|_{\frac{2r}{r+1}} \\ \text{subject to} \quad & \tilde{\mathbf{A}}\mathbf{x} + \mathbf{e} = \mathbf{y} \end{aligned} \tag{62}$$

and then incorporating the constraint into the cost function yielding the LASSO-like optimization problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad g = \left\|\mathbf{y} - \tilde{\mathbf{A}}\mathbf{x}\right\|_2 + \sum_{k=1}^{K}\sqrt{\frac{v_k}{N}}\|\mathbf{x}_k\|_{\frac{2r}{r+1}} \tag{63}$$

which is a group-version of the weighted square-root LASSO [27]. Also in this case there exists a hyperparameter, λ_k, for each group, which is, by construction, chosen as $\lambda_k = \sqrt{v_k/N}$ for the proposed homoscedastic group-SPICE. We may conclude that the distinction between hetero- and homoscedasticity in the covariance fitting model results in different LASSO formulations, where the misfit cost term is either the ℓ_1- or ℓ_2-norm. This implies that the heteroscedastic group-LASSO offers more robustness than its homoscedastic counterpart, as the model allows for data anomalies, e.g., outliers, to be modeled as independent noise samples. Also, we have shown that by changing the variable representation of the covariance fitting problem, we may equivalently formulate it as a SPICE

or a LASSO problem, which is illustrated in Table 1. We also believe it is of interest to note that, despite the different sets of assumptions made in the covariance matching and sparse regression frameworks, respectively, both formulations turn out to be equivalent.

5 Considerations for hyperparameter-free estimation with group-SPICE

In this work, we have introduced group-SPICE by applying Hölder's inequality to the second term of the covariance fitting criteria in (14), which holds true for any r-norm, $r \geq 1$. In this section, we examine what may constitute a suitable choice of this design-parameter. In connection with the equivalent LASSO-type estimators in (58) and (63), the choice of r affects which norm is used in the penalty functions. Define

$$q \triangleq \frac{2r}{r+1} \tag{64}$$

as the penalty norm for these group-LASSOs. The constraint $r \geq 1$ implies that group-SPICE is equivalent to a LASSO formulation fulfilling

$$1 \leq q < 2 \tag{65}$$

where on the lower bound, one obtains the non-grouped LASSO, whereas on the upper bound one obtains the ℓ_2 group-LASSO described in the earlier literature. With the purpose of achieving group sparsity, $q \to 2$ is an intuitive choice of design, as the ℓ_2-norm does not promote sparsity within a group, whereas the sum of such norms does promote group-sparsity among them, as intended. In some cases, however, it may be reasonable to assume that not all components within a candidate group are represented in the data, and thus the group-LASSO for sparse groups was introduced in [35]. It does so by introducing a second penalty function, and solving

$$\underset{\mathbf{x}}{\text{minimize}} \quad g = \left\| \mathbf{y} - \tilde{\mathbf{A}}\mathbf{x} \right\|_2^2 + \lambda \left(\mu \left\| \mathbf{x} \right\|_1 + (1-\mu) \sum_{k=1}^{K} \sqrt{L_k} \left\| \mathbf{x}_k \right\|_2 \right) \tag{66}$$

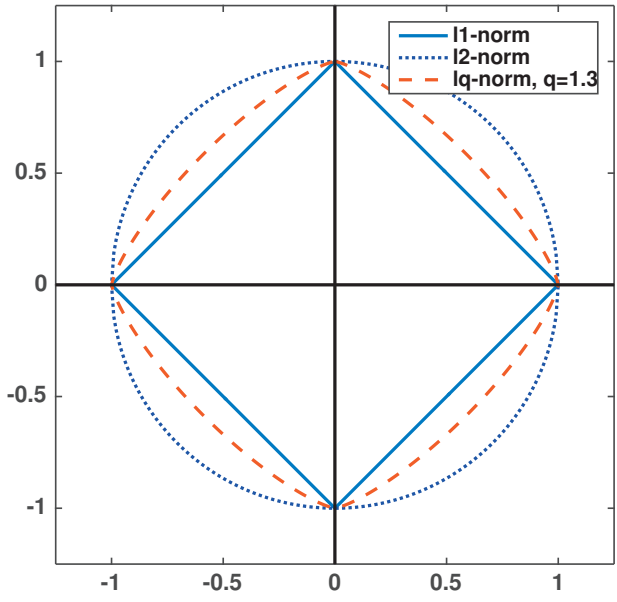


Figure 1: Illustration of the LASSO-type penalty for different choice of ℓ_q-norms for a group with two components, $x_{k,1}$ and $x_{k,2}$. The sparsifying effect of the group-LASSO for sparse groups corresponds to a choice $1 < q < 2$, which is equivalent of setting $1 < r < \infty$ in group-SPICE.

i.e., by using a combination of penalty terms with ℓ_1- and ℓ_2-norms, where the hyperparameter $\mu \in [0, 1]$ prioritizes between regular and group sparsity, and where λ sets the level of sparsity. For this LASSO-type, the authors in [55] show that, for the components within a group, the penalty's constraint region lies between that of the ℓ_1- and ℓ_2-norms, as illustrated in Figure 1. For group-SPICE, this implies that one would achieve similar group sparsity with sparse groups for $1 < q < 2$, i.e., $1 < r < \infty$. Next, it is worth examining the inherent choice of hyperparameter for the homo- and heteroscedastic group-SPICE methods. First, we note that any scaling of the terms in g, such as with γ^2 in (46), will not affect the

hyperparameter. This becomes apparent in (55) for the scaled g', as

$$g'(\mathbf{x},\mathbf{e}) = \sum_{k=1}^{K} \sqrt{\gamma^2 v_k}\, \|\mathbf{x}_k\|_{\frac{2r}{r+1}} + \sum_{k=K+1}^{K+N} \sqrt{\gamma^2 v_k}|e_{k-K}| = \gamma\, g(\mathbf{x},\mathbf{e}) \tag{67}$$

which when minimized thus attains the same optimal point in $\mathbf{x}$ as g.

Second, let λ_k be the regularization term for the k:th group for the group-LASSO reformulations descibed herein. Then, using (58) and (63), one may conclude that

$$\lambda_k = \sqrt{\frac{v_k}{N^\nu}} = \sqrt{\frac{1}{N^\nu}} \left\| \begin{bmatrix} \|\mathbf{a}_{k,1}\|_2^2 & \cdots & \|\mathbf{a}_{k,L_k}\|_2^2 \end{bmatrix}^\top \right\|_{\frac{2r}{r-1}} \tag{68}$$

with $\nu = 0$ and $\nu = 1$ for the hetero- and homoscedastic noise cases, respectively. As an example, for many applications, the dictionary $\tilde{\mathbf{A}}$ is constructed such that is has normalized atoms, i.e., $\|\mathbf{a}_{k,\ell}\|_2 = 1, \forall(k,\ell)$. Thus, we obtain

$$\lambda_k = \sqrt{\frac{{L_k}^{r-1/r}}{N^\nu}} \tag{69}$$

and, e.g., for the homoscedastic group-SPICE with $r \to \infty$, it is equivalent to a square-root group-LASSO with regularization $\lambda_k = \sqrt{L_k/N}$.

6 Numerical results

In this section, we evaluate the performance of the group-SPICE methods presented in this paper. For simulated signals, we establish the results that group-SPICE performs as well as an optimally regularized group-LASSO, and has preferable performance to both standard SPICE and common greedy group sparse methods. We perform these evaluations under different levels of noise power, sample size, and dictionary coherence. We also show that the implicit regularization level of group-SPICE in comparison to the square-root group-LASSO is adequate. First, we consider Gaussian dictionaries.

6.1 Signals from Gaussian dictionaries

We commence by examining the problem of recovering a group sparse signal generated from a real-valued dictionary with regressor candidates drawn from an

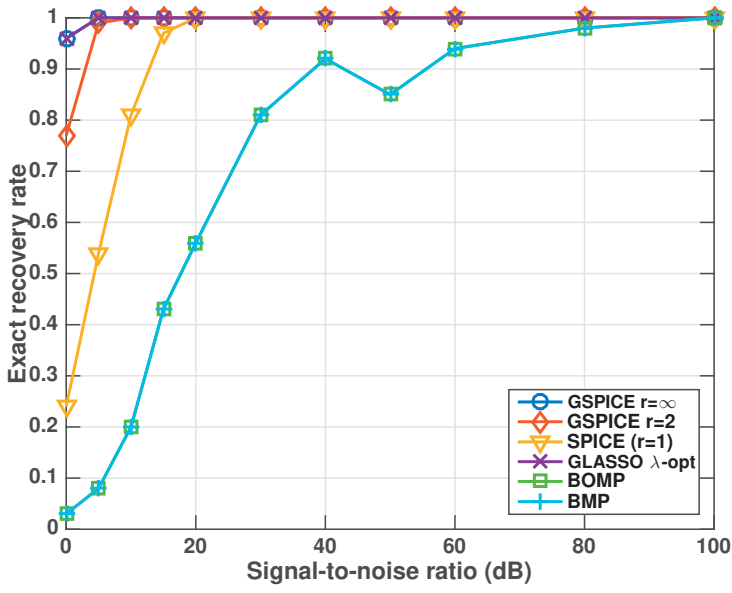


Figure 2: Exact recovery rates from 100 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario one, where an incoherent Gaussian dictionary is used, with $C = 3$ active groups, for varying SNR levels.

independent Gaussian distribution, i.e.,

$$\mathbf{a}_{k,\ell} \in \mathcal{N}\left(\mathbf{0}, \xi_{k,\ell}\mathbf{I}\right) \tag{70}$$

$\forall(k,\ell)$, where $\xi_{k,\ell}$ is chosen such that $\left\|\mathbf{a}_{k,\ell}\right\|_2 = 1$. Such a dictionary is known to be incoherent with high probability [56]. Furthermore, we let the signal be corrupted by an equivariance circular symmetric Gaussian noise. We compare the performance of the homoscedastic group-SPICE to the standard group-LASSO [30], the standard SPICE, as well as two greedy methods, namely the Block Orthogonal Matching Pursuit (BOMP) [57], and the Block Matching Pursuit (BMP) [58]. In the simulations, the group-LASSO is allowed oracle knowledge of the noise variance, and the hyperparameter is selected as $\lambda = \sigma\sqrt{L}$. In this setting, the group-LASSO will illustrate a soft upper performance bound for the group-SPICE estimator. Furthermore, without loss of generality, we choose $L_1 = \cdots = L_K = L$. The signal is generated by randomly selecting C groups

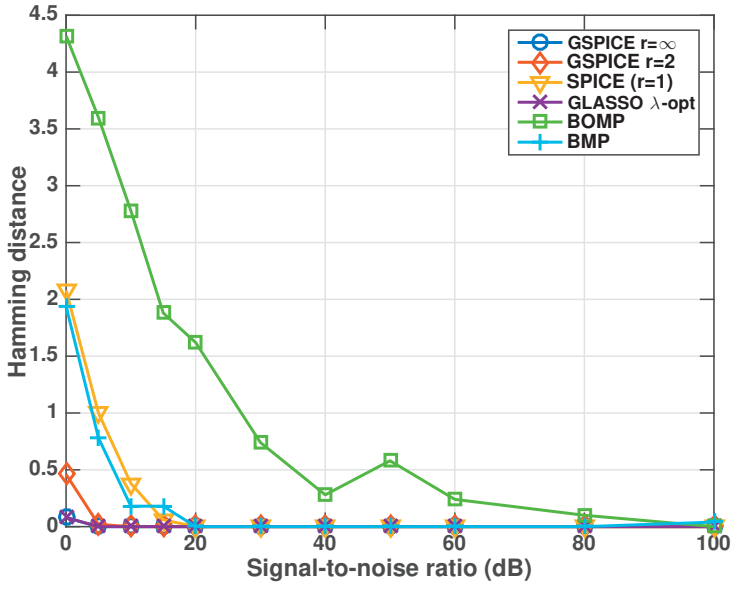


Figure 3: Hamming distances from 100 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario one, where an incoherent Gaussian dictionary is used, with $C = 3$ active groups, for varying SNR levels.

from the dictionary to make up the signal, where each group randomly consists of between $\lceil L/2 \rceil$ and L active components, with $\lceil \cdot \rceil$ denoting the ceiling operator. Thus, the active groups have on average a smaller support than the dictionary groups, illustrating the realistic scenario of not precisely knowing the model orders. For each active group, we set the parameter value $\mathbf{x}_c = \mathbf{1}, c = 1, \ldots, C$. In the first simulation scenario, we examine the performance over different levels of the signal-to-noise ratio, defined as $\text{SNR} = 10 \log(\sigma^2_{\text{sig}}/\sigma^2_e)$, where σ^2_{sig} and σ^2_e denote the power of the signal and the noise, respectively. Here, $N = 200$ samples, $P = 200$ candidate groups, $L = 10$, $C = 3$ active groups. To obtain statistics, we perform $\text{MC} = 100$ Monte-Carlo simulations on which parameter estimation is performed using the above mentioned methods. To measure the estimation performance, we use the exact recovery rate (ERR), defined as

$$\text{ERR}^{(i)} = 1\left\{\hat{\mathcal{I}}_C^{(i)} = \mathcal{I}_C^{(i)}\right\} \tag{71}$$

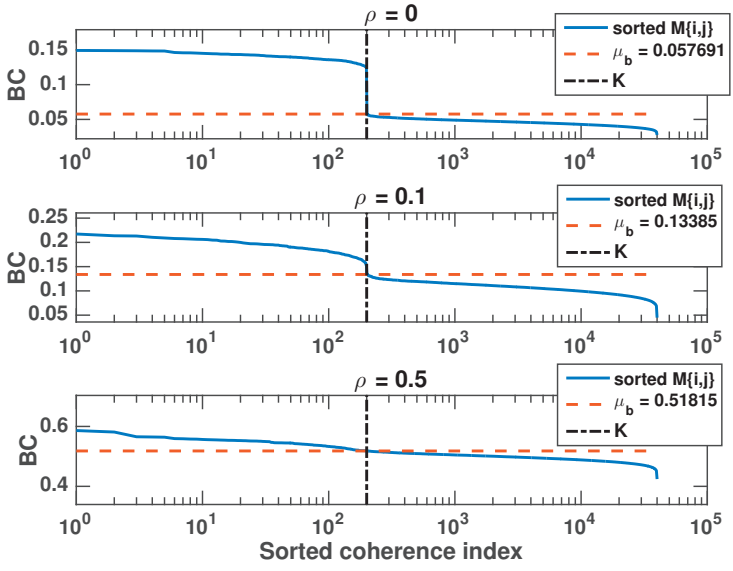


Figure 4: Plot of block coherence (BC) elements in the vector vec(**M**), sorted from high to low, where the first K elements correspond to the diagonal elements of **M**, indicated by the vertical dash-dotted line, whereafter the last $K^2 - K$ elements correspond to the off-diagonal coherences. The BC measure μ_b corresponds to the largest off-diagonal element, i.e., the $(K+1)$:th element in the plot, indicated by the horizontal dashed line.

for the i:th Monte-Carlo realisation, where $\hat{\mathcal{I}}_S$ denotes the set of group indices whose estimated parameters have the largest ℓ_2-norm, and $\mathcal{I}_C$ is the set of true group indices. In other words, ERR measures whether the C largest groups in the estimate have the same support as the ground truth. Secondly, we use the hamming distance (HMD) for groups, defined as the number of binary flips $0 \to 1$ or $1 \to 0$ required to obtain the correct group support, i.e.,

$$\text{HMD}^{(i)} = \sum_{k \in \hat{\mathcal{I}}_C^{(i)}} 1\left\{k \notin \mathcal{I}_C^{(i)}\right\} + \sum_{k' \in \mathcal{I}_C^{(i)}} 1\left\{k' \notin \hat{\mathcal{I}}_C^{(i)}\right\} \tag{72}$$

implying that $0 \leq \text{HMD} \leq 2C$. These measures for the first scenario are shown in Figures 2 - 3, respectively.

We examine the performance of group-SPICE with both $r = \infty$, corres-

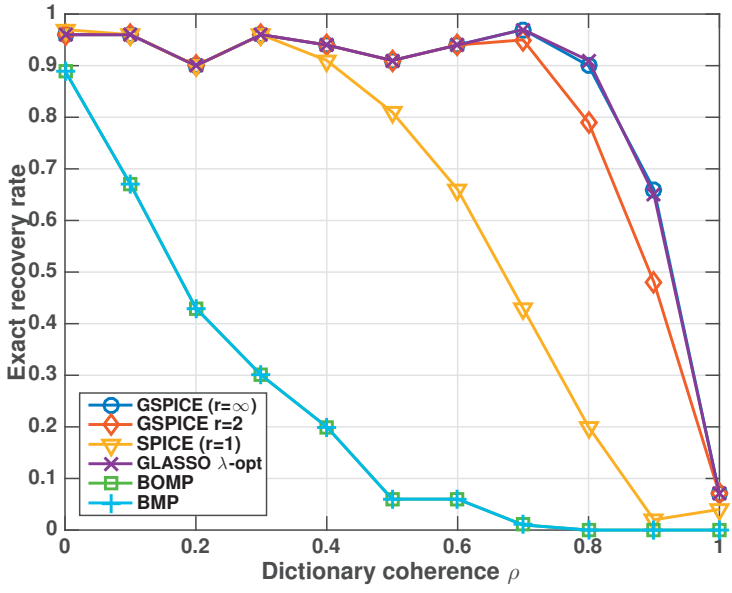


Figure 5: Exact recovery rates from 200 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario two, where Gaussian dictionaries of different levels of coherence are used, with $C = 3$ active groups.

ponding to the standard group-LASSO, as well as with $r = 2$, corresponding to a mix between group- and component-wise sparsity as described above. As can be seen from the figures, the non-greedy estimators outperform the greedy estimators, whereas group-SPICE outperforms the standard SPICE estimator, indicating that imposing group sparsity improves estimation performance. It is worth noting that, as expected, the hyperparameter-free group-SPICE performs on par with an optimally regularized group-LASSO. One may also note that BOMP and BMP performs equally in terms of ERR, but in terms of HMD, BMP performs as well as SPICE, indicating that BMP have determined some groups correctly, although not all.

In principle, as may be seen in the first scenario, even a non-grouping sparse estimator often finds the correct group support if the dictionary is sufficiently orthonormal. Support recovery results for such dictionaries has been shown (see, e.g., [11]), but intuitively it also makes sense, considering that an orthonormal

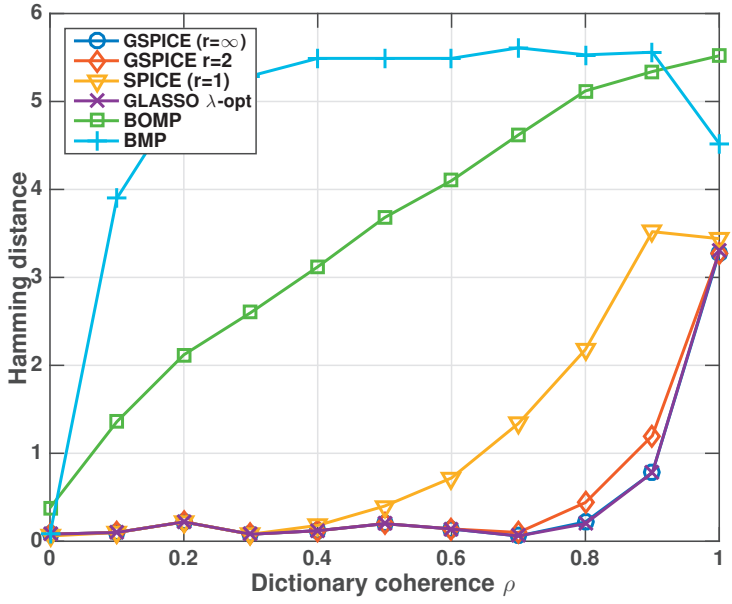


Figure 6: Hamming distances from 200 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario two, where Gaussian dictionaries of different levels of coherence are used, with $C = 3$ active groups.

dictionary has components which uniquely describe the data, in which no support mismatch may occur. In several applications, e.g., the multi-pitch estimation problem, there are different possible groups having single components that may partly model the signal, however only one group which model all the components in a source. For such dictionaries, referred to as being coherent or having overlapping groups, standard sparse regression methods (such as, e.g., SPICE or LASSO) may not differentiate the correct group support from its spuriously matching components, whereas their group-sparse estimation counterparts would select the group which corresponds to the best clustering of components. Furthermore, a consequence of such dictionary designs is that the true sources may also have components which are coherent, why greedy estimators, which estimate the sources serially rather than jointly, are likely to estimate the wrong group support. Therefore, in the second scenario, coherence is added between dictionary

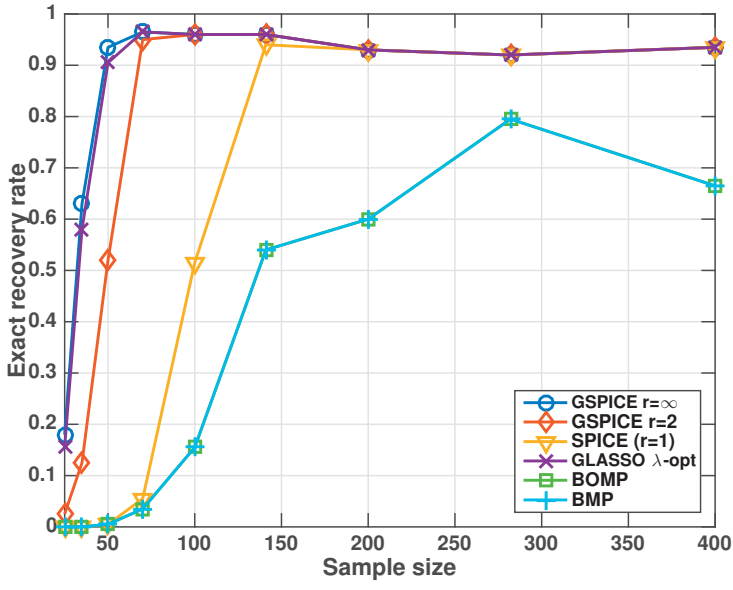


Figure 7: Exact recovery rates from 100 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario three, where Gaussian dictionaries with coherence $\rho = 0.1$ are used, with $C = 3$ active groups, for different sample lenghts.

components. To that end, we let

$$\mathbf{a}_{k,\ell} = \sum_{\mathcal{I}^{\rho}_{k,\ell}} \mathbf{b}_{k',\ell'}, \quad \mathbf{b}_{k',\ell'} \in \mathcal{N}\left(\mathbf{0}, \xi_{k',\ell'}\mathbf{I}\right) \tag{73}$$

where $\mathbf{b}_{k,\ell}, \forall(k,\ell)$ are the regressors of the incoherent dictionary used above, and where $\mathcal{I}^{\rho}_{k,\ell}$ is an index set of size n equal to a random sample from a binomial distribution, i.e., $n \in \text{Bin}\big((K-1)L, \rho\big)$, where these indices are uniformly drawn from

$$(k',\ell') \in \{k : 1 \leq k \leq K, k' \neq k\} \times \{1 \leq \ell \leq L\} \tag{74}$$

Thus, there are no coherent components within each group, but for every component in a group, there will on average be $(K-1)L\rho$ components in other groups to which it is coherent. To quantize the amount of coherency between

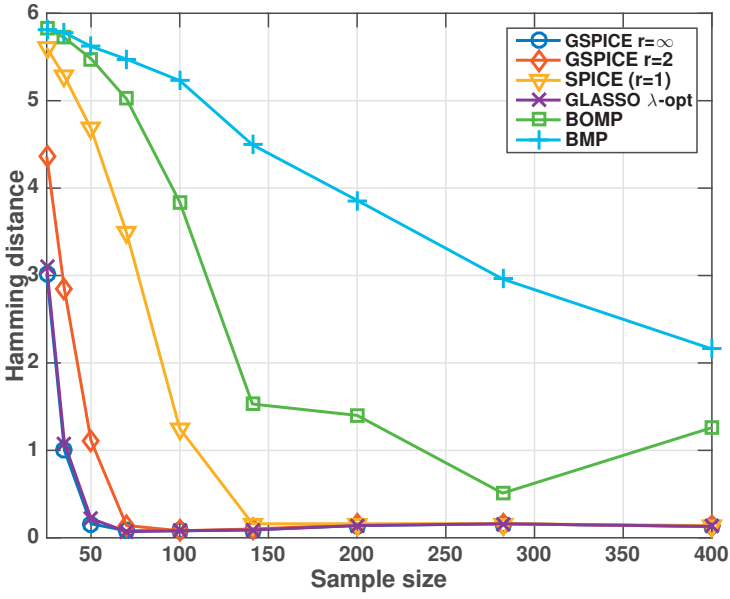


Figure 8: Hamming distances from 100 Monte-Carlo samples estimated with group-SPICE in comparison to other methods under simulation scenario three, where Gaussian dictionaries with coherence $\rho = 0.1$ are used, with $C = 3$ active groups, for different sample lenghts.

groups in the dictionary, we use the block-coherence measure, μ_b, defined as [57]

$$\mu_b = \max_{i \neq j} \mathbf{M}\{i,j\}, \quad \mathbf{M}\{i,j\} = L^{-1} \left\| \mathbf{A}_i{}^H \mathbf{A}_j \right\|_2 \tag{75}$$

where L is the maximal group size, and $\mathbf{B}\{i,j\}$ denotes the (i,j):th matrix element and $\|\mathbf{B}\|_2$ the spectral norm for a matrix $\mathbf{B}$. To illustrate the difference between the incoherent case, $\rho = 0$, and coherent dictionary designs where $\rho > 0$, we examine the distribution of elements in vec($\mathbf{M}$), where vec($\cdot$) denotes vector stacking, by ordering them from high to low. The ordered sample of vec($\mathbf{M}$) is seen in Figure 4 for three realizations of Gaussian dictionaries with different levels of coherence. As ρ increases, so does the block coherence measure, but the figure also illustrates the distribution of coherence values in $\mathbf{M}$; The diagonal elements, i.e., $\mathbf{M}\{i,i\}, i = 1,\ldots,P$, denotes the co-dependence of a group with itself, whereas off-diagonal, i.e., $\mathbf{M}\{i,j\}, i \neq j$, denotes the co-dependence with the

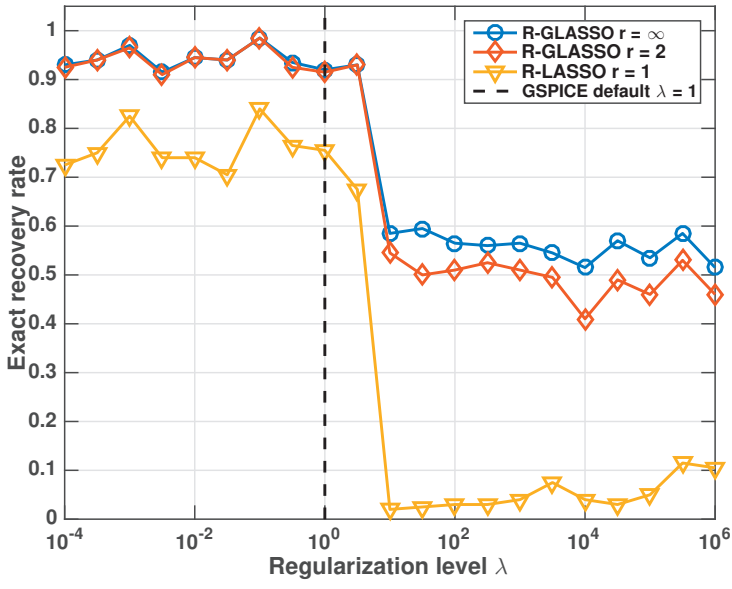


Figure 9: Exact recovery rates from 200 Monte-Carlo samples estimated with square-root group LASSO for different group-norms r, i.e., with penalties $\|\mathbf{x}_k\|_{\frac{2r}{r-1}}$, at different levels of regularization. Gaussian dictionaries with coherence $\rho = 0.1$ are used, where $C = 3$ active groups and SNR $= 10$ dB. Here, $\lambda = 1$ corresponds to the group-SPICE regularization level, illustrated by the vertical dashed line.

other candidate groups, which should ideally be low. As can be seen in Figure 4, when the block-coherence is low, the coherence is much higher on diagonal than off the diagonal, whereas when block-coherence increases, the differences become negligible. Thus, as $\rho \to 1$, many groups are almost as co-dependent with other candidates as with themselves, which intuitively makes sparse estimation increasingly difficult. In a second simulation scenario, we evaluate the performance at different levels of coherence, ρ, using the same settings as above, with SNR $= 40$ dB. Figures 5 - 6 illustrate the comparison between the methods measured in ERR and HMD, respectively. Here, we have the same performance relationships as before, although in this case BOMP can be seen to outperform BMP, likely due to its orthogonalization feature which slightly offsets the dictionary coherency. It also becomes apparent that group-SPICE finds the correct group subset,

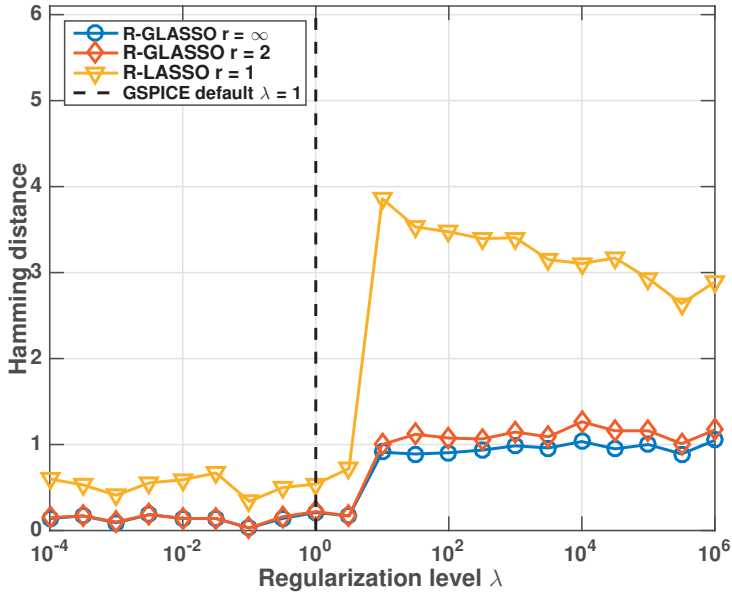


Figure 10: Hamming distances from 200 Monte-Carlo samples estimated with square-root group LASSO for different group-norms r, i.e., with penalties $\|\mathbf{x}_k\|_{\frac{2r}{r-1}}$, at different levels of regularization. Gaussian dictionaries with coherence $\rho = 0.1$ are used, where $C = 3$ active groups and SNR $= 10$ dB. Here, $\lambda = 1$ corresponds to the group-SPICE regularization level, illustrated by the vertical dashed line.

even with very high dictionary coherence. In a third scenario, we examine the performance under varying sample size. Otherwise similar to the first scenario, here we use SNR $= 40$ dB and $\rho = 0.1$, i.e., with some added coherence. The results are shown in Figures 7 - 8, indicating similar results as seen in the previous scenarios. Next, we examine how the choice of regularization level affects the estimation performance, to assess whether the model-based choice of regularization in group-SPICE is optimal. To that end, we form estimates using the square-root group-LASSO estimator at different regularization levels. To simplify comparison, we reparametrize its hyperparameter using (70) as

$$\lambda = \mu \sqrt{\frac{L^{r-1/r}}{N}} \tag{76}$$

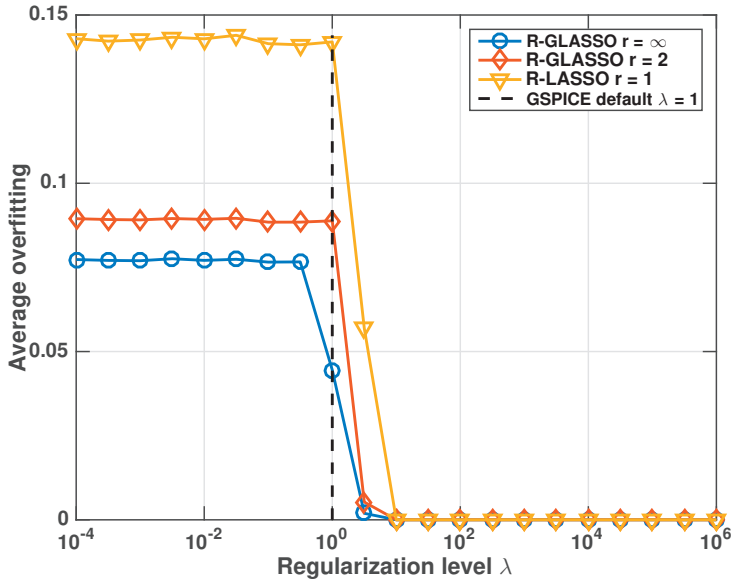


Figure 11: Average overfitting from 200 Monte-Carlo samples estimated with square-root group LASSO for different group-norms r, i.e., with penalties $\|\mathbf{x}_k\|_{\frac{2r}{r-1}}$, at different levels of regularization. Gaussian dictionaries with coherence $\rho = 0.1$ are used, where $C = 3$ active groups and SNR $= 10$ dB. Here, $\lambda = 1$ corresponds to the group-SPICE regularization level, illustrated by the vertical dashed line.

where $\mu > 0$, such that the group-SPICE estimator is obtained when $\mu = 1$. Here, we use the settings from the first scenario, SNR $= 10$ dB, and $\rho = 0.1$. Figures 9 - 11 illustrate the resulting performance, also showing the average overfitting measure

$$\varepsilon(\mathbf{x}, C) = \frac{1}{K - C} \sum_{k \notin \hat{\mathcal{J}}_C} \|\mathbf{x}_k\|_2 \tag{77}$$

such that ε measures the average ℓ_2-norm of the parameters outside the designated signal set, indicating power leakage in the estimates. In Figures 9 and 10, one may note that at some just point above $\mu = 1$, ERR falls significantly, as a result of too much regularization, which thereby yields a (nearly) zero solution. At exactly how much above the group-SPICE regularization this happens depends

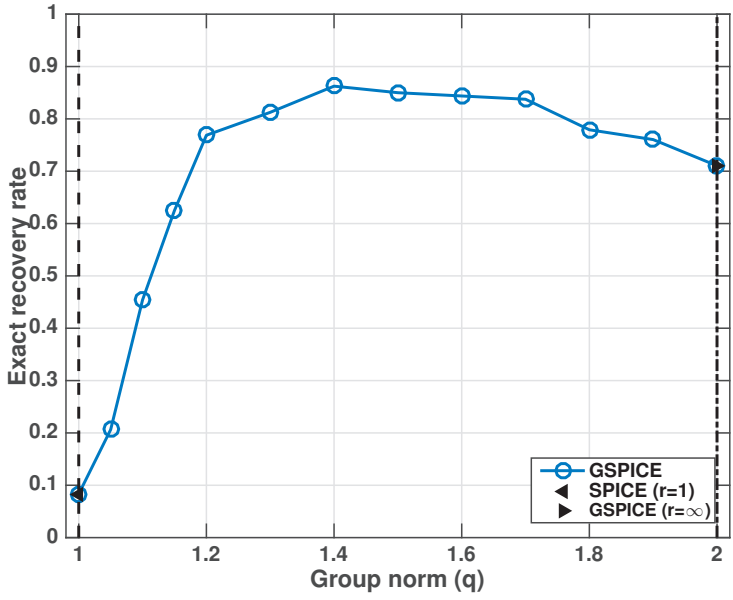


Figure 12: Exact recovery rates from 480 Monte-Carlo samples estimated with group-SPICE under different choices of grouping norm q. Gaussian dictionaries with coherence $\rho = 0.1$ are used, where $C = 3$ active groups and SNR $= 40$ dB. Here, the lower endpoint $q = 1$ corresponds to the regular SPICE method, and subsequently $q = 2$ gives the standard ℓ_2-norm commonly used in group-sparse regression.

on the studied problem; in all examined examples, we have observed a similar effect occurring at, or at most one order of magnitude above, $\mu = 1$. Another property of varying the regularization level is shown in Figure 11, where one may note that too low regularization gives estimates which are too dense, corresponding poorly to the true level of sparsity in the signal, which occurs at, or slightly below, $\mu = 1$. In conclusion, we observe that the inherent group-SPICE regularization offers a trade-off between sparsity and model fit, just small enough not to give a zero solution, although large enough to avoid excessive overfitting. In the final scenario for Gaussian dictionaries, we examine the choice of the grouping norm, r. We use the same setting as in the fourth scenario, with $N = 100$, and SNR $= 40$ dB, although the dictionary group sizes are on purpose made much too large. We thus set $L = 40$, whereas in the true blocks only between 5 and

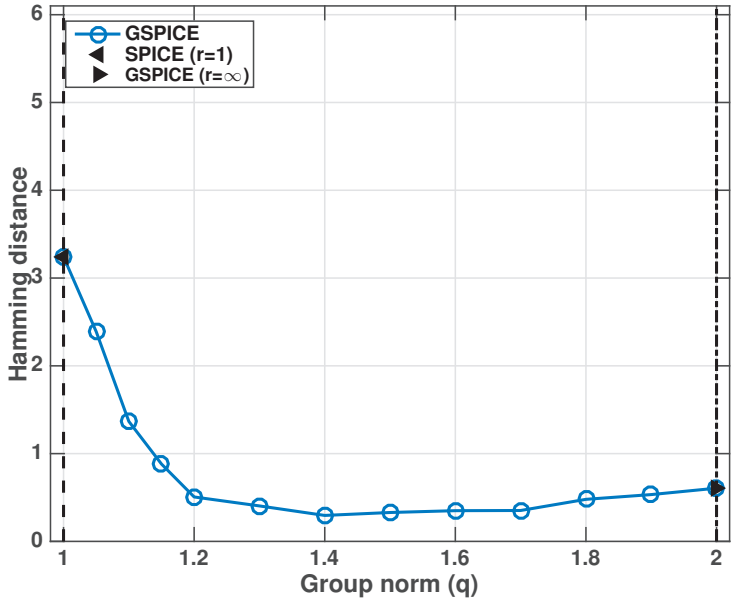


Figure 13: Hamming distances from 480 Monte-Carlo samples estimated with group-SPICE under different choices of grouping norm q. Gaussian dictionaries with coherence $\rho = 0.1$ are used, where $C = 3$ active groups and SNR $= 40$ dB. Here, the lower endpoint $q = 1$ corresponds to the regular SPICE method, and subsequently $q = 2$ gives the standard ℓ_2-norm commonly used in group-sparse regression.

10 elements of these, randomly chosen, are present in the signal. The result can be seen in Figures 12 - 13, where it is clear that estimation performance is maximized for $q = 1.6$, i.e., $r = 4$. Note however that even with 1/8 to 1/4 sparse blocks as in this scenario, the performance degradation is not large at the upper endpoint. Therefore, the selection of r seems not to be critical, and choosing the upper endpoint, i.e., $r \to \infty$, always seems to be a reasonable initial choice. In fact, as seen in the previous scenarios, this is also often the best choice.

6.2 Multi-pitch signals

We proceed to examine results for group-SPICE when applied to the multi-pitch estimation problem, i.e., when each noise-free group is assumed to be on the form

$$\mathbf{s}_c(t) = \sum_{\ell=1}^{L_c} x_{\vartheta_c,\ell}\, e^{i2\pi\vartheta_c \ell t/f_s} \tag{78}$$

for $t = 1, \ldots, N$, for some fundamental frequency ϑ_c and sampling frequency f_s, thus being a sum of complex exponentials with frequencies at an integer multiplicity of the fundamental, being the harmonics of the pitch signal. The sparse modeling approach to this problem is to linearize the non-linear signal model (a sum of groups each parametrized in (78)), by defining a grid of possible fundamental frequencies, $\vartheta_k, k = 1, \ldots, K$, and a group size $L \geq \max_c L_c$, from which the dictionary is constructed. Thus, we choose the regressor $\mathbf{a}_{k,\ell}$ as the ℓ:th harmonic of the k:th candidate pitch, i.e.,

$$\mathbf{a}_{k,\ell} = \frac{1}{\sqrt{N}} \begin{bmatrix} e^{i2\pi\vartheta_k \ell \cdot 1} & \ldots & e^{i2\pi\vartheta_k \ell \cdot N} \end{bmatrix}^\top \tag{79}$$

where the scaling by $\sqrt{N}$ gives the regressor unit ℓ_2-norm, i.e., $\left\|\mathbf{a}_{k,\ell}\right\|_2 = 1$. In the simulation, we examine a mix of $C = 3$ pitch signals with fundamental frequencies uniformly selected from the continuous interval $\Theta = \{\vartheta : 100 \leq \vartheta < 400\}$ Hz. We select the dictionary to parametrize candidate pitches with $K = 300$ fundamental frequencies uniformly spaced on Θ. The true fundamentals are thus always somewhat off-grid, and can therefore partly be modeled by either of its neighboring fundamentals present in the dictionary. To account for such off-grid effects in the performance metrics, we use the approximate recovery rate (ARR), defined as

$$\mathrm{ARR}^{(i)} = 1\left\{\left|\hat{\mathcal{I}}_C^{(i)} - \mathcal{I}_C^{(i)}\right| \leq \delta\right\} \tag{80}$$

for some limit $\delta \in \mathbb{N}$. For example, one may choose $\delta = 1$ if a match to either of a pitch's two closest grid points is deemed acceptable. In the simulation scenario, we set $\delta = 1$, as well as SNR $= 20$ dB, $N = 200$, $f_s = 8$ kHz, $L = 10$, and where the true number of harmonics in each pitch is randomly chosen between $\lceil L/2 \rceil$ and L.

The results can be seen in Figures 14 - 15 with respect to ARR and HMD, respectively, illustrating how the group-SPICE estimator performs on par with

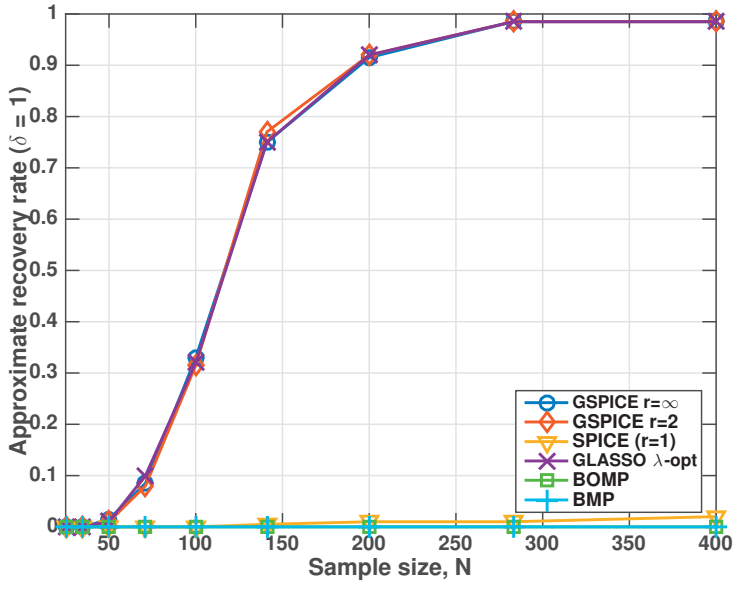


Figure 14: Approximate recovery rates (for recoveries ± 1 gridpoint from ground truth) from 200 Monte-Carlo samples estimated with group-SPICE and comparable estimators at different sample lengths. Here, we have sampled a mix of $C = 3$ pitch signals from the multi-pitch model, with fundamental frequencies randomly chosen on the interval $[100, 400)$ Hz, measured in noise with SNR $= 40$ dB. We thus make use of a multi-pitch dictionary with $K = 300$ candidate fundamentals, uniformly spaced on $[100, 400]$ Hz, where each pitch group contains $L = 10$ harmonics.

the ideally regularized group-LASSO, whereas standard SPICE and group-sparse greedy estimators completely fail to find the true support. Most likely, SPICE, BMP, and BOMP failed estimations are due to the inherently large block-coherence of the multi-pitch dictionary, as illustrated in Figure 16, for $N = 25, 100$, and 400. It can be seen that whereas μ_b decreases when N grows, the ratio between on-diagonal and off-diagonal elements remains unchanged, illuminating the difficulty of the multi-pitch estimation problem. Finally, we conclude the numerical section by a small example of how well group-SPICE works for multi-pitch signals recorded *in natura*. Here, we use a rather simple signal, namely a mixed recording of three trumpets, playing the musical notes A4 (≈ 440 Hz), B4 (≈ 493.883 Hz),

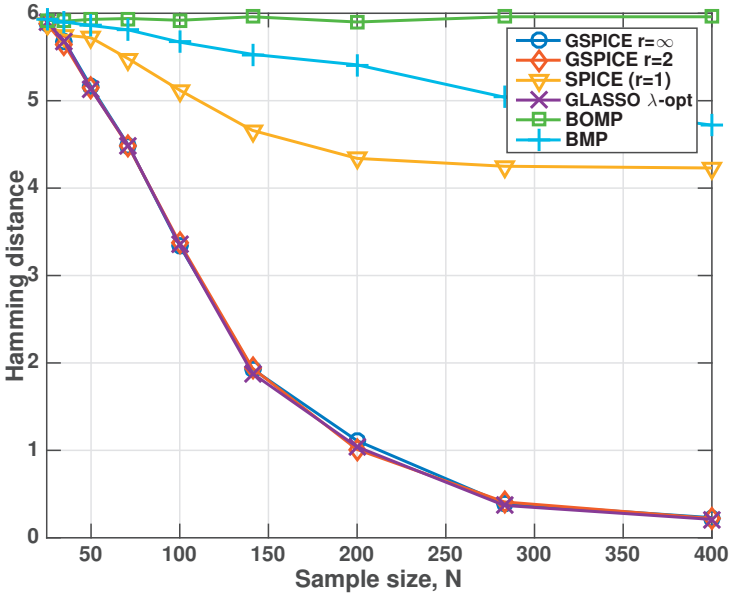


Figure 15: Hamming distances from 200 Monte-Carlo samples estimated with group-SPICE and comparable estimators at different sample lengths. Here, we have sampled a mix of $C = 3$ pitch signals from the multi-pitch model, with fundamental frequencies randomly chosen on the interval $[100, 400)$ Hz, measured in noise with SNR $= 40$ dB. We thus make use of a multi-pitch dictionary with $K = 300$ candidate fundamentals, uniformly spaced on $[100, 400]$ Hz, where each pitch group contains $L = 10$ harmonics.

and C♯5 (≈ 554.365 Hz), respectively, for a duration of less than five seconds. The signal is decimated from 44 kHz to $f_s = 8$ kHz to reduce complexity, which, as its spectral content is largely confined to lower frequencies, typically does not effect estimation performance noticably. Estimation is performed on individual 30 ms frames ($N = 240$) which are 50 % overlapping, using a dictionary with $K = 350$ candidate fundamental frequencies uniformly spaced on the interval $[100, 800)$ Hz, with $L = 20$. We have, however, limited the group size for each candidate group individually, such that no frequency in the dictionary becomes larger than the Nyquist sampling frequency $f_s/2$, e.g., $L_K = \lfloor f_s/\vartheta_K \rfloor = 5$ for the largest fundamental, where $\lfloor \cdot \rfloor$ denotes the flooring operator. Thus, the group sizes, L_k, vary from $L_1 = 20$ down to $L_K = 5$. The estimation results can be

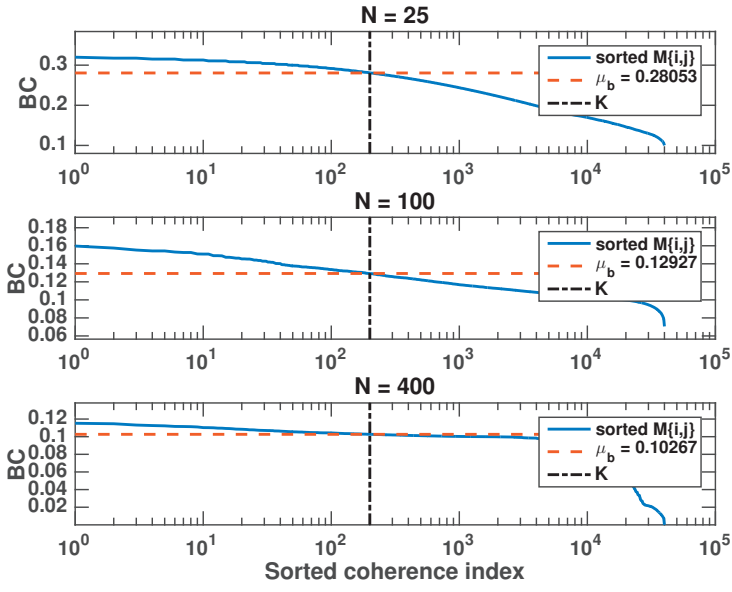


Figure 16: Plot of block coherence (BC) elements in the vector vec(**M**), sorted from high to low, where the first K elements correspond to the diagonal elements of **M**, indicated by the vertical dash-dotted line, whereafter the last $K^2 - K$ elements correspond to the off-diagonal coherences. The BC measure μ_b corresponds to the largest off-diagonal element, i.e., the $(K+1)$:th element in the plot, indicated by the horizontal dashed line.

seen in Figures 17 - 18, illustrating the estimates of group-SPICE ($r = \infty$ and $r = 4$), group-LASSO, SPICE, BOMP, and BMP. For the group-LASSO, the regularization level cannot be chosen as done earlier due to the noise variance being unknown. Instead, we here chose it such that the largest dynamic range of the estimates becomes $\eta = 30$ dB, i.e., the difference in power between the strongest and the weakest non-zero group in the estimate. Thus, we set $\lambda = \lambda_{\max}\sqrt{10^{-\eta/10}}$, where $\lambda_{\max}$ is the smallest regularization level which gives the null solution. It becomes apparent from the figures that the multi-pitch estimation problem is not trivial; the group-LASSO should be able to give a solution comparable to group-SPICE, but here it clearly needs further tuning of the hyperparameter to give satisfying estimates. Instead it exhibits the so-called suboctave error, where half or the quarter fundamental frequency is found in lieu of the true one. The standard

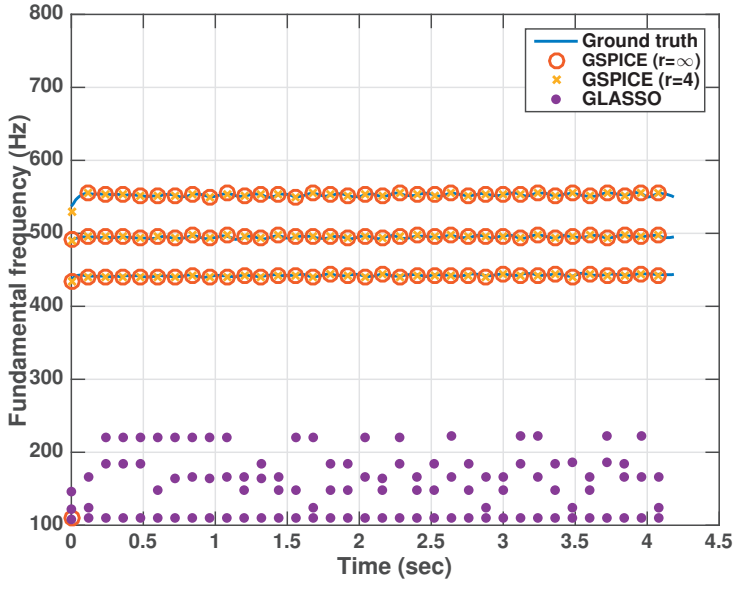


Figure 17: Fundamental frequency tracks from estimation on a mix with three trumpets playing three different musical notes. Ground truth is illustrated by solid lines, whereas estimates are, for clarity of presentation, given by markers for every 8:th estimated frame.

SPICE also perform poorly, the lack of grouping results in unpredictable estimates where dominant individual frequencies are preferred to pitches. The greedy estimators perform likewise; BMP also illustrate suboctave errors, whereas BOMP entirely fails to function with the highly coherent dictionary. In conclusion, for this example only group-SPICE is found to deliver high performance estimates.

It is also worth noting that there exists a myriad of estimators specifically created for, and fine-tuned to solve the multi-pitch problem (see, e.g., [59, 60] for an overview). In this paper, we have confined ourselves to comparisons with more general estimators based on sparse regression, and a detailed comparison with current state-of-the-art multi-pitch estimators is thus beyond the scope of this work.

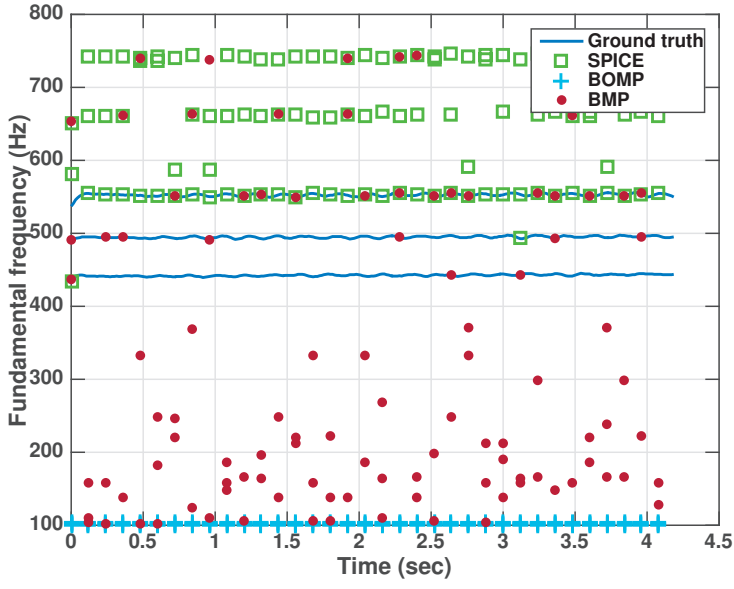


Figure 18: Fundamental frequency tracks from estimation on a three trumpet mix. Ground truth is illustrated by solid lines, whereas estimates are, for clarity of presentation, given by markers for every 8:th estimated frame.

7 Conclusions

In this work, we have presented the hetero- and homoscedastic group-SPICE methods, and propose using them for group-sparse regression problems, as they circumvent cumbersome model order estimation, while being hyperparameter-free. We have also shown the connection between homoscedastic group-SPICE and the SR group-LASSO, as well as between the heteroscedastic group-SPICE and the LAD group-LASSO, thereby endowing these with optimal regularization strategies. Furthermore, the group-SPICE formulation allows, without necessitating, the setting of a hyperparameter which improves performance when sparsity also occurs within the active groups. From simulation results, we have illustrated how group-SPICE shows robustness against dictionary coherence, achieving performance as good as the group-LASSO when allowed oracle regularization, and far outperforms comparable greedy estimation methods. We have also verified

these results when applied to the multi-pitch estimation problem, both for synthetic and recorded audio data.

References

[1] M. Elad, *Sparse and Redundant Representations*. Springer, 2010.

[2] D. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, pp. 1289–1306, 2006.

[3] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman and Hall/CRC, 2015.

[4] D. Donoho and I. M. Johnstone, "Ideal Spatial Adaptation by Wavelet Shrinkage," *Biometrika*, pp. 425–455, 1994.

[5] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[6] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic Decomposition by Basis Pursuit," *SIAM Review*, vol. 43, pp. 129–159, 2001.

[7] D. Donoho, M. Elad, and V. Temlyakov, "Stable Recovery of Sparse Overcomplete Representations in the Presence of Noise," *IEEE Transactions on Information Theory*, vol. 52, pp. 6–18, Jan 2006.

[8] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet D2ecomposition," in *Signals, Systems and Computers. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pp. 40–44 vol.1, Nov 1993.

[9] J. Fan and R. Li, "Variable selection via non-concave penalized likelihood and its oracle properties," *Journal of the Amer. Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, 2001.

[10] J. Tropp, "Just Relax: Convex Programming Methods for Identifying Sparse Signals in Noise," *IEEE Transactions on Information Theory*, vol. 52, pp. 1030–1051, March 2006.

[11] E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, vol. 52, pp. 489–509, Feb. 2006.

[12] H. Zou and T. Hastie, "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[13] I. CVX Research, "CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta." `http://cvxr.com/cvx`, Sept. 2012.

[14] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, (Seoul), pp. 10225–10229, July 2008.

[15] B. Ottersten, P. Stoica, and R. Roy, "Covariance matching estimation techniques for array signal processing applications," *Digit. Signal Process.*, vol. 8, pp. 185–210, 1998.

[16] P. Stoica, P. Babu, and J. Li, "New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data," *IEEE Transactions on Signal Processing*, vol. 59, pp. 35–47, Jan 2011.

[17] P. Stoica, P. Babu, and J. Li, "SPICE : a novel covariance-based sparse estimation method for array processing," *IEEE Transactions on Signal Processing*, vol. 59, pp. 629 –638, Feb. 2011.

[18] I. F. Gorodnitsky and B. D. Rao, "Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm," *IEEE Transactions on Signal Processing*, vol. 45, pp. 600–616, March 1997.

[19] D. Malioutov, M. Cetin, and A. S. Willsky, "A Sparse Signal Reconstruction Perspective for Source Localization With Sensor Arrays," *IEEE Transactions on Signal Processing*, vol. 53, pp. 3010–3022, August 2005.

[20] X. Tan, W. Roberts, J. Li, and P. Stoica, "Sparse Learning via Iterative Minimization With Application to MIMO Radar Imaging," *IEEE Transactions on Signal Processing*, vol. 59, pp. 1088–1101, March 2011.

[21] R. Gribonval and E. Bacry, "Harmonic decomposition of audio signals with matching pursuit," *IEEE Transactions on Signal Processing*, vol. 51, pp. 101–111, jan. 2003.

[22] M. D. Plumbley, S. A. Abdallah, T. Blumensath, and M. E. Davies, "Sparse representations of polyphonic music," *Signal Processing*, vol. 86, pp. 417–431, March 2006.

[23] P. Babu, *Spectral Analysis of Nonuniformly Sampled Data and Applications*. PhD thesis, Uppsala University, 2012.

[24] C. R. Rojas, D. Katselis, and H. Hjalmarsson, "A Note on the SPICE Method," *IEEE Transactions on Signal Processing*, vol. 61, pp. 4545–4551, Sept. 2013.

[25] P. Stoica, D. Zachariah, and L. Li, "Weighted SPICE: A Unified Approach for Hyperparameter-Free Sparse Estimation," *Digit. Signal Process.*, vol. 33, pp. 1–12, October 2014.

[26] O. Arslan, "Weighted LAD-LASSO Method for Robust Parameter Estimation and Variable Selection in Regression," *Computational Statistics & Data Analysis*, vol. 56, no. 6, pp. 1952 – 1965, 2012.

[27] A. Belloni, V. Chernozhukov, and L. Wang, "Square-Root LASSO: Pivotal Recovery of Sparse Signals via Conic Programming," *Biometrika*, vol. 98, no. 4, pp. 791–806, 2011.

[28] S. Bakin, *Adaptive regression and model selection in data mining problems*. PhD thesis, School of Mathematical Sciences, Australian National University, 1999.

[29] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[30] X. Lv, G. Bi, and C. Wan, "The Group Lasso for Stable Recovery of Block-Sparse Signal Representations," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1371–1382, 2011.

[31] M. Stojnic, F. Parvaresh, and B. Hassibi, "On the Reconstruction of Block-Sparse Signals with an Optimal Number of Measurements," *IEEE Transactions on Signal Processing*, vol. 57, no. 8, pp. 3075–3085, 2009.

[32] P. Zhao, G. Rocha, and B. Yu, "The Composite Absolute Penalties for Grouped and Hierarchical Variable Selection," *The Annals of Statistics*, vol. 39, pp. 3468–3497, 2009.

[33] L. Meier, S. van de Geer, and P. Buhlman, "The Group Lasso for Logistic Regression," *Journal of the Royal Statistical Society, Series B*, 2008.

[34] G. Obozinski, M. J. Wainwright, and M. I. Jordan, "Support Union Recovery in High-Dimensional Multivariate Regression," *The Annals of Statistics*, vol. 39, pp. 1–47, 2011.

[35] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A Sparse-Group Lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[36] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, pp. 91–108, January 2005.

[37] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[38] F. Elvander, T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization," *Elsevier Signal Processing*, vol. 127, pp. 56–70, October 2016.

[39] S. I. Adalbjörnsson, T. Kronvall, S. Burgess, K. Åström, and A. Jakobsson, "Sparse Localization of Harmonic Audio Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 117–129, January 2016.

[40] T. Kronvall, M.Juhlin, J. Swärd, S. Adalbjörnsson, and A. Jakobsson, "Sparse modeling of chroma features," *Signal Process.*, vol. 130, pp. 105–117, Jan. 2017.

[41] L. Jacob, G. Obozinski, and J.-P. Vert, "Group Lasso with Overlap and Graph Lasso," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, (New York, NY, USA), pp. 433–440, ACM, 2009.

[42] E. Grave, G. Obozinski, and F. Bach, "Trace Lasso: A Trace Norm Regularization for Correlated Designs," in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, (USA), pp. 2187–2195, Curran Associates Inc., 2011.

[43] M. Osborne, B. Presnell, and B. Turlach, "A New Approach to Variable Selection in Least Squares Problems," *IMA Journal of Numerical Analysis*, vol. 20, no. 3, pp. 389–403, 2000.

[44] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, pp. 407–499, April 2004.

[45] T. Kronvall, F. Elvander, S. Adalbjörnsson, and A. Jakobsson, "Multi-Pitch Estimation via Fast Group Sparse Learning," in *24rd European Signal Processing Conference*, (Budapest, Hungary), 2016.

[46] C. D. Austin, R. L. Moses, J. N. Ash, and E. Ertin, "On the Relation Between Sparse Reconstruction and Parameter Estimation With Model Order Selection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 560–570, 2010.

[47] X. Xu and M. Ghosh, "Bayesian Variable Selection and Estimation for Group Lasso," *Bayesian Anal.*, vol. 10, pp. 909–936, 12 2015.

[48] P. Stoica and P. Babu, "SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation," *Signal Processing*, vol. 92, pp. 1580–1590, July 2012.

[49] Y. Yu, "Monotonic Convergence of a General Algorithm for Computing Optimal Designs," *The Annals of Statistics*, vol. 38, pp. 1593–1606, 2010.

[50] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[51] F. Bunea, J. Lederer, and Y. She, "The Group Square-Root Lasso: Theoretical Properties and Fast Algorithms," *IEEE Trans. Inf. Theor.*, vol. 60, pp. 1313–1325, Feb. 2014.

[52] P. Stoica and P. Babu, "Sparse Estimation of Spectral Lines: Grid Selection Problems and Their Solutions," *IEEE Transactions on Signal Processing*, vol. 60, pp. 962–967, Feb. 2012.

[53] P. Stoica and R. Moses, *Spectral Analysis of Signals*. Upper Saddle River, N.J.: Prentice Hall, 2005.

[54] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*. Englewood Cliffs, N.J.: Prentice-Hall, 1993.

[55] J. Friedman, T. Hastie, and R. Tibshirani, "A Note on the Group Lasso and a Sparse Group Lasso." Unpublished manuscript, 2010.

[56] E. Elhamifar and R. Vidal, "Block-Sparse Recovery via Convex Optimization," *IEEE Transactions on Signal Processing*, vol. 60, pp. 4094–4107, Aug 2012.

[57] Y. V. Eldar, P. Kuppinger, and H. Bolcskei, "Block-Sparse Signals: Uncertainty Relations and Efficient Recovery," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, 2010.

[58] L. Peotta and P. Vandergheynst, "Matching Pursuit with Block Incoherent Dictionaries," *IEEE Transactions on Signal Processing*, vol. 55, pp. 4549–4557, Sept 2007.

[59] M. G. Christensen, P. Stoica, A. Jakobsson, and S. H. Jensen, "Multi-pitch estimation," *Signal Processing*, vol. 88, pp. 972–983, April 2008.

[60] M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard, "Signal Processing for Music Analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1088–1110, 2011.

E

Paper E

Online Group-Sparse Estimation Using the Covariance Fitting Criterion

Ted Kronvall, Stefan Ingi Adalbjörnsson, Santhosh Nadig, and Andreas Jakobsson

Abstract

In this paper, we present a time-recursive implementation of a recent hyperparameter-free group-sparse estimation technique. This is achieved by reformulating the original method, termed group-SPICE, as a square-root group-LASSO with a suitable regularization level, for which a time-recursive implementation is derived. Using a proximal gradient step for lowering the computational cost, the proposed method may effectively cope with data sequences consisting of both stationary and non-stationary signals, such as transients, and/or amplitude modulated signals. Numerical examples illustrates the efficacy of the proposed method for both coherent Gaussian dictionaries and for the multi-pitch estimation problem.

Keywords: Online estimation, covariance fitting, group sparsity, multi-pitch estimation.

1 Introduction

Estimating a sparse parameter support for a high-dimensional regression problem has been the focus of much scientific attention during the last two decades, as this methodology has shown its usefulness in many applications, ranging from spectral analysis [1–3], array- [4–6] and audio processing [7–9], to biomedical modeling [10], and magnetic resonance imaging [11, 12]. In its vanguard, notable contributions were done by, among others, Donoho et al. [13] and Tibshirani et al. [14]. Their methods are effectively equivalent but are termed differently; the basis pursuit de-noising (BPDN) and the least absolute selection and shrinkage operator (LASSO), respectively, are nowadays a common component in the standard scientific toolboxes. These methods will estimate a parameter vector which reconstructs the signal using only a small number of regressors from the regressor matrix, i.e., a small number of columns from an (often highly underdetermined) linear system. More recently, a methodology termed the group-LASSO [15] was developed for modeling a signal where the sparse parameter support is assumed to be clustered into pre-defined groups, with the justification that some signal sources are better modeled by a group of regressors rather than just one. The above mentioned methods, as well as the vast majority of sparse estimators, have in common the requirement of selecting one or several hyperparameters, controlling the degree of sparsity in the solution. This may be done using, e.g., application-specific heuristics, cross-validation, or using some information criteria, which may often be computational burdensome and/or inaccurate. The discussed sparse estimation approaches typically assume having access to one or more offline frames of data, each having time-stationary signal support. For many applications, such as, for instance, audio processing, data is often generated online, with large correlation between consecutive frames, and with a varying degree of non-stationarity. To better accommodate these conditions, one may use a sparse recursive least squares (RLS) approach (see, e.g., [16, 17]), such as the one derived in [18] for the multi-pitch estimation problem. In a recent effort, the sparse iterative covariance-based estimator (SPICE) [19] utilizes a criteria for covariance fitting, originally developed within array processing, to form sparse estimates without the need of selecting hyperparameters. In fact, SPICE may shown to be equivalent to the square root (SR) LASSO [20]; in a covariance fitting sense, SPICE may as a result be viewed as the, in some sense, optimal selection of the SR LASSO hyperparameter [21]. In this paper, we extend the method proposed in [22], which generalizes SPICE for grouped variables, along the lines of [23] to

form recursive estimates in an online-fashion, reminiscent to the approach used in [24]. By first reformulating group-SPICE as an SR-LASSO, we then derive an efficient method for sparse recursive estimation formed via proximal gradient iterations, enabling recursive estimation of non-stationary signals. We justify the proposed method accordingly by numerical examples, illustrating its performance as on par with group-SPICE for stationary signals, and outperforming an online SPICE for group-sparse non-stationary signals.

2 Notational conventions

In this paper, we use the mathematical convention of letting lower-case letters, e.g., y, denote scalars, while lower-case bold-font letters, $\mathbf{y}$, denote column vectors and upper-case bold-font letters, $\mathbf{Y}$, denote matrices. Furthermore, E denotes the expectation operator, ∇ the first order derivative, and $(\cdot)^\top$ and $(\cdot)^H$ the transpose and hermitian transpose, respectively. Also, $|\cdot|$ denotes the absolute value of a complex number, while $\|\cdot\|_q$ and $\|\cdot\|_F$ denotes the ℓ_q-norm for $q \geq 1$ and the Frobenius norm, respectively. We let $\mathrm{diag}(\mathbf{a})$ denote the diagonal matrix with diagonal vector $\mathbf{a}$, and $\mathrm{tr}(\mathbf{A})$ the matrix trace of $\mathbf{A}$. We describe the structure of a matrix or vector by ordering elements within hard brackets, e.g., $\mathbf{y} = \begin{bmatrix} y(1) & y(2) \end{bmatrix}^\top$, while a set of elements is described using curly brackets, e.g., $\mathbb{N} = \{1, 2, \ldots\}$ denotes the set of natural numbers. We use subscripts to denote a subgroup of a vector or matrix, while time indices are indicated within parentheses, e.g., $\mathbf{x}_k(t)$ denotes the variables in subgroup $\mathbf{x}_k$ at time t. Superscript typically denotes a power operation, except for when the exponent is within parentheses, e.g., $x^{(j)}$, which denotes the j:th iteration of x. Finally, we also make use of notations $(x)_+ = \max(0, x)$, $\mathrm{sign}(\mathbf{x}) = \mathbf{x}/\|\mathbf{x}\|_2$, and $x \in \mathrm{Bin}(n, p)$, where the latter denotes that x is binomally distributed with n independent trials and probability parameter p.

3 Group-sparse estimation via the covariance fitting criterion

Here, we consider an N sample signal frame which may be reasonably well approximated by a select few variables in the linear signal model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{1}$$

where $\mathbf{A} \in \mathbb{C}^{N \times M}$ and $\mathbf{x} \in \mathbb{C}^{M}$ denote the regressor matrix (or dictionary) and the response variable vector, respectively, and where $\mathbf{e}$ denotes the approximation error and noise. In our signal model, we assume that a possible signal source is represented by a sum of column vectors from the dictionary rather than just one, such that it may be clustered into K predefined groups,

$$\mathbf{A} = \left[\begin{array}{ccc} \mathbf{A}_1 & \ldots & \mathbf{A}_K \end{array} \right] \tag{2}$$

$$\mathbf{A}_k = \left[\begin{array}{ccc} \mathbf{a}_{k,1} & \ldots & \mathbf{a}_{k,L_k} \end{array} \right] \tag{3}$$

where the k:th group has L_k basis vectors, and consequently the dictionary has altogether $M = \sum_{k=1}^{K} L_k$ columns. By construction, we consider a group-sparse regression problem, where only a small number of the K possible groups are represented in the signal. We assume that $\mathbf{e}$ is reasonably homoscedastic, i.e., $E(\mathbf{e}\mathbf{e}^H) = \sigma\mathbf{I}$, as well as that the variables in $\mathbf{x}$ are independent and identically distributed with a random phase, uniformly distributed over $[0, 2\pi)$. The covariance matrix may thus be expressed as

$$\mathbf{R} = E(\mathbf{y}\mathbf{y}^H) = \mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma\mathbf{I} \tag{4}$$

where $\mathbf{P}$ is a diagonal matrix with the diagonal vector

$$\mathbf{p} = \left[\begin{array}{ccc} \mathbf{p}_1 & \ldots & \mathbf{p}_K \end{array} \right]^\top \tag{5}$$

$$\mathbf{p}_k = \left[\begin{array}{ccc} p_{k,1} & \ldots & p_{k,L_k} \end{array} \right]^\top \tag{6}$$

which corresponds to the squared magnitude of the response variables, i.e.,

$$p_{k,\ell} = |x_{k,\ell}|^2 \tag{7}$$

for the ℓ:th component in the k:th dictionary group. To account for the group-sparse structure, we have relaxed the orginal covariance fitting criterion used in [19] by following the lines of [22] and thus seek to minimize

$$g(\mathbf{p}, \sigma) = \mathbf{y}^H \left(\mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma\mathbf{I} \right)^{-1} \mathbf{y} + \sum_{k=1}^{K} v_k \left\| \mathbf{p}_k \right\|_\infty \tag{8}$$

with respect to the unknown variables $\mathbf{p}$ and σ, where

$$v_k = \sqrt{\mathrm{tr}(\mathbf{A}_k^H \mathbf{A}_k)} = \left\| \mathbf{A}_k \right\|_F \tag{9}$$

Following the derivations in [22], minimizing (8) with respect to $\mathbf{p}$ and σ is equivalent of minimizing

$$g(\mathbf{x}) = \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 + \sum_{k=1}^{K} \sqrt{\frac{v_k}{N}} \|\mathbf{x}_k\|_2 \tag{10}$$

with respect to the original variable $\mathbf{x}$, which is a square root group-LASSO [25] with the regularization parameter individually set for each group as $\sqrt{v_k/N}$.

4 Recursive estimation via proximal gradient

To allow for a recursive estimation of $\mathbf{x}$, which can be improved or changed adaptively as new samples are added, let $\mathbf{x}(n)$ denote a linear filter for some time point $n \in \{1 \leq n \leq N\}$. Also, we reformulate the first term in (10), here denoted $q(\cdot)$, such that a forgetting factor, $0 < \lambda \leq 1$, is utilized to give older samples less importance than newer samples, i.e.,

$$q(\mathbf{y}(n), \mathbf{x}(n)) = \sqrt{\sum_{t=1}^{n} \lambda^{n-t} |y(t) - \boldsymbol{\alpha}(t)^\top \mathbf{x}(t)|^2} \tag{11}$$

where $\mathbf{y}(n)$ and $\boldsymbol{\alpha}(t)^\top$ denote the vector of samples up to n and the t:th row of $\mathbf{A}$, respectively. On matrix form, $q(\cdot)$ may be equivalently formulated as

$$q(\mathbf{y}(n), \mathbf{x}(n)) = \left\| \sqrt{\boldsymbol{\Lambda}(n)} \Big(\mathbf{y}(n) - \mathbf{A}(n)\mathbf{x}(n) \Big) \right\|_2 \tag{12}$$

where $\mathbf{A}(n) = \begin{bmatrix} \boldsymbol{\alpha}(1) & \dots & \boldsymbol{\alpha}(n) \end{bmatrix}^\top$ denotes the first n rows in $\mathbf{A}$, and where $\boldsymbol{\Lambda}(n) = \text{diag}(\begin{bmatrix} \lambda^{n-1} & \dots & \lambda^0 \end{bmatrix})$. Our aim is to implement a proximal gradient algorithm reminiscent of [26] to estimate $\mathbf{x}(n), \forall n$. To that end, one may iteratively upper-bound $q(\cdot)$ by centering it around the previous iteration's estimate, $\mathbf{x}^{(j-1)}(n)$, i.e.,

$$\begin{aligned} q(\mathbf{y}(n), \mathbf{x}^{(j)}(n)) \leq\ & q(\mathbf{y}(n), \mathbf{x}^{(j-1)}(n)) \\ & + (\mathbf{x}^{(j)}(n)) - \mathbf{x}^{(j-1)}(n)))^T \nabla q(\mathbf{y}(n), \mathbf{x}^{(j)}(n))) \\ & + \frac{1}{2h} \left\| \mathbf{x}^{(j)}(n)) - \mathbf{x}^{(j-1)}(n)) \right\|_2^2 \end{aligned} \tag{13}$$

Algorithm 1 The proposed online group-SPICE algorithm

Intitiate $n \leftarrow 0, \mathbf{R} \leftarrow \mathbf{0}, \mathbf{r} \leftarrow \mathbf{0}, \gamma \leftarrow 0$ and set $\mathbf{x}(n) = \mathbf{0}$
while $n < (N - \tau)$ **do**
  Reset $j \leftarrow 0$ and warm start $\mathbf{u}^{(j)} \leftarrow \mathbf{x}(n)$
  Add τ new samples and set $n \leftarrow n + \tau$
  Update $\mathbf{R}$, $\mathbf{r}$, and γ using (23)
  repeat {proximal gradient iterations}
    Update gradient $\nabla q(\mathbf{y}(n), \mathbf{u}^{(j)}))$ using (18)
    Take a gradient step, from $\mathbf{u}^{(j)}$ to $\mathbf{z}$, using (16)
    Apply group-wise shrinkage $\mathbf{u}_k^{(j+1)}$ using (15)
    $j \leftarrow j + 1$
  until convergence
  Save $\mathbf{x}(n) = \mathbf{u}_k^{(j)}$
end while

for some step size $h > 0$, and instead of minimizing (10) one may equivalently instead iteratively minimize [26]

$$\tilde{g} = \frac{1}{2h} \left\| \mathbf{x}^{(j)}(n)) - h\nabla q(\mathbf{y}(n), \mathbf{x}^{(j-1)}(n)) \right\|_2^2 + \sum_{k=1}^{K} \mu_k \left\| \mathbf{x}_k^{(j)}(n) \right\|_2 \tag{14}$$

for a suitable choice of regularization μ_k. By solving the subgradient equations of (14), previously shown in, e.g., [27] and here omitted due to page restrictions, one obtains the closed-form solution for the k:th group as

$$\mathbf{x}_k^{(j)} = \left(\|\mathbf{z}_k\| - h\mu_k \right)_+ \operatorname{sign}(\mathbf{z}_k) \tag{15}$$

where $\mathbf{z} = \begin{bmatrix} \mathbf{z}_1^\top & \dots & \mathbf{z}_K^\top \end{bmatrix}^\top$ is formed as

$$\mathbf{z} = \mathbf{x}^{(j-1)}(n) - h\nabla q \left(\mathbf{y}(n), \mathbf{x}^{(j-1)}(n) \right) \tag{16}$$

where the gradient ∇q becomes

$$\nabla q(\mathbf{y}(n), \mathbf{x}(n)) = \frac{-\mathbf{A}(n)^H \mathbf{\Lambda}(n) \left(\mathbf{y}(n) - \mathbf{A}(n)\mathbf{x}(n) \right)}{\left\| \sqrt{\mathbf{\Lambda}(n)} \left(\mathbf{y}(n) - \mathbf{A}(n)\mathbf{x}(n) \right) \right\|_2} \tag{17}$$

wherein the superscript of $\mathbf{x}^{(j-1)}(n)$ was temporarily omitted for notational convenience.

5 Efficient recursive updates for new samples

One may facilitate an efficient estimation process when new samples are introduced by reusing old computations. To that end, the derivative (17) may be expressed as

$$\frac{\mathbf{R}(n)\mathbf{x}(n) - \mathbf{r}(n)}{\sqrt{\boldsymbol{\gamma}(n) - 2\,\Re\left(\mathbf{r}(n)^H\mathbf{x}(n)\right) + \mathbf{x}(n)\mathbf{R}(n)\mathbf{x}(n)}} \tag{18}$$

where

$$\begin{aligned}\mathbf{r}(n) &= \mathbf{A}(n)^H\boldsymbol{\Lambda}(n)\mathbf{y}(n)\\ \mathbf{R}(n) &= \mathbf{A}(n)^H\boldsymbol{\Lambda}(n)\mathbf{A}(n)\\ \boldsymbol{\gamma}(n) &= \mathbf{y}(n)^H\boldsymbol{\Lambda}(n)\mathbf{y}(n)\end{aligned} \tag{19}$$

Let $\left[\begin{array}{ccc} y(n+1) & \dots & y(n+\tau)\end{array}\right]^\top$, $\tau \in \mathbb{N}$ denote a vector of τ new samples available for estimation, and $(+\tau)$ the time indices from $n+1$ to $n+\tau$. Then,

$$\mathbf{y}(n+\tau) = \left[\begin{array}{cc}\mathbf{y}(n)^\top & \mathbf{y}(+\tau)^\top\end{array}\right]^\top \tag{20}$$

$$\mathbf{A}(n+\tau) = \left[\begin{array}{c}\mathbf{A}(n)\\ \mathbf{A}(+\tau)\end{array}\right] \tag{21}$$

$$\boldsymbol{\Lambda}(n+\tau) = \left[\begin{array}{cc}\lambda^\tau\boldsymbol{\Lambda}(n) & \mathbf{0}\\ \mathbf{0}^\top & \boldsymbol{\Lambda}(\tau)\end{array}\right] \tag{22}$$

which, if inserted into (19), yields the updating formulas

$$\begin{aligned}\mathbf{r}(n+\tau) &= \lambda^\tau\mathbf{r}(n) + \mathbf{A}(+\tau)^H\boldsymbol{\Lambda}(\tau)\mathbf{y}(+\tau)\\ \mathbf{R}(n+\tau) &= \lambda^\tau\mathbf{R}(n) + \mathbf{A}(+\tau)^H\boldsymbol{\Lambda}(\tau)\mathbf{A}(+\tau)\\ \boldsymbol{\gamma}(n+\tau) &= \lambda^\tau\boldsymbol{\gamma}(n) + \mathbf{y}(+\tau)^H\boldsymbol{\Lambda}(\tau)\mathbf{y}(+\tau)\end{aligned} \tag{23}$$

The hyperparameters, μ_k, from (15), are in (10) defined as $\mu_k = \sqrt{v_k/N}$. In a time-recursive scheme, however, when new samples are added and older samples are given smaller importance, one must choose μ_k accordingly. As the sample

size and dictionary matrix increase, governed the forgetting factor, one may select $\mu_k(n)$ as

$$\mu_k(n) = \sqrt{\frac{\sqrt{\mathrm{tr}\big(\mathbf{R}_{k,k}(n)\big)}}{(\lambda^n - 1)/(\lambda - 1)}} \tag{24}$$

where the denominator results from the geometric sum $\sum_{t=0}^{n-1} \lambda^t$ and $\mathbf{R}_{k,k}(t) = \mathbf{A}_k(t)^H \mathbf{\Lambda}(n)\mathbf{A}_k(n)$, which is obtained by choosing a submatrix of the recursively updated $\mathbf{R}(t)$ with rows and columns corresponding to group k. The step size, h, may, e.g., be chosen along the lines of [27]. Algorithm 1 summarizes the proposed method, which has computational complexity $\mathcal{O}(M^2)$. The main cost occurs at line 5 and is independent of the sample size, n.

6 Numerical results

In this section, we compare the proposed estimator to relevant estimators for some different scenarios. We begin by examining the case of a cohrerent Gaussian dictionary, which is constructed by letting

$$\mathbf{a}_{k,\ell} = \sum_{\mathcal{I}^{\rho}_{k,\ell}} \mathbf{b}_{k',\ell'}, \quad \mathbf{b}_{k',\ell'} \in \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{25}$$

where $\mathbf{b}_{k,\ell}, \forall (k,\ell)$ are independent and identically distributed Gaussian vectors with zero mean and unit variance. The set $\mathcal{I}^{\rho}_{k,\ell}$ selects a mix of $n \in \mathrm{Bin}\big(M - L_k, \rho\big)$ of these vectors, the indices of which are uniformly drawn from

$$(k',\ell') \in \{k : 1 \leq k \leq K, k' \neq k\} \times \{1 \leq \ell \leq L_k\} \tag{26}$$

This results in a dictionary of mixed Gaussian regressors, with no linearly dependent components within the groups, but for every component in a group, there will on average be $(M - L_k)\rho$ components in other groups to which it is linearly dependent. The parameter ρ thus controls the degree of regressor coherence, $0 \leq \rho \leq 1$. Figure 1 verifies the stationary performance of the proposed method in comparison with the non-recursive group-SPICE, the standard SPICE, the group-LASSO with a manually optimized choice of hyperparameter, as well as the (greedy) block matching pursuit [28] and block orthogonal matching pursuit [29]. The results are based on 500 Monte-Carlo (MC) simulations of

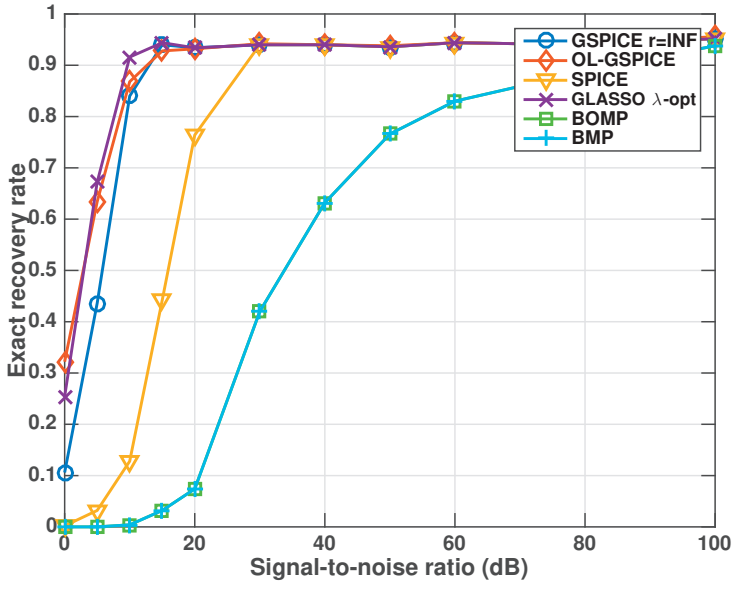


Figure 1: Exact recovery rates from 500 Monte-Carlo samples estimated with online group-SPICE (OL-GSPICE) in comparison to other methods, where an coherent Gaussian dictionary is used, with $C = 3$ active groups.

$N = 100$ samples with $C = 3$ groups, having $L_1 = L_2 = L_3 = 10$ components, randomly drawn from a dictionary with $K = 200$ blocks of size $L = 10$, with dictionary coherence $\rho = 0.1$. To measure performance, we use the exact recovery rate (ERR) metric, defined as the rate of correct support recovery, i.e.,

$$\text{ERR}^{(i)} = 1\left\{\hat{\mathcal{I}}_C^{(i)} = \mathcal{I}_C^{(i)}\right\} \tag{27}$$

for the i:th MC simulation, averaged over all simulations, where $\mathcal{I}_C^{(i)}$ and $\hat{\mathcal{I}}_C^{(i)}$ denote the true and the estimated support, respectively. To be able to make comparisons with the abovementioned stationary estimators, we use $\lambda = 1$ for the proposed method. As can be seen from the figure, the online group-SPICE performs on par with a group-LASSO, which has been given the oracle hyper-parameter, whereas SPICE (without grouping) yields significantly poorer results. Next, we examine estimation results for a multi-pitch dictionary, where the k:th

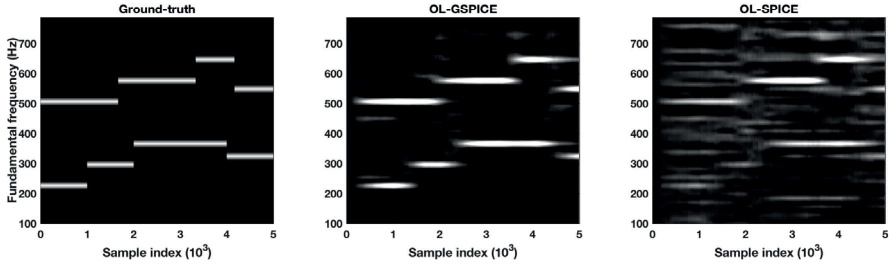


Figure 2: True parameters for a simulated non-stationary multi-pitch signal (left), with corresponding estimates of the proposed method (middle), in comparison with the online-SPICE estimator (right).

candidate dictionary group in the dictionary is

$$\mathbf{a}_k(t) = \begin{bmatrix} e^{i2\pi f_k/f_s 1 t)} & \dots & e^{i2\pi f_k/f_s L_k t} \end{bmatrix}$$

at sample point t, i.e., where the regressors are Fourier vectors with frequencies at an integer multiple of the fundamental frequency candidate f_k. Here, we simulate a non-stationary signal by allowing $C = 2$ sources to have a dynamic support changing at random locations over a frame $N = 5 \cdot 10^3$ samples. We let the dictionary contain $K = 50$ candidate fundamental frequencies, f_k, uniformly spaced on $[100,\ 800)$ Hz, with $f_s = 44$ kHz. Figure 2 illustrates the true signal (left), the estimates of the proposed estimator (middle), and the estimates of the online SPICE (right). The figure clearly shows favorable performance of the online group-SPICE, whereas online SPICE is prone to misclassification. This is likely due to the harmonic structure of the multi-pitch dictionary, making it highly coherent, with the consequence that many erroneous candidate groups partly fit the signal. It may however be noted that the estimates for the proposed method are slightly too dense, even if an ocular inspection clearly shows the two signal sources.

References

[1] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, (Seoul), pp. 10225–10229, July 2008.

[2] S. Bourguignon, H. Carfantan, and J. Idier, "A sparsity-based method for the estimation of spectral lines from irregularly sampled data," *IEEE J. Sel. Topics Signal Process.*, December 2007.

[3] J. Fang, F. Wang, Y. Shen, H. Li, and R. S. Blum, "Super-Resolution Compressed Sensing for Line Spectral Estimation: An Iterative Reweighted Approach," *IEEE Trans. Signal Process.*, vol. 64, pp. 4649–4662, September 2016.

[4] 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, pp. 600–616, March 1997.

[5] D. Malioutov, M. Cetin, and A. S. Willsky, "A Sparse Signal Reconstruction Perspective for Source Localization With Sensor Arrays," *IEEE Trans. Signal Process.*, vol. 53, pp. 3010–3022, August 2005.

[6] S. I. Adalbjörnsson, T. Kronvall, S. Burgess, K. Åström, and A. Jakobsson, "Sparse Localization of Harmonic Audio Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 117–129, January 2016.

[7] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[8] T. Kronvall, M. Juhlin, S. I. Adalbjörnsson, and A. Jakobsson, "Sparse Chroma Estimation for Harmonic Audio," in *40th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, (Brisbane), Apr. 19-24 2015.

[9] F. Elvander, T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization," *Elsevier Signal Processing*, vol. 127, pp. 56–70, October 2016.

[10] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, pp. 91–108, January 2005.

[11] A. S. Stern, D. L. Donoho, and J. C. Hoch, "NMR data processing using iterative thresholding and minimum l1-norm reconstruction," *J. Magn. Reson.*, vol. 188, no. 2, pp. 295–300, 2007.

[12] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying N-dimensional Signals," *Elsevier Signal Processing*, vol. 128, pp. 309–317, Nov 2016.

[13] D. Donoho, M. Elad, and V. Temlyakov, "Stable Recovery of Sparse Overcomplete Representations in the Presence of Noise," *IEEE Transactions on Information Theory*, vol. 52, pp. 6–18, Jan 2006.

[14] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[15] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[16] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, "Online Adaptive Estimation of Sparse Signals: Where RLS meets the ℓ_1-Norm," *IEEE Trans. Signal Process.*, vol. 58, 2010.

[17] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The Sparse RLS Algorithm," *IEEE Trans. Signal Process.*, vol. 58, pp. 4013–4025, August 2010.

[18] F. Elvander, J. Swärd, and A. Jakobsson, "Online Estimation of Multiple Harmonic Signals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, pp. 273–284, February 2017.

[19] P. Stoica, P. Babu, and J. Li, "New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data," *IEEE Trans. Signal Process.*, vol. 59, pp. 35–47, Jan 2011.

[20] A. Belloni, V. Chernozhukov, and L. Wang, "Square-Root LASSO: Pivotal Recovery of Sparse Signals via Conic Programming," *Biometrika*, vol. 98, no. 4, pp. 791–806, 2011.

[21] C. R. Rojas, D. Katselis, and H. Hjalmarsson, "A Note on the SPICE Method," *IEEE Trans. Signal Process.*, vol. 61, pp. 4545–4551, Sept. 2013.

[22] T. Kronvall, S. I. Adalbjörnsson, S. Nadig, and A. Jakobsson, "Group-Sparse Regression Using the Covariance Fitting Criterion," *Elsevier Signal Processing*, vol. 139, pp. 116 – 130, 2017.

[23] D. Zachariah and P. Stoica, "Online Hyperparameter-Free Sparse Estimation Method," *IEEE Trans. Signal Process.*, vol. 63, pp. 3348–3359, July 2015.

[24] E. Eksioglu, "Group sparse RLS algorithms," *International Journal of Adaptive Control and Signal Processing*, vol. 28, pp. 1398–1412, 2014.

[25] F. Bunea, J. Lederer, and Y. She, "The Group Square-Root Lasso: Theoretical Properties and Fast Algorithms," *IEEE Trans. Inf. Theor.*, vol. 60, pp. 1313–1325, Feb. 2014.

[26] N. Parikh and S. Boyd, "Proximal Algorithms," *Found. Trends Optim.*, vol. 1, pp. 127–239, 2014.

[27] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani, "A Sparse-Group Lasso," *Journal of Computational and Graphical Statistics*, vol. 22, no. 2, pp. 231–245, 2013.

[28] L. Peotta and P. Vandergheynst, "Matching Pursuit with Block Incoherent Dictionaries," *IEEE Transactions on Signal Processing*, vol. 55, pp. 4549–4557, Sept 2007.

[29] Y. V. Eldar, P. Kuppinger, and H. Bolcskei, "Block-Sparse Signals: Uncertainty Relations and Efficient Recovery," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, 2010.

F

Paper F

Hyperparameter-Selection for Group-Sparse Regression: A Probablistic Approach

Ted Kronvall and Andreas Jakobsson

Abstract

This work analyzes the effects on support recovery for different choices of hyper- or regularization parameters in LASSO-like sparse and group-sparse regression problems. For these problems, the hyperparameters implicitly select the model order of the solution, and are typically set using cross-validation (CV). This may be computationally prohibitive for large problems, and also often results in an overestimation of the model order, as CV optimizes the prediction error rather than the support recovery directly. In this work, we propose a probablistic approach to select the hyperparameters, specifically aimed at support recovery, by quantifying the type I error (false positive rate) using extreme value analysis. Using Monte Carlo simulations, one may draw inference on the upper tail of the distribution of the spurious parameter estimates, such that the regularization level is selected to yield an appropriate detection quantile. Solving the scaled group-LASSO problem, our proposed choice of hyperparameters becomes independent of the noise variance, which is also estimated and thus decouples from the regularization level. Furthermore, we analyze the effects on the false positive rate caused by collinearity in the dictionary, and discuss different strategies to circumvent this. The proposed method is compared to other methods for selecting the hyperparameters, in terms of the rates of support recovery, false positive rate, false negative rate, and computational complexity. Simulated data illustrate how the proposed method outperforms both cross-validation and the Bayesian Information Criterion in terms of computational complexity and support recovery.

Keywords: sparse estimation, group-LASSO, regularization, model order estimation, sparsistency, extreme value analysis, detection threshold

1 Introduction

Estimating the sparse parameter support for a high-dimensional regression problem has been the focus of much scientific attention during the past two decades, as this methodology has shown its usefulness in a wide array of applications, ranging from spectral analysis [1–3], array- [4–6] and audio processing [7–9], to biomedical modeling [10], magnetic resonance imaging [11, 12], and more. For many of these and for other applications, the retrieved data may be well explained using a highly underdetermined regression model, in which only a small subset of the explanatory variables are required to represent the data. The approach is typically referred to as sparse regression; the individual regressors are called atoms, and the entire regressor matrix the dictionary, which is typically customized for a particular application. The common approach of inferring sparsity on the estimates is to solve a regularized regression problem, i.e., appending the fit term with a regularization term that increases as variables become active (or non-zero). Much of the work in the research area springs from extensions on the seminal work by Tibshirani et al., wherein the least absolute selection and shrinkage operator (LASSO) [13] was introduced. The LASSO is a regularized regression problem where an ℓ_1-norm on the variable vector is used as regularizer, which in signal processing is also referred to as the basis pursuit denoising (BPDN) method [14]. Another early alternative to the LASSO problem is the penalized likelihood problem, introduced in [15].

In this paper, we focus on a generalization of the sparse regression problem, wherein the atoms of the dictionary exhibit some form of grouping behavior which is defined *a priori*. This follows the notion that a particular data feature is modeled not only using a single atom, but instead by a group of atoms, such that each atom has an unknown (and possibly independent) response variable, but where the entire group is assumed to be either present or not present in the data. This is achieved in the group-LASSO [16] by utilizing an ℓ_1/ℓ_2-regularizer, but other approaches have also been successful, such as in, e.g., [9, 10]. Being a generalization of the LASSO, the group-LASSO reverts back to the standard LASSO when the group sizes in the dictionary all have size one. Typically, results which hold for the group-LASSO thus also hold for the LASSO. One reason behind the success of LASSO-like approaches is that these are typically cast as a convex optimization problems, for which there exists strong theoretical results for convergence and recovery guarantees (see, e.g., [17–19], and the references therein). For convex problems, there also exist user-friendly scientific software for simple

experimentation and investigation of new regularizers [20].

The sparse regression problems described here, being a subset of the regularized regression problems, have in common the requirement of selecting one or several hyperparameters, which have the role of controlling the degree of sparsity in the solution by adjusting the level of regularization in relation to the fit term. Thus, sparsity is subject to user control, and must therefore be chosen adequately for each problem. To that end, the least angle regression (LARS) algorithm [21] calculates the entire (so called) path of solutions on an interval of values for the hyperparameter of a LASSO-like problem, and at a computational cost similar to solving the LASSO for a single value of the hyperparameter. However, by using warm-starts, a solution path may also be calculated quickly using some appropriate implementation of the group-LASSO. Even so, a single point on the solution path must still be selected; often, this is done using cross-validation (CV), as was done, for instance, in [22], for the multi-pitch estimation problem. However, due to the computationally burdensome process of CV, one often instead reverts to using less consistent heuristic approaches, or choosing the hyperparameter based on some information criteria (see, e.g., [23]). One approach to simplify the choice of hyperparameter is the scaled LASSO [24], wherein an auxilliary variable describing the standard deviation of the model residual is introduced. This has the effect that the regularization level may be selected (somewhat) independently of the noise variance, often simplifying matters for the heuristic approaches.

With a heritage from array processing, the sparse iterative covariance-based estimation (SPICE) method yields a relatively sparse parameter support by matching the observed covariance matrix and a covariance matrix parametrized by a dictionary, with an implicitly made choice of the hyperparameter. The method has been shown to work well for a variety of applications, especially those pertaining to estimation of line spectra and directions-of-arrival (see, e.g., [25]). In subsequent publications (see, e.g., [25, 26]), SPICE was shown to be equivalent to either the least absolute deviation (LAD) LASSO under a heterscedastic noise assumption, or the square root (SR) LASSO under a homoscedastic noise assumption, both for particular choices of the hyperparameter. It may be shown that the SR LASSO and the scaled LASSO are equivalent, and we conclude that SPICE is a robust (and possibly heuristic) approach of fixing the hyperparameter (somewhat) independently of the noise level. In a recent effort, the SPICE approach was extended for group sparsity [27], showing promising results, e.g., for multi-pitch estimation, but also illustrating how the fixed hyperparameter yields

estimates which are not as sparse as one may typically expect.

A valid argument in defence of the SPICE approach is that the measure of 'good' in sparse estimation is not entirely straightforward, and not sparse enough may still be good enough. Borrowing some terminology from detection theory [28], one way of measuring performance is to calculate the false negatives (FNs), i.e., whether atoms pertaining to the true support of the signal (those atoms of which the data is truly composed) are estimated as zero for some choice of the hyperparameter. As the SPICE regularization level is typically set too low, the possibility of FNs is consequently also low, which for some applications may be the focus. Conversely, for some applications, the focus may be to eliminate the false positives (FPs), i.e., when noise components are falsely set to be non-zero while not being in the true support set. The FPs and FNs are also sometimes referred to as the type I and type II errors, respectively. In addition, a metric called sparsistency is sometimes used, measuring the binary output of whether the estimated and the true supports are identical, which is the complement of the union between FN and FP [29]. Sparsistency might also be unobtainable for a certain problem; avoiding FPs requires selecting the hyperparameter so large that FNs will arise, and similarly avoiding FNs will result in more FPs. To the best of our knowledge, there exists no method of choosing the regularization level formulated with regards to FPs and FNs. There have, however, been other related works on hyperparameter selection; in [30], a covariance test statistic was used to determine whether to include every new regressor along a path of regularization values, whereas in [31], the regularization level was selected using a maximum-a-posteriori approach, which was estimated alongside the regression variables by appropriately selecting a hyperprior for it.

In this work, we take a probabilistic approach to hyperparameter selection. By analyzing how the noise component propagates into the parameter estimates for different estimators and different choices of the hyperparameters, we seek to increase the sparsistency of the group-LASSO estimate by means of optimizing the FP rate. By making assumptions on the noise distribution and then sampling from the corresponding extreme value distribution using the Monte Carlo method, the hyperparameter is chosen as an appropriate quantile of the largest anticipated noise components. Avoiding FPs can never be guaranteed without maximizing the regularization level, thereby setting the entire solution to zero, but the risk may be quantified. By specifying the type I error, the sparsistency rate is also indirectly controlled, whenever this is feasible. Furthermore, for

Gaussian noise, we show that the distribution for the maximum noise components follows a type I extreme value distribution (Gumbel), from which a parametric quantile may be obtained at a low computational cost.

For coherent dictionaries, i.e., where there is a high degree of collinearity between the atoms, many of the theoretical guarantees for sparse estimation will fail to hold, along with a few of the methods themselves. The effects on the estimates for the collinear atoms are difficulty to discern; depending on the problem either all of them, or just a few of them, become non-zero. Coherence therefore typically results in FPs, if the regularization level is not increased, which in turn might yield FNs. There exists some approaches of dealing with coherent dictionaries. The elastic net uses a combination of ℓ_1 and ℓ_2 penalties [32], with the effect of increasing the inclusion of coherent components, thereby avoiding some FNs, but still not decreasing the number of FPs. Another popular approach is the reweighted LASSO [33], which solves a series of LASSO problems where the regularization level is individually set for each atom using its previous estimate. This approach approximates the use of a (non-convex) logarithmic regularizer, which allows the estimates to better reallocate power to the strongest of the coherent atoms. The proposed approach does not account for the leakage of power between coherent components in the true support, but only for the coherence effects on the assumed noise. As a remedy, the proposed method instead solves the reweighted LASSO problem at the chosen regularization level.

To illustrate the achievable performance of the proposed method, numerical results show how the proposed method for selecting the hyperparameter is both much less computationally demanding and at the same time far more accurate than both CV and the Bayesian information criterion (BIC). These results are obtained for both the sparse and the group-sparse regression problems.

The remainder of this paper is organized as follows: Section II defines the mathematical notation used throughout the paper, whereafter Section III describes some background on the group-LASSO and the scaled group-LASSO problems, also including an implementation of the cyclic coordinate descent solver for these problems. Section IV desribes the proposed method of selecting the regularization level. Section V then describes how the estimate of the noise standard deviation may be improved, and section VI how FPs due to coherence may be dealt with. Thereafter, Section VII describes the competition; the CV and BIC methods. Section VIII shows some numerical results, and, finally, Section 9 concludes upon the presented results.

2 Notational conventions

In this paper, we use the mathematical convention of letting lower-case letters, e.g., y, denote scalars, while lower-case bold-font letters, $\mathbf{y}$, denote column vectors wheras upper-case bold-font letters, $\mathbf{Y}$, denote matrices. Furthermore, E, V, D, and P denote the expectation, variance, standard deviation, and probability of a random variable or vector, respectively. We let $|\cdot|$ denote the absolute value of a complex-valued number, while $\|\cdot\|_p$ and $\|\cdot\|_\infty$ denote the p-norm and the maximum-norm, respectively. Furthermore, $\text{diag}(\mathbf{a}) = \mathbf{A}$ denotes the diagonal matrix with diagonal vector $\mathbf{a}$, although $\text{diag}(\mathbf{A}) = \mathbf{a}$ is also denoting the diagonal vector of a square matrix. As is conventional, we let $(\cdot)^\top$ and $(\cdot)^H$ denote the transpose and hermitian transpose, respectively. Subscripts are used to denote a subgroup of a vector or matrix, and superscript typically denotes a power operation, except when the exponent is within parentheses or hard brackets, which we use to denote iteration number, e.g., $x^{(j)}$, is j:th iteration of x, and the index of a random sample, e.g., $x^{[j]}$ denotes the j:th realization of the random variable x. We also make use of the notations $(x)_+ = \max(0, x)$, $\text{sign}(\mathbf{x}) = \mathbf{x}/\|\mathbf{x}\|_2$, and $x \sim F$, which states that the random variable x has distribution function F. Finally, we let $\varnothing$ denote the empty set.

3 Group-sparse regression via coordinate descent

Consider a noisy N-sample complex-valued vector $\mathbf{y}$, which may be well described using the linear regression model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{1}$$

where $\mathbf{A} \in \mathbb{C}^{N \times M}$, where $M \gg N$, is the (possibly highly) underdetermined regressor matrix, or dictionary, constructed from a set of normalized regressors, i.e., $\mathbf{a}_i^H \mathbf{a}_i = 1,\ i = 1, \ldots, M$, with $\mathbf{a}_i$ denoting the i:th column of $\mathbf{A}$. The unknown parameter vector $\mathbf{x}$ is assumed to have a C/M-sparse parameter support, i.e., only $C < N$ of the parameters in $\mathbf{x}$ are assumed to be non-zero. In this paper, we consider the generalized case where the dictionary may contain groups of regressors whose components are linked in a modeling sense, such that the model parametrizes a superpositioning of objects, each of which is modeled by a group of regressors rather than just one. Therefore, we let the dictionary be constructed such that the M regressors are collected into K groups with L_k regressors in the

k:th group, i.e.,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \dots & \mathbf{A}_K \end{bmatrix} \tag{2}$$

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{a}_{k,1} & \dots & \mathbf{a}_{k,L_k} \end{bmatrix} \tag{3}$$

and where, similarly,

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1^\top & \dots & \mathbf{x}_K^\top \end{bmatrix}^\top \tag{4}$$

Furthermore, we assume that the observation noise, $\mathbf{e}$, may be well modeled as an i.i.d. multivariate random variable, such that $\mathbf{e} = \sigma\mathbf{w}$, where $\mathbf{w} \sim F$, for some sufficiently symmetric distribution F with unit variance. Let $\hat{\mathbf{x}}(\lambda)$ denote the solution to the convex optimization problem

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) = \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_{k=1}^{K} \|\mathbf{x}_k\|_2 \tag{5}$$

for some hyperparameter $\lambda > 0$. This is the group-LASSO estimate, for which we briefly outline the corresponding cyclic coordinate descent (CCD) algorithm. In its essence, CCD updates the parameters in $\mathbf{x}$ one at a time, by iteratively minimizing $f(\mathbf{x})$ for each x_i, $i = 1, \dots, M$, in random order. As $\mathbf{x}$ is complex-valued, and as $f(\mathbf{x})$ is non-differentiable for $\mathbf{x}_k = 0$, for any k, we exploit Wirtinger calculus and subgradient analysis to form derivatives of f. Let $\mathbf{r}_k = \mathbf{y} - \mathbf{A}_{-k}\mathbf{x}_{-k}$ be the residual vector where the effect of the k:th variable group has been left out, i.e., such that $\mathbf{A}_{-k}$ and $\mathbf{x}_{-k}$ omit the k:th regressor and variable group, respectively. Thus, one may find the derivative of $f(\mathbf{x})$ with respect to $\mathbf{x}_k$ and set it to zero, i.e.,

$$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_k} = -\mathbf{A}_k^H (\mathbf{r}_k - \mathbf{A}_k\mathbf{x}_k) + \lambda \mathbf{u}_k = \mathbf{0} \tag{6}$$

where

$$\mathbf{u}_k = \begin{cases} \operatorname{sign}(\mathbf{x}_k) & \mathbf{x}_k \neq \mathbf{0} \\ \{\mathbf{u}_k : \|\mathbf{u}_k\|_2 \leq 1\} & \mathbf{x}_k = \mathbf{0} \end{cases} \tag{7}$$

Under the assumption that $\mathbf{A}_k^H \mathbf{A}_k = \mathbf{I}$, a closed-form solution may be found as

$$\hat{\mathbf{x}}_k(\lambda) = \mathcal{T}(\mathbf{A}_k^H \mathbf{r}_k, \lambda) \tag{8}$$

where $\mathcal{T}(\mathbf{z}, \alpha) = \text{sign}(\mathbf{z}) \left(\|\mathbf{z}\|_2 - \alpha \right)_+$ denotes the group-shrinkage operator. The group-LASSO estimate is thus formed by the inner product between the residual and regressor groups, albeit where the groups' ℓ_2-norms are reduced by λ. Therefore, the estimate $\hat{\mathbf{x}}$ will be biased towards zero. Similarly, sparsity in groups is induced as the groups having an inner product with ℓ_2-norm smaller than λ are set to zero. The regularization parameter thus serves as an implicit model order selector. In particular, the zeroth model order, $\hat{\mathbf{x}} = \mathbf{0}$, is obtained for $\lambda \geq \lambda_0 = \max_k \left\| \mathbf{A}_k^H \mathbf{y} \right\|_2$. Let the true support set be denoted by $\mathcal{I} = \{k : \|\mathbf{x}_k\| \neq 0\}$. When decreasing λ, the model order grows, introducing the parameter group $k \in \hat{\mathcal{I}}(\lambda) \Leftrightarrow \left\| \mathbf{A}_k^H \mathbf{r}_k \right\|_2 > \lambda$. As a consequence, parameter groups are included in the support set in an order determined by their magnitude. In the case that $\left\| \mathbf{A}_k^H \mathbf{r}_k \right\| > \left\| \mathbf{A}_{k'}^H \mathbf{r}_{k'} \right\|_2$, then the implication $k' \in \hat{\mathcal{I}}(\lambda) \Rightarrow k \in \hat{\mathcal{I}}(\lambda)$ is always true, and a parameter group with some smaller ℓ_2-norm is never in the solution set if another one with a larger ℓ_2-norm is not. Selecting an appropriate regularization level is thus important; if set too large, the solution will have omitted parts of the sought signal, if set too small, the solution will include noise components and be too dense. Recently, the scaled LASSO was introduced, solving the optimization problem (here in group-version) [24]

$$\underset{\mathbf{x}, \sigma > 0}{\text{minimize}} \; g(\mathbf{x}, \sigma) = \frac{1}{\sigma} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + N\sigma + \mu \sum_{k=1}^{K} \|\mathbf{x}_k\|_2 \tag{9}$$

i.e., a modification of the group-LASSO where the auxilliary variable σ, representing the residual standard deviation, scales the least squares term, and where $\mu > 0$ is the regularization parameter. Again, utilizing a CCD approach, σ may be included in the cyclic optimization scheme, which has the closed-form solution

$$\hat{\sigma}(\mu) = \frac{\|\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}(\mu)\|_2}{\sqrt{N}} \tag{10}$$

Similar to the derivations above, the $\mathbf{x}_k$s may be iteratively estimated as

$$\hat{\mathbf{x}}_k(\mu) = \mathcal{T}(\mathbf{A}_k^H \mathbf{r}_k, \hat{\sigma}\mu) \tag{11}$$

which are thus regularized by $\sigma\mu$, making μ seemingly independent of the noise power. However, the estimate of σ is itself clearly affected by μ. For too low values of μ, typically $\hat{\sigma}(\mu) < \sigma$, i.e., smaller than the true noise standard deviation, as too much of the noise components will be modeled by $\hat{\mathbf{x}}(\mu)$. Similarly, when μ is too

Algorithm 1 Scaled group-LASSO via cyclic coordinate descent

Intialize $\mathbf{x}^{(0)} = \mathbf{0}$, $\mathbf{r} = \mathbf{y}$, and $j = 1$
while $j < j_{\max}$ **do**
  $\sigma \leftarrow \|\mathbf{r}\|_2 / \sqrt{N}$
  $I_j = \mathcal{U}(1, \ldots, K)$
  for $i \in I_j$ **do**
    $\mathbf{r} \leftarrow \mathbf{r} + \mathbf{A}_i \mathbf{x}_i^{(j-1)}$
    $\mathbf{x}_i^{(j)} = \mathcal{T}(\mathbf{A}_i^H \mathbf{r}, \sigma\mu)$
    $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{A}_i \mathbf{x}_i^{(j)}$
  end for
  if $\left\|\mathbf{x}^{(j)} - \mathbf{x}^{(j-1)}\right\|_2 \leq \varkappa_{\text{tol}}$ **then**
    break
  end if
  $j \leftarrow j + 1$
end while

large, $\hat{\sigma}(\mu) > \sigma$ as it will also model part of the signal variability. However, even when the regularization level is chosen appropriately, one still has $\hat{\sigma}(\mu) \geq \sigma$ in general due to the estimation bias in $\hat{\mathbf{x}}(\mu)$. It is also worth noting that by inserting $\hat{\sigma}$ into $g(\mathbf{x}, \sigma)$, one obtains the equivalent optimization problem

$$\underset{\mathbf{x}}{\text{minimize}}\; g(\mathbf{x}) = 2\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 + \frac{\mu}{\sqrt{N}} \sum_{k=1}^{K} \|\mathbf{x}_k\|_2 \tag{12}$$

which may be identified as the square-root group-LASSO [34]. Algorithm 1 outlines the CCD solver for the scaled group-LASSO problem at some regularization level μ, where $j_{\max}$, $\varkappa_{\text{tol}}$, and $\mathcal{U}(\cdot)$ denote the maximum number of iterations, the convergence tolerance, and a random permutation of indices, respectively. Here, when comparing (11) to (8), it becomes apparent that the group-LASSO and the scaled group-LASSO will yield identical solutions when $\lambda = \sigma\mu$. The motivation behind using the scaled group-LASSO is instead that the regularization level may be chosen independently of the noise variance. In the next section, we make use of this property.

4 A probabilistic approach to regularization

Consider the overall aim of selecting the hyperparameter in order to maximize sparsistency, i.e., selecting λ such that the estimated support coincides with the true support,

$$\lambda = \{\lambda : \hat{\mathcal{I}}(\lambda) = \mathcal{I}\} \tag{13}$$

From the perspective of detection theory, whenever the support recovery fails, at least one of the following errors have occurred:

- False positive (FP) or type-I error: the regularization level is set too low and the estimated support contains indices which are not in the true support; $\left(\hat{\mathcal{I}}(\lambda) \cap \mathcal{I}^c\right) \neq \varnothing$, where $\mathcal{I}^c$ denotes the complement of the support set.
- False negative (FN) or type-II error: the regularization level is set too high and the estimated support set does not contain all indices in the true support; $\left(\hat{\mathcal{I}}^c(\lambda) \cap \mathcal{I}\right) \neq \varnothing$.

One may therefore seek to maximize sparsistency by minimizing the FP and FN probabilities simultaneously, which for the group-LASSO means finding a regularization level which offers a compromise between FPs and FNs. To that end, let $\Lambda^* = [\lambda_{\text{min}}, \lambda_{\text{max}}]$ denote the interval for which any $\lambda \in \Lambda^*$ for the group-LASSO estimator fulfills (13), where

$$\lambda_{\text{min}} = \inf\{\lambda : \max_{i \notin \mathcal{I}} \left\|\mathbf{A}_i^H \mathbf{r}_i\right\|_2 \leq \lambda\} \tag{14}$$

$$\lambda_{\text{max}} = \sup\{\lambda : \min_{i \in \mathcal{I}} \left\|\mathbf{A}_i^H \mathbf{r}_i\right\|_2 \geq \lambda\} \tag{15}$$

Thus, λ_{min} is the smallest λ possible which does not incur FPs, whereas λ_{max} is the largest λ possible without incurring FNs in the solution. Therefore, support recovery is only possible if $\lambda_{\text{min}} \leq \lambda_{\text{max}}$. Clearly, the converse might occur, for instance, if the observations are very noisy, and the largest noise component becomes larger than the smallest signal component, and, as a result $\Lambda^* = \varnothing$.

4.1 Support recovery as a detection problem

The i:th parameter group is included in the estimated support if the ℓ_2-norm of the inner product between the i:th dictionary group and the residual is larger than

the regularization level. The group-LASSO estimate for each group can thus can be seen as a detection problem, with λ acting as the global detection threshold. Support recovery can therefore be seen as a detection test; successful if λ can be selected such that all detection problems (for each and every group) are solved. We therefore begin by examining the statistical properties of the inner product between the i:th dictionary group and the data model. We here assume that the observations consist of a deteministic signal-of-interest and a random noise. Thus,

$$E(\mathbf{y}) = \mathbf{A}\mathbf{x}, \quad V(\mathbf{y}) = E(\mathbf{e}\mathbf{e}^H) = \sigma^2\mathbf{I} \tag{16}$$

The inner product between the dictionary and the data, $\mathbf{A}^H\mathbf{y}$, yields a vector in which each element constitutes a linear combination of the data elements. Under the assumption that $M > N$, the variability in the data vector is spread among the elements in a larger vector. Let $\mathbf{u} = \mathbf{A}^H\mathbf{y}$ denote this M element vector, which has the statistical properties

$$E(\mathbf{u}) = \mathbf{A}^H\mathbf{A}\mathbf{x}, \quad V(\mathbf{u}) = \sigma^2\mathbf{A}^H\mathbf{A} \tag{17}$$

and while the elements in $\mathbf{y}$ are statistically independent, the elements in $\mathbf{u}$ are generally not, as $\mathbf{A}^H\mathbf{A} \neq \mathbf{I}$ for $M > N$. By examining the i:th group in $\mathbf{u}$, one may note that

$$E(\mathbf{u}_i) = \begin{cases} \sum_{j\in\mathcal{I}} \mathbf{A}_i^H\mathbf{A}_j\mathbf{x}_j, & i \notin \mathcal{I} \\ \mathbf{x}_i, & i \in \mathcal{I} \end{cases} \tag{18}$$

where it is also assumed that the components in the true support are independent, $\mathbf{A}_i^H\mathbf{A}_j = \mathbf{0}$, for $i \neq j, (i,j) \in \mathcal{I}$. One may note how the true variables are mixed amongst the elements in $\mathbf{u}$; they appear consistently in the true support, while also leaking into the other variables, proportionally to the coherence between the groups, as quantified by $\mathbf{A}^H\mathbf{A}$. In the CCD updates for the group-LASSO, the i:th component becomes active if

$$\lambda < \left\|\mathbf{A}_i^H\mathbf{r}_i\right\|_2 \tag{19}$$

$$= \left\|\mathbf{A}_i^H\left(\mathbf{A}\mathbf{x} + \mathbf{e} - \mathbf{A}_{-i}\mathbf{x}_{-i}\right)\right\|_2 \tag{20}$$

$$= \left\|\mathbf{A}_i^H\mathbf{A}_i\mathbf{x}_i + \sum_{j\in\mathcal{I}} \mathbf{A}_i^H\mathbf{A}_j(\mathbf{x}_j - \hat{\mathbf{x}}_j) + \mathbf{A}_i^H\mathbf{e}\right\|_2 \tag{21}$$

$$= \begin{cases} \left\| \sum_{j \in \mathcal{I}} \mathbf{A}_i^H \mathbf{A}_j (\mathbf{x}_j - \hat{\mathbf{x}}_j) + \mathbf{A}_i^H \mathbf{e} \right\|_2, & i \notin \mathcal{I} \\ \left\| \mathbf{x}_i + \mathbf{A}_i^H \mathbf{e} \right\|_2, & i \in \mathcal{I} \end{cases} \tag{22}$$

This result provides some insight into choosing the regularization level; this must be set such that with high probability

$$\lambda > \left\| \sum_{j \in \mathcal{I}} \mathbf{A}_i^H \mathbf{A}_j (\mathbf{x}_j - \hat{\mathbf{x}}_j) + \mathbf{A}_i^H \mathbf{e} \right\|_2, \qquad \forall i \notin \mathcal{I} \tag{23}$$

$$\lambda < \left\| \mathbf{x}_i + \mathbf{A}_i^H \mathbf{e} \right\|_2, \qquad \forall i \in \mathcal{I} \tag{24}$$

where if (23) is not fulfilled, FPs enters the solution, whereas if (24) does not hold, FNs will enter the solution. Certainly, the true $\mathbf{x}$ is unknown, as is $\hat{\mathbf{x}}$ before the estimation starts, at which point λ must be selected. If there is coherence in the dictionary, it is not well defined how the data's variability is explained among the dependent variables, due to the bias resulting from the regularizers used in the LASSO-methods. Our proposition is thus to, when selecting the regularization level, focus on the noise part, while leaving the leakage of the $\mathbf{x}_i$:s into the other components be, dealing with them in a later refinement step. To that end, consider an hypothesis test examining whether the observed data contains the signal-of-interest or not, i.e.,

$$\begin{aligned} \mathcal{H}_0 &: \quad \mathbf{y} = \mathbf{e} \\ \mathcal{H}_1 &: \quad \mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \end{aligned} \tag{25}$$

Under the null hypothesis, $\mathcal{H}_0$, $\mathcal{I} = \varnothing$. In this case, (23) and (24) reduces to

$$\lambda > \left\| \mathbf{A}_i^H \mathbf{e} \right\|_2, \quad \forall i \tag{26}$$

which should be fulfilled with a high probability for all groups. Thus, one may chose the regularization level with regards to the maximum noise component, i.e.,

$$P\left(\max_i \left\| \mathbf{A}_i^H \mathbf{e} \right\|_2 \leq \lambda_\alpha \right) = 1 - \alpha \tag{27}$$

such that λ_α denotes the α-quantile of the maximum ℓ_2-norm of the inner product between the dictionary and the noise. This regularization level can be seen as a lower bound approximation of $\lambda_{\min}$, where FPs due to leakage from the true support are disregarded.

4.2 Model selection via extreme value analysis

In order to determine λ_α, we need to find the distribution in (27), which is an extreme value distribution determined by the underlying noise distribution. To that end, let

$$z_i = \left\|\mathbf{A}_i^H \mathbf{e}\right\|_2^2 / \sigma^2 \tag{28}$$

denote the (squared) ℓ_2-norm of the inner product between the i:th dictionary group and the noise, scaled by the noise variance. For the scaled group-LASSO, where $\lambda = \mu\hat{\sigma}$, one may express the sought extreme value distribution, denoted $\bar{F}$, as

$$P\left(\max_i \left\|\mathbf{A}_i^H \mathbf{e}\right\|_2 < \mu\hat{\sigma}\right) = P\left(\max_i \sigma\sqrt{z_i} < \mu\hat{\sigma}\right) \tag{29}$$

$$= P\left(\max_i z_i < \mu^2 \left(\frac{\hat{\sigma}}{\sigma}\right)^2\right) \tag{30}$$

$$= \bar{F}\left(\mu^2 \left(\frac{\hat{\sigma}}{\sigma}\right)^2\right) \tag{31}$$

Thus, if $\hat{\sigma}/\sigma \approx 1$, one may seek μ instead of λ, providing a method for finding a regularization level independent of the unknown noise variance. We thus propose selecting μ as the α-quantile from the extreme value distrubution $\bar{F}$ which may be obtained as

$$\mu_\alpha = \frac{\sigma}{\hat{\sigma}}\sqrt{\bar{F}^{-1}(1-\alpha)} \approx \sqrt{\bar{F}^{-1}(1-\alpha)} \tag{32}$$

It is, however, difficult to find closed-form expressions for extremes of dependent sequences; $z_1, \ldots, z_K$ become dependent as the underlying sequence, $\mathbf{u}$ (from (17)) from which the z_i:s are formed, is dependent. As a comparison, let $\tilde{z}_1, \ldots, \tilde{z}_K$ denote a sequence of variables with the same distribution as the z_i:s, although being independent of each other. One may then form the bound

$$P\left(\max_i z_i \leq \mu^2\right) \geq P\left(\max_i \tilde{z}_i \leq \mu^2 \left(\frac{\hat{\sigma}}{\sigma}\right)^2\right) \tag{33}$$

$$= P\left(\tilde{z}_i \leq \mu^2 \left(\frac{\hat{\sigma}}{\sigma}\right)^2\right)^K \tag{34}$$

$$= G\left(\mu^2\left(\frac{\hat{\sigma}}{\sigma}\right)^2\right)^K \tag{35}$$

$\forall \mu > 0$, where the independence of the parameters was used to form (34). One may thus form an upper bound on the sought quantile as

$$\mu_\alpha \leq \tilde{\mu}_\alpha = \frac{\hat{\sigma}}{\sigma}\sqrt{G^{-1}\left((1-\alpha)^{m^{-1}}\right)} \tag{36}$$

where G is the distribution of z_i, assumed to be equal $\forall i$. Indeed, μ_α is thus constructed such that the null hypothesis, $\mathcal{H}_0$, is falsely rejected with probability α. It does not, however, automatically mean that the probability for FPs, as described in (24), is also α. As argued above, the FP probability is typically larger than α, but, as will be illustrated below, can be shown to yield regularization levels that give high sparsistency.

4.3 Inference using Monte Carlo sampling

The proposed method for choosing the regularization level, as presented in this paper, only requires knowledge of the distribution family for the noise. We might then sample from the corresponding extreme value distribution, $\bar{F}$, using the Monte Carlo method. Consider $\mathbf{w}^{[j]}$ to be the j:th draw from the noise distribution F which has unit variance. A sample from the sought distribution, $\bar{F}$, is then obtained by calculating

$$\max_i z_i^{[j]} \sim \bar{F}(z) \tag{37}$$

where $z_i^{[j]} = \mathbf{w}^{[j]\,H}\mathbf{A}_i\mathbf{A}_i^H\mathbf{w}^{[j]}$. By randomly drawing N_{sim} such samples from $\bar{F}$, the quantile μ_α may be chosen either using a parametric quantile, or using the empirical distribution function, i.e.,

$$\mu_\alpha = \sqrt{\Psi_{\bar{F}}^{-1}(1-\alpha)} \tag{38}$$

where $\Psi_{\bar{F}}$ is the empirical distribution function of $\bar{F}$. For small α, the empirical approach may be computationally burdensome as

$$\mu_\alpha^2 \leq \max_{j=1,\ldots N_{\text{sim}}}\left(\max_i z_i^{[j]}\right) \Rightarrow N_{\text{sim}} \geq \lfloor \alpha^{-1} \rfloor \tag{39}$$

and one might then prefer to use a parametric quantile instead. Luckily, as the noise distribution F is assumed to be known, or may be estimated using standard methods, it is often feasible to derive which distribution family $\bar{F}$ belongs to. By then estimating the parameters of the distribution using the gathered Monte Carlo samples, a parametric quantile μ_α may be obtained using much fewer samples than using the corresponding empirical quantiles.

4.4 The Gaussian noise case

A common assumption is to model the noise as a zero-mean circular-symmetric i.i.d. complex-valued Gaussian process with some unknown variance, σ^2, i.e., $\mathbf{e} = \sigma\mathbf{w}$, where $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. For the i:th group, one then obtains

$$\mathbf{A}_i^H \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \Rightarrow \quad z_i \sim \chi^2(2L_i) \tag{40}$$

as it is assumed that $\mathbf{A}_i^H \mathbf{A}_i = \mathbf{I}, \forall i$. Thus, as z_i is a sum of L_i independent squared $\mathcal{N}(0,1)$ variables, it becomes χ^2 distributed with $2L_i$ degrees of freedom. In such a case, one may use (36) to directly find a closed-form upper bound on the regularization parameter. Alternatively, to find a more accurate quantile, one may draw inference on the Monte Carlo samples obtained when sampling from $\bar{F}$ in (37) instead. Classical extreme value theory states that the maximum domain of attraction for the Gamma distribution (of which χ^2 is a special case) is the type I extreme value distribution, i.e., the Gumbel distribution [35]. By estimating the scale and location parameters of the Gumbel distribution, one may obtain a more accurate tail estimate from the z_i:s than the empirical distribution yields. Thus, it holds that

$$\max_i z_i \sim \bar{F}(z) = \exp\left(e^{-\frac{z-\gamma}{\beta}}\right) \tag{41}$$

where the parameters γ and β are obtained using maximum likelihood estimation on the samples $\max_i z_i^{[1]}, \ldots, \max_i z_i^{[N_{\text{sim}}]}$. The regularization parameter μ_α can then be calculated using (32).

5 Correcting the σ-estimate for the scaled group-LASSO

The scaled LASSO framework provides a way of choosing the regularization level independently of the noise variance. By introducing σ as an auxiliary variable in the LASSO minimization objective, it may be estimated along with $\mathbf{x}$.

In the CCD solver, the estimate of the noise standard deviation is obtained in closed-form, from (10), as the residual standard deviation estimate, i.e., $\hat{\sigma}(\mu) = \|\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}(\mu)\|_2/\sqrt{N}$. There are two aspects in how well $\hat{\sigma}$ approximates the true noise standard deviation; firstly, as $\hat{\sigma}(\mu)$ models the residual, it will depend on the sparsity level of $\hat{\mathbf{x}}(\mu)$, such that

$$\hat{\sigma}(\mu) \to \sqrt{\mathbf{y}^H\mathbf{y}/N}, \quad \text{as } \mu \to \mu_0 \tag{42}$$

$$\hat{\sigma}(\mu) \to 0, \quad \text{as } \mu \to 0 \tag{43}$$

where μ_0 is the smallest μ which yields the zeroth solution, i.e.,

$$\mu_0 = \frac{\max\limits_i \left\|\mathbf{A}_i^H\mathbf{y}\right\|_2}{\sqrt{\mathbf{y}^H\mathbf{y}/N}} \tag{44}$$

Therefore, if μ is chosen too large, such that it underestimates the model order, $\hat{\sigma}$ is overestimated, whereas if μ is chosen too small, and too many components are included in the model, $\hat{\sigma}$ becomes underestimated. The second aspect is that the estimate models the residual standard deviation for the LASSO estimator, where the magnitudes of the elements in $\mathbf{x}$ are always biased towards zero, and will thus overestimate $\hat{\sigma}$ even when the regularization level is selected such that the true support is obtained, i.e.,

$$\hat{\sigma}(\mu) = \frac{\|\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}(\mu)\|}{N} \geq \frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|}{N} = \sigma, \quad \mu \in M^* \tag{45}$$

where M^* is the interval over μ which yields the true support estimate. These aspects have a profound effect on the regularization level. As a result of the approximation in (32), the chosen α will not yield the actual FP rate of the hypothesis test under $\mathcal{H}_0$; let the true FP rate be denoted by α^*. The relation between the chosen quantile μ_α and the true quantile μ_{α^*} is then

$$\mu_\alpha = \sqrt{\bar{F}^{-1}(1-\alpha)} = \frac{\hat{\sigma}(\mu_\alpha)}{\sigma}\sqrt{\bar{F}^{-1}(1-\alpha^*)} = \frac{\hat{\sigma}(\mu_\alpha)}{\sigma}\mu_{\alpha^*} \tag{46}$$

and subsequently the true FP rate becomes

$$\alpha^* = 1 - \bar{F}\left(\left(\frac{\sigma}{\hat{\sigma}(\mu_\alpha)}\right)^2 \bar{F}^{-1}(1-\alpha)\right) \tag{47}$$

One may therefore deduce that when $\hat{\sigma}$ is over- or underestimated, the FP rate becomes over- or underestimated, respectively; i.e.,

$$\sigma(\mu_\alpha) > \sigma \Rightarrow \alpha^* > \alpha \tag{48}$$

$$\sigma(\mu_\alpha) < \sigma \Rightarrow \alpha^* < \alpha \tag{49}$$

while if the standard deviation is correctly estimated, $\alpha^* = \alpha$. This may be attempted by selecting α small, such that the model order reasonably reflects the true model order, and then estimate the noise standard deviation using an unbiased method instead of via the LASSO. One may then undertake the following steps to improve the estimate of the noise standard deviation:

1. Estimate $\mathbf{x}$ and σ by solving the scaled group-LASSO problem (9) with regularization parameter μ_α, given by (32) for some α.

2. Re-estimate σ using a least squares estimate of the non-zero variables obtained in Step 1),
   $x_i \in \hat{\mathcal{I}}$, i.e., $\hat{\sigma}_{\text{LS}} = \left\| \left(\mathbf{I} - \mathbf{A}_{\hat{\mathcal{I}}} \mathbf{A}_{\hat{\mathcal{I}}}^{\dagger} \right) \mathbf{y} \right\|_2 / \sqrt{N}$.

3. Estimate $\mathbf{x}$ by solving the (regular) group-LASSO problem in (5) with the regularization parameter selected as $\lambda = \mu_\alpha \hat{\sigma}_{\text{LS}}$.

6 Marginalizing the effect of coherence-based leakage

The proposed method calculates a regularization level by quantifying the FP error probability for the hypothesis testing of whether the noisy data observations also contain the signal-of-interest, $\mathbf{Ax}$, or not. This FP rate is used to approximate the FP rate for finding the correct support, which is a slightly different quantity. The regularization level is set by analyzing how the noise propagates into the estimate of $\mathbf{x}$, and selects a level larger than the magnitude of the maximum noise component. In relation to the hypothesis test in (25), when the signal-of-interest is present in the signal, the group-LASSO estimate suffers from spurious non-zero estimates outside of the support set, as described in (23). Thus, even if the choice of μ_α drowns out the noise part with probability $1 - \alpha$, it does not necessarily zero out the spurious signal components, if the dictionary coherence is non-negligible. The true support is thereby not recovered, and the sparsistency rate is lower than $1 - \alpha$. One remedy is to set the regularization level higher, but it is inherently

difficult to quantify how the variability of the signal component is divided among its coherent dictionary atoms with LASSO-like estimators, and therefore difficult to assess how much higher it should be selected. For low SNR observations, the choice of the regularization level is sensitive; if set too high, the estimate will suffer from FNs. One should therefore to keep the regularization level as the proposed quantile μ_α, but instead to modify the sparse regression as to promote more sparsity among coherent components, and thereby possibly increasing the sparsistency rate. One such method is to solve the reweighted group-LASSO problem, where one at the j:th iteration obtains $\hat{\mathbf{x}}^{(j)}$ by solving

$$\underset{\mathbf{x}}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \sum_{k=1}^{K} \frac{\|\mathbf{x}_k\|_2}{\left\|\hat{\mathbf{x}}_k^{(j-1)}\right\|_2 + \varepsilon} \tag{50}$$

where ε is a small positive constant used to avoid numerical instability. Thus, the regularization level is iteratively updated using the ℓ_2-norm of the old estimate, which has the effect that the individual regularizer

$$\lambda_k = \frac{\lambda}{\left\|\mathbf{x}_k^{(j-1)}\right\|_2 + \varepsilon} \searrow, \quad \|\hat{\mathbf{x}}_k\|_2 > 1 \quad \text{(is large)} \tag{51}$$

$$\lambda_k = \frac{\lambda}{\left\|\mathbf{x}_k^{(j-1)}\right\|_2 + \varepsilon} \nearrow, \quad \|\mathbf{x}_k\|_2 < 1 \quad \text{(is small)} \tag{52}$$

Thus, the best (largest) component among the coherent variables will be less and less regularized, while the weaker components will be more and more regularized, until they are omitted altogether. By solving (50) iteratively, the solver approximates a (non-convex) sparse regression problem with a logaritmic regularizer, which is more sparsifying than the ℓ_1-regularizer. We thus propose to modify Step 3) in the σ-corrected approach discusses above with the reweighted group-LASSO, using $\lambda = \mu_\alpha \hat{\sigma}_{\text{LS}}$.

7 In comparison: Hyperparameter-selection using information criterias

Commonly, the statistical approach to finding the regularization level is to solve the LASSO problem for a grid $\lambda \in \Lambda$, typically selected as N_λ points uniformly

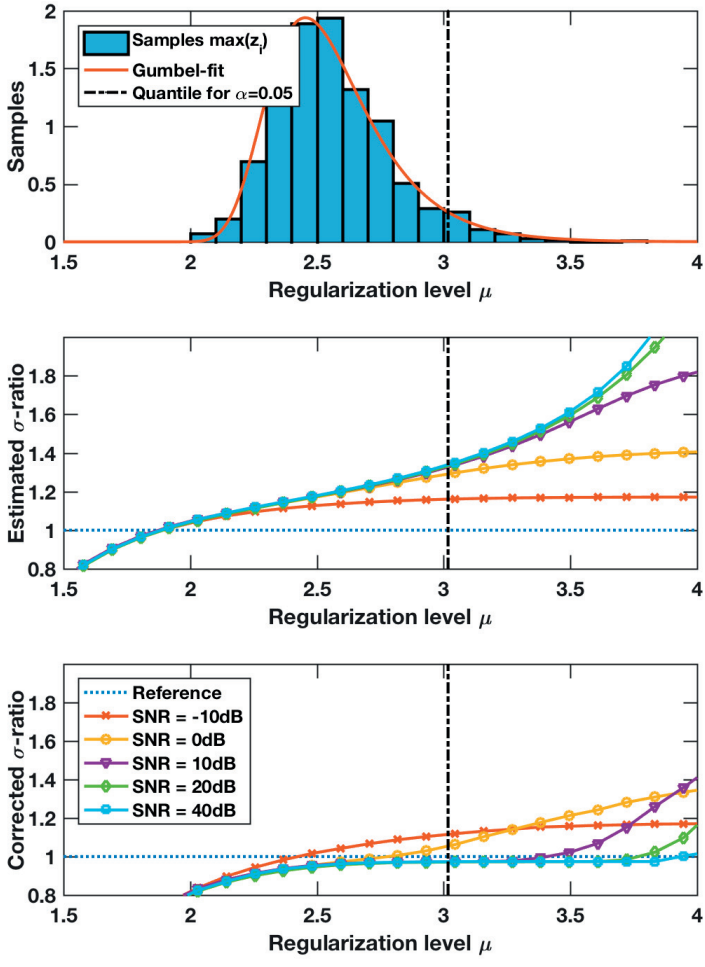


Figure 1: Results for estimation of σ for different levels of the regularization level, μ; the top plot illustrates how the ℓ_2-norm of the maximum nuisance component is distributed, the middle and bottom plots illustrate the ratio between the estimated $\hat{\sigma}$ and the true σ, for different levels of regularizations, using the scaled LASSO estimator. The different curves show the ratio estimates for different levels of SNR, i.e., σ. In the bottom plot, the σ-correction step has been applied to the estimation.

chosen on $(0,\ \lambda_0]$. Commonly, $\mathbf{x}(\Lambda)$ is then referred to as the solution path. For each point, λ_j, on the solution path, one can obtain the model order, $\hat{k}_j = \left\|\hat{\mathbf{x}}(\lambda_j)\right\|_0$ and the statistical likelihood of the observed data given the assumed distribution of the parameter estimate, $L(\mathbf{y}, \hat{\mathbf{x}}(\lambda_j))$, which when used to calculate the Bayesian Information Criteria (BIC), i.e.,

$$\mathrm{BIC}(\lambda_j) = \log(N)\hat{k}(\lambda_j) - 2L(\mathbf{y}, \hat{\mathbf{x}}(\lambda_j)) \tag{53}$$

yields the model order estimate $\hat{\lambda}_{\mathrm{BIC}} = \mathrm{argmin}_j\, \mathrm{BIC}(\lambda_j)$. Certainly, this procedure may prove costly, as it requires solving the LASSO N_λ times. Typically, BIC also tends to overestimate the model order, thereby underestimating $\hat{\lambda}_{\mathrm{BIC}}$. Another commonly used method for selecting the regularization level is to perform cross-validation (CV) on the observed data. As there exist many different variations of CV, many of which are computationally infeasible for typical problems, this paper describes the popular R-fold CV variant [36], in which one shall:

1. Split the observed data into R disjoint random partitions.
2. Omit the r:th partition from estimation and use the remaining partitions to estimate a solution path $\mathbf{x}_r(\Lambda)$. Repeat for all partitions.
3. Calculate the prediction residual variance $\mathbf{r}(\lambda, r) = \|\mathbf{y}(r) - \mathbf{A}(r)\mathbf{x}_r(\lambda_i)\|_2^2$ and use this to calculate

$$\mathrm{CV}(\lambda_j) = \sum_{r=1}^{R} \mathbf{r}(\lambda_i, r) n^{-1} \tag{54}$$

$$\mathrm{SE}(\lambda_j) = D\left(\{\mathbf{r}(\lambda_j, r)\}_{r=1}^{R}\right) R^{-1/2} \tag{55}$$

4. Let $\lambda^* = \mathrm{argmin}_j\ CV(\lambda_j)$. Utilizing the one standard error rule, one then finds $\lambda_{\mathrm{CV}} = \sup_j \lambda_j$ such that $\mathrm{CV}(\lambda_j) \leq \mathrm{CV}(\lambda^*) + \mathrm{SE}(\lambda^*)$.
5. Calculate the solution $\hat{\mathbf{x}}(\lambda_{\mathrm{CV}})$ using the entire data set.

When R is large enough, CV has been shown to asymptotically approximate the Akaike Information Criterion (AIC) [37]. However, CV is generally computationally burdensome, requiring solving the LASSO $(N_\lambda R + 1)$ times. It should be noted that CV forms the model order estimate by selecting the λ which yields the smallest prediction error. This undoubtedly discourages overfitting, but does not specifically target support recovery, which often is the main objective of sparse estimation. Thus, CV tends to set λ too low, which reduces the bias for the correct variables of $\hat{\mathbf{x}}(\lambda)$, but which also introduces FPs.

8 Numerical results

To illustrate the efficacy of the proposed method, termed the PRObabilistic regularization approach for SParse Regression (PROSPR) for selecting a regularization level, we test it under a few test scenarios, in comparison to the BIC and CV methods. However, before doing so, we illustrate the distribution of the maximum noise component over μ (from (32)), and make analysis on how σ is estimated in the scaled LASSO for these levels of the regularization parameter, with and without the σ-correction step described in Section 5. We thus simulate $N_{\text{MC}} = 200$ Monte Carlo simulations of $\mathbf{y}$, such that

$$\mathbf{y}^{[n]} = \mathbf{A}^{[n]}\mathbf{x}^{[n]} + \sigma\mathbf{w}^{[n]} \tag{56}$$

for the n:th simulation, where the elements in the dictionary consists of i.i.d. draws from the complex-valued Gaussian distribution, $\mathcal{N}(0, 1)$, and wherein $\mathbf{x}$ are $S = 5$ non-zero elements with unit magnitude, at randomly selected indices. In this test scenario, we consider the standard (non-grouped) regression problem. In each simulation, $N = 100$ data samples are retrieved, and the number of regressors is set to $M = 500$, and thus equally many groups, $K = 500$, such that $L = 1$. The signal-of-interest is here corrupted by an i.i.d. complex-valued Gaussian noise, such that $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Figure 1 illustrates the distribution of $z_i^{[n]} = \max_i (\mathbf{a}_i^{[n]})^H \mathbf{w}^{[n]}$, for $n = 1, \ldots, N_{\text{sim}}$, in the top figure, where the density function for a fitted Gumbel distribution is overlaid. The dash-dotted line illustrates the quantile value for $\alpha = 0.05$, which thus corresponds to the regularization level used with the proposed method for that α. The middle plot illustrates the paths of the ratios $\hat{\sigma}(\mu)/\sigma$, when estimated using the scaled LASSO, wherein each of the four lines illustrates the estimated ratio when the true σ used in (56) is selected such that the signal-to-noise ratio (SNR) is $-10, 0, 10, 20$, and 40 dB, respectively. Depending on μ, the LASSO estimate will include either only noise, or both noise and the signal-of-interest, and the ratios thus grow as $\mu \to \mu_0$. When the SNR is low, the ratio contains much of the signal-of-interest even at a low regularization level, while, as the SNR is low, the signal-of-interest is weak. Also, one may note that the ratios levels out as $\mu \to \mu_0$, at which point $\hat{\mathbf{x}} = \mathbf{0}$. The choice of α, if selected too small, in tandem with a low SNR, will therefore yield the zero solution. One sees this in the middle figure, as the ratio has leveled out for SNR = -10 dB, whereas the ratios are still growing for the other noise levels. Most important, however, is how the ratios affect the choice of μ in (32).

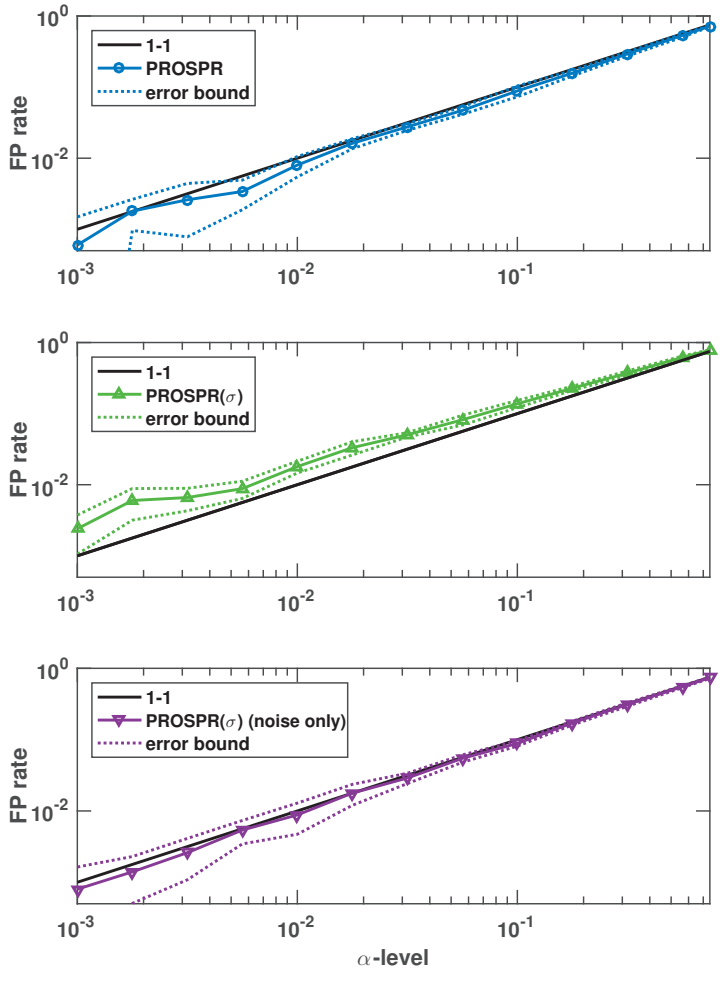


Figure 2: The subfigures illustrate the FP rate of support recovery for different levels of α. For all figures, the filled curves shows the preferred 1-1 line, the filed lines with symbols shows the estimate, and the dashed line shows the estimates $\pm$ one standard error. The top figure illustrate the FP rates for the PROSPR method, in the middle figure PROSPR with σ-correction is used, and the bottom figure shows the FP rate when the data is noise-only.

As the method assumes that the ratio is close to one, the effect when it grows becomes, as given by (48), that the actual α becomes larger than the selected value, which decreases sparsistency as FPs enter the solution. A remedy to this may be seen in the bottom figure, where σ is re-estimated along the lines described in Section 5. By using the least squares estimate, the ratio becomes larger than one only if the components in the true support has already been excluded from the estimated support, which can be seen for SNR = -10 and 0 dB. It may be noted that for the other SNR levels, the ratio is approximately one in the upper tail of the Gumbel distribution, and α approximates the true FP rate for inclusion of components due to noise.

As noted above, this is not necessarily equal to the FP rate of support recovery, as is illustrated in Figure 2, where the estimated FP rate for support recovery,

$$\frac{1}{N_{\text{MC}}} \sum_{n=1}^{N_{\text{MC}}} 1 \left\{ \left(\hat{\mathcal{I}} \cap \mathcal{I}^c \right) \neq \varnothing \right\} \tag{57}$$

is shown, where $1\{\cdot\}$ denotes the binary indicator function, which is one if the specified condition is fullfilled (and zero otherwise); the condition being that there exists elements in estimated support which are not in the true support. One may also note from the top figure that SPICE, which is shown in, e.g, [27], to have the regularization level fixed at $\mu = 1$, is very unlikely to avoid FPs.

The middle plot illustrates the obtained FP rate for different choices of α, when σ-corrected PROSPR is used. The filled line with sitting triangles shows the mean value, and the dashed lines shows the mean value $\pm$ one standard deviation. In this scenario, we have used the same parameter settings as above, although with $N_{\text{MC}} = 5000$ Monte Carlo simulations, at SNR = 20 dB. To select the regularization level in each simulation, we let PROSPR use $N_{\text{sim}} = 500$ simulations of the noise, $\mathbf{w}^{[j]}$, and use a parametric quantile from a Gumbel distribution fitted to the obtained draws of $\max_i z_i^{[j]}$. One note from the simulation results that in the middle figure, the estimated FP rate is consistenly higher than the selected α. This is a result of the dictionary coherence, which makes the signal power in the true support leak into the other variables, as shown in (22). To verify this claim, the bottom figure shows the same estimation scenario, when applied to the noise-only signal, i.e., $\mathbf{y} = \sigma\mathbf{w}$, where thus $\mathcal{I} = \varnothing$, and FPs occur whenever $\hat{\mathbf{x}}(\mu_\alpha) \neq \mathbf{0}$. As may be seen in the figure, the FP rate follows the α-level well for this scenario. To remedy the overestimated FP rate, one should set the regularization level

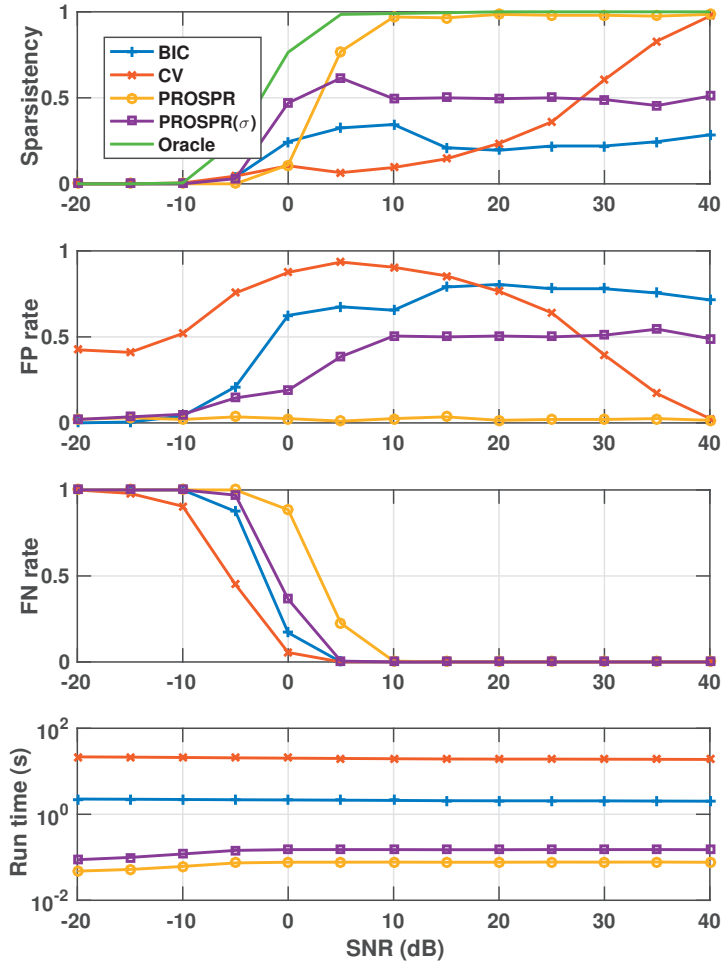


Figure 3: Compared performance results with LASSO using PROSPR (with and without σ-correction), CV, BIC, and in the top plot, an oracle method, illustrating the best result achievable at any regularization level. The top plot shows sparsistency (or support recovery rate), the second plot shows the FP rate, the third shows the FN rate, and the bottom plot shows the average run times for each method.

somewhat higher, in order to account for the signal leakage, too. Generally, we have found that when σ-correction is not used, and the estimated σ is too large, as discussed above, this results in a regularization level that is set too high, i.e.,

$$\lambda_\alpha = \mu_\alpha \hat{\sigma}(\mu_\alpha) > \mu_\alpha \sigma = \lambda_{\alpha^*} \tag{58}$$

which may cancel out the unwanted effect of having a too large FP rate. For this estimation scenario, this indeed becomes the case, as can be seen in the top figure, where the FP rate follows the specified α-level well. Next, we compare the proposed method, with and without σ-correction, to the BIC and CV methods for hyperparameter selection, in terms of sparsistency

$$\frac{1}{N_{\text{MC}}} \sum_{n=1}^{N_{\text{MC}}} 1\left\{\hat{\mathcal{I}} = \mathcal{I}\right\} \tag{59}$$

FP rate from (57), FN rate

$$\frac{1}{N_{\text{MC}}} \sum_{n=1}^{N_{\text{MC}}} 1\left\{\left(\hat{\mathcal{I}}^c \cap \mathcal{I}\right) \neq \varnothing\right\} \tag{60}$$

and average run time in seconds, when implemented in Matlab using the CCD solver on a 2013 Intel Core i7 MacBook Pro, for $N_{\text{MC}} = 200$ simulations. In the top plot in Figure 3, illustrating the sparsistency results at different levels of SNR, we have also included the oracle support recovery, which illustrates the maximum rate of support recovery achievable using an oracle choice of λ. For CV and BIC, the LASSO is solved on a grid of $N_\lambda = 50$ regularization levels, uniformly spaced on $(0, \lambda_0]$, and $R = 10$ folds were used for CV. For the PROSPR methods, $\alpha = 0.05$ was used. From the FP and FN results, one sees the trade-off between FPs and FNs, such that, on average, CV is the approach which selects the lowest regularization level, benefitting the FN rate, but at the cost of often incurring FPs. PROSPR, on the other hand, selects the highest regularization level, which results in approximately 5 % FPs, with the result that FNs are more frequent than with CV. However, if sparsistency is the focus of the regularization, the proposed methods fairs the best, outperforming both CV and BIC. An advantage with CV is that it chooses the regularization level with respect to both the signal and the noise components, and thus improves as SNR increases, whereas PROSPR yields similar FP rates independently of the SNR. One may therefore,

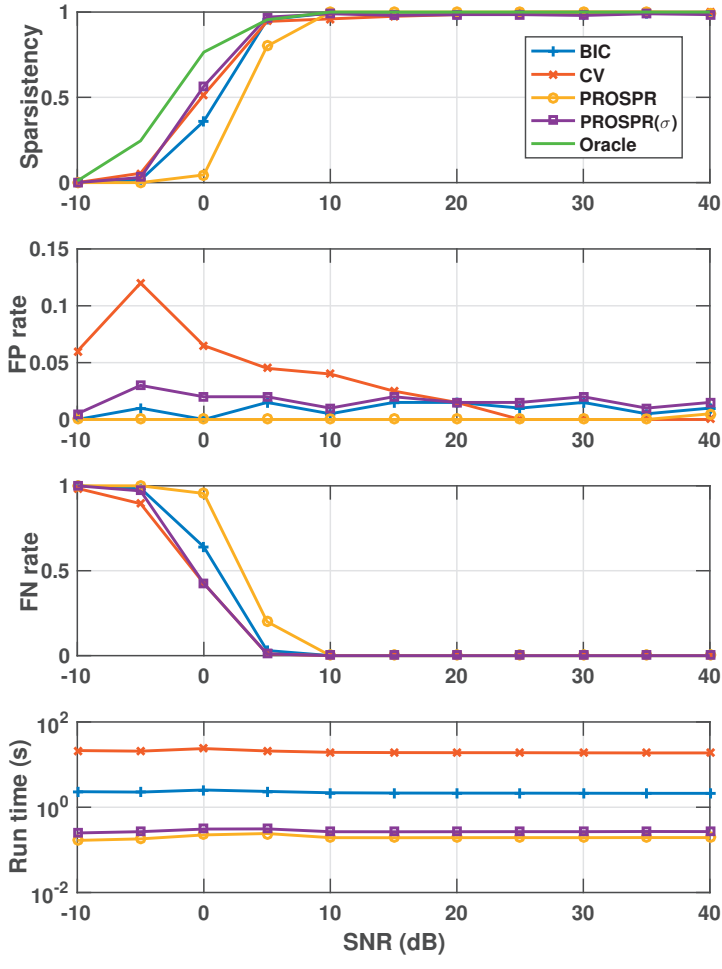


Figure 4: Compared performance results with reweighted LASSO using PROSPR (with and without σ-correction), CV, BIC, and in the top plot, an oracle method, illustrating the best result achievable at any regularization level. The top plot shows sparsistency (or support recovery rate), the second plot shows the FP rate, the third shows the FN rate, and the bottom plot shows the average run times for each method.

if the SNR is high, choose an α smaller than $\alpha = 0.05$. As also verified in Figure 2, the FP-rate for the σ-corrected estimate becomes smaller than α; here at most 0.5, whereas PROSPR without σ-correction performs best from SNR = 5 dB and higher. Most impressive are the run times; CV should be at most $N_\lambda R = 2.5 \cdot 10^2$ times slower than the proposed method - a gap which is slightly narrowed as CV uses warm-starts and as PROSPR's regularization level still requires some computational effort. Still, the PROSPR methods are significantly faster than CV. By comparison, BIC seems to fair somewhere in between; it is faster than CV, but also performs worse than CV for high levels of SNR.

As discussed in Section 6, the effect of coherence-based leakage from the signal components may be lessened by using a reweighted LASSO, where the (group-) LASSO problems is solved several times, with the regularization level being individually and iteratively selected for each group using the old estimate. The approach approximates a non-convex logarithmic penalty, which is sparser than the convex ℓ_1 or ℓ_2/ℓ_1 regularizers for the LASSO and group-LASSO, respectively. Typically, the reweighted (group-) LASSO handles FPs very well, pushing these towards zero, while FNs remains unchanged. As seen from Figure 3, the main error incurred in the PROSPR methods is FNs, why instead of using $\alpha = 0.05$ (i.e., using a low probability of FP), one might select a larger quantile, such that the FN rate decreases, at the expense of a higher FP rate. Figure 4 illustrates the estimation performance when the reweighted LASSO has been used at the regularization levels by the methods, where $\alpha = 0.5$ is used for the PROSPR-methods. One may note at this level, PROSPR with σ-correction follows CV well, with the FPs being dealt to a large extent. Although performing similarly to CV in terms of sparsistency, the proposed method still has a substantial computational advantage.

Next, we then analyze estimation performance for the group-sparse regression problem. We simulate $N_{\text{MC}} = 200$ Monte Carlo simulations of the signal in (56), with $N = 100$ observations in each, using a dictionary with $M = 1000$ atoms collected into $K = 200$ groups with $L = 5$ atoms in each. The true signal-of-interest consists of $S = 3$ groups, with indices randomly selected, and where $\mathbf{x}_i = \mathbf{1}$, for $i \in \mathcal{I}$.

Otherwise, the settings are identical to the standard sparse regression setup. Figure 5 shows the estimation performance for this simulation scenario, for different levels of SNR. One may note that CV does not perform as well as for the standard sparse regression model, as the FP rate does not decrease when SNR increase. As before, PROSPR performs better without σ-correction than with, for

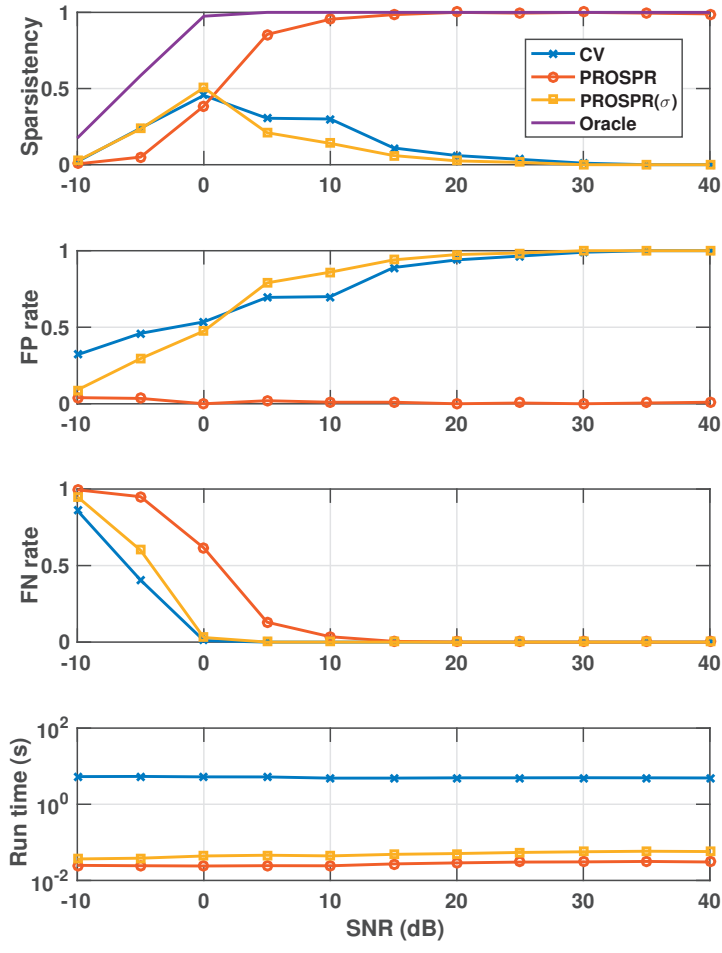


Figure 5: Compared performance results with group-LASSO using PROSPR (with and without σ-correction), CV, and in the top plot, an oracle method, illustrating the best result achievable at any regularization level. The top plot shows sparsistency (or support recovery rate), the second plot shows the FP rate, the third shows the FN rate, and the bottom plot shows the average run times for each method.

high levels of SNR. Unlike before, the BIC criterion for groups is not straightforward, as the degrees of freedom in the estimation is not well-defined; we have therefore decided to omit it from comparison in the group-sparse regression scenario.

Finally, similar to Figure 4, Figure 6 compares the estimation results for the reweighted group-LASSO estimator for the compared methods, for different levels of SNR. One may note that the proposed method with σ-correction now outperforms CV, approaching the oracle performance, while the proposed method without correction performs on par with CV. It therefore seems that CV, for the group-sparse problem, sets the regularization level relatively higher than for the non-grouped case. Still selecting $\alpha = 0.5$ heuristically seems to be good for the reweighted approach; it is set low enough to avoid FNs, while the reweighting manages to lessen the FPs. Again, the computational complexity can be seen to be significantly lower than for CV.

9 Conclusions

This paper has studied the selection of regularization level for sparse and group-sparse regression problems. As an implicit model order selection, it has a profound effect on support recovery; by changing the regularization level, one obtains supports with sizes ranging from very dense to completely empty. If support recovery is the main objective, selecting the regularization level carefully is of utmost importance. The group-regression problem, includes or excludes components from the estimated support depending on how the ℓ_2-norm of the inner-product between the dictionary group and the modeling residual compares to the regularization level. Intuitively, one therefore wishes to select the regularization level larger than the noise components, as to exclude them, but smaller than the signal components, as to include these. As the regularization level is selected prior to estimation, when the signal components are still unknown, we have instead studied the effect of the unit-variance observation noise, and how it propagates into the parameter estimates. Via extreme value analysis and by virtue of Monte Carlo simulations, we sample from the distribution of the maximal noise component, and may therefore select the regularization level as a quantile from the distribution. With the implicit assumption that the signal components are larger than the noise components, the quantile level, if chosen too large, may incur FNs, and if set too small, may incur FPs.

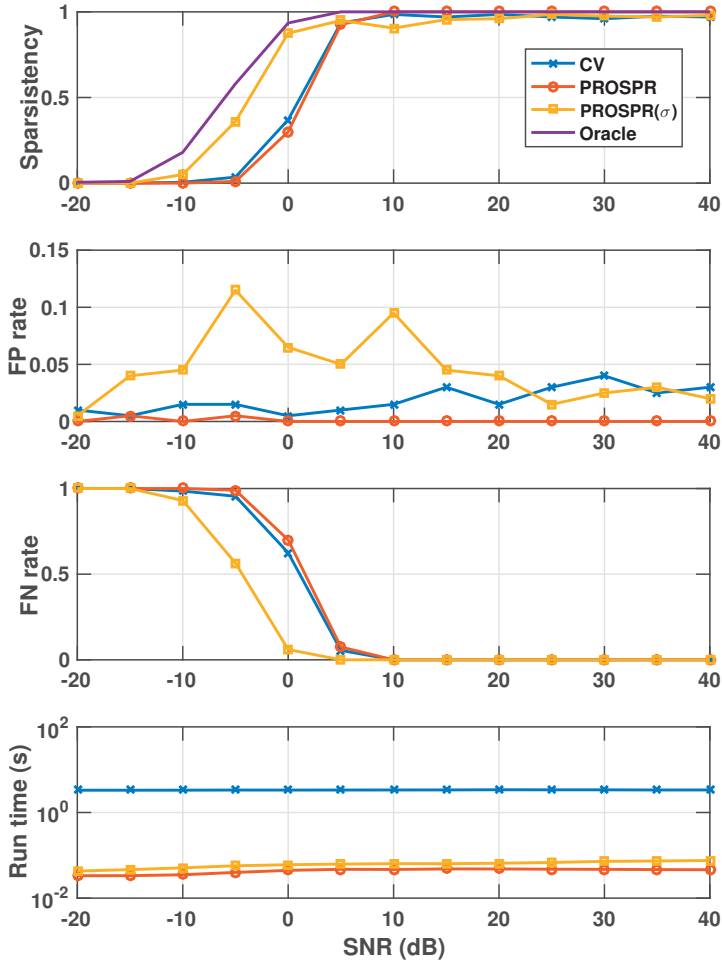


Figure 6: Compared performance results with reweighted group-LASSO using PROSPR (with and without σ-correction), CV, and in the top plot, an oracle method, illustrating the best result achievable at any regularization level. The top plot shows sparsistency (or support recovery rate), the second plot shows the FP rate, the third shows the FN rate, and the bottom plot shows the average run times for each method.

The proposed method is thus not hyperparameter-free, and in some sense merely replaces one hyperparameter by another. However, the sparse regression model does not contain enough information to be hyperparameter-free on its own; other methods for hyperparameter-selection will also require assumptions on the model, e.g., CV assumes that the optimal regularization level is the one yielding the smallest prediction error. Similarly, SPICE simply selects $\mu = 1$. In this work, we have shown that by selecting $0 < \alpha < 1$ in μ_α, one approximately select the FP rate for support recovery. If set too generous, FPs are likely but for low SNRs, FNs become less likely, and conversely, if set too small, the solution is likely to be sparse, but might omit parts of the sought support. We argue that α is relatively easy to set heuristically, whereas the regularization level, λ, is much more difficult to set appropriately. We have also shown that the median quantile, i.e., $\alpha = 0.5$, will approximate the CV's regularization level, which, when used for the reweighted LASSO problem, gives high rates of support recovery.

The great virtue of the proposed method lies in the computational complexity; CV is often computationally burdensome, even infeasible for some applications, solving the LASSO problem again and again for different regularization levels, while the proposed method is independent of the examined data. It only requires knowing the approximate shape of the noise distribution. This may often be found using secondary noise-only data, or using some standard estimation procedure. Furthermore, the family of the noise distribution is not required to be specifically known, as it may suffice to draw samples from its empirical distribution function.

References

[1] J. J. Fuchs, "On the Use of Sparse Representations in the Identification of Line Spectra," in *17th World Congress IFAC*, Seoul, July 2008, pp. 10 225–10 229.

[2] S. Bourguignon, H. Carfantan, and J. Idier, "A sparsity-based method for the estimation of spectral lines from irregularly sampled data," *IEEE Journal of Selected Topics in Signal Processing*, December 2007.

[3] J. Fang, F. Wang, Y. Shen, H. Li, and R. S. Blum, "Super-Resolution Compressed Sensing for Line Spectral Estimation: An Iterative Reweighted Approach," *IEEE Trans. Signal Process.*, vol. 64, no. 18, pp. 4649–4662, September 2016.

[4] I. F. Gorodnitsky and B. D. Rao, "Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, March 1997.

[5] D. Malioutov, M. Cetin, and A. S. Willsky, "A Sparse Signal Reconstruction Perspective for Source Localization With Sensor Arrays," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3010–3022, August 2005.

[6] S. I. Adalbjörnsson, T. Kronvall, S. Burgess, K. Åström, and A. Jakobsson, "Sparse Localization of Harmonic Audio Sources," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 117–129, January 2016.

[7] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-Pitch Estimation Exploiting Block Sparsity," *Elsevier Signal Processing*, vol. 109, pp. 236–247, April 2015.

[8] T. Kronvall, M. Juhlin, S. I. Adalbjörnsson, and A. Jakobsson, "Sparse Chroma Estimation for Harmonic Audio," in *40th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Brisbane, Apr. 19-24 2015.

[9] F. Elvander, T. Kronvall, S. I. Adalbjörnsson, and A. Jakobsson, "An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization," *Elsevier Signal Processing*, vol. 127, pp. 56–70, October 2016.

[10] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society B*, vol. 67, no. 1, pp. 91–108, January 2005.

[11] A. S. Stern, D. L. Donoho, and J. C. Hoch, "NMR data processing using iterative thresholding and minimum l1-norm reconstruction," *J. Magn. Reson.*, vol. 188, no. 2, pp. 295–300, 2007.

[12] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying N-dimensional Signals," *Elsevier Signal Processing*, vol. 128, pp. 309–317, Nov 2016.

[13] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[14] D. Donoho, M. Elad, and V. Temlyakov, "Stable Recovery of Sparse Overcomplete Representations in the Presence of Noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, Jan 2006.

[15] J. Fan and R. Li, "Variable selection via non-concave penalized likelihood and its oracle properties," *Journal of the Amer. Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, 2001.

[16] M. Yuan and Y. Lin, "Model Selection and Estimation in Regression with Grouped Variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9868.2005.00532.x

[17] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[18] E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[19] Y. V. Eldar, P. Kuppinger, and H. Bolcskei, "Block-Sparse Signals: Uncertainty Relations and Efficient Recovery," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, 2010.

[20] I. CVX Research, "CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta," `http://cvxr.com/cvx`, Sep. 2012.

[21] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, April 2004.

[22] T. Kronvall, F. Elvander, S. Adalbjörnsson, and A. Jakobsson, "Multi-Pitch Estimation via Fast Group Sparse Learning," in *24rd European Signal Processing Conference*, Budapest, Hungary, 2016.

[23] C. D. Austin, R. L. Moses, J. N. Ash, and E. Ertin, "On the Relation Between Sparse Reconstruction and Parameter Estimation With Model Order Selection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 560–570, 2010.

[24] T. Sun and C. H. Zhang, "Scaled sparse linear regression," *Biometrika*, vol. 99, no. 4, p. 879, 2012. [Online]. Available: +http://dx.doi.org/10.1093/biomet/ass043

[25] P. Stoica, D. Zachariah, and L. Li, "Weighted SPICE: A Unified Approach for Hyperparameter-Free Sparse Estimation," *Digit. Signal Process.*, vol. 33, pp. 1–12, October 2014.

[26] C. R. Rojas, D. Katselis, and H. Hjalmarsson, "A Note on the SPICE Method," *IEEE Transactions on Signal Processing*, vol. 61, no. 18, pp. 4545–4551, Sept. 2013.

[27] T. Kronvall, S. I. Adalbjörnsson, S. Nadig, and A. Jakobsson, "Group-Sparse Regression Using the Covariance Fitting Criterion," *Elsevier Signal Processing*, vol. 139, pp. 116 – 130, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168417301202

[28] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*. Englewood Cliffs, N.J.: Prentice-Hall, 1998.

[29] Y. Li, J. Scarlett, P. Ravikumar, and V. Cevher, "Sparsistency of l1-Regularized M-Estimators," *Journal of Machine Learning Research*, vol. 38, pp. 644–652, 2015.

[30] R. Lockhart, J. Taylor, R. Tibshirani, and R. Tibshirani, "A Significance Test for the LASSO," *Ann. Statist.*, vol. 42, no. 2, pp. 413–468, 04 2014. [Online]. Available: http://dx.doi.org/10.1214/13-AOS1175

[31] M. Pereyra, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Maximum-a-posteriori Estimation With Unknown Regularisation Parameters," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, Aug 2015, pp. 230–234.

[32] H. Zou and T. Hastie, "Regularization and Variable Selection via the Elastic Net," *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[33] E. J. Candès, M. B. Wakin, and S. Boyd, "Enhancing Sparsity by Reweighted l_1 Minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec. 2008.

[34] F. Bunea, J. Lederer, and Y. She, "The Group Square-Root Lasso: Theoretical Properties and Fast Algorithms," *IEEE Trans. Inf. Theor.*, vol. 60, no. 2, pp. 1313–1325, Feb. 2014. [Online]. Available: http://dx.doi.org/10.1109/TIT.2013.2290040

[35] P. Embrechts, T. Mikosch, and C. Klüppelberg, *Modelling extremal events for insurance and finance*. New York : Springer, 1997, formerly published in series: Applications of mathematics v 34.

[36] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman and Hall/CRC, 2015.

[37] M. Stone, "An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 44–47, 1977. [Online]. Available: http://www.jstor.org/stable/2984877

Ted Kronvall, visiting the Lone Pine Koala Sanctuary, during ICASSP 2015 in Brisbane, Australia.

Printed by Media-Tryck, Lund 2017 NORDIC ECOLABEL, 3041 0903

Doctoral Theses in Mathematical Sciences 2017:7
ISBN 978-91-7753-417-4
LUTFMS-1044-2017
ISSN 1404-0034