Parameter Estimation - in sparsity we trust

Swärd, Johan

2017

Document Version:
Publisher's PDF, also known as Version of record

Parameter Estimation
-in sparsity we trust

Johan Swärd

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 2017:6
ISSN 1404-0034

ISBN 978-91-7753-353-5 (print), 978-91-7753-354-2 (pdf)
LUTFMS-1043-2017

Printed in Sweden by Media-Tryck, Lund 2017

Contents

Abstract

This thesis is based on nine papers, all concerned with parameter estimation. The thesis aims at solving problems related to real-world applications such as spectroscopy, DNA sequencing, and audio processing, using sparse modeling heuristics. For the problems considered in this thesis, one is not only concerned with finding the parameters in the signal model, but also to determine the number of signal components present in the measurements. In recent years, developments in sparse modeling have allowed for methods that jointly estimate the parameters in the model and the model order. Based on these achievements, the approach often taken in this thesis is as follows. First, a parametric model of the considered signal is derived, containing different parameters that capture the important characteristics of the signal. When the signal model has been determined, an optimization problem is formed aimed at finding the parameters in the model as well as the model order. An important aspect when formulating the optimization problem is to include the characteristics and properties inherent in the signal model. For instance, if we know that the true set of parameters are smooth, this should also be a requirement reflected in the optimization problem. In the ideal case, the optimization problem is convex, in which case powerful solvers exist that may be used for finding the solution. In many cases, however, the original optimization problem is rather complex and definitely not convex. In this case, a common approach is to use a convex relaxation that approximates the original problem. In papers A, B, C, E, F, and H, this approach is utilized, however in different variations and for different applications. Paper A deals with estimation of periodic signals in symbolic sequences used in DNA sequences, paper B looks at the estimation of multi-dimensional sinusoids for NMR data, paper C considers the estimation of an unknown number of chirps for audio signals, papers E and F study pitch estimation, where the first paper considers online estimation and where the second paper proposes an off-grid method. Paper D proposes a generalization of a popular estimation method, whereas paper G introduces a new approach to frequency estimation. Paper I investigates how to sample a partially know signal to minimize the number of samples needed given a lower bound on the desired estimation performance. In all papers, the proposed methods are examined using simulated

and/or measured data and compared to competing state-of-the-art methods.

Acknowledgements

Roughly five years ago, I received a Skype call from Prof. Andreas Jakobsson. It was a sunny day in Osaka, Japan, where Ted Kronvall and I were in the middle of writing our master thesis. Andreas called us to let us know that he was going to offer us the two PhD-positions we had, a couple of months earlier, applied for. I contemplated long and hard before accepting. Excited as I was for my first job, I still had a small doubt; was it really that interesting and fun to be a PhD-student? Today, I can answer that question in the affirmative. The past five years have been fantastic. I have met a lot of interesting people, have had the privilege to visit some really wonderful places, and, most importantly, learned a lot. Prof. Andreas Jakobsson has, of course, played an integral part in making these five years so great for me. His optimism and endless enthusiasm did not only inspire me as a PhD-student but, indeed, as a person. I could not have asked for a better supervisor. The same goes for my co-authors: Stefan Ingi Adalbjörnsson, Filip Elvander, Ted Kronvall, Johan Brynolfsson, and Maksim Butsenko. We have spent many days working hard on different projects; sharing moments of happiness as well as despair. We have also created a lot of great memories on our many travels, eating good food, and drinking nice wine, discussing subjects such as skepticism, politics, evolution, and Counter Strike. To you all, an enormous thank you.

I would also like to extend a warm thank you to all my colleagues at Mathematical Statistics in Lund; there is always some interesting subject being discussed over our Fika-times. Special thanks to Mona Forsler, James Hakim, Maria Lövgren, Joakim Lübeck, Lise-Lotte Mörner, and Natasa Olsson for helping out with all the practical matters, where I owe Lise-Lotte Mörner extra gratitude for helping me out with all the paper work and preparations for my US visit.

I would also like to thank Prof. Hongbin Li at Stevens Institute of Technology, Hoboken, New Jersey, for inviting me to his laboratory. I learned a lot from my visit and from Prof. Li. My stay in the US would not have been as great as it was if I had not met with Xin Zhang, Xiaonan Guo, and Jian Liu. Together, we were "the Jersey Boys", and we had a great time exploring the many restaurants in Hoboken and Manhattan. Thank you guys!

To my parents, it is impossible to thank you for everything you have done for me; for all the support and love through out my life. Thank you!

Finally, to my girlfriend Cheri. You are the best thing that ever happened to me. Thank you for all the support and love that you have given me. You are truly the sunshine of my life!

Lund, 2017 *Johan Swärd*

List of papers

This thesis is based on the following papers:

- S. I. Adalbjörnsson, J. Swärd, J. Wallin, and A. Jakobsson, “Estimating Periodicities in Symbolic Sequences Using Sparse Modeling”, *IEEE Transactions on Signal Processing*, Vol. 63, No. 8, pp. 2142-2150, April 2015.
- J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, “High Resolution Sparse Estimation of Exponentially Decaying N-dimensional Signals”, *Elsevier Signal Processing Journal*, Vol. 128, pp. 309-317, November 2016.
- J. Swärd, J. Brynolfsson, A. Jakobsson, and M. Hansson-Sandsten, “Sparse Semi-Parametric Estimation of Harmonic Chirp Signals”, *IEEE Transactions on Signal Processing*, Vol. 64, No 7, pp. 1798-1807, April 2016.
- J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson,“Generalized Sparse Covariance-based Estimation”, submitted.
- F. Elvander, J. Swärd, and A. Jakobsson, “Online Estimation of Multiple Harmonic Signals”, *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 25, No. 2, pp. 273-284, February, 2017.
- J. Swärd, H. Li, and A. Jakobsson, "Off-grid Fundamental Frequency Estimation", submitted.
- M. Butsenko, J. Swärd, and A. Jakobsson, “Estimating Sparse Signals Using Integrated Wideband Dictionaries”, submitted.
- F. Elvander, J. Swärd, and A. Jakobsson, “Multi-dimensional Grid-less Estimation of Saturated Signals”, submitted.
- J. Swärd, F. Elvander, and A. Jakobsson, “Designing Optimal Sampling Schemes”, submitted.

Additional papers not included in the thesis:

1. F. Elvander, J. Swärd, and A. Jakobsson, "Grid-less Estimation of Saturated Signals", to be presented at *51st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, October 29- November 1, 2017.

2. J. Swärd, F. Elvander, and A. Jakobsson, "Designing Optimal Sampling Schemes for Multi-Dimensional Data", to be presented at *51st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, October 29-November 1, 2017.

3. M. Butsenko, J. Swärd, and A. Jakobsson, "The Zoomed Iterative Adaptive Approach", submitted.

4. X. Zhang, J. Swärd, H. Li, A. Jakobsson, and B. Himed, "A Passive Multistatic Detector Explotting the IO Waveform Sparsity", submitted.

5. J. Swärd, F. Elvander, and A. Jakobsson, "Designing Optimal Sampling Schemes", to be presented at *25th European Signal Processing Conference*, Kos, Greece, 2017.

6. M. Klasson, S. I. Adalbjörnsson, J. Swärd, and S. V. Andersen, "Conjugate-Prior-Regularized Multinomial pLSA for Collaborative Filtering", to be presented at *25th European Signal Processing Conference*, Kos, Greece, 2017.

7. J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Generalization of the Sparse iterative Covariance-based Estimator", *The 42nd International Conference on Acoustics, Speech and Signal Processing*, New Orleans, USA, 2017.

8. M. Butsenko, J. Swärd, and A. Jakobsson, "Estimating Sparse Signals Using Integrated Wide-Band Dictionaries", *The 42nd International Conference on Acoustics, Speech and Signal Processing*, New Orleans, USA, 2017.

9. T. Kronvall, M. Juhlin, J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Sparse Modeling of Chroma Features", *Elsevier Signal Processing Journal*, Vol. 130, pp. 105-117, January 2017.

10. S. Lei, F. Elvander, J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Computationally Efficient Multi-Pitch Estimation Using Sparsity", *11th IMA International Conference on Mathematics in Signal Processing*, Birmingham, England, 2016.

11. S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, “Enhancing Smoothness in Amplitude Modulated Sparse Signals”, *11th IMA International Conference on Mathematics in Signal Processing*, Birmingham, England, 2016.

12. F. Elvander, J. Swärd, and A. Jakobsson, “Time-recursive multi-pitch estimation using group sparse recursive least squares”, *50th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, Nov. 6-9, 2016.

13. S. I. Adalbjörnsson, J. Swärd, M. Ö Berg, S. V. Anderson, and A. Jakobsson, “Conjugate Priors for Gaussian Emission PLSA Recommender Systems”, *The 24th European Signal Processing Conference*, Budapest, Hungary, 2016.

14. J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, “Computationally Efficient Estimation of Multi-Dimensional Spectral Lines”, *The 41st International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China, 2016.

15. M. Juhlin, T. Kronvall, J. Swärd, and A. Jakobsson, “Smooth Spectral Estimation Using Covariance Fitting”, *The 23nd European Signal Processing Conference*, Nice, France, 2015.

16. J. Swärd, J. Brynolfsson, A. Jakobsson, and M. Sandsten, "Sparse Semiparametric Chirp Estimation", *48th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, Nov. 2-5, 2014.

17. S. I. Adalbjörnsson, J. Swärd, A. Jakobsson, and T. Kronvall, “A Sparse Approach for Estimation of Amplitude Modulated Sinusoids”, *48th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, Nov. 2-5, 2014.

18. J. Swärd, J. Brynolfsson, A. Jakobsson, M. Sandsten, “Smooth 2D Frequency Estimation Using Covariance Fitting”, *22nd European Signal Processing Conference* , Lisbon, Portugal, September 1-5, 2014.

19. S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, “High resolution sparse estimation of exponentially decaying two-dimensional-signals”, *22nd European Signal Processing Conference* , Lisbon, Portugal, September 1-5, 2014.

20. J. Swärd and A. Jakobsson, “Canceling Stationary Interference Signals Exploiting Secondary Data”, *22nd European Signal Processing Conference* , Lisbon, Portugal, September 1-5, 2014.

21. J. Swärd, S. I. Adalbjörnsson, A. Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying Signals", *The 39th International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, May 4-9, 2014.

22. J. Brynolfsson, J. Swärd, A. Jakobsson, and M. Sandsten , "Smooth Time-Frequency Estimation using Covariance Fitting", *The 39th International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, May 4-9, 2014.

23. S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, "Likelihood-based Estimation of Periodicities in Symbolic Sequences", *The 21st European Signal Processing Conference*, Marrakech, Marocco, September 9-13, 2013.

24. J. Swärd, and A. Jakobsson, "Subspace-based Estimation of Symbolic Periodicities", *The 38th International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 26-31, 2013.

25. T. Kronvall, J. Swärd, and A. Jakobsson, "Non-Parametric Data-Dependent Estimation of Spectroscopic' Echo-Train Signals", *The 38th International Conference on Acoustics, Speech, and Signal Processing*, Vancouver, Canada, May 26-31, 2013.

Introduction

This thesis covers contributions from nine papers, spanning different research subjects; the first deals with the estimation of periodicities in symbolic sequences; the second outlines a method for estimating an unknown number of damped sinusoids in N-dimensional space; the third treats estimation of harmonically related chirp-signals in the time-frequency domain. The fourth paper is a more theoretical work, where a popular estimator is analyzed and improved, and in the fifth paper, we derive an online estimator of multi-pitch signals. In the sixth paper, we address the problem of off-grid estimates and propose a fast and easy method for estimating multi-pitch signals off-grid. In the seventh paper, we introduce a new signal candidate dictionary, spanning larger parts of the parameter space, which can be used to speed up the estimation procedure. In the eighth paper, we look into the problem of estimating signals which have been saturated, for instance due to limitations in the dynamic span of the analog-to-digital (AD) converter, and finally, in the ninth paper, we investigate how to efficiently sample a partly known signal; a problem that is of great interest in, e.g., the chemistry and physics community. Although at a first glance, it may seem that these papers have little in common, they all share many common characteristics. For instance, in each paper, a signal model is derived which tries to explain the behavior of the signal of interest. The models are detailed by a range of parameters, where each explains some aspects of the signal, e.g., the frequency parameter in a sinusoidal model explains the rate of oscillation in the signal, whereas the amplitude parameter explains the power of the corresponding frequency. The goal of these papers is to estimate the parameters detailing the signal and thereby allow for an accurate analysis of the signal at hand. Below follows a short introduction to some of the mathematical results used in this thesis.

1 Non-parametric methods

The herein discussed methods may be divided into three general areas; parametric estimation, semi-parametric estimation, and non-parametric estimation. In the

following, we will proceed to discuss each of these areas, and reflect on their characteristics.

The non-parametric methods make no or only minor assumptions on how the signal is constructed and are thus very general. The most famous non-parametric spectral estimation method is the periodogram, which estimates the frequency content of a signal, such that

$$\Phi(f) = \frac{1}{N}\left|\sum_{t=1}^{N} y(t)\mathrm{e}^{-2i\pi ft}\right|^2 \tag{1}$$

where N denotes the number of samples, $y(t)$ the observed sample at time t, and f the frequency. Evaluating (1) for a range of frequencies yields a spectrum, in which one can see the contribution of each frequency in the signal. Due to the efficient implementation of the fast Fourier Transform (FFT), with computational complexity as low as about $\mathcal{O}\left(N\log(N)\right)$, together with its simple interpretation, the periodogram has become one of the most widely used methods within the signal processing community. The FFT-algorithm has a long history and the implementation most used today is credited to Cooley and Tukey [1], but already Johann Carl Friedrich Gauss formulated a method that resembles the modern FFT-algorithm [2]. Although justly celebrated, the periodogram has severe limitations. The estimates are not consistent, i.e., the variance of the estimates does not tend to zero as the number of samples goes to infinity, and the spectral resolution is often inadequate for many applications due to the smearing effect [3]. As an example, Figure 1 shows the resulting periodogram estimate of a signal containing three sinusoids, with frequencies $f_1 = 0.2$, $f_2 = 0.5$, and $f_3 = 0.7$, all with unit amplitude, together with additive Gaussian white noise with a signal to noise ratio (SNR) of 10 dB. As is clear from the figure, the periodogram easily resolves all the three peaks, having a noise floor which is much below the amplitudes of the signal components. However, the amplitudes are poorly estimated, and the relative amplitude between the peaks are far from correct. This is rather common for non-parametric estimators. They are in general robust but suffer from notable variance and poor resolution. It should be noted that the resulting peaks are quite wide, which can be problematic if there are peaks that are closely located, as the two peaks might look like a single peak in the periodogram.

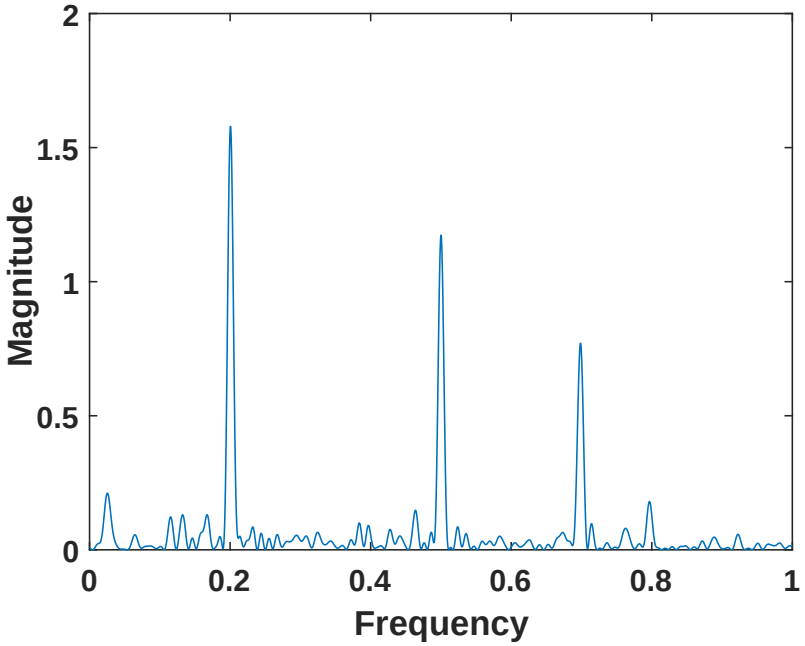

Figure 1: The periodogram estimate of three sinusoids embedded in noise.

2 Parametric methods

If one is given any prior knowledge about the signal, e.g., one knows that the signal contains three sinusoids, it may be preferable to make use of this additional information, forming a parametric estimate instead. The prior information about the signal is utilized by deriving a signal model which contains a number of parameters that are then estimated. These parameters thus explain different characteristics of the signal and can be very useful when analyzing the signal. If given accurate model information, this form of methods often experiences high resolution and low estimation variance. However, the methods are often very sensitive to model errors such as interference, or errors in the model assumptions. There are many different methods for estimating parameters. Depending on the complexity of the signal model at hand, different methods may be more suitable than others. As a simple example, consider the case when the parameters are linear arguments

in the model. One may then estimate the parameters using the least squares (LS) method. Given the output signal $y(t)$ and the input signals $\mathbf{x}(t) \in \mathbb{C}^{K\times 1}$, one is then typically interested in estimating some regression parameters $\psi \in \mathbb{C}^{1\times K}$, for $t = 0, \ldots, N-1$ such that

$$\hat{\psi}_{\text{LS}} = \underset{\psi}{\operatorname{argmin}} \sum_{t=0}^{N-1} |y(t) - \psi \mathbf{x}(t)|^2 \tag{2}$$

This minimization has an analytic solution, which is found by differentiating the sum of squares with respect to ψ and setting it equal to zero.

If the parameters instead depend on the signal in a non-linear fashion, one may similarly estimate the parameters using non-linear least squares (NLS) [3]. Given a vector of observations $\mathbf{y}$, one is then interested in estimating a set of parameters ϑ, which are detailing the function g. The NLS takes the form

$$\hat{\vartheta}_{\text{NLS}} = \underset{\vartheta}{\operatorname{argmin}} \quad ||\mathbf{y} - g(\vartheta)||_2^2 \tag{3}$$

An often interesting case is when

$$g(\vartheta) = \begin{bmatrix} \sum_{k=1}^{K} \alpha_k & \sum_{k=1}^{K} \alpha_k e^{i\varphi_k + 2i\pi f_k} & \cdots \sum_{k=1}^{K} \alpha_k e^{i\varphi_k + 2i\pi f_k (N-1)} \end{bmatrix}^T \tag{4}$$

i.e., when the signal contains K sinusoids with frequency f_k, amplitude α_k, and phase φ_k, for $k = 1, \ldots, K$. By stacking the K complex amplitudes in a vector

$$\mathbf{a} = \begin{bmatrix} \alpha_1 e^{i\varphi_1} & \ldots & \alpha_K e^{i\varphi_K} \end{bmatrix}^T \tag{5}$$

where $(\cdot)^T$ denotes the transpose, (3) may be expressed as

$$\begin{bmatrix} \hat{\vartheta}, & \hat{\mathbf{a}} \end{bmatrix} = \underset{\vartheta, \mathbf{a}}{\operatorname{argmin}} \quad \frac{1}{2} ||\mathbf{y} - \mathbf{Z}\mathbf{a}||_2^2 \tag{6}$$

where

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 & \ldots & \mathbf{z}_K \end{bmatrix} \tag{7}$$

$$\mathbf{z}_k = \begin{bmatrix} 1 & \mathrm{e}^{2i\pi f_k} & \ldots & \mathrm{e}^{2i\pi f_k (N-1)} \end{bmatrix}^T \tag{8}$$

$$\vartheta = \begin{bmatrix} f_1 & \ldots & f_K \end{bmatrix} \tag{9}$$

Noting that (6) is linear in $\mathbf{a}$, one may simplify (6) by substituting $\mathbf{a}$ with the LS solution of (6)

$$\hat{\mathbf{a}}_{\text{LS}} = \left(\mathbf{Z}^H\mathbf{Z}\right)^{-1}\mathbf{Z}^H\mathbf{y} \tag{10}$$

yielding

$$\hat{\boldsymbol{\vartheta}} = \underset{\boldsymbol{\vartheta}}{\text{argmin}} \quad \frac{1}{2}||\mathbf{y} - \mathbf{Z}\hat{\mathbf{a}}_{\text{LS}}||_2^2 \tag{11}$$

where $(\cdot)^H$ denotes the complex conjugate transpose (the Hermitian). It is worth noting that the cost function in (11) now only depends on $\boldsymbol{\vartheta}$, and may thus, possibly, be solved using a gradient method, e.g., Newton's method [4], or by, simply evaluating the cost function over a grid of candidates $\boldsymbol{\vartheta}$, although such a solution is only feasible if the dimension of $\boldsymbol{\vartheta}$ is low.

3 Semi-parametric methods

So far, one may conclude that the parametric and the non-parametric methods are somewhat the opposite of each other; the non-parametric methods are often robust to model assumptions, whereas the parametric methods are commonly fragile. On the other hand, the parametric methods generally outperform the non-parametric methods when it comes to estimation variance and bias, if using accurate model information. Now one might ask if there is a way of combining these two ideas and thereby get the best of each. Sometimes the answer to that question is yes as will be briefly discussed below.

4 Sparsity

As the word suggests, a sparse vector is a vector with only a few non-zero valued elements. The basic idea with sparse estimation methods is to set up an under-determined system and force the solution to be sparse, i.e., only a few of the candidates for the solution are selected. Let $\mathbf{y}$ be the signal on vector form and $\mathbf{D}$ be a matrix where the columns are candidate signal components for $\mathbf{y}$, i.e., we assume that a combination of the columns of $\mathbf{D}$ can well approximate $\mathbf{y}$, which yields the signal model

$$\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{e} \tag{12}$$

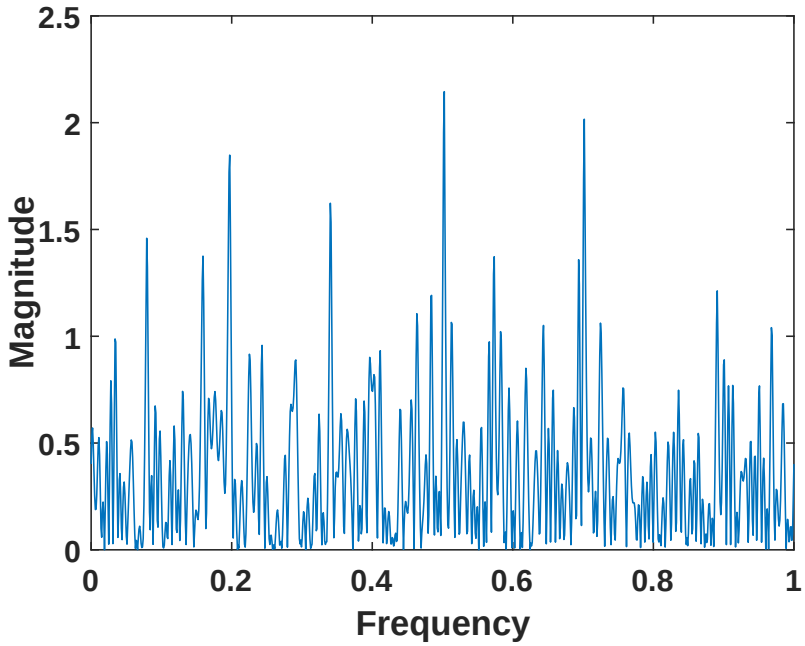

Figure 2: The estimate using LS when solving (13).

where **e** denotes the white Gaussian noise with the same dimension as **y**. The most straight forward approach for finding these columns in **D** that best approximate **y** would be to solve the LS problem

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \quad \frac{1}{2} ||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 \tag{13}$$

The solution of (13) will not be sparse and would typically suggest that a linear combination of (almost) all of the columns in **D** would yield the best approximation. Especially, if the signal **y** contains a high level of noise, (13) would generally find a solution that also models the noise. An example of this is depicted in Figure 2, where the same signal as before is used, but with amplitudes of magnitude 10 and with a much higher noise presence, SNR $= -5$ dB. As is clear from the figure, the LS estimate is very sensitive to the noise, and the peaks of the signal are hardly recognizable. A better way of solving this problem is to introduce a penalty

that enforces the solution to be sparse. A natural choice of a sparsity enforcing penalty would be one that counts the number of non-zero elements in $\mathbf{x}$ and ensures that this number is small. This penalty is often referred to as the ℓ_0-"norm", $||\mathbf{x}||_0$, and is defined as the number of the non-zero elements in $\mathbf{x}$. It should be stressed that although commonly called so, $||\mathbf{x}||_0$ is not a norm, as it does not fulfill the definition of a norm (it is not homogeneous). The problem will now be on the form

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \quad \frac{1}{2}||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 + \lambda||\mathbf{x}||_0 \tag{14}$$

where $\lambda > 0$ is a tuning parameter governing the amount of sparsity in $\mathbf{x}$. To be able to solve (14), one has to conduct an combinatorial search, which, if the size of $\mathbf{x}$ is large, is essentially infeasible. Instead, one often relaxes the ℓ_0-constraint and replace it with the ℓ_1-norm,

$$||\mathbf{x}||_1 = \sum_{i=1}^{P} |x_i| \tag{15}$$

such that the problem is reformulated as

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \quad \frac{1}{2}||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 + \lambda||\mathbf{x}||_1 \tag{16}$$

With this relaxation, we have ended up with the celebrated Lasso method [5], which was introduced in 1996. However, ideas that resembles the Lasso had been introduced much earlier, e.g. in seismology [6], and are today present in many different research areas, e.g. in portfolio optimization [7], connected graphs [8], and compressed sensing [9–11].

It is important to remember that sparsity is not uniquely defined for vectors; on the contrary, one may consider sparsity for matrices and tensors as well. In Figure 3, the Lasso estimate is shown for the signal examined in Figure 2. Comparing this estimate to the LS estimate depicted in Figure 2, one sees that the Lasso estimate is much sparser, and that the three peaks of the signal are clearly visible, here together with two spurious noise peaks. A prominent feature of the Lasso method is that it is convex. A convex problem is a problem where the cost function (the function to minimize) is convex and where the set over which the function is minimized is also convex (meaning that the equality constraints are affine and the inequality constraints are convex functions) [12]. An important

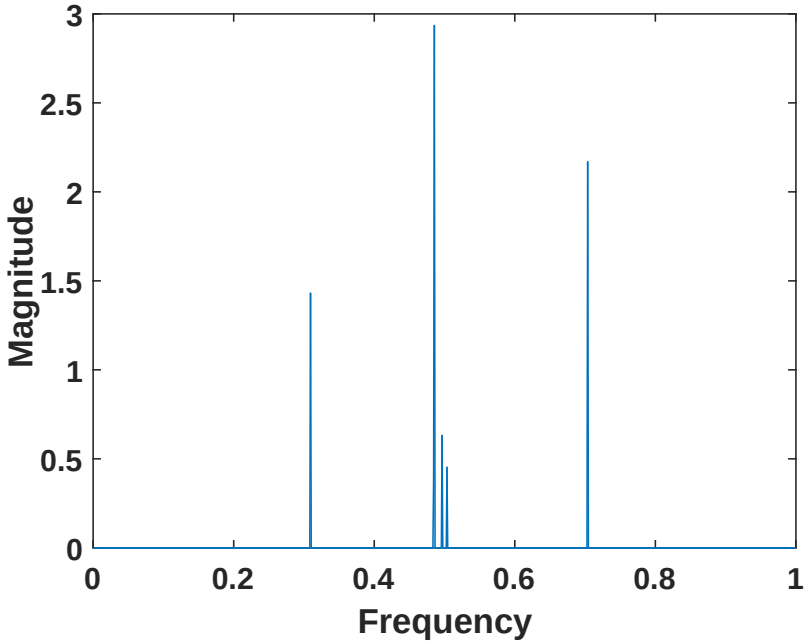

Figure 3: The Lasso estimate of the three sinusoids.

property of a convex problem is that, if one finds a locally optimal point, the point is also globally optimal. The Lasso can be seen to be a combination of the parametric and the non-parametric methods due to the fact that in (16), one uses the information of which kind of components the signal is made up by, but one does not specify how many of these components there are. This approach is thus less sensitive to model errors than many parametric approaches, but it still often manages to yield a lower estimation variance as compared to the non-parametric approaches. As a result of (16), a user parameter λ is introduced. This parameter governs the amount of sparsity allowed in the solution and is in practice often difficult to choose. Common approaches to find a suitable λ include cross-validation [13] or using some data dependent rule of thumb. In this thesis, the latter approach has been commonly used where λ has often been chosen such that $\lambda = \alpha||\mathbf{D}^H\mathbf{y}||_\infty$, with $\alpha \in [0, 1]$, thus only allowing the peaks that are at least as big as α of the largest inner-product of the dictionary and the data. In [14],

Algorithm 1 Reweighted ℓ_1

1: Initiate $\mathbf{W}^{(0)} = \mathbf{I}$ and set $k = 0$;
2: **repeat**
3: Solve for $\mathbf{x}^{(k)}$ using (17) with $\mathbf{W}^{(k)}$
4: Update $\mathbf{W}^{(k+1)}$ using (18)
5: Set $k = k + 1$
6: **until** convergence or $k_{\max}$ is reached

a method for enhancing sparsity by reweighting the ℓ_1 penalty was introduced. This will also decrease the sensitivity to the choice of λ, as long as λ is not chosen too large. The basic idea is to replace the original ℓ_1 penalty and reformulating the problem. For (16), this reformulation becomes

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \quad \frac{1}{2} ||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 + \lambda ||\mathbf{W}\mathbf{x}||_1 \tag{17}$$

where $\mathbf{W}$ is a diagonal matrix with weights $w_1, w_2, \ldots, w_P$ on the diagonal entries. A natural way of choosing the weights would be to set them equal to the magnitude of the corresponding elements in $\mathbf{x}$. This is, of course, impossible since one does not know the true $\mathbf{x}$ vector in advance (which would render the problem meaningless), but it suggests that one may find the weights iteratively by alternating the estimation of $\mathbf{x}$ using (17) and updating the weights according to

$$w_i^{(k+1)} = \frac{1}{|x_i^{(k)}| + \varepsilon} \tag{18}$$

where $\varepsilon > 0$ is a parameter introduced to ensure numerical stability, and where $x_i^{(k)}$ denotes the ith element in $\mathbf{x}$ at the kth iteration. Algorithm 1 summarizes the discussed scheme, where $\mathbf{I}$ denotes the $P \times P$ identity matrix. The algorithm is a variant of the Majorization-Minimization (MM) algorithm and resembles the log-sum penalty function $\sum_{i=1}^{P} \log(|x_i| + \varepsilon)$, which introduces a larger sparsity penalty than the original ℓ_1 penalty (see e.g. [14] for a thorough discussion on the subject). Figure 4 shows the resulting estimate for the reweighted Lasso. Comparing to the previous methods, it is clear that the reweighted Lasso estimate is the sparsest, and that the method is able to estimate the sinusoids correctly. Furthermore, as the method only details the signal part, the noise floor is fully reduced to zero.

The sparse based methods often introduce bias in the amplitude estimates. This can also be seen in Figure 4, where the peaks are somewhat offset from their

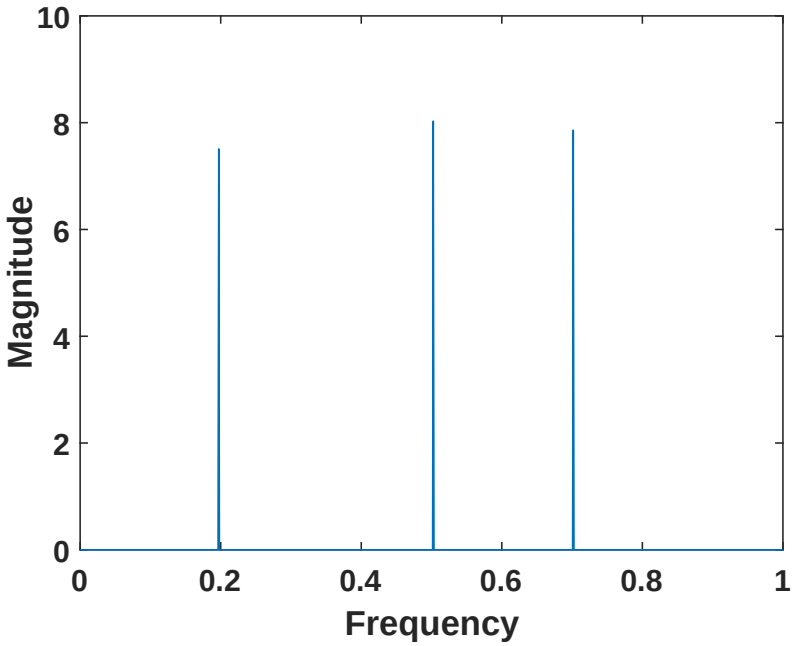

Figure 4: The reweighted Lasso estimate of the three sinusoids.

true magnitudes (being 10 for all three components). However, this is often not of serious concern since once the correct support is found, thus the frequencies in the sinusoidal case, the amplitude estimates may be refined using, e.g. NLS.

5 Efficient implementation

In recent years, convex optimization has become very popular and many researchers have contributed to the area. Today, powerful convex solvers, such as SDPT3 [15] and SeDuMi [16], are freely available to the public. This has further accelerated the interest for convex optimization in the research community. Even though the available solvers are very general and powerful, they are also (or therefore) complex and suffers from high computational complexity. In many cases, it is, therefore, advisable to implement a faster solver directly for the problem at hand and instead use the powerful solvers as benchmarks. A method for such an

efficient implementation that is used many times in this thesis is the Alternating Direction Method of Multipliers (ADMM), which is both simple and powerful. It works by separating the original large problem into smaller subproblems, which are then iteratively solved one by one, and then coordinated to find a solution to the original problem. This approach makes the ADMM suitable for distributed optimization, where the subproblems are distributed to different processes, thus parallelizing the computation, allowing for shorter computation time. Another beneficial feature is that given some mild assumptions (such that the problem is convex and that there exists a solution), the ADMM will converge to the true solution [17–19]. In general, the ADMM solves problems in the form

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) + g(\mathbf{z}) \\ &\text{subject to} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c} \end{aligned} \tag{19}$$

where $f(\cdot)$ and $g(\cdot)$ are convex functions and the matrices $\mathbf{A}$ and $\mathbf{B}$ and the vector $\mathbf{c}$ are all known. The ADMM solves the problem by solving for each variable separately, in an iteratively fashion. To derive the steps in the algorithm, the augmented Lagrangian is formed

$$L(\mathbf{x}, \mathbf{z}, \mathbf{y}) = f(\mathbf{x}) + g(\mathbf{z}) + \mathbf{y}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}) + \frac{\rho}{2}||\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}||_2^2 \tag{20}$$

where $\mathbf{y}$ denotes the dual variable and ρ the augmented Lagrangian parameter. If one defines the scaled dual variable as $\mathbf{u} = (1/\rho)\mathbf{y}$, one ends up with the simpler form

$$L(\mathbf{x}, \mathbf{z}, \mathbf{u}) = f(\mathbf{x}) + g(\mathbf{z}) + \frac{\rho}{2}||\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c} + \mathbf{u}||_2^2 \tag{21}$$

The steps in the ADMM are then derived by first minimizing (21) with respect to $\mathbf{x}$ and then $\mathbf{z}$, yielding, for iteration $k+1$,

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x}}{\operatorname{argmin}} \left(f(\mathbf{x}) + \frac{\rho}{2}||\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}^{(k)} - \mathbf{c} + \mathbf{u}^{(k)}||_2^2 \right) \tag{22}$$

$$\mathbf{z}^{(k+1)} = \underset{\mathbf{z}}{\operatorname{argmin}} \left(g(\mathbf{z}) + \frac{\rho}{2}||\mathbf{A}\mathbf{x}^{(k+1)} + \mathbf{B}\mathbf{z} - \mathbf{c} + \mathbf{u}^{(k)}||_2^2 \right) \tag{23}$$

and then update the scaled dual variable

$$\mathbf{u}^{(k+1)} = \mathbf{A}\mathbf{x}^{(k+1)} + \mathbf{B}\mathbf{z}^{(k+1)} + \mathbf{u}^{(k)} - \mathbf{c} \tag{24}$$

As an example, the steps in ADMM for the Lasso problem in (16) are derived for a real valued problem. First, the variable $\mathbf{x}$ is split into two new variables, here denoted $\mathbf{x}$ and $\mathbf{z}$, yielding the optimization problem

$$\underset{\mathbf{x},\mathbf{z}}{\text{minimize}} \quad \frac{1}{2}||\mathbf{y}-\mathbf{D}\mathbf{x}||_2^2+\lambda||\mathbf{z}||_1 \tag{25}$$

Noting that $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{c}$ in (21) are, for (25), $\mathbf{A}=\mathbf{I}$, $\mathbf{B}=-\mathbf{I}$, and $\mathbf{c}=0$, the augmented Lagrangian for (25) is

$$L(\mathbf{x},\mathbf{z},\mathbf{u})=\frac{1}{2}||\mathbf{y}-\mathbf{D}\mathbf{x}||_2^2+\lambda||\mathbf{z}||_1+\frac{\rho}{2}||\mathbf{x}-\mathbf{z}+\mathbf{u}||_2^2 \tag{26}$$

The ADMM step for $\mathbf{x}$ follows from differentiating (26) with respect to $\mathbf{x}$ and setting it equal to zero, yielding

$$\mathbf{x}^{(k+1)}=\left(\mathbf{D}^T\mathbf{D}+\rho\mathbf{I}\right)^{-1}\left(\mathbf{D}^T\mathbf{y}+\rho(\mathbf{z}^{(k)}-\mathbf{u}^{(k)})\right) \tag{27}$$

For $\mathbf{z}$ it is more complicated, since $||\mathbf{z}||_1$ is not differentiable if one element is equal to zero, but using sub-gradients and letting

$$J(\mathbf{z})=\lambda||\mathbf{z}||_1+\frac{\rho}{2}||\mathbf{x}-\mathbf{z}+\mathbf{u}||_2^2 \tag{28}$$

we may express $\frac{\partial J(\mathbf{z})}{\partial z_j}$ as

$$\frac{\partial J(\mathbf{z})}{\partial z_j}=\begin{cases}\rho(z_j-x_j-u_j)-\lambda & \text{if } z_j<0\\ \{\rho(z_j-x_j-u_j)-\lambda,\rho(z_j-x_j-u_j)+\lambda\} & \text{if } z_j=0\\ \rho(z_j-x_j-u_j)+\lambda & \text{if } z_j>0\end{cases} \tag{29}$$

By then setting (29) equal to zero and solving for $\mathbf{z}$, the solution may be compactly expressed using the soft threshold

$$S(\mathbf{v},\alpha)_i=\frac{v_i}{|v_i|}\max\{0,|v_i|-\alpha\}, \quad \text{for } i=1,\ldots,P \tag{30}$$

where $\mathbf{v}=\mathbf{x}^{(k+1)}+\mathbf{u}^{(k)}$, and $\alpha=\lambda/\rho$. The last step of the ADMM is then to update the scaled dual variable using (24).

6 Off-grid estimation

In all the above presented examples, we have assumed that the dictionaries have contained the actual signal frequencies. This is of course the ideal situation when using a grid-based method for estimation. However, the probability that the grid actually contains the true parameters is generally small, at least when dealing with signals from real world applications. Instead, one has to hope that the grid-points are close enough to the true parameters [20]. In an effort to close the gap between the closest grid point and the true parameter, it is tempting to increase the grid size. Even though this makes intuitive sense, it has two major drawbacks. First, when increasing the grid size, the size of the problem will become larger, and the computational cost will grow. Since many of the off-the-shell convex solvers, like, e.g., CVX [21], scale badly with the number of grid points, this can result in problems that practically takes too long time to solve. The second drawback is the problem that the dictionary matrix becomes coherent, which means that the columns in the matrix become more correlated as the grid spacing decreases. This may in turn result in a decrease in performance, especially for signal reconstruction problems [22, 23].

Recently, there has been notable attention directed to solve the problem with grid mismatch [22]. One idea that has been vigorously studied is using adaptive grids. The idea behind adaptive grids is to let the grid points be a part of the optimization problem. For the here studied Lasso method in (16), this would mean that both $\mathbf{x}$ and the frequency grid are optimization variables, thus solving

$$\underset{\mathbf{x},\boldsymbol{\vartheta}}{\text{minimize}} \quad ||\mathbf{y} - \mathbf{A}(\boldsymbol{\vartheta})\mathbf{x}||_2^2 + \lambda||\mathbf{x}||_1 \tag{31}$$

At a first glance, this looks like an awful problem to solve; not only do we have to find $\mathbf{x}$, we also have to change the columns of the dictionary, which in turn affects $\mathbf{x}$. Fortunately, these problems often allow for separating the optimization problem, such that one may first solve for $\mathbf{x}$ using a coarse grid, and then update the grid. This is then iterated until the method has converged or until some predefined stopping criteria is fulfilled. Algorithm 2 shows the underlying idea of the method, where $F(\mathbf{x}, \boldsymbol{\vartheta})$ denotes the cost function to be minimized, $\mathbf{x}$ the original optimization variable, and $\boldsymbol{\vartheta}$ the vector containing the adaptive grid points. The adaptive grid methods are often quite easy to use, but they have their limitations. First, one has to make sure that the initial grid is not too coarse, such that true signal components are missed. For instance, if the initial grid is too coarse, one

Algorithm 2 Adaptive Grid Method

1: Given a dictionary matrix, $\mathbf{A}(\boldsymbol{\vartheta}^{(0)})$, with initial grid $\boldsymbol{\vartheta}^{(0)}$, and an upper limit on the number of iterations $I_{\max} \in \mathbb{Z}_+$.
2: **for** $i = 1, \ldots$ **do**
3: Compute $\mathbf{x}^{(i)} = \underset{\mathbf{x}}{\operatorname{argmin}}\, F(\mathbf{x}, \boldsymbol{\vartheta}^{(i-1)})$ as a function of $\boldsymbol{\vartheta}^{(i-1)}$.
4: Compute $\boldsymbol{\vartheta}^{(i)} = \underset{\boldsymbol{\vartheta}}{\operatorname{argmin}}\, F(\mathbf{x}^{(i)}, \boldsymbol{\vartheta})$.
5: Terminate after convergence or if $i \geq I_{\max}$.
6: **end for**

may find that one and the same grid point is the closest grid point to two signal components. This can cause the grid point to get stuck in between the two true signal components, and thus one of the signal components will be lost, whereas the other one is poorly estimated. Another disadvantage is that the optimization problem is no longer convex, and thus one can no longer guarantee that a local optimum is also the global optimum.

An alternative to the adaptive grid approach is to use an infinite grid. This can be done by solving an atomic norm minimization problem [24]. The idea with using the atomic norm is to specify the building blocks that make up the signal, the so called atoms. For the sinusoidal case, these become [25]

$$\mathbf{a}(f, \varphi) = \begin{bmatrix} \mathrm{e}^{i\varphi} & \mathrm{e}^{2i\pi f + i\varphi} & \ldots & \mathrm{e}^{2i\pi f(N-1)+i\varphi} \end{bmatrix}^T \tag{32}$$

The set of all these atoms is thus

$$\mathcal{A} = \{\mathbf{a}(f, \varphi) : f \in [0, 1], \varphi \in [0, 2\pi]\} \tag{33}$$

The signal model can now be represented as a linear combination of the atoms

$$\mathbf{y} = \sum_{k=1}^{K} \tilde{c}_k \mathbf{a}(f_k, \varphi_k) = \sum_{k=1}^{K} c_k \mathbf{a}(f_k, 0), \;\; \mathbf{a}(f_k, \varphi_k) \in \mathcal{A} \tag{34}$$

where $c_k = \tilde{c}_k \mathrm{e}^{i\varphi_k}$. The atomic norm is defined as

$$||\mathbf{y}||_{\mathcal{A}} \triangleq \inf\{t > 0 : \mathbf{y} \in t \operatorname{conv}(\mathcal{A})\} \tag{35}$$

$$= \inf\left\{\sum_k c_k : \mathbf{y} = \sum_k c_k \mathbf{a}_k, c_k > 0, \mathbf{a}_k \in \mathcal{A}\right\} \tag{36}$$

where conv($\mathcal{A}$) denotes the convex hull of the set $\mathcal{A}$. The atomic norm is a proper norm if conv ($\mathcal{A}$) is compact, centrally symmetric, and contains a ball of radius ε around the origin for some $\varepsilon > 0$. To minimize (35) in the sinusoidal case is thus equivalent to finding the smallest sum of magnitudes for all linear combinations of sinusoids that fully explains the signal. In [26] it is shown that for the sinusoidal case, minimizing (35) is equivalent to

$$||\mathbf{y}||_{\mathcal{A}} = \inf\left\{\frac{1}{2}(x + \mathbf{T}_{1,1}) : \begin{bmatrix} \mathbf{T} & \mathbf{y} \\ \mathbf{y}^H & x \end{bmatrix} \geq 0\right\} \tag{37}$$

where $\mathbf{T} \in \mathbb{T}$, with $\mathbb{T}$ denoting the set of all Hermitian Toeplitz matrices, and where $\mathbf{T}_{1,1}$ denotes the first element of the first row of $\mathbf{T}$. To prove (37), we first need the following lemma [26] (Caratheodory-Toeplitz)

Lemma 6.1. *Any positive semidefinite Toeplitz matrix* $\mathbf{T}$ *can be represented as*

$$\mathbf{T} = \mathbf{APA}^H$$

where

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{a}(f_1, 0) & \dots & \mathbf{a}(f_r, 0) \end{bmatrix} \\ \mathbf{P} &= \text{diag}\left([d_1 \dots d_r]\right) \end{aligned} \tag{38}$$

where $d_k > 0$ *are real numbers, and* $r = \text{rank}(\mathbf{P})$.

For the sake of completeness, the proof of (37), which was first given in [26], will be presented.
Proof of (37):
Let the right hand side of (37) be denoted $\Gamma(\mathbf{y})$. First assume that $\mathbf{y} = \sum_k c_k \mathbf{a}(f_k, \varphi_k)$ with $c_k > 0$ and that the signal has been sampled at $t = 0, \dots, N-1$. Define $\mathbf{T} = \sum_k c_k \mathbf{a}(f_k, \varphi_k)\mathbf{a}(f_k, \varphi_k)^H$ and let $x = \sum_k c_k$. Then,

$$\begin{bmatrix} \mathbf{T} & \mathbf{y} \\ \mathbf{y}^H & x \end{bmatrix} = \sum_k c_k \begin{bmatrix} \mathbf{a}(f_k, \varphi_k) \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{a}(f_k, \varphi_k) \\ 1 \end{bmatrix}^H \geq 0 \tag{39}$$

Thus,

$$\mathbf{T}_{1,1} = x = \sum_k c_k \Rightarrow \Gamma(\mathbf{y}) \leq \sum_k c_k \tag{40}$$

This holds for any decomposition of $\mathbf{y}$, thus

$$||\mathbf{y}||_{\mathcal{A}} \geq \Gamma(\mathbf{y}) \tag{41}$$

Conversely, suppose that for some Toeplitz matrix $\mathbf{T} \geq 0$ and some complex vector $\mathbf{y}$, we have

$$\begin{bmatrix} \mathbf{T} & \mathbf{y} \\ \mathbf{y}^H & x \end{bmatrix} \geq 0 \tag{42}$$

From lemma 6, we have that

$$\mathbf{T} = \mathbf{A}\mathbf{P}\mathbf{A}^H = \sum_k p_k \mathbf{a}(f_k, 0)\mathbf{a}^H(f_k, 0) \tag{43}$$

thus, $\frac{1}{N}\text{trace}\,(\mathbf{T}) = \text{trace}\,(\mathbf{P})$, since $||\mathbf{a}(f_k, 0)||_2 = \sqrt{N}$. Further, due to the Toeplitz structure, we have that $\frac{1}{N}\text{trace}\,(\mathbf{T}) = \mathbf{T}_{1,1}$. Using the Vandermonde decomposition and (42), it follows that $\mathbf{y}$ is in the range of $\mathbf{T}$, and thus also in the range of $\mathbf{A}$. This means that

$$\mathbf{y} = \sum_k w_k \mathbf{a}(f_k, 0) = \mathbf{A}\mathbf{w} \tag{44}$$

for some complex vector $\mathbf{w}$. Moreover, the Schur complement yields

$$\begin{bmatrix} \mathbf{T} & \mathbf{y} \\ \mathbf{y}^H & x \end{bmatrix} \geq 0 \iff \mathbf{T} \geq x^{-1}\mathbf{y}\mathbf{y}^H \tag{45}$$

resulting in that $\mathbf{A}\mathbf{P}\mathbf{A}^H \geq x^{-1}\mathbf{A}\mathbf{w}\mathbf{w}^H\mathbf{A}^H$. Since $\mathbf{A}$ has full rank, there exists a vector $\mathbf{q}$ such that $\mathbf{A}^H\mathbf{q} = \text{sign}(\mathbf{w})$. Then,

$$x\mathbf{T}_{1,1} = \text{trace}\,(\mathbf{P})\,x = x\mathbf{q}^H\mathbf{A}\mathbf{P}\mathbf{A}^H\mathbf{q} \geq \mathbf{q}^H\mathbf{A}\mathbf{w}\mathbf{w}^H\mathbf{A}^H\mathbf{q} \tag{46}$$

$$= (\text{sign}\,(\mathbf{w})\,\mathbf{w}))^H \text{sign}\,(\mathbf{w})\,\mathbf{w} = \left(\sum_k |w_k|\right)^2 \tag{47}$$

yielding $x\mathbf{T}_{1,1} \geq \left(\sum_k |w_k|\right)^2$. By the arithmetic geometric mean inequality [27], we have

$$\frac{1}{2}\left(\mathbf{T}_{1,1} + x\right) \geq \sqrt{\mathbf{T}_{1,1}x} \geq \sum_k |w_k| \geq ||\mathbf{y}||_{\mathcal{A}} \tag{48}$$

implying that $\Gamma(\mathbf{y}) \geq ||\mathbf{y}||_{\mathcal{A}}$. We have now shown that $\Gamma(\mathbf{y}) \leq ||\mathbf{y}||_{\mathcal{A}} \leq \Gamma(\mathbf{y})$ meaning that $||\mathbf{y}||_{\mathcal{A}} = \Gamma(\mathbf{y})$, which concludes the proof. □

In the context of this presentation, the atomic norm has two interesting applications. As was mentioned before, it allows one to estimate the signal frequencies without the use of a grid, and thus serves as a grid-less frequency estimator. The second application is missing data. We can use the atomic norm formulation to recover the true signal from only a subset of the elements in the vector. If we denote Ω as the set of elements corresponding to the observed samples in the signal vector $\mathbf{s}$, we may form the optimization problem as [26]

$$\begin{aligned} \underset{\mathbf{y},\mathbf{T}\in\mathbb{T},x}{\text{minimize}} \quad & \frac{1}{2}\left(\mathbf{T}_{1,1}+x\right) \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{T} & \mathbf{y} \\ \mathbf{y}^H & x \end{bmatrix} \geq 0 \\ & \mathbf{y}_\Omega = \mathbf{s}_\Omega \end{aligned} \tag{49}$$

where $\mathbf{y}_\Omega$ selects the elements in $\mathbf{y}$ corresponding to Ω. The atomic norm formulation we have seen above assumes that the observed signal is noise free. To accommodate for noisy signals as well, one may instead solve [25, 28]

$$\begin{aligned} \underset{\mathbf{z},\mathbf{T}\in\mathbb{T},x}{\text{minimize}} \quad & \frac{\tau}{2}\left(\mathbf{T}_{1,1}+x\right) + \frac{1}{2}||\mathbf{s}_\Omega - \mathbf{z}_\Omega||_2^2 \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{T} & \mathbf{z} \\ \mathbf{z}^H & x \end{bmatrix} \geq 0 \end{aligned} \tag{50}$$

where τ, similar to λ in (16), is a hyper-parameter governing the allowed sparsity in the solution.

One question still remains: when we have solved (49) or (50), how do we find the frequency estimates? There are two answers to this question. The first one finds the frequencies by using the fact that the optimal $\mathbf{T}$ is a Vandermonde matrix. Thus, using the Vandermonde decomposition [26], one may retrieve the frequencies. The second approach to retrieve the frequencies is via the dual problem [26]. In this thesis, the first approach has been used and we refer the interested reader to [26] for a discussion on how to use the dual problem to find the frequencies.

This chapter is concluded by demonstrating the results from solving (50) as compared to the Lasso formulation in (16) and the reweighted Lasso in (17).

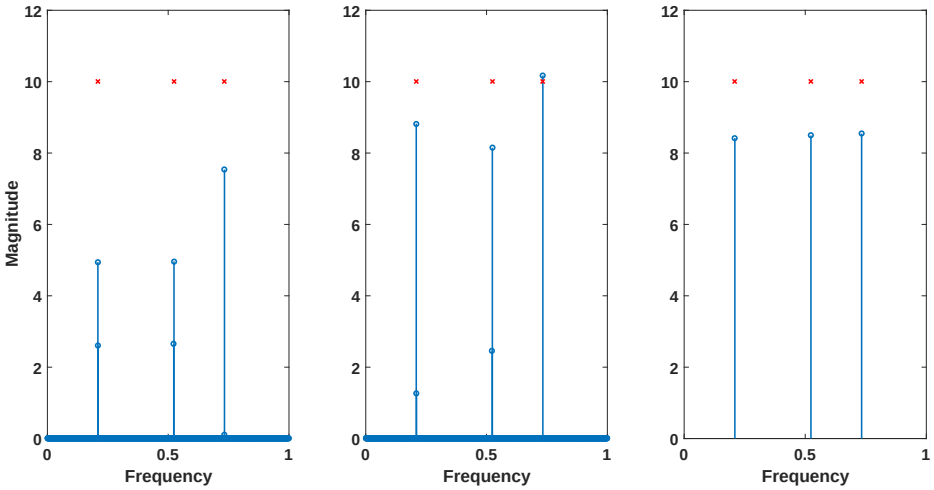

Figure 5: Left: the Lasso estimate. Middle: the reweighted Lasso estimate. Right: the atomic norm estimate.

This time, the frequencies are deliberately selected not on the frequency grid, being selected as

$$\mathbf{f} = \begin{bmatrix} 0.2 & 0.5 & 0.7 \end{bmatrix} \pi/3 \tag{51}$$

Further, the SNR level is increased to 10 dB. Figure 5 shows the resulting estimates from the three methods. The left figure shows the result from the Lasso, where it can be seen that the power of the true peaks has been split to the closest grid points. In the middle figure, the results from the reweighted Lasso in shown. Here the amplitude estimates have become better, but the splitting of the peaks are still visible. In the right figure, the estimates from the atomic norm method are shown. Since this method does not depend on any grid structure, there are no splitting of the peaks, and the resulting frequency estimates are closer to the true frequencies compared to the other two methods. Note that the amplitude estimates are biased. As mentioned above, this can easily be corrected for by reestimating the amplitudes using, e.g., NLS, once the frequencies are found.

7 Outline of the papers

Paper A: Estimating Periodicities in Symbolic Sequences Using Sparse Modeling

In the first paper, the task of finding hidden periodicities in symbolic sequences is considered. Previously, the by far most commonly used approach to determine the periodicities has been to map the symbols into a numerical representation and then apply standard frequency estimation techniques on the transformed data. In this paper, we formulate a likelihood-based sparse logistic regression model, which models the probability of each symbol being present at the considered periodic index sets. We present two different methods for maximizing the likelihood. The first one is a greedy approach, where each index set is added to the likelihood, one at a time. The procedure is terminated when it is statistically unlikely that the signal contains an additional periodicity. The second method is a cyclic coordinate descent algorithm that maximizes the penalized likelihood. The methods are evaluated on simulated and real symbolic data, showing superior performance as compared to competing methods. The work in paper A has been published in part as

> Stefan Ingi Adalbjörnsson, Johan Swärd, Andreas Jakobsson, "Likelihood-based Estimation of Periodicities in Symbolic Sequences", 21st European Signal Processing Conference, Marrakech, Morocco, September 9-13, 2013.

and has been published in full as

> Stefan Ingi Adalbjörnsson, Johan Swärd, Jonas Wallin, and Andreas Jakobsson, "Estimating Periodicities in Symbolic Sequences Using Sparse Modeling", *IEEE Transactions on Signal Processing*, Vol. 63, No. 8, pp. 2142-2150, April 2015.

Paper B: High Resolution Sparse Estimation of Exponentially Decaying N-D Signals

In the second paper, we set out to estimate the parameters detailing an unknown number of N dimensional exponentially decaying sinusoids. For small dimensional problems, a common approach is to form a dictionary containing finely spaced parameter candidates for the signal at hand. If either of the parameter space and/or the signal space are large, the dictionary easily becomes vast, even if

the candidate parameters are sparsely distributed on the grid. In this paper, we propose a method that exploits the Kronecker structure inherent in the model, thereby drastically decreasing the computational complexity. Furthermore, we introduce a novel dictionary learning approach that iteratively refines each found component, allowing for off-grid estimation as well as for smaller dictionaries. This approach is based on the fact that it is often an easier task to find the frequency parameter than the damping parameter. Therefore, first a rough frequency estimate is found by constructing a dictionary containing a grid of frequencies, while the damping parameter is fixed. The damping parameter, as well as a refined estimate of the frequency, are then found by treating each component at the time. The method achieves, at medium to high SNR-levels, the same level of performance as statistically efficient parametric methods with oracle model order knowledge, for well-separated components. Furthermore, the proposed method is shown to produce superior resolution as compared to the zero-padded periodogram for closely spaced components. The work in paper B has been published in part as

> Johan Swärd, Stefan Ingi Adalbjörnsson, Andreas Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying Signals", *39th International Conference on Acoustics, Speech, and Signal Processing*, Florence, Italy, May 4-9, 2014.
>
> Stefan Ingi Adalbjörnsson, Johan Swärd, Andreas Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying Two-Dimensional Signals", *22nd European Signal Processing Conference*, Lisbon, Portugal, September 1-5, 2014.

and has been published in full as

> Johan Swärd, Stefan Iingi Adalbjörnsson, and Andreas Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying N-dimensional Signals", *Elsevier Signal Processing Journal*, Vol. 128, pp. 309-317, November 2016.

Paper C: Sparse Semi-parametric Estimation of Harmonic Chirp Signals

In the third paper, we introduce a novel harmonic chirp estimator which allows for non-uniformly sampled data. Based on sparse modeling, a dictionary of can-

didate chirps are constructed, and an estimate of the chirp components are found utilizing the harmonic relation. The estimates are then refined using an iterative approach, where each component is added to the residual, refined, and then subtracted again, all while the harmonic relation is held intact. The proposed method is evaluated on simulated data, as well as on an actual recording of a bat sound, showing preferable performance as compared to other chirp estimators. Furthermore, the proposed method is shown to attain the Cramér-Rao lower bound, at least for medium and high SNR-levels. The work in paper C has been published in part as

> Johan Swärd, Johan Brynolfsson, Andreas Jakobsson, and Maria Hansson-Sandsten, "A Sparse Semi-Paramtric Chirp Estimator", *48th Asilomar Conference on Signals, Systems, and Computers*, Asilomar, USA, November 2-5, 2014.

and has been published in full as

> Johan Swärd, Johan Brynolfsson, Andreas Jakobsson, and Maria Hansson-Sandsten, "Sparse Semi-Parametric Estimation of Harmonic Chirp Signals", *IEEE Transactions on Signal Processing*, Vol. 64, No 7, pp. 1798-1807, April 2016.

Paper D: Generalized Sparse Covariance-based Estimation

In this paper, we extend the sparse iterative covariance-based estimator (SPICE), by generalizing the formulation to allow for different norm constraints on the signal and noise parameters in the covariance model. We show that by using this new formulation, one may expect sparser solutions than with the original SPICE method, which is known for producing spurious peaks. We also show that the extended SPICE formulation is equivalent to a certain family of penalized regression problems, for which the proposed method presents itself as a computationally attractive solver. Furthermore, we also provide a gridless formulation of the proposed method for the case of sinusoidal signals, based on the recent atomic norm framework. The numerical evaluations show the preferred performance of the proposed method. The work in paper D has been published in part as

> Johan Swärd, Stefan Ingi Adalbjörnsson, Andreas Jakobsson, "A Generalization of the Sparse Iterative Covariance-based Estimator", *42nd International Conference on Acoustics, Speech, and Signal Processing*, New Orleans, Louisiana, USA, March 5-9, 2017.

and is submitted for possible publication as

> Johan Swärd, Stefan Ingi Adalbjörnsson, Andreas Jakobsson, "Generalized Sparse Covariance-based Estimation".

Paper E: Online Estimation of Multiple Harmonic Signals

In this paper, we consider time-recursive estimation of the fundamental frequencies of multi-pitch signals with an unknown number of signal components using sparse modeling techniques. By using signal-adaptive penalties that induce a group structure, we show that the proposed method is capable of multi-pitch estimation without requiring model order information, i.e., without knowing the number of pitches or harmonics present in the signal. By using a signal-adaptive dictionary updating technique, we also show that the proposed methods are able to track frequency modulated signals. The amplitudes of the active pitches are also recursively updated, allowing for a smooth and more accurate representation. When evaluated on a data set of real audio signals, the proposed method outperforms state-of-the-art methods in either estimation accuracy or computational speed. The work in paper E has been published in part as

> Filip Elvander, Johan Swärd, and Andreas Jakobsson, "Time-Recursive Multi-Pitch Estimation Using Group Sparse Recursive Least Squares", *50th Asilom'ar Conference on Signals, Systems, and Computers*, Asilomar, USA, November 6-9, 2016.

and has been published in full as

> Filip Elvander, Johan Swärd, and Andreas Jakobsson, "Online Estimation of Multiple Harmonic Signals", *IEEE Transactions on Audio, Speech and Language Processing*, Vol. 25, No. 2, pp. 273-284, February, 2017.

Paper F: Off-grid Fundamental Frequency Estimation

This paper addresses the problem of off-grid estimation of fundamental frequencies in multi-pitch signals. First, a non-convex optimization problem is proposed that allows for sparser solutions than the traditional convex cost functions, having both the amplitude and the frequency grid points as variables. This non-convex problem is then relaxed using a majorization-minimization approach, where, in

each iteration, a simpler surrogate function, based on the latest estimates, is minimized. In each iteration, the amplitudes may be found in closed-form, whereas the fundamental frequencies may be found using, e.g., a gradient descent method. The dictionary is in each iteration pruned, such that the grid points that are deemed obsolete are removed, thus decreasing the total computational cost for solving the problem. The proposed algorithm is shown to perform similar to state-of-the-art transcription methods that have been trained on the instruments, whereas the proposed method does not require any such training, and is therefore also more robust to any prior signal assumptions. The work in paper F has been submitted for possible publication as

> Johan Swärd, Hongbin Li, and Andreas Jakobsson, "Off-grid Fundamental Frequency Estimation".

Paper G: Estimating Sparse Signals Using Integrated Wideband Dictionaries

In this paper, we introduce new dictionary elements for, primarily, frequency estimation. Instead of following the traditional approach, where the dictionary is composed of sinusoids, we consider elements covering larger bands in the frequency domain. Using these bands, we may form problems where the number of parameters is smaller but still covers the whole spectrum. Thus, we may discard large parts of the parameter space when using these bands. In the paper, we propose an iterative zooming procedure, where in each iteration the parts of the spectrum that were not activated in the previous iteration are discarded, and the active bands are refined to create thinner bands in the next iteration. This approach makes it possible to limit the number of parameters in the dictionary, thus allowing for faster computations. The work in paper G has been published in part as

- Maksim Butsenko, Johan Swärd, and Andreas Jakobsson, “Estimating Sparse Signals Using Integrated Wide-band Dictionaries”, *42nd International Conference on Acoustics, Speech, and Signal Processing*, New Orleans, Louisiana, USA, March 5-9, 2017.

and submitted for possible publication as

> Maksim Butsenko, Johan Swärd, and Andreas Jakobsson, “Estimating Sparse Signals Using Integrated Wide-band Dictionaries”.

Paper H: Grid-less Estimation of Saturated Signals

This paper addresses the problem of frequency and amplitude estimation when the measured signal has been subjected to clipping, which happens when the measured signal is saturated at its maximum and/or minimum values. Traditionally, these samples have been treated as missing and have completely been disregarded from the estimation. In this paper, we incorporate the information available in the saturated samples as well as including robustness to noise effects and propose a sparse reconstruction algorithm based on the atomic norm framework. We provide a formulation enabling multidimensional estimation and show the preferred performance of the proposed method on 1-D and 2-D data. Furthermore, we also present a refinement procedure for the amplitude estimates which also, robustly to the noise effects, incorporates the information inherent in the clipped samples. This work has been submitted for possible publication as

> Filip Elvander, Johan Swärd, and Andreas Jakobsson, "Multi-dimensional Grid-less Estimation of Saturated Signals".

Paper I: Designing Optimal Sampling Schemes

In this paper, we propose a method for finding good sampling schemes for multidimensional data. In many experiments, sampling is associated with high costs of both time and money. In these situations, it is of importance to know how to sample a signal to minimize the acquisition time but, at the same time, avoiding losing too much information. In this paper, we propose a convex optimization problem that minimizes the number of samples subject to an upper bound on the variances of the parameters of interest. The method takes any *a-priori* information about the signal into account and also provides a simple approach for giving more importance to one or many parameters. The proposed method outperforms other state-of-the-art sampling schemes and is shown to provide a better sampling scheme, much faster, as compared to using a random sampling approach. The work in paper I has been published in part as

> Johan Swärd, Filip Elvander, and Andreas Jakobsson, "Designing Optimal Sampling Schemes", *25th European Signal Processing Conference*, Kos island, Greece, 28 August - 2 September, 2017.

and submitted for possible publication as

Filip Elvander, Johan Swärd, and Andreas Jakobsson, “Designing Sampling Schemes for Multi-Dimensional Data”.

References

[1] J. Cooley and J. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Math. Computation*, vol. 19, pp. 297–301, 1965.

[2] M. T. Heideman, D. H. Johnson, and C. S. Burrus, "Gauss and the History of the Fast Fourier Transform," *IEEE ASSP Magazine*, vol. 1, pp. 14–21, 1984.

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

[4] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2 edition, 2009.

[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] H. L. Taylor, S. C. Banks, and J. F. McCoy, "Deconvolution with the ℓ_1 norm," *Geophysics*, vol. 44, no. 1, pp. 39–52, 1979.

[7] M. Lobo, M. Fazel, and S. Boyd, "Portfolio Optimization with Linear and Fixed Transaction Costs," *Annals of Operations Research*, vol. 152, pp. 341–365, 2006.

[8] A. Ghosh and S. Boyd, "Growing Well-Connected Graphs," in *Procedings IEEE Conference on Decision and Control*, Dec 2006, pp. 6605–6611.

[9] 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.

[10] E. J. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406 – 5425, Dec. 2006.

[11] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

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

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] J. Eckstein and D.P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, April 1992.

[19] 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.

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

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

[22] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

[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 J. Sel. Topics Signal Process.*, vol. 4, pp. 560–570, 2010.

[24] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The Convex Geometry of Linear Inverse Problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, Dec 2012.

[25] B. N. Bhaskar, G. Tang, and B. Recht, "Atomic Norm Denoising with Applications to Line Spectral Estimation," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5987 – 5999, July 2013.

[26] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed Sensing Off the Grid," *IEEE Trans. Inform. Theory*, vol. 59, no. 11, pp. 7465–4790, Nov 2013.

[27] J. M. Steele, *The Cauchy-Schwarz Master Class*, Cambridge University Press, 2004.

[28] Z. Yang and L. Xie, "On Gridless Sparse Methods for Line Spectral Estimation From Complete and Incomplete Data," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3139–3153, June 2015.


A

Paper A

Estimating Periodicities in Symbolic Sequences Using Sparse Modeling

Stefan Ingi Adalbjörnsson[1], Johan Swärd[1], Jonas Wallin[2], and Andreas Jakobsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Department of Mathematical Sciences, Chalmers University of Technology, Gothenburg, Sweden*

Abstract

In this work, we propose a method for estimating statistical periodicities in symbolic sequences. Different from other common approaches used for the estimation of periodicities of sequences of arbitrary, finite, symbol sets, that often map the symbolic sequence to a numerical representation, we here exploit a likelihood-based formulation in a sparse modeling framework to represent the periodic behavior of the sequence. The resulting criterion includes a restriction on the cardinality of the solution; two approximate solutions are suggested, one greedy and one using an iterative convex relaxation strategy to ease the cardinality restriction. The performance of the proposed methods are illustrated using both simulated and real DNA data, showing a notable performance gain as compared to other common estimators.

Key words: Periodicity, symbolic sequences, spectral estimation, data analysis, DNA

1 Introduction

Sequences formed from a finite set of symbols, or *alphabet*, occur in a variety of fields, such as, for instance, in genomics, semantic analysis, and categorical time series [1, 2]. Frequently, there is an interest in determining reoccurring patterns, periodicities, in such sequences. For instance, in DNA analysis, the latent periodicities in DNA sequences, commonly assumed to be stationary in short time intervals, have been found to be correlated with various forms of functional roles of importance [3–11]. Traditional spectral estimation techniques are not suitable for this problem as symbolic sequences lack algebraic structures. For DNA analysis, there is no natural ordering among the four occurring symbols, A, C, G, and T. In earlier literature, several authors have addressed the problem of estimating symbolic periodicity using heuristic mappings from the symbol set to sets of complex numbers. After the transformation the periodicities are estimated through standard estimation methods like, for instance, the periodogram. However, such estimates will suffer from the well-known high variability and/or poor resolution inherent to the periodogram [12]. Other examples of methods that use a mapping to transform the symbolic data include PAM- or QPSK-based mappings, minimum entropy mapping, mapping equivalences, or other transformations [4–7, 9, 10, 13, 14]. Generally, these mappings are computationally intensive, and/or suffer from difficulties expanding to a larger symbol sets, and often inadvertently impose a non-existing structure on the symbols. In this work, we instead use a probabilistic approach, modeling the symbolic sequences using a categorical distribution for each observation and try to infer not only the unknown probabilities but also the unknown indices where the distribution differs, resulting in a likelihood ratio test, which, for a given index set, is equivalent with the well studied problem of testing for independence in $2 \times J$ contingency tables, where J denotes the number of categories, see, e.g., [2]. Ideally, an estimator for this problem should be able to discern not only whether the distribution differs at a certain periodicity, but also how many indices have differing distributions. If more than one statistical periodicity is considered at the same time, the number of possible combinations of index sets grows rapidly and an exact test will in many cases be computationally infeasible. By formulating the estimation of the unknown index sets, and the unknown probabilities, as a sparse logistic regression problem, we devise two approximate solutions to the combinatorial problem using sparse heuristics. Namely, one greedy approach which builds up the solution by adding the sets in a sequential manner, and one using a convex relaxation of

the cardinality constraint, resulting in the well-known (reweighted) Lasso problem. The resulting methods are firmly based in statistical theory, and also easily generalized to any finite symbol set.
The remainder of the paper is organized as follows: in the next section, we introduce the considered data model and show how the problem of choosing which indices that show a periodic change in the distribution can be interpreted as a sparse estimation problem. Then, in section III, we introduce a greedy algorithm that approximately solves the sparse problem, as well as a convex relaxation of the original problem, which may be efficiently solved using convex optimization algorithms. Then, in section IV, we outline some implementation issues, including a cyclic coordinate descent algorithm for solving the resulting convex relaxation problem. In section V, we examine the performance of the discussed estimators, showing the benefits of the proposed approach as compared to previously published methods. Finally, we conclude on the work in section VI.

2 Probabilistic model for symbolic sequences

Consider a symbolic sequence, $\{s_k\}_{k=1}^N$, where each symbol, s_k, is a stochastic variable drawn from a finite set, $\mathcal{A} = \{\alpha_1, \ldots, \alpha_B\}$, where B denotes the size of the alphabet. Assume that the symbols in the sequence are independent and identically distributed, such that

$$p_j \triangleq \text{Prob}(s_k = \alpha_j) \tag{1}$$

Then, if gathering a sequence of observations, $x_1, \ldots, x_N$, into the vector $\mathbf{x}$, the probability mass function (PMF) of $\mathbf{x}$ is given as

$$\begin{aligned} q_0(\mathbf{x}|\mathbf{p}) &\triangleq \text{Prob}(\mathbf{s} = \mathbf{x}) \\ &= \prod_{j=1}^{N} \prod_{\ell=1}^{B} p_\ell^{[x_j=\alpha_\ell]} = \prod_{\ell=1}^{B} p_\ell^{G_\ell} \end{aligned} \tag{2}$$

where $[\cdot]$ denotes the Iverson's bracket, which equals one if the statement inside the brackets is true, and zero otherwise, with each of the symbols appearing G_k times, and where $\mathbf{p}$ and $\mathbf{s}$ denote the vector of probabilities and the sequence of random variables, respectively, i.e.,

$$\mathbf{p} = \begin{bmatrix} p_1 & \cdots & p_B \end{bmatrix}^T \tag{3}$$

$$\mathbf{s} = \begin{bmatrix} s_1 & \dots & s_N \end{bmatrix}^T \tag{4}$$

with $(\cdot)^T$ denoting the transpose. As a result, the PMF is a function depending only on the number of times each symbol appears, and on the probability given to each symbol. In general, the probabilities, p_k, are unknown and need to be estimated from the observed sequence. This can be done using the maximum likelihood (ML) estimate, formed as

$$\hat{p}_j = \frac{G_j}{N} \tag{5}$$

for $j = 1, \dots, B$, which is an unbiased and asymptotically efficient estimate (see, e.g., [15, p. 475]). Furthermore, note that a symbol $\alpha \in \mathcal{A}$, occurring with periodicity m, i.e., with the symbol appearing at every mth index in the sequence, implies that all elements of the sequence should be equal to the symbol α in one of the m possible (disjoint) index sets

$$I(m, \ell) = \left\{ \ell, \ell + m, \dots, \ell + \left\lfloor \frac{N - \ell}{m} \right\rfloor m \right\} \tag{6}$$

for all offsets $\ell \in \{1, \dots, m\}$, where $\lfloor \cdot \rfloor$ denotes the rounding down operation. This means that if a periodicity m is present in a sequence, the sequence is clearly also periodic on the subharmonics i.e., for every mr:th symbol, for all natural numbers r [8]. To avoid ambiguity, we here refer to the period as the lowest possible such periodicity. Considering a sequence, $\mathbf{s}$, with a periodicity m in the symbol α, with offset n, this implies that all the symbols in the sequence at index k, will equal α, for $k \in I(m, n)$. Thus, it is a deterministic and not a statistical problem to determine if such a (deterministic) periodicity is present. However, of more interest are typically the statistical periodicities that occur in many forms of symbolic sequences, such as, e.g., DNA sequences. These are characterized by certain index sets having different distributions, such that the sequence may contain the periodicity over only a limited interval, and/or with some of the periodically occurring symbols occasionally being replaced by some other symbol, which may occur, for example, due to the presence of measurement noise, coding errors, or some, perhaps unknown, functional equivalence between symbols [3]. In such cases, the PMF for a symbolic sequence might instead be formed from two distributions, one for the indices, say I_1, corresponding to some unknown periodic index set $I(m, l)$, and another distribution for the complement

index set, here denoted I_0. In this case, the PMF is

$$q_1(\mathbf{x}|\mathbf{p}_0, \mathbf{p}_1) \triangleq \prod_{j=1}^{N} \prod_{\ell=1}^{B} p_{0,\ell}^{[x_j=\alpha_\ell][j\in I_0]} p_{1,\ell}^{[x_j=\alpha_\ell][j\in I_1]}$$
$$= \prod_{\ell=1}^{B} p_{0,\ell}^{G_{0,\ell}} p_{1,\ell}^{G_{1,\ell}} \tag{7}$$

where $\mathbf{p}_0$, and similarly for $\mathbf{p}_1$, is a parameter vector containing the probabilities $p_{0,k}$, denoting the probability of a symbol, α_k, occurring in the index set I_0, and with $G_{0,k}$ and $G_{1,k}$ denoting the number of times the symbol α_k occurs in the set $I(m, n)$ and in its complement, respectively. The corresponding ML estimates are found as

$$\hat{p}_{0,j} = \frac{G_{0,j}}{|I_0|} \tag{8}$$

$$\hat{p}_{1,j} = \frac{G_{1,j}}{|I_1|} \tag{9}$$

for $j = 1, \ldots, B$, where $|S|$ denotes the cardinality of a set S, i.e., the number of elements in S. In a similar fashion, the addition of more than one periodicity can be accomplished by defining the distribution on more index sets, e.g. if one considers M disjoint index sets, $I_0, \ldots, I_{M-1}$, so that their union corresponds to the entire sequence, the PMF is

$$q_1(\mathbf{x}|\mathbf{p}_0, \ldots, \mathbf{p}_{M-1}) \triangleq$$
$$\prod_{m=0}^{M-1} \prod_{k=1}^{B} p_{m,k}^{G_{m,k}} \tag{10}$$

where $G_{m,k}$ denotes the number of times the symbol α_k occurs in the set I_m. Comparing the likelihood above with (2), it can be seen that (10) corresponds to a likelihood for i.i.d. categorical variables, within each of the M index sets. However, note that this does not assume that the sequence consists of i.i.d. variables, only that knowing the index sets we can split the sequence into sub sequences of i.i.d. variables.

A similar model was considered in [8], although there they defined a statistical periodicity, say k, to be present when all index set $I(k, \ell)$, for $\ell = 1 \ldots, k$, have different distributions, and then set out to find the periodicity, k, by maximizing

the log-likelihood using an information-theoretic criterion penalty term to select the correct periodicity. If doing so, and the signal has a periodicity of k, then each index set corresponding to a different offset also has a unique distribution, implying a subdivision of the data into $\lfloor N/k \rfloor$ disjoint data sets, resulting in less data to be used to estimate these probabilities. For multiple periodicities, i.e., several index sets with different distributions, this results in a necessity to consider the overall periodicity of the sequence, i.e., if periods l and k are present, then the sequence will have a periodicity of lk, resulting in the need for substantially more data to achieve a similar performance as if only a single periodicity was present, as well as the need to perform on additional analysis to identify the factors constituting lk. Furthermore, in the case when the sequence contains more than two periodicities, the problem quickly becomes infeasible. We instead want to find the index sets where the distributions differ as much as possible from the rest of the sequence. To that end, we recast the estimation problem in a sparse modeling framework. To do so, we note that one can interpret (10) as a multi-response logistic regression problem, which, as we will show, will be particularly useful for the case of several simultaneous periodicities. Furthermore, this mapping allows us to consider sequences one symbol at a time, which is particularly useful when the periodicity in a certain symbol is sought, or if the distribution of a particular symbol deviates especially much on a given index set. This, when applicable, decreases the variance of the estimated probabilities, thus improving the detection of periodicities only occurring in one symbol, or one subset of symbols. Rewriting (10) using logistic regression is accomplished by modeling the probability of each observation separately using a logistic function to map a linear model to the interval $[0,\ 1]$. To clarify the exposition, we first consider the case of a binary symbol set, a special case which will be shown to be particularly useful. Thus, consider a binary sequence which has a statistical periodicity on the indices I_1, and some other distribution on the indices I_0, so that the PMF may be expressed as

$$q_1(\mathbf{x}|\boldsymbol{\gamma}(\mathbf{c})) \triangleq \prod_{k=1}^{N} \gamma_k(\mathbf{c})^{x_k}(1-\gamma_k(\mathbf{c}))^{1-x_k} \tag{11}$$

where $\boldsymbol{\gamma}(\mathbf{c}) \in \mathbf{R}^N$ is a vector of probabilities, such that

$$Pr(s_k = 1) = \gamma_k(\mathbf{c}) \tag{12}$$

and the vector $\mathbf{c} \in \mathbf{R}^2$ models the probabilities for the index sets I_1 and its com-

plement, I_0, such that

$$\boldsymbol{\gamma}(\mathbf{c}) = \left[\begin{array}{ccc} \gamma_1(\mathbf{c}) & \ldots & \gamma_N(\mathbf{c}) \end{array} \right]^T \tag{13}$$

$$\gamma_k(\mathbf{c}) = \frac{\mathrm{e}^{\mathbf{h}_k^T \mathbf{c}}}{1 + \mathrm{e}^{\mathbf{h}_k^T \mathbf{c}}} \tag{14}$$

where

$$\mathbf{h}_k = \begin{cases} \left[\begin{array}{cc} 1 & 1 \end{array} \right]^T & \text{if } k \in I_1 \\ \left[\begin{array}{cc} 1 & 0 \end{array} \right]^T & \text{if } k \notin I_1 \end{cases} \tag{15}$$

Thus, there is a simple relationship between the parameters $p_{0,1}$ and $p_{1,1}$ in the original model in (7), i.e.,

$$P(s_k = 1) = p_{0,1} \quad \text{for } k \in I_0 \tag{16}$$

$$P(s_k = 1) = p_{1,1} \quad \text{for } k \in I_1 \tag{17}$$

and the parameter vector, $\mathbf{c}$, introduced in (11), i.e.,

$$\log\left(\frac{p_{0,1}}{1 - p_{0,1}}\right) = \left[\begin{array}{cc} 1 & 0 \end{array} \right]^T \mathbf{c} \tag{18}$$

$$\log\left(\frac{p_{1,1}}{1 - p_{1,1}}\right) = \left[\begin{array}{cc} 1 & 1 \end{array} \right]^T \mathbf{c} \tag{19}$$

It should be noted that (18) implies that the probability of a symbol appearing in the set I_0 is given by the first element of the vector $\mathbf{c}$, and, similarly, one may by substituting (18) into (19) and simplifying, note that

$$\log\left(\frac{p_{1,1}}{1 - p_{1,1}}\right) - \log\left(\frac{p_{0,1}}{1 - p_{0,1}}\right) = \left[\begin{array}{cc} 0 & 1 \end{array} \right]^T \mathbf{c} \tag{20}$$

Thus, the second element in $\mathbf{h}_k$ control the change in probability on the index set, I_1, as compared to the indices in the set, I_0, e.g., if the second element is zero, then the probabilities are the same for both sets, whereas a positive or negative second element implies higher or lower probabilities on the set I_1, respectively. Extending the model to allow for the possibility of several periodicities using the logistic

regression parameterization can be achieved by adding elements to the $\mathbf{c}$ vector such that each new element adjusts the probability for an additional index set. To that end, consider the case with M index sets, I_j, for $j = 1, \ldots, M$, corresponding to some specific periodicities with their different offsets, then $\mathbf{c} \in \mathbf{R}^M$ and every element of $\mathbf{h}_k^T \in \mathbf{R}^M$ is zero except the first element and the elements where k is in the corresponding index set, i.e.,

$$h_{k,j} = \begin{cases} 1 & k \in I_j \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

for $j = 1, \ldots, M$, and $d_{k,j}$ denotes element j of the vector $\mathbf{d_k}$. The resulting model can then be seen as the solution of the following optimization criterion

$$\begin{aligned} \underset{\mathbf{c}}{\text{maximize}} \quad & \prod_{k=1}^{N} \gamma_k(\mathbf{c})^{x_k}(1-\gamma_k(\mathbf{c}))^{1-x_k} \\ \text{subject to} \quad & \begin{cases} ||\mathbf{c}||_0 \leq L \\ \gamma_k(\mathbf{c}) = \frac{e^{\mathbf{h}_k^T\mathbf{c}}}{1+e^{\mathbf{h}_k^T\mathbf{c}}} \end{cases} \end{aligned} \tag{22}$$

where $||\cdot||_0$ denotes the ℓ_0 (pseudo) norm, which counts the number of nonzero elements of a vector, and L is the maximum number of periodicities that will be included in the model. It is worth noting that the expression for $\gamma_k(\mathbf{c})$ does not pose a restriction to the minimization, but has been included to emphasize that the probabilities for each observation are being modeled explicitly. Solving (22) for a given L, i.e., finding the maximum allowed number of simultaneous periodic sets, can be accomplished using an exhaustive search, since for each fixed k there are $(M)!/\left((M-j)!j!\right)$ index sets. For each such set, the ML estimates may then be found using (5). However, the dimension of the parameter vector will grow quadratically with the maximum periodicity considered, since

$$M = \sum_{k=1}^{m_{max}} k = \frac{m_{max}(m_{max}+1)}{2} \tag{23}$$

where m_{max} is the maximum allowed periodicity, since each period k has k corresponding index sets, one for each possible offset. Thus, to evaluate the likelihood for all combinations of index sets will soon lead to a computationally infeasible problem. Generalization to larger symbol sets may be carried out in a similar

manner, leading to the multi-response logistic regression model (see, e.g., [2] for a further discussion on multi-response logistic regression). The corresponding optimization problem is therefore given as the maximum of the log-likelihood with a cardinality constraint [16]

$$\begin{aligned} \underset{\mathbf{c}_1,\ldots,\mathbf{c}_B}{\text{maximize}} \quad & \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{\ell=1}^{B} x_{i\ell}(\mathbf{h}_i^T\mathbf{c}_\ell) - \log\left(\sum_{\ell=1}^{B} \mathrm{e}^{\mathbf{h}_i^T\mathbf{c}_\ell}\right)\right] \\ \text{subject to} \quad & ||\mathbf{C}_{k\cdot}||_0 \leq L, \quad \text{for } k = 1,\ldots,R \end{aligned} \tag{24}$$

where $\mathbf{C}$ is a matrix constructed such that its k:th column is formed by the vector $\mathbf{c}_k$, and R is the number of considered index sets, with $\mathbf{C}_{k\cdot}$ denoting the restriction that $||\mathbf{C}_{k\cdot}||_0$ forces the solution to adjust the B parameters corresponding to every index set simultaneously. Thus, the distributions can be changed on at most L index sets. As a result, the framework allows for flexibility in what is deemed a periodicity, e.g., one might test for a high probability of a certain symbol appearing, or even for if some symbols appear with low probability. Both of these ideas will be explored further in the following, where we outline a couple of possible algorithms for estimating periodicities for some commonly occurring situations, namely, estimation of an unknown periodicity, detection of an unknown periodicity, and, finally, estimation of multiple periodicities.

3 Relaxation of the cardinality constraint

For cardinality constrained, or sparse, least squares problems, there are a wide range of tools for forming approximate solutions, with many methods falling into two broad categories, namely greedy methods that build up a solution one variable at a time until either fitting criterion is satisfied, or the number of variables reaches the constraint, or methods that replace the cardinality constraint with a penalty function that promotes solutions that have few non-zero variables [17]. This implies that the optimization can be carried out without the combinatorial computation complexity inherent in cardinality constrained optimization problems. Typically, the penalty function is selected as the ℓ_1 norm, leading to a simple convex optimization problem. In the following two subsections, we propose both kinds of algorithms, first a greedy approach and then an iterative convex relaxation.

3.1 Greedy approach

In order to form a greedy estimate of the minimization in (24), one may note the analogy between this formulation and that of simple hypothesis test for testing if a distribution is different on some index sets (see also [3]). Thus, one may form a test to determine the hypothesis that a given sequence has a different distribution for the indices corresponding to $I(m, \ell)$, i.e., that the PMF is formed using (7), against the null hypothesis that the entire sequence has the same categorical distribution, such that the PMF instead follows (2), i.e.,

$$\mathrm{H}_0 : \mathbf{p}_0 = \mathbf{p}_1 \tag{25}$$

$$\mathrm{H}_1 : \mathbf{p}_0 \neq \mathbf{p}_1 \tag{26}$$

Such a test may be formed as a likelihood ratio (LR) test (see, e.g., [18, p. 375])

$$\lambda_{m,\ell}(\mathbf{x}_N) = \frac{q_0(\mathbf{x}_N|\mathbf{p}_0, \mathrm{H}_0)}{q_1(\mathbf{x}|\mathbf{p}_0, \mathbf{p}_1, \mathrm{H}_1)} \tag{27}$$

where the probabilities are determined using (5) under H_0, and using (8) and (9) under H_1. Thus, if one only seek to find a single index set, a suitable choice would be the one minimizing the LR, i.e.,

$$\arg\min_{m,\ell} \quad \lambda_{m,\ell}(\mathbf{x}_N) \tag{28}$$

If the number of periodicities is unknown, i.e., the problem is one of detection and not estimation, one can allow for the possibility of no set being added by considering that if H_0 is true, it holds asymptotically that [18, p. 489]

$$-2\log(\lambda_{m,\ell}(\mathbf{x}_N)) \xrightarrow{d} \chi^2_{B-1} \tag{29}$$

where $\xrightarrow{d}$ denotes convergence in distribution and χ^2_k denotes the chi-squared distribution with k degrees of freedom. Thus, if no periodicity is present, a critical value, denoted T_α, for the likelihood ratio, below which no periodicity is deemed to be present, can be constructed for the likelihood ratio for each of the tests. Since M tests are formed in order to compute (28), and if assuming that these are independent, the critical value may be well approximated using extreme value theory as a quantile of the random variable

$$\psi = max\,(z_1, \ldots, z_M) \tag{30}$$

where each z_k is χ^2 distributed, implying that ψ will follow a Gumbel distribution (see, e.g., [19, p. 156]). In the case when multiple periodicities may be present, one can extend this procedure using a step-wise approach. To do so, first define I_1 as the index set containing all the indices in the sequence. Then, the initial step is performed by using the above algorithm to determine an index set $I_2 = I_{m_1,\ell_1}$, where m_1 and ℓ_1 denote the initially estimated periodicity and offset, respectively, found in the minimization of (28). In order to determine the next periodicity, the H_0 distribution is formed from (10), using one distribution for the found index set I_2 and one for all the other indices, $I_1 \setminus I_2$, where $\setminus$ denotes set subtraction operation. The second phase, m_2, and periodicity, ℓ_2, may be determined using (28). This procedure can then be repeated until the zero hypothesis can not be rejected using a suitable quantile of (30), i.e., at iteration s the corresponding likelihood ratio test may be formed as

$$\lambda^{(s)}_{m,\ell}(\mathbf{x}_N) = \frac{q_0(\mathbf{x}_N|\mathbf{p}_0,\ldots,\mathbf{p}_{s-1},\mathrm{H}_0)}{q_1(\mathbf{x}|\mathbf{p}_0,\ldots,\mathbf{p}_s,\mathrm{H}_1)} \tag{31}$$

Note that this assumes that the sets I_k being added to the zero hypothesis are disjoint, otherwise the likelihood would include some data points more than once. To ensure this we propose to only consider the indices that have not already been added to H_0 when evaluating $q_1(\mathbf{x}|\mathbf{p}_0,\mathbf{p}_1,\mathrm{H}_1)$ in (27), i.e., at iteration k the sets $I(m,\ell)$ are replaced with $I(m,l) \leftarrow I(m,l) \setminus I_{k-1}$, for all m and ℓ, where $\leftarrow$ denotes that the quantity on the left is replaced with the one on the right. The resulting greedy algorithm, here termed the greedy *P*eriodicity *E*stimation of *C*ategorical *S*equences (PECS$_G$) estimator, is outlined in Algorithm 1 below, with each iteration requiring at most $\mathcal{O}(Bm_{max}N)$ operations.

3.2 Iterative Convex Relaxation

It is worth noting that the optimization criterion in (22) is not convex as it restricts the parameter space to lie in a non-convex set. A commonly used relaxation for problems of this kind is to replace the ℓ_0 restriction with the convex ℓ_1 ball, which by taking the negative logarithm and using the Lagrange duality, results in the relaxed convex optimization criterion

$$\underset{\mathbf{c}}{\text{minimize}} \quad \sum_{k=1}^{N} -x_k\mathbf{h}_k^T\mathbf{c} + \log(1+\mathrm{e}^{\mathbf{h}_k^T\mathbf{c}}) + \lambda||\mathbf{c}||_1 \tag{32}$$

Algorithm 1 The PECS_{G} estimator

Given a categorical sequence, $\mathbf{x}$ of length N
$I_0 = \{1, \ldots, N\}$
for $s = 1, \ldots$ **do**
  $\{m_s, \ell_s\} = \arg\max_{m,\ell} \quad \lambda_{m,\ell}(\mathbf{x}_N)$
  if $\lambda_{m,\ell}(\mathbf{x}_N) > C_\alpha$ **then**
    $I_s = I_{m_s,\ell_s}$
  else
    break
  end if
  $I(m, l) \leftarrow I(m, l) \setminus I_s$ for all m and l
  $I_0 \leftarrow I_0 \setminus I_s$
  H_0 distribution is replaced with (10) using $I_0, \ldots, I_s$
end for

where we have exploited the equality constraint for $p_k(\mathbf{c})$ and where $\lambda > 0$ is a tuning parameter, which may be set using, for example, cross validation (see e.g., [20]), or by an heuristic choice using the observation following equation (42). Some adjustments may be done to this criterion; firstly, the penalty on $\mathbf{c}$ includes the first element. This is not appropriate since the first element controls the probability for all observations, and we have no reason to want to bias that probability towards $1/2$. This is easily accomplished by only penalizing the other elements of the vector, i.e., replacing $||\mathbf{c}||_1$ with $||\underline{\mathbf{c}}||_1$, where $\underline{\mathbf{c}}$ denotes the resulting vector once the first element of $\mathbf{c}$ is removed. However, the resulting expression will also have an undesirable ambiguity due to the lack of distinction being made between if the probability is higher or lower on the periodic indices. For instance, consider a case when every third index starting with 1 has the probability 0.1 of being 1, and all other indices have probability 0.9 of being 1. Should this be considered two periodicities of 3 with probability 0.9, or one periodicity of 3 with probability 0.1?

Such a distinction is of course not a problem specific for this model. However, since one is commonly interested in finding periodic indices where the probability is either higher or lower, such an ambiguous result would result in a non-consistent interpretation of the estimates. Fortunately, this can be easily handled by adding a constraint on $\underline{\mathbf{c}}$, ensuring that only periodicities with greater probab-

ility of a symbol appearing are considered, i.e., $c_k > 0$, for $k = 2, \ldots, M$, where c_i is the i:th element of the vector $\mathbf{c}$. This yields

$$\begin{aligned} \underset{\mathbf{c}}{\text{minimize}} \quad & \sum_{k=1}^{N} -x_k \mathbf{h}_k^T \mathbf{c} + \log(1 + \mathrm{e}^{\mathbf{h}_k^T \mathbf{c}}) + \lambda ||\underline{\mathbf{c}}||_1 \\ \text{subject to} \quad & c_k \geq 0 \quad \text{for } k = 2, \ldots, M \end{aligned} \tag{33}$$

The resulting optimization is thus a sum of an affine function and the logarithm of a sum of exponential functions, and is thus a convex function. (see, e.g., [21, p. 93]). Thus, since the constraints can be seen as inequalities involving inner products with the Cartesian coordinate basis vectors, they are affine, and therefore convex functions, and the criterion is as a result a convex optimization problem in the standard form, as defined in [21, p. 136]. However, the criterion in (33) will not yield sufficiently sparse estimates, as a result of the rather coarse approximation of the ℓ_1 norm to the desired ℓ_0 norm. Recently, interest in non-convex penalties that are closer, in some sense, to the ℓ_0 norm have been suggested, such as the use of the ℓ_q norm, for $0 < q < 1$ (see e.g., [22, 23]). Herein, we consider an alternative approach where the ℓ_1 penalty is replaced with the concave $\log(\cdot)$ penalty. The resulting optimization is then solved with an iteratively re-weighted ℓ_1 minimization, using a technique suggested in [24]. The resulting algorithm thus solves, at iteration $j + 1$, the minimization

$$\begin{aligned} \min_{\mathbf{c}} \quad & \sum_{k=1}^{N} -x_k \mathbf{h}_k^T \mathbf{c} + \log(1 + \mathrm{e}^{\mathbf{h}_k^T \mathbf{c}}) + \lambda \sum_{k=2}^{M} \frac{|c_k|}{|\hat{c}_k^{(j)}| + \varepsilon} \\ \text{s. t.} \quad & c_k \geq 0 \quad \text{for } k = 2, \ldots, M \end{aligned} \tag{34}$$

where $\hat{c}_k^{(j)}$ is the k:th element of the $\mathbf{c}$ estimate resulting from the j:th iteration, and ε is set as a small number to avoid numerical problems as well as to enable zero valued elements of $\mathbf{c}$ to transition from zero to non-zero values (see also [24]). The resulting sequence of convex minimizations yields a sufficiently sparse estimate of the periodicities (although at a high a computational complexity if implemented directly using a standard interior point-based solver). The resulting estimator is in the following referred to as the *P*eriodicity *E*stimation of *C*ategorical *S*equences using Logistic regression, PECS$_L$.

Comparing the two methods, PECS$_G$ offers a faster solution, whereas PECS$_L$ yields better results in the case of multiple periodicities. This is due to the fact

that the iterative greedy procedure in PECS$_G$ does not take into account the overlap between the two index sets, e.g., the index sets $I(k,1) \cap I(l,1) = I(kl,1)$, whereas, the logistic regression approach also takes the overlap into account in the estimation procedure.

4 Efficient implementation

In order to form an efficient solver for the minimization in (34), we proceed to develop a cyclic coordinate descent (CCD) algorithm. The CCD algorithm minimize the cost function in (34) one variable at a time, in a cyclical fashion, holding the other variables fixed at their most recent estimates. This will thus transform the $M-$dimensional optimization problem into a scheme where one instead repeatedly solves simpler one-dimensional problems.

It should be noted that such an approach is, in general, converging notoriously slowly, or in some cases, not at all. However, for the optimization problems encountered in sparse modeling, this does no longer hold, as in fact, convergence proofs exist [20, 25], and in many applications, CCD implementations have emperically been shown to be the fastest algorithms available [16, 26]. Below, we outline the steps involved in a CCD algorithm for the case of $c_k \geq 0$, with the other case being handled in a similar manner. Thus, consider $c_i^{(r)}$ as the r:th estimate of element i of the vector $\mathbf{c}$, then, for $i > 1$,

$$\begin{aligned} c_i^{(r+1)} &= \arg\min_{c_i} \sum_{k=1}^{N} -x_k \mathbf{h}_k^T \mathbf{c} + \log(1 + \mathrm{e}^{\mathbf{h}_k^T \mathbf{c}}) + \lambda ||\underline{\mathbf{c}}||_1 \\ &= \arg\min_{c_i} -\mathbf{x}^T \mathbf{H}_{(\cdot,i)} c_i + \lambda |c_i| + \sum_{k=1}^{N} \log(1 + a_{k,i} \mathrm{e}^{h_{k,i} c_i}) \end{aligned} \tag{35}$$

The notation $\mathbf{H}_{(\cdot,i)}$ denotes the i:th column of the matrix $\mathbf{H}$, $h_{k,i}$ the i:th element of the vector $\mathbf{h}_k$, and

$$\mathbf{x} = \begin{bmatrix} x_1 & \dots & x_N \end{bmatrix}^T \tag{36}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \dots & \mathbf{h}_N \end{bmatrix}^T \tag{37}$$

$$\mathbf{c} = \begin{bmatrix} c_1^{(r+1)} & \dots & c_{(i-1)}^{(r+1)} & c_i^{(r)} & \dots & c_N^{(r)} \end{bmatrix}^T \tag{38}$$

Algorithm 2 The PECS$_L$ estimator

Initiate $\mathbf{c} = \mathbf{c}_0$
for $r = 1, \ldots$ **do**
  for $i = 1, \ldots, M$ **do**
    if maximum of (40) ≥ 0 **then**
      $c_i^{(r)} = 0$
    else
      Update $c_i^{(r)}$ according to (35)
    end if
  end for
end for

$$a_{k,i} = \exp\left(\sum_{j,j\neq i} h_{k,j}c_j\right) \tag{39}$$

If the maximum value of the subdifferential set

$$\partial f_0 = -\mathbf{x}^T\mathbf{H}_{(\cdot,i)} + \lambda w + \sum_{k=1}^{N} \frac{a_{k,i}h_{k,i}e^{h_{k,i}c_i}}{1 + a_{k,i}e^{h_{k,i}c_i}} \tag{40}$$

with $c_i = 0$ is positive and $\{w \in [-1, 1]\}$, then the optimum is attained at $c_i = 0$ for the constrained optimization problem. On the other hand, if the maximum is negative, the stationary point may be found using a gradient approach (since the cost function is differentiable for all positive c_i). Note that this analysis gives insight into both the sparsity promoting effect of the ℓ_1 norm as well as the role of the tuning parameter λ, in fact, rewriting (40) as

$$\partial f_0 = -\mathbf{x}^T\mathbf{H}_{(\cdot,i)} + \lambda w + \mathbf{r}_i^T\mathbf{H}_{(\cdot,i)} \tag{41}$$

where $\mathbf{r}_i = \begin{bmatrix} \frac{a_{1,i}}{1+a_{1,i}} & \cdots & \frac{a_{N,i}}{1+a_{N,i}} \end{bmatrix}$ can be interpreted as probabilities for each index. Furthermore, $\mathbf{r}_i^T\mathbf{H}_{(\cdot,i)}$ is the expected number of symbols on the periodicity corresponding to i and $\mathbf{x}^T\mathbf{H}_{(\cdot,i)}$ is the observed number of symbols on that periodicity, thus if

$$|\mathbf{r}_i^T\mathbf{H}_{(\cdot,i)} - \mathbf{x}^T\mathbf{H}_{(\cdot,i)}| < \lambda \tag{42}$$

implying that, if the expectation for the model with $c_i = 0$ is closer than λ to the observed number in the data, then set $c_i^{(r+1)} = 0$. The resulting CCD algorithm

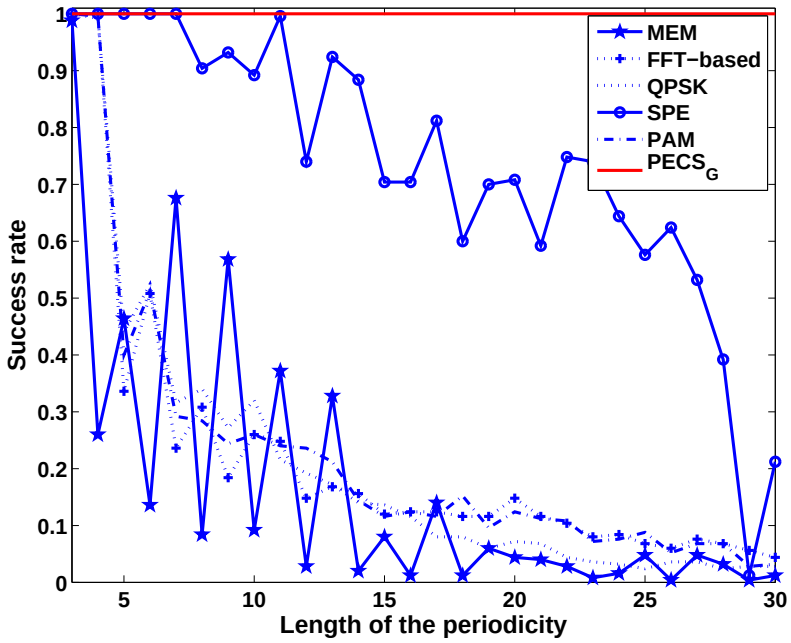

Figure 1: Rate of success in estimating deterministic periods.

is outlined in Algorithm 2. The computational cost of one iteration of the outer loop is $\mathcal{O}(m_{max}^2 N)$. Note that a significant performance increase is often possible in batch applications, where a recursive algorithm is needed, by the so called *active set strategy* [20]. The strategy simply involves not updating the parameters that are currently zero in every iteration, and perhaps only doing so once every tenth iteration or so.

5 Numerical results

We proceed to examine the performance of the proposed likelihood-based estimators using simulated DNA sequences, binary sequences, and measured DNA data. For DNA sequences, only $B = 4$ different symbols are present, namely A,

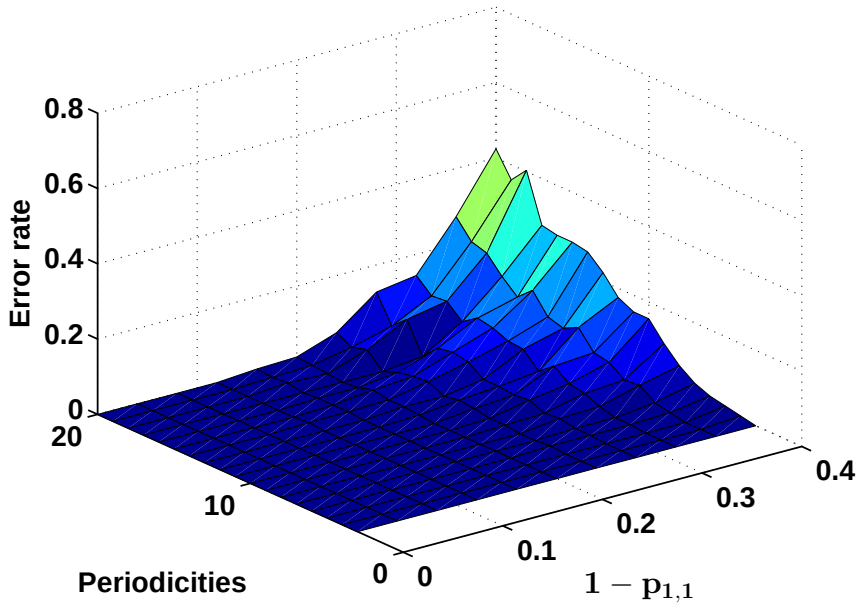

Figure 2: The error rate of finding the periodicity as a function of $1 - p_{1,1}$, and the periodicity for the proposed PECS$_G$ method.

C, G, and T. Initially, we examine a simulated DNA sequence containing one deterministic periodicity. Figure 1 illustrates the rate of successfully determining this periodicity as a function of the length of the periodicity, comparing the proposed PECS$_G$ estimator with the MEM [10], PAM [7], QSPK [5], and SPE [27] estimators, as well as with a Fourier-based estimator detailed in [27]. As the simulated sequence is stationary, the window length used for the DFT-based methods were selected to be equal to the length of the sequence.

Here, and in the following, the success rate has been determined using 250 Monte-Carlo simulations using $N = 1000$ equiprobable symbols, with the sought periodicity being inserted appropriately. As is clear from Figure 1, the proposed estimator succeeds in successfully determining all the considered periodicities, whereas all the other methods lose performance as the length of the periodicity

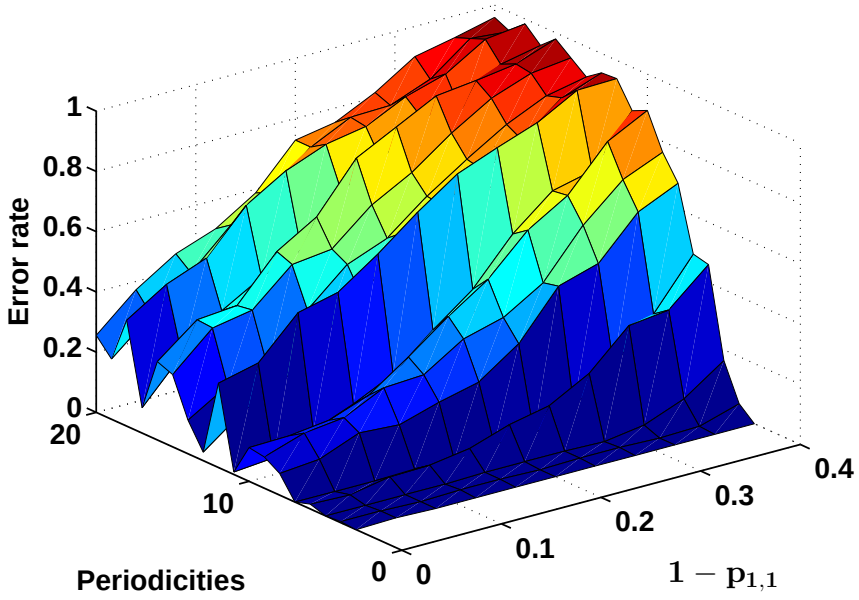

Figure 3: The error rate of finding the periodicity as a function of the negative probability, $1 - p_{1,1}$, and the periodicity for the SPE algorithm.

grows. Of the other examined estimators, the SPE estimator seems to offer the second best performance, and we will for this reason only show the results for this estimator in the following comparisons, noting that all the other discussed estimators exhibits a notably worse performance than the SPE estimator in all the considered cases (see also [1]). Proceeding to examine also statistical periodicities, we vary $p_{1,1}$ for the index set corresponding to the generated periodicity, with $p_{0,1} = 1/4$ on the complement set. It may be noted that $p_{1,1} = 1$ corresponds to a perfect periodicity, whereas $p_{1,1} < 1$ corresponds to a statistical periodicity with a probability of each symbol being eroded, i.e., a non-perfect periodicity, being $1 - p_{1,1}$. Similarly, $p_{0,1}$ is the corresponding probability for the complement set. Figures 2 and 3 show the resulting success rate for the PECS$_G$ and SPE estimators as a function of the periodicity and the probability $p_{1,1}$, again clearly illustrating

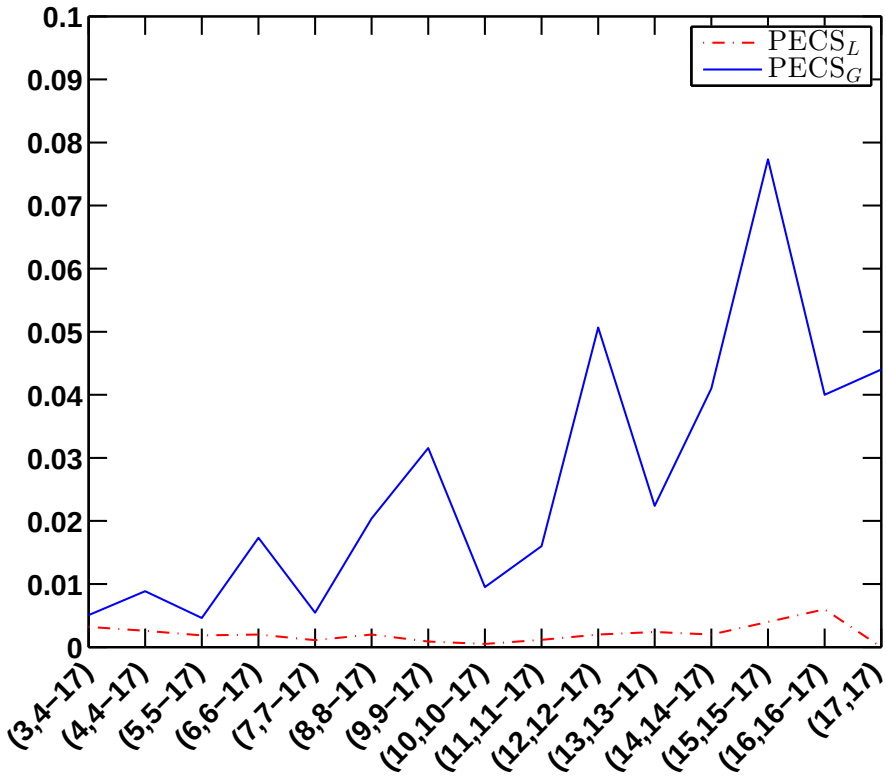

Figure 4: The proportion of incorrect estimations of two periodicities for the PECS algorithms. Each point on the x-axis represent average error for all combination of that point and smaller (or equal) periodicities.

how PECS$_G$ outperform SPE (and thus also all the other mentioned estimators) for all periodicities and $p_{1,1}$.

Next, we investigate how well PECS$_G$ and PECS$_L$ are able to resolve two periodicities in a binary sequence. In this case, some care needs to be taken when setting up the simulations, as when generating two periodicities, these may overlap or combine to create a new periodicity, e.g., if generating two periodicities of period six, these may be placed such that they instead form just a single periodicity with period three. Similarly, two periodicities with period four and twelve

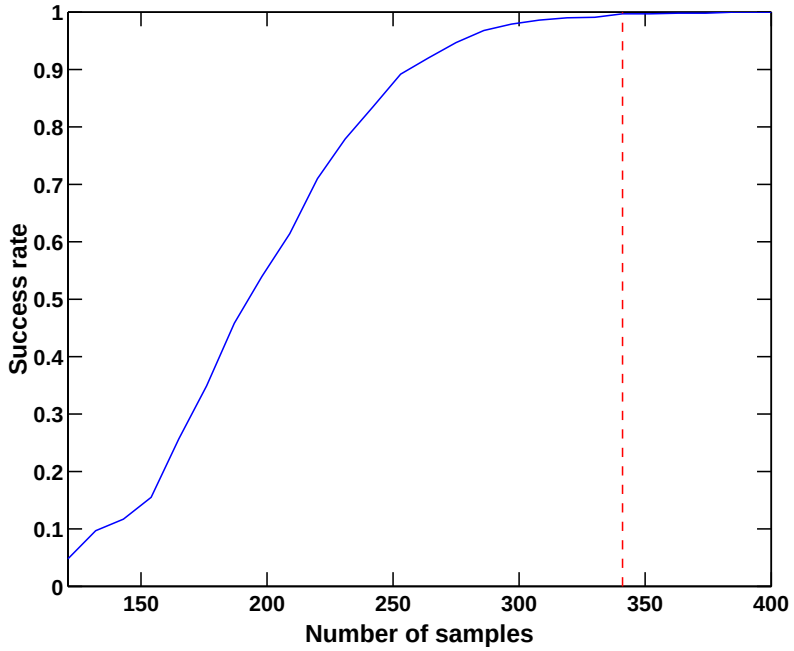

Figure 5: Rate of success for PECS$_G$ in estimating the periodicities of a signal with periodicities at 11 and 31, as a function of signal length. The dashed line denotes the minimum data needed for using [8].

may cause the resulting sequence to have only a single periodicity of four. In order to avoid such ambiguities in the resulting performance measure, the test data has been generated such that it avoids this form of ambiguities. Figure 4 illustrates the success rate of determining both periodicities correctly, as a function of the length of the two periodicities, with $N = 500$ and again using $p_{1,1} = 3/4$ and $p_{0,1} = 1/4$. Each point on the x-axis should be interpreted as the average error for all combinations of periodicities within the brackets, i.e., for instance $(14, 14 - 17)$ denotes all combinations $(14, 14), (14, 15), (14, 16)$ and $(14, 17)$. As may be seen from the figure, even when the sequence contains two periodicities of lengths up to 12, when most of the other discussed estimators completely

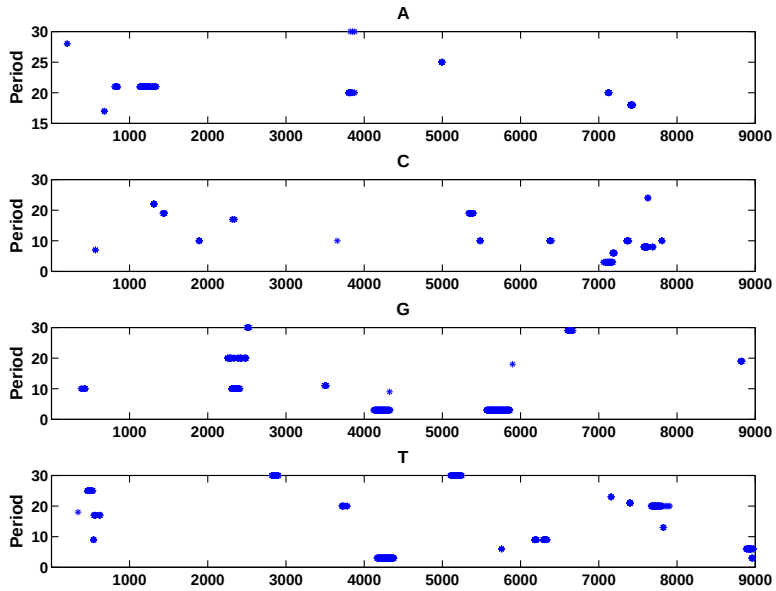

Figure 6: The periodicities of each symbol in the gene C.elegans F56F11.4 computed using a sliding window.

fail to find even a single perfect periodicity, both PECS algorithms have a very low proportion of errors. From the figure, one can also observe that, as expected, the PECS_L outperforms the PECS_G when there is more than one periodicity present in the sequence. For the last simulated data experiment, we recreate a simulation experiment similar to the one that was used in [8], where a deterministic periodicity of 11 and 31 are present simultaneously in a signal generated from a 4 element set being uniformly distributed on the other indices. As can be seen in Figure 5, the PECS_G estimator achieves almost 100 % success rate even before the method presented in [8] can start to be used, since it requires a minimum of $11 \times 31 = 341$ data points. Finally, we examine the performance of the PECS_G estimator on measured genomic data, in the form of the gene C. elegans F56F11.4 [28]. Since genomic data is generally not stationary, the estimate has

been formed using a sliding window with length $N = 360$. The results obtained by PECS$_G$ are shown in Figure 6, where the periodicities with a likelihood ratio greater than the 95% quantile of the maximum of $M = 465$ χ^2 distributed random variables are shown for each symbol. In earlier work, such as [10] and [27], a period of three was found at around index 7000. This period was also found when using PECS$_G$, but when looking at the corresponding $\tilde{p}$, one may note that this periodicity is actually constituted by the lack of the symbol C, i.e., this period is detected since the symbols A, G, and T are alternating in a non-periodic fashion, and since C is always absent at these indices, this apparently causes the Fourier based methods to indicate a periodicity of three. If one is not interested in finding these sorts of periodicities, one may restrict $p_{1,1}$ to be in $[1/2, 1]$, in the same manner as mentioned above. This will ensure that PECS$_G$ only finds periodicities that are made up by an increased probability in the presence of a symbol.

6 Conclusion

In this work, we have presented a likelihood-based approach for modeling periodicities in symbolic sequences. Modeling the observations using a categorical distribution with periodic indices, possibly having a different distribution, leads to a difficult combinatorial problem. Here, we have proposed two algorithms to relax the problem using sparse heuristics: namely, one fast greedy approach which builds up the solution set in an iterative fashion, and one based on convex relaxation ideas, which has the benefit of a more efficient usage of the data. Finally, we show the benefits of the proposed algorithms as compared to previously published methods using simulation experiments as well as with real DNA data examples.

7 Acknowledgement

The authors would like to thank Prof. Lorenzo Galleani and Dr. Roberto Garello at Politecnico di Torino, Italy, for providing us with the their implementation of MEM-algorithm detailed in [10].

References

[1] S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, "Likelihood-based Estimation of Periodicities in Symbolic Sequences," in *Proceedings of the 21th European Signal Processing Conference*, Marrakesh, 2013.

[2] A. Agresti, *Categorical Data Analysis*, John Wiley & Sons, second edition, 2007.

[3] M. B. Chaley, E. V. Korotkov, and K. G. Skryabin, "Method Revealing Latent Periodicity of the Nucleotide Sequences Modified for a Case of Small Samples," *DNA Res.*, vol. 6, no. 3, pp. 153–163, 1999.

[4] E. Korotkov and N. Kudryaschov, "Latent periodicity of many genes," *Genome Informatics*, vol. 12, pp. 437–439, 2001.

[5] D. Anastassiou, "Genomic Signal Processing," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 8–20, July 2001.

[6] W. Wang and D. H. Johnson, "Computing linear transforms of symbolic signals," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 628–634, March 2002.

[7] G. L. Rosen, *Signal Processing for Biologically-Inspired Gradient Source Localization and DNA Sequence Analysis,*, Ph.D. thesis, Georgia Institute of Technology, 2006.

[8] R. Arora, W. A. Sethares, and J. A. Bucklew, "Latent Periodicities in Genome Sequences," *IEEE J. Sel. Topics in Signal Processing*, vol. 2, no. 3, pp. 332–342, June 2008.

[9] L. Wang and D. Schonfeld, "Mapping Equivalence for Symbolic Sequences: Theory and Applications," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4895–4905, Dec. 2009.

[10] L. Galleani and R. Garello, "The Minimum Entropy Mapping Spectrum of a DNA Sequence," *IEEE Trans. Inf. Theory*, vol. 56, no. 2, pp. 771–783, Feb. 2010.

[11] J. Epps, H. Ying, and G. Huttley, "Statistical methods for detecting periodic fragments in DNA sequence data," *Biology Direct*, vol. 6, no. 21, pp. 1–16, 2011.

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

[13] D. D. Muresan and T. W. Parks, "Orthogonal, exactly periodic subspace decomposition," *IEEE Transactions on Signal Processing*, vol. 51, no. 9, pp. 2270–2279, Sept. 2003.

[14] W. A. Sethares and T. W. Staley, "Periodicity transforms," *IEEE Transactions on Signal Processing*, vol. 47, no. 11, pp. 2953–2964, Nov 1999.

[15] E. L. Lehmann and G. Casella, *Theory of Point Estimation (Springer Texts in Statistics)*, Springer, 2nd edition, 1998.

[16] 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.

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

[18] G. Casella and R. Berger, *Statistical Inference*, Duxbury, 2nd edition, 2002.

[19] P. Emberchts, C. Klüppelberg, and T. Mikosch, "Fluctuations of Maxima," in *Modelling Extremal Events*, vol. 33 of *Applications of Mathematics*, pp. 113–179. Springer Berlin Heidelberg, 1997.

[20] P. G. Bühlmann and S. van de Geer, *Statistics for High-Dimensional Data*, Springer Series in Statistics. Springer, 2011.

[21] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[22] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 707–710, Oct. 2007.

[23] 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.

[24] 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.

[25] 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.

[26] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani, "Pathwise Coordinate Optimization," *The Annals of Applied Statistics*, vol. 1, no. 2, pp. 302–332, 2007.

[27] J. Swärd and A. Jakobsson, "Subspace-based estimation of symbolic periodicities," in *38th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Vancouver, Canada, May 26-31 2013.

[28] National Center for Biotechnology Information, "Genome sequence of the nematode C. elegans: a platform for investigating biology," http://www.ncbi.nlm.nih.gov/nuccore/FO081497.1.

B

Paper B

High Resolution Sparse Estimation of Exponentially Decaying N-D Signals

Johan Swärd, Stefan Ingi Adalbjörnsson, and Andreas Jakobsson

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

In this work, we consider the problem of high-resolution estimation of the parameters detailing an N-dimensional (N-D) signal consisting of an unknown number of exponentially decaying sinusoidal components. Since such signals are not sparse in an oversampled Fourier matrix, earlier approaches typically exploit large dictionary matrices that include not only a finely spaced frequency grid, but also a grid over the considered damping factors. Even in the 2-D case, the resulting dictionary is typically very large, resulting in a computationally cumbersome optimization problem. Here, we introduce a sparse modeling framework for N-dimensional exponentially damped sinusoids using the Kronecker structure inherent in the model. Furthermore, we introduce a novel dictionary learning approach that iteratively refines the estimate of the candidate frequency and damping coefficients for each component, thus allowing for smaller dictionaries, and for frequency and damping parameter that are not restricted to a grid. The performance of the proposed method is illustrated using simulated data, clearly showing the improved performance as compared to previous techniques.

Key words: Sparse signal modeling, spectral analysis, sparse reconstruction, parameter estimation, dictionary learning, damped sinusoids.

1 Introduction

High-dimensional decaying sinusoidal signals occur in a wide variety of fields, such as spectroscopy, geology, sonar, and radar, and given the importance of such signals in a variety of applications, the topic has attracted notable attention in the recent literature (see, e.g. [1–11]). Common solutions include subspace-based algorithms [3–8], which are typically making relatively strong model assumptions, or the use of high-dimensional representations necessitating an iterative zooming procedure over multiple dimensions, such as the technique introduced in [11]. These kind of approaches often suffer from high complexity and sub-optimal performance, typically requiring an accurate initialization or model order information to yield reliable results, information which is commonly not available in many of the discussed applications.

Often, the measurements are also assumed to be uniformly sampled, which may well be undesired in applications such as, for instance, spectroscopy. Furthermore, the number of modes present in the signal is generally unknown, or may vary over time, typically necessitating some form of model order selection decision. Given such difficulties, it is often of interest to formulate non-parametric or semi-parametric modeling techniques, imposing only mild assumptions of the *a priori* knowledge of the signal structure. Popular solutions include the so-called dCapon, dAPES, and dIAA spectral estimators, which all form generalized spectral estimates of the signal, constructing spectral representations over both the frequency and damping dimensions [12, 13] (see also [14, 15]). Although this form of techniques are robust to the made model order assumptions, they suffer difficulties in separating closely spaced modes from each other, and typically require notable computational efforts if not implemented carefully [15].

As an alternative, one may use sparse modeling of the signal, forming a large dictionary of all potential frequencies and damping candidates, thus generally having vastly more columns than rows. For a given signal and the resulting dictionary matrix, one thus wishes to find the sparsest solution to the resulting linear set of equations, mapping the signal to a linear combination of a few of the columns of the dictionary. Such techniques have successfully been applied to line spectral data, and the topic has attracted notable attention in the recent literature (see, e.g., [16–22]). Although these algorithms appear quite different from each other, they share the property that the considered dictionary grid should be selected sufficiently fine to allow for a sparse signal representation (see also [23, 24]), which, if extended to also consider damped modes, necessitates a large dictionary

matrix containing elements with a sufficiently fine grid over the range of both the potential frequencies and damping candidates (see, e.g., [13, 25, 26]); this will be particularly noticeable if treating large data sets, or data sets with multiple measurement dimensions. In order to mitigate this problem, we here introduce a tensor representation of the signal model, allowing us to exploit the resulting inherent Kronecker structure, which may be exploited to significantly reduce the required complexity as compared to a naive implementation of the sparse modeling framework.

Furthermore, we propose a novel dictionary learning approach, wherein one iteratively decomposes the signal with a fixed small dictionary, adaptively learning the dictionary elements best suited to enhance sparsity. To this effect, we initially form a coarsely spaced dictionary with undamped modes over the range of considered frequency candidates, iteratively adapting both the frequency and damping settings for the dictionary elements, thereby also allowing for both a reduction and an expansion of the number of dictionary elements considered in each iteration of the optimization. In order to further reduce complexity, we propose a computationally efficient implementation based on the concept of the alternating direction method of multipliers (ADMM) (see, e.g., [27]), where the Kronecker structure of the resulting dictionary matrices may be exploited to dramatically decrease the cost of each iteration.

The remainder of the paper is organized as follows: in the next section, we introduce the considered data model. Then, in Section 3, we introduce the idea behind decoupling the search dimensions. Section 4 introduces the ADMM formulation of the estimator, and Section 5 illustrates the performance of the proposed estimator using simulated data. Finally, Section 6 contains our conclusions. In the remainder of the paper, we use the following notation: scalars are represented using lower case letters, whereas vectors are represented with lower case bold-face letters. Matrices are represented with capital bold-face letters, tensors with capital bold Euler script letter, $(\cdot)^T$ denotes the transpose, and $(\cdot)^H$ the conjugate transpose.

2 N-dimensional signal model

Consider an N-dimensional signal consisting of a sum of K modes, i.e., K N-dimensional damped sinusoids such that observation $x_{\boldsymbol{\tau}}$ at a sampling point $\boldsymbol{\tau}$,

where

$$\boldsymbol{\tau} = \left[\begin{array}{cccc} t_{i_1}^{(1)} & t_{i_2}^{(2)} & \dots & t_{i_N}^{(N)} \end{array} \right]^T \tag{1}$$

and $t_{i_\ell}^{(\ell)}$ denotes the i_ℓ:th sampling point in dimension ℓ, may be well modeled as

$$x_{\boldsymbol{\tau}} = \sum_{k=1}^{K} g_k \prod_{\ell=1}^{N} \xi_{k,\ell}^{t_{i_\ell}^{(\ell)}} + \varepsilon_{\boldsymbol{\tau}} \tag{2}$$

where

$$\xi_{k,\ell} = \mathrm{e}^{j\omega_k^{(\ell)} - \beta_k^{(\ell)}} \tag{3}$$

and with g_k denoting the complex amplitude of mode k, and $\varepsilon_{\boldsymbol{\tau}}$ is an additive noise term, here for simplicity assumed to be an independent identically distributed circularly symmetric Gaussian random variable. Assuming the signal is observed over $t_{i_n}^{(n)}$, for $i_n = 1, \dots, I_n$, and $n = 1, \dots, N$, the entire sequence may be stored in an N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$. It is worth noting that this formulation makes no restriction on any of the dimensions to have a sampling scheme that is equidistant, thus encompassing both missing data scenarios as well as irregular sampling. The entire model may thus be written in tensor format as the sum of K rank one tensors, such that

$$\boldsymbol{\mathcal{X}} = \sum_{k=1}^{K} g_k \tilde{\mathbf{a}}^{(1)}(k) \circ \tilde{\mathbf{a}}^{(2)}(k) \cdots \circ \tilde{\mathbf{a}}^{(N)}(k) + \boldsymbol{\mathcal{E}} \tag{4}$$

where $\circ$ denotes the outer product defined such that element $\boldsymbol{\tau}$ of $\boldsymbol{\mathcal{X}}$ corresponds to equation (2), $\boldsymbol{\mathcal{E}}$ is the tensor containing the noise terms, and

$$\tilde{\mathbf{a}}^{(n)}(k) = \left[\begin{array}{ccc} \xi_{k,n}^{t_1^{(n)}} & \dots & \xi_{k,n}^{t_{I_n}^{(n)}} \end{array} \right]^T \tag{5}$$

For an overview of tensor algebra sufficient for the here discussed results see, e.g., [28], which also use a notation consistent with the one used in this article. The model thus contain $(2N+1)K + 1$ unknown parameters, namely

$$\boldsymbol{\vartheta} \triangleq \left[\begin{array}{cc} \{\{\omega_k^{(n)}, \beta_k^{(n)}\}_{n=1}^{N}, g_k\}_{k=1}^{K}, & K \end{array} \right]^T \tag{6}$$

of which $2NK$ are non-linear parameters. Clearly, one could, in theory, form a non-linear least squares (LS) minimization over these parameters, as well as form

a model order estimate from the resulting model order residuals for varying possible candidate model sizes. However, such a solution would in most practical situations be computationally unfeasible, even for low dimensional data sets, especially as the optimization is well known to have numerous local minima [29]. To avoid this, we introduce a sparse modeling heuristic to approximate the model. This can be done by creating a large dictionary of candidate parameters, selected from a grid fine enough such that each true parameter lies sufficiently close to some grid point. For instance, if, to simplify our notation, one considers a single N-dimensional sinusoid and fix all but the first frequency and damping coefficients, then one may approximate (4) using a dictionary containing P_1 and J_1 candidate elements along the (first) frequency and damping dimension, respectively, such as

$$\boldsymbol{\mathcal{X}} \approx \sum_{p=1}^{P_1} \sum_{j=1}^{J_1} g_{p,j} \mathbf{a}^{(1)}_{\omega_p}(\beta_j) \circ \mathbf{a}^{(2)}_{\omega_2}(\beta_2) \circ \cdots \circ \mathbf{a}^{(N)}_{\omega_N}(\beta_N) \tag{7}$$

where $\omega_2, \ldots, \omega_N$ and $\beta_2, \ldots, \beta_N$ denote the (for simplicity) fixed frequency and damping coefficients along the 2nd to N:th dimensions,

$$\mathbf{a}^{(n)}_{\omega}(\beta) = \begin{bmatrix} \xi_n^{1^{(n)}} & \ldots & \xi_n^{I_n^{(n)}} \end{bmatrix}^T$$

where

$$\xi_n = \mathrm{e}^{j\omega^{(n)} - \beta^{(n)}} \tag{8}$$

and $g_{k,\ell}$ denotes the contribution of each of these dictionary elements in the approximation. Thus, as long as P_1 and J_1 are selected sufficiently large to allow for a grid of dictionary elements such that the true frequency and damping coefficients lie close to one of the grid points, only one $g_{p,j}$ should be non-zero for each of the K modes. By similarly extending the dictionary for each of the frequency and damping dimensions, such that $g_{p_1,\ldots,p_N,j_1,\ldots,j_N}$ denotes the contribution of the corresponding dictionary elements for the p_q:th and j_r:th frequency and damping dictionary elements, where $q, r \in \{1, \ldots, N\}$, the resulting (very large) dictionary would allow for a sparse approximative solution of the unknown parameters, such that most of the dictionary elements would not contribute to the approximation. Given such an approximative solution, the number of modes, K, may be estimated as the number of elements with non-zero contribution to the approximation. The non-linear parameters may then be estimated correspondingly, such

that for any non-zero variables, e.g., $g_{h_1,\ldots,h_N,i_1,\ldots,i_N}$, the non-linear parameters are estimated as the frequency and damping coefficient that correspond to the found coefficients. Such a solution may be obtained by reformulating the problem using the *vec* operator, defined here for tensors such that it is the usual *vec* operation on the mode-1 matricization, or unfolding (see also [28]), of a given tensor, i.e.,

$$vec\left(\boldsymbol{\mathcal{X}}\right) \triangleq vec\left(\mathbf{X}_{(1)}\right) \tag{9}$$

This allows for a sparse LS solution to be found by solving

$$\min_{\tilde{\mathbf{g}}} \|vec\left(\boldsymbol{\mathcal{X}}\right) - \tilde{\mathbf{A}}\tilde{\mathbf{g}}\|_2^2 + \rho(\tilde{\mathbf{g}}) \tag{10}$$

where $\tilde{\mathbf{g}} = vec\left(\boldsymbol{\mathcal{G}}\right)$, with $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{P_1 \times \cdots \times J_N}$ denoting the tensor formed from the amplitudes of all of the dictionary elements, and the i:th column of $\tilde{\mathbf{A}}$ is formed as

$$\tilde{\mathbf{A}}_{:i} = vec\left(\mathbf{a}^{(1)}_{\omega_{p_1}}(\beta_{j_1}) \circ \mathbf{a}^{(2)}_{\omega_{p_2}}(\beta_{j_2}) \cdots \circ \mathbf{a}^{(N)}_{\omega_{p_N}}(\beta_{j_N})\right) \tag{11}$$

where the notation $\mathbf{A}_{:i}$ denotes the ith column of the matrix $\mathbf{A}$. The penalty term $\rho(\cdot)$ is added in (10) as the grid is typically chosen such that the number of elements in $vec(\boldsymbol{\mathcal{X}})$ is smaller than the number of columns in $\tilde{\mathbf{A}}$; thus, if assuming that $\tilde{\mathbf{A}}$ is of full rank, the system of equations is under-determined, with infinitely many solution, out of which one is interested in finding a solution that appropriately weighs sparsity and model fit. Ideally, $\rho(\cdot)$ could be chosen as a function counting the number of non-zero elements. However, the resulting optimization problem is well known to be combinatorial in nature and will be unfeasible to solve even for moderate problem sizes. Common approximative choices include the scaled ℓ_1 norm [17,30], ℓ_q penalties [16,31], and the reweighted ℓ_1 approach, which may be seen to correspond to the log penalty [32]. Herein, we consider the ℓ_1 and the log penalty. It is worth noting that the above sparsity restrictions allow for solutions having multiple damping coefficients for a given frequency. Such solutions imply that the component is not an exponentially damped sinusoid; as this is not relevant for the here considered application, we proceed to refine the constraint such that it will only yield unique frequency-damping pairs for each component. To this end, we propose an iterative dictionary learning approach such that the damping parameters for each sinusoidal component is held fixed during the sparse LS step, after which the damping parameters are found using

the residual from the sparse LS step, one mode at the time, thus allowing for damping and frequency estimation to be performed with a non-linear optimization algorithm, e.g., Newton's method. Thus, we initially fix all damping parameters to zero, modifying (7) such that the dictionary is only formed over the unknown frequencies, i.e.,

$$\boldsymbol{\mathcal{X}} \approx \sum_{p_1=1}^{P_1} \cdots \sum_{p_N=1}^{P_N} g_{p_1,\ldots,p_N} \mathbf{a}^{(1)}_{\omega_{p_1}}(\beta_{p_1}) \circ \cdots \circ \mathbf{a}^{(N)}_{\omega_{p_N}}(\beta_{p_N}) \tag{12}$$

The resulting minimization with respect to the K unknown frequencies, which may then be used to estimate the damping components, iteratively finding each of the set of estimates. To allow for a computationally efficient solution, the considered frequency and damping grids, respectively, are updated in each iteration, such that the dictionary is refined in each step of the iteration. However, even with such a reduction in complexity, the iterative optimization problems are clearly daunting, being formed over $J_1 \times \cdots \times J_N$ and $P_1 \times \cdots \times P_N$ dimensions, respectively. In the next two sections, we therefore proceed to examine how these minimizations may be performed in an efficient manner utilizing the Kronecker structure of the dictionary matrices for the sparse LS step, and by solving the non-linear damping parameter estimation one mode at a time.

3 ADMM implementation

The minimization problem considered in (10) may be solved using an approximation of the form

$$\min_{\tilde{\mathbf{g}}} \|vec(\boldsymbol{\mathcal{X}}) - \tilde{\mathbf{A}}\tilde{\mathbf{g}}\|_2^2 + \sum_{k=1}^{P_1 \times \cdots \times J_N} \lambda_k |\tilde{g}_k| \tag{13}$$

where λ_k denotes a set of tuning parameters, for $k = 1, \ldots, P_1 \times \cdots \times J_N$.

In case these tuning parameters are all selected equal and the penalty is included as an inequality constraint, the resulting minimization is equivalent with the regular ℓ_1 penalized LS problem, often called basis pursuit denoising [33], or the Lasso [30]. For highly correlated dictionary elements, as may be required for high resolution N-D spectra, one may obtain sparser solutions using a reweighted Lasso formulation [32], such that the λ_k:s are instead selected as

$$\lambda_k = \frac{\varphi}{|\tilde{g}_k(\ell)| + \varepsilon} \tag{14}$$

Algorithm 1 Sparse LS via ADMM

1: Initiate $\mathbf{z} = \mathbf{z}(0), \mathbf{u} = \mathbf{u}(0)$, and $\ell = 0$
2: **repeat**
3: $\quad \mathbf{z}(\ell+1) = \left(\tilde{\mathbf{A}}^H\tilde{\mathbf{A}} + \mu\mathbf{I}\right)^{-1}\left(\tilde{\mathbf{A}}^H\mathbf{y} + \mu(\mathbf{u}(\ell) - \mathbf{d}(\ell))\right)$
4: $\quad \mathbf{u}(\ell+1) = \Psi\left(\mathbf{z}(\ell+1) + \mathbf{d}(\ell+1), \frac{\lambda}{\mu}\right)$
5: $\quad \mathbf{d}(\ell+1) = \mathbf{d}(\ell) + \mathbf{z}(\ell+1) - \mathbf{u}(\ell+1)$
6: $\quad \ell \leftarrow \ell + 1$
7: **until** convergence

where the constant ε is included to avoid numerical problems when $g_k(\ell)$ is close to zero. Here, $\tilde{g}_k(\ell)$ denotes the value of g_k at iteration ℓ, and with $\varphi > 0$ denoting a tuning parameter controlling the sparsity at the solution. A general efficient iterative algorithm for solving problems such as (10), using an ADMM implementation was proposed in [27], and may be easily adapted to the here considered reweighted scenario. The steps involved are summarized in Algorithm 1, where the Ψ operator is a shrinkage operator, defined as

$$\Psi(\mathbf{x}, \gamma) = \mathbf{x}(1 - \gamma/|\mathbf{x}|)^+ \tag{15}$$

where $(\cdot)^+$ denotes the positive part of a scalar. In Algorithm 1, $\tilde{\mathbf{g}}$ has been split up into two separate variables $\mathbf{z}$ and $\mathbf{u}$. Furthermore, $\mathbf{d}$ denotes the scaled dual variable, (see, e.g., [27] for a detailed discussion). The complexity of each iteration in the resulting algorithm is approximately $\mathcal{O}(n^2p)$, where p and n denote the columns and rows of $\mathbf{a}$, respectively. This is about the same as the computational cost for many Lasso solvers (see e.g. [34]). In the N-dimensional case, the overall computational complexity is about $\mathcal{O}(\prod_{n=1}^{N} J_n P_n \prod_{n=1}^{N} I_n^2)$, implying that even a 3-dimensional problem with 100 grid points in each dimension would result in a cost of approximately $100^{12} I_1 I_2$ operations, in each step, where I_n denotes the number of samples in dimension n. Fortunately, this complexity may be significantly reduced by exploiting the inherent Kronecker structure of the model. In order to do so, we rewrite (4) using tensor products as

$$\boldsymbol{\mathcal{X}} \;=\; \boldsymbol{\mathcal{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)} + \boldsymbol{\mathcal{E}} \tag{16}$$

where the operator $\times_n$ represents the n-mode product of a tensor with a matrix, and the dictionary matrix for dimension n is given as

$$\mathbf{A}^{(n)} \triangleq \left[\begin{array}{ccc} \mathbf{a}^{(n)}_{\omega_{k_1}}(\beta_{k_1}) & \cdots & \mathbf{a}^{(n)}_{\omega_{K_1}}(\beta_{K_1}) \end{array}\right] \tag{17}$$

Expressed in this form, one may note that the matricization may be accomplished via Kronecker products instead (see, e.g., [28], [35]), yielding

$$\mathbf{X}_{(1)} = \mathbf{A}^{(1)}\mathbf{G}_{(1)}\left(\mathbf{A}^{(N)} \otimes \mathbf{A}^{(N-1)} \otimes \cdots \otimes \mathbf{A}^{(2)}\right)^T \tag{18}$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{X}_{(1)} \in \mathbb{C}^{I_1 \times \prod_{n=2}^{N} I_n}$ is obtained by stacking all the mode-1 slices of $\boldsymbol{\mathcal{X}}$, and with $\mathbf{G}_{(1)}$ defined similarly. Vectorizing the resulting mode-1 slices yields (see, e.g., [36]),

$$\text{vec}\left(\mathbf{X}_{(1)}\right) = \left(\mathbf{A}^{(N)} \otimes \cdots \otimes \mathbf{A}^{(2)} \otimes \mathbf{A}^{(1)}\right) \text{vec}\left(\mathbf{G}_{(1)}\right) \tag{19}$$

allowing us to express the parameters in (10) as

$$\tilde{\mathbf{g}} \triangleq \text{vec}\left(\mathbf{G}_{(1)}\right) \in \mathbb{C}^{\tilde{K} \times 1} \tag{20}$$

$$\tilde{\mathbf{A}} \triangleq \left(\mathbf{A}^{(N)} \otimes \cdots \otimes \mathbf{A}^{(2)} \otimes \mathbf{A}^{(1)}\right) \in \mathbb{C}^{\tilde{I} \times \tilde{K}} \tag{21}$$

As a result, the full $\tilde{\mathbf{A}}$ matrix does not need to be formed, and vector multiplication of the form $\tilde{\mathbf{A}}\mathbf{x}$ and $\tilde{\mathbf{A}}^H\mathbf{y}$, for any appropriately sized vector $\mathbf{x}$ and $\mathbf{y}$, may be computed iteratively by each sub-matrix $\mathbf{A}^{(n)}$, and by then reshaping the resulting elements (see, e.g., [37, p. 28] for further details). This allows for a dramatic complexity reduction. To illustrate this, consider the case where each $\mathbf{A}^{(\ell)}$ matrix is $n \times n$. Then, the operation $\tilde{\mathbf{A}}\mathbf{x}$, which would require about about n^{2N} multiplications if first forming $\tilde{\mathbf{A}}$ and then computing the inner-product using this matrix, may instead be formed using only about Nn^{N+1} (see, e.g., [38]) operations. Furthermore, the LS step in the ADMM algorithm for solving (10) may also be computed significantly cheaper by utilizing its Kronecker structure, simply by calculating the singular value decomposition of each sub-matrix $\mathbf{A}^{(n)} = \mathbf{U}_n\boldsymbol{\Sigma}_n\mathbf{V}_n^H$, and then utilizing that the singular value decomposition of $\tilde{\mathbf{A}}$ is given by (see, e.g., [36, p. 246])

$$\tilde{\mathbf{A}} = \mathbf{U}_{\tilde{\mathbf{A}}}\boldsymbol{\Sigma}_{\tilde{\mathbf{A}}}\mathbf{V}_{\tilde{\mathbf{A}}}^H \tag{22}$$

where

$$\mathbf{U}_{\tilde{\mathbf{A}}} = \mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_N \tag{23}$$

$$\boldsymbol{\Sigma}_{\tilde{\mathbf{A}}} = \boldsymbol{\Sigma}_1 \otimes \cdots \otimes \boldsymbol{\Sigma}_N \tag{24}$$

$$\mathbf{V}_{\tilde{\mathbf{A}}}^H = \mathbf{V}_1^H \otimes \cdots \otimes \mathbf{V}_N^H \tag{25}$$

As a result, one may solve step 3 in Algorithm 1 by solving the equivalent LS problem

$$\min_{\tilde{\mathbf{z}}} \left\| \begin{bmatrix} \mathbf{U}_{\tilde{\mathbf{A}}}^H \mathbf{y} \\ \mathbf{V}_{\tilde{\mathbf{A}}}^H \boldsymbol{\xi} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\Sigma}_{\tilde{\mathbf{A}}} \\ \sqrt{\mu}\mathbf{I} \end{bmatrix} \tilde{\mathbf{z}} \right\|_2^2 \tag{26}$$

where

$$\tilde{\mathbf{z}} = \left(\boldsymbol{\Sigma}_{\tilde{\mathbf{A}}}^2 + \mu \mathbf{I} \right)^{-1} \left(\boldsymbol{\Sigma}_{\tilde{\mathbf{A}}} \mathbf{U}_{\tilde{\mathbf{A}}}^H \mathbf{y} + \sqrt{\mu} \mathbf{V}_{\tilde{\mathbf{A}}}^H \boldsymbol{\xi} \right) \tag{27}$$

with $\tilde{\mathbf{z}} = \mathbf{V}_{\tilde{\mathbf{A}}}^H \mathbf{z}$ and $\boldsymbol{\xi} = \sqrt{\mu}(\mathbf{u}(\ell) - \mathbf{d}(\ell))$. Thus, the LS step can be solved by three matrix vector multiplications, two Hadamard products between vectors, one scalar multiplication of a vector, and a vector-vector addition, which may all be calculated using their inherent Kronecker structure, significantly reducing the computational cost. For example if each $\mathbf{A}^{(\ell)}$ is $n \times n$, the cost for our approach is approximately about Nn^{N+1} versus approximately n^{3N} for a solution that does not use the inherent structure of the equations.

4 Sparse dictionary learning

As noted above, the considered grid over the candidate frequency and damping coefficients are updated in alternating fashion. Let $\hat{K}$ denote the number of non-zero amplitudes after the sparse LS step. Then, the dictionary learning may be done by forming the residual[1]

$$\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)} \tag{28}$$

Using a relaxation-based procedure (see also [39]), one then iteratively adds back one mode at a time to the residual in (28), and form an estimate of the frequency and damping of this mode using an N-dimensional single mode solver, such as, for instance, the standard nonlinear least squares estimator or, in the case of uniformly sampled data, an estimator such as the PUMA estimator [40]. Using the refined parameter estimates, the mode is then subtracted again, and the next mode is refined similarly. The procedure is summarized in Algorithm 2. Using the refined modes, the dictionary is then updated, such that it is separated into N dictionaries, one over each dimension, with each dictionary being centered in a fine grid around each of the found frequencies. As a result, the unused dictionary

[1] To simplify our notation, we have here suppressed the dependencies on the frequency ω and the damping β.

Algorithm 2 Mode estimation

1: Initialize all damping coefficients to zero and use (10) to form initial estimates $\left\{g_{\boldsymbol{\Upsilon}_k}\right\}_{k=1}^{\hat{K}}$
2: **for** $i = 1, \ldots, iter_{max}$ **do**
3: Compute the residual according to (12)
4: **for** $k = 1, \ldots, \hat{K}$ **do**
5: Add the current mode to the residual: $\boldsymbol{\mathcal{Y}}_k = \boldsymbol{\mathcal{R}}_k + g_{\boldsymbol{\Upsilon}_k} \mathbf{a}_k^{(1)} \circ \cdots \circ \mathbf{a}_k^{(N)}$
6: Estimate the frequencies and the dampings for the mode
7: Remove the current mode: $\boldsymbol{\mathcal{R}}_k = \boldsymbol{\mathcal{Y}}_k - g_{\boldsymbol{\Upsilon}_k} \mathbf{a}_k^{(1)} \circ \cdots \circ \mathbf{a}_k^{(N)}$
8: **end for**
9: Use the found frequencies and damping coefficient to create new dictionaries and re-solve (10).
10: **end for**

elements, having zero-amplitudes, are excluded from the updated dictionary (unless being close to one of the found modes). This also implies that closely spaced modes may yield overlapping dictionary elements; such duplicated dictionary elements are removed to avoid collinearity in the dictionary. For each grid point, the dictionary element is scaled according to the found damping coefficient of the corresponding mode, to ensure that all dictionary elements have the same norm, thus refining the dictionary iteratively over both frequencies and damping coefficients. We coin the resulting method the Sparse Exponential Mode Analysis (SEMA) algorithm.

5 Numerical examples

We proceed to examine the performance of the proposed method using simulated data. To simplify the presentation, we focus on the 1-D, 2-D, and 3-D cases, since problems of these dimensions offer more intuitive results that are also easier to analyze. Considering first the 1-D case, we illustrate the performance of the proposed method using simulated data. We initially consider a data vector containing $N = 128$ samples of a three mode signal, where the frequencies and damping parameters are chosen uniformly over $[0,\ 1]$ and $[0,\ 0.025]$, re-

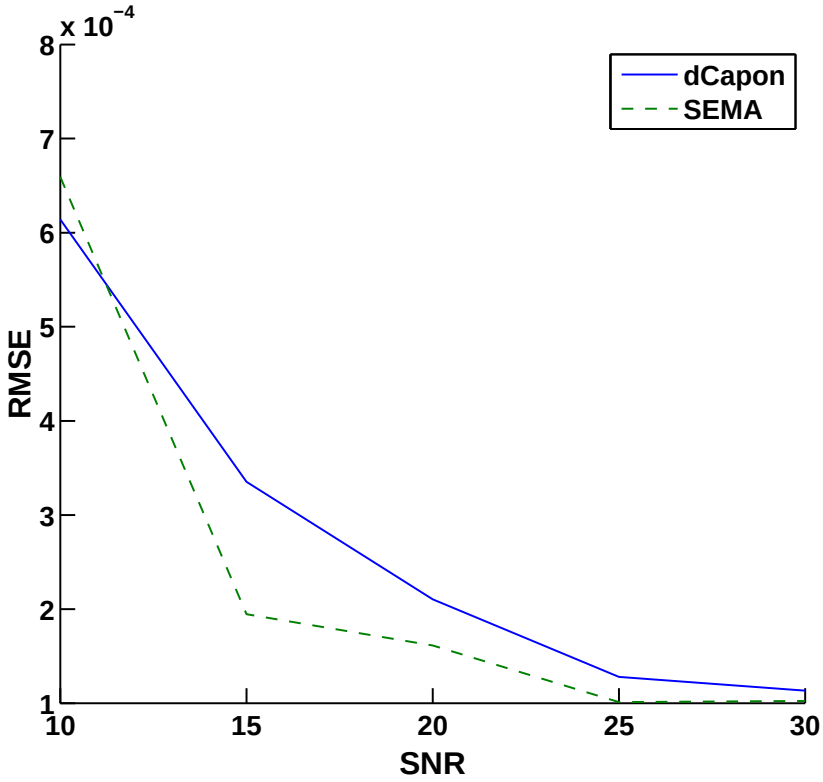

Figure 1: The RMSE of the frequency estimation as a function of SNR.

spectively. We note that we here use normalized frequencies, lying in the interval $[0,\ 1]$, denoted by the letter f. For now, we ensure that no modes are closer in frequency than $1/N$. Figures 1 and 2 depict the resulting performance of the SEMA algorithm, as compared to the non-parametric damped-Capon (dCapon) estimate [12, 15], as a function of the signal-to-noise-ratio (SNR), defined as $\log_{10}(||\mathbf{y}||_2^2/N\sigma^2)$, where σ^2 denotes the variance of the noise. The two figures show the root mean squared error (RMSE) of the frequency and damping estimates, defined as

$$\text{RMSE} = \sqrt{\frac{1}{MK}\sum_{m=1}^{M}\sum_{k=1}^{K}\left(\vartheta_{m,k} - \hat{\vartheta}_{m,k}\right)^2} \tag{29}$$

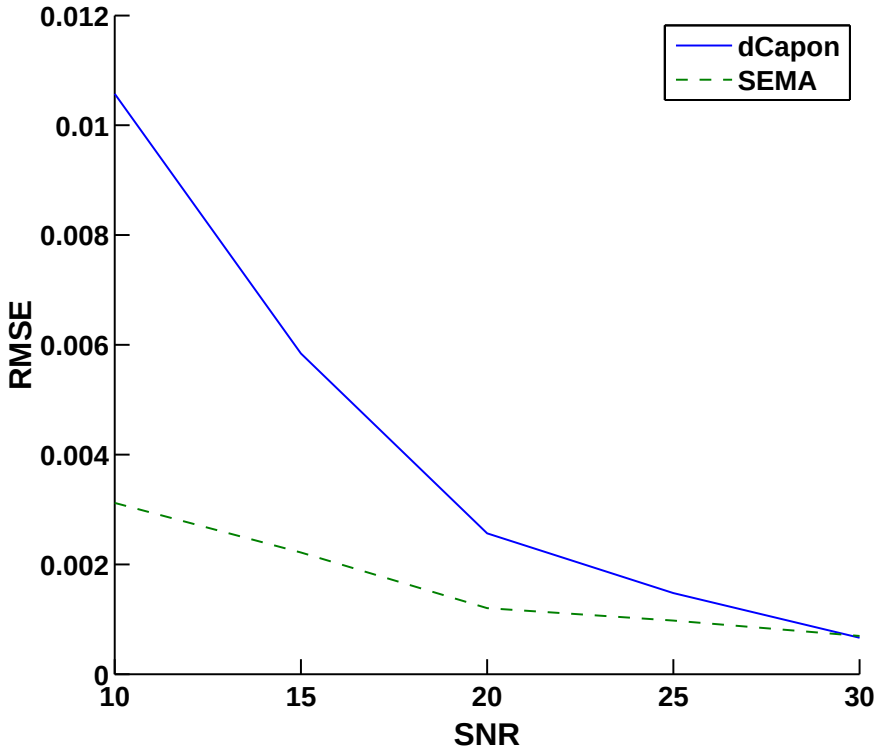

Figure 2: The RMSE of the damping estimation as a function of SNR.

where $\vartheta_{m,k}$ denotes the estimate of either the frequency or the damping of mode k for Monte-Carlo simulation m, M is the total number of Monte-Carlo simulations, and K the number of modes. These results have been obtained using $M = 175$ Monte-Carlo simulations. In this example, dCapon has a frequency grid that is selected to be 6000×6000, uniformly covering frequencies and damping factors in $[0, 1]$ and $[0, 0.025]$, respectively, and where the recommended filter length of $N/4$ is used. The SEMA algorithm on the other hand uses a dictionary containing only 128 elements in the first iteration, and, thereafter, uses only 40 grid points for each found mode when updating the dictionary in each subsequent iteration. As can be seen from the figures, the proposed SEMA algorithm yields notably better estimates than the dCapon estimator, without requiring a

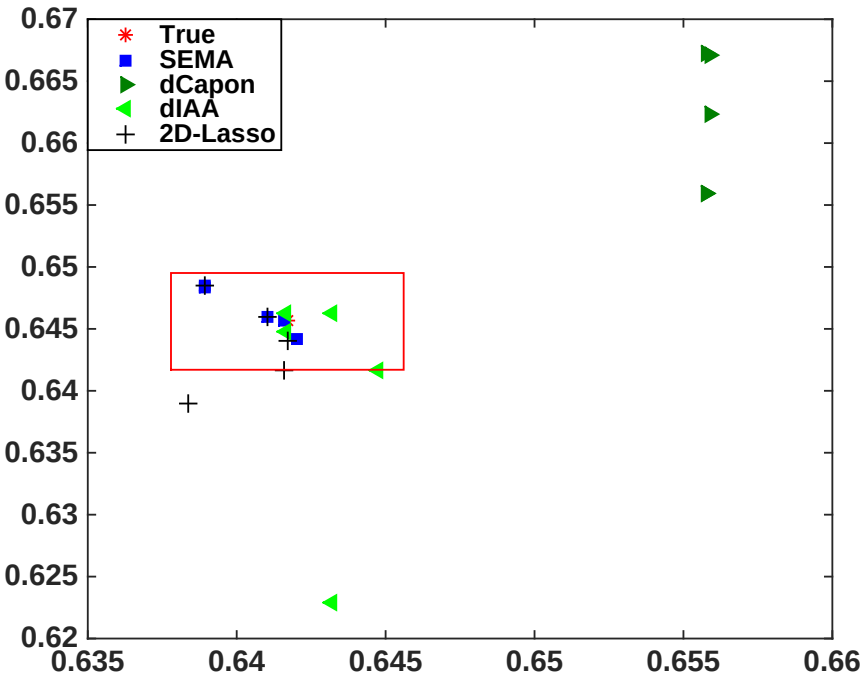

Figure 3: The result of resolving two closely spaced spectral peaks. The (red) square indicates the distance $1/(2N)$ from the true frequencies.

large dictionary grid over both dimensions, thereby allowing for a substantially faster implementation. It is also worth noting that the dCapon estimation errors are here larger than the smallest possible error that is attainable given the current grid size, implying that the grid size does not in itself limit the quality of the estimates.

Next, we examine the ability of the methods to resolve two closely spaced spectral lines. In this case, we consider a signal containing two sinusoidal components with frequencies, $f_1 = 0.6417$ and $f_2 = 0.6456$, i.e., separated by $0.5/N$, with damping constants being 0.010 for both modes. Figure 3 illustrates the resulting frequency estimates as obtained from 5 Monte-Carlo simulations, and SNR $= 20$ dB. For comparison, the figure also shows the estimates obtained us-

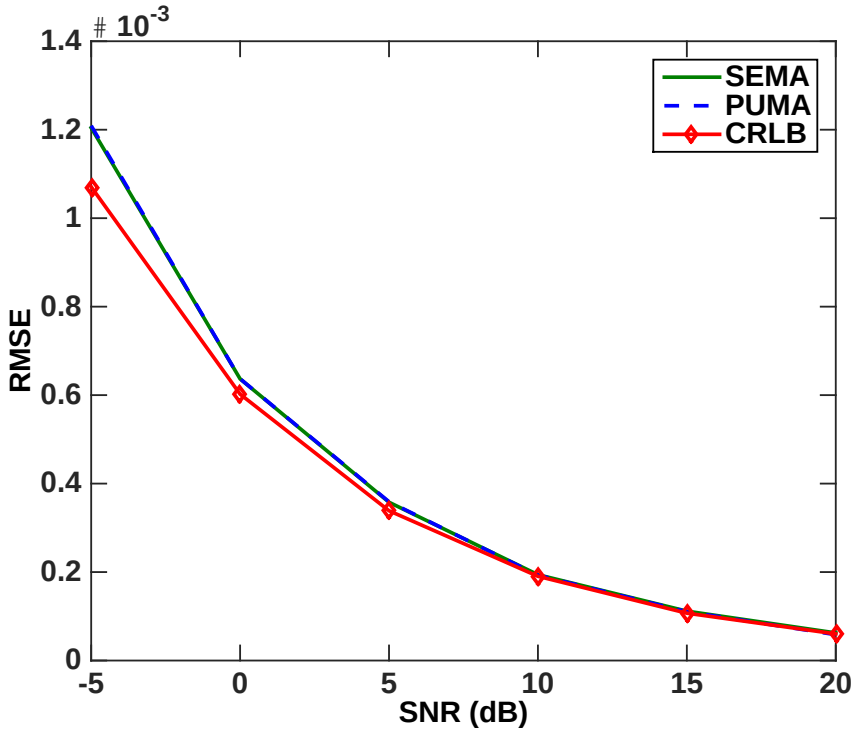

Figure 4: The average RMSE of $f_1^{(1)}$ and $f_2^{(1)}$ as a function of SNR.

ing 1-D SEMA, dCapon, dIAA [41], and for a Lasso method with a dictionary containing both frequencies and damping factors, and exploiting a zooming similar to the one used in SEMA. Here, to speed-up the computations, the frequency grid for dCapon has been selected to only be formed on $[0.63, 0.67]$, allowing the method notable *a priori* information on the frequency region of interest. The damping grid ranges over $[0, 0.025]$ and has size 500 for all methods, except for the used Lasso method, where, due to complexity reasons, it is set to 10. As seen in the figure, the proposed method clearly manages to resolve the two peaks, whereas the Lasso and dIAA estimates are only partly succeeding to do so, while dCapon yields noticeably biased estimates. In the figure, the (red) square indicates the region $1/(2N)$ centered around the true frequencies.

We proceed to examine the performance of the SEMA algorithm for 2-*D*

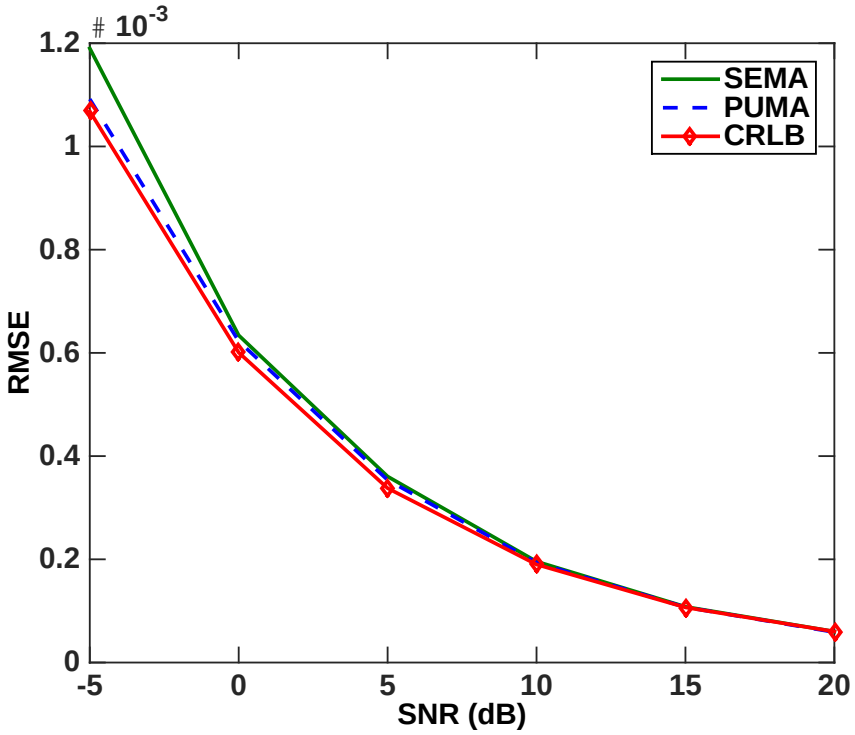

Figure 5: The average RMSE of $f_1^{(2)}$ and $f_2^{(2)}$ as a function of SNR.

simulated data, examining the RMSE of two well separated peaks, showing that the proposed method has similar performance to the statistically efficient PUMA method [7], using simulated data mimicking a 2-D Nuclear Magnetic Resonance (NMR) signal, simulated using (2), containing two damped sinusoids and having 33×31 samples. Figures 4-7 illustrates the performance of the SEMA estimator as compared to the parametric PUMA estimator and the corresponding Cramér-Rao lower bound (CRLB) [42]. The frequencies were randomly selected in the interval from 0 to 1 in normalized frequencies, and selected such that components were separated by at least $3/N$ in each dimension. If the spacing between the peaks is smaller, the estimation will degenerate for all methods. The damping parameters were set to $\beta_1 = (0.05 \quad 0.02)$ and $\beta_2 = (0.01 \quad 0.04)$ for all simulations. Each mode was normalized in amplitude, thus making sure that both peaks

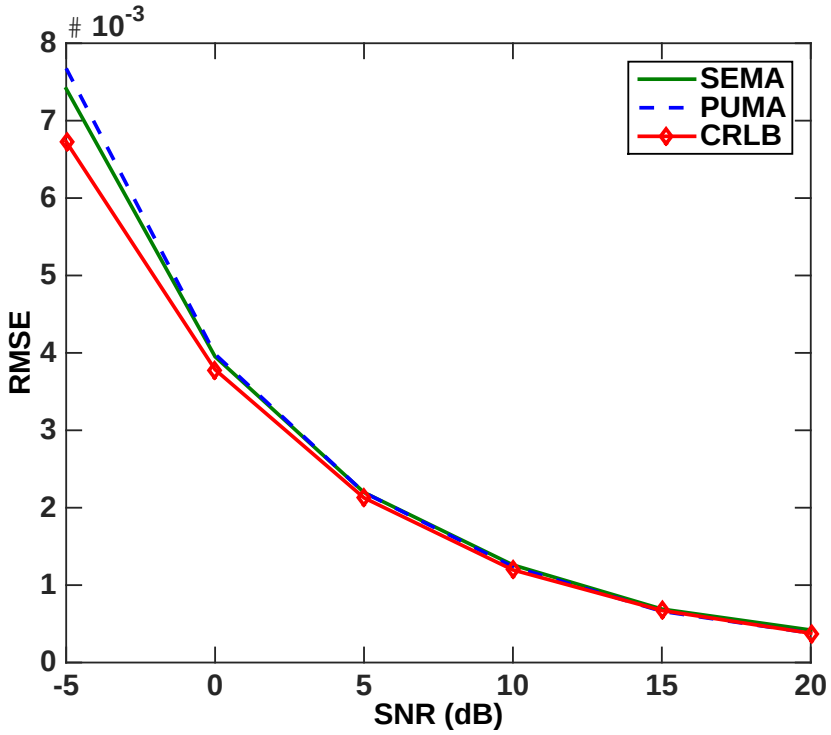

Figure 6: The average RMSE of $\beta_1^{(1)}$ and $\beta_2^{(1)}$ as a function of SNR.

were equally dominant. The PUMA algorithm was, as for all examples, allowed 100 iterations, as well as oracle model order information, and the initial grid for the proposed 2-D method was, as for the following examples, set to 100. The proposed method was allowed two iterations and used 33 grid points to zoom in on each found mode.

The choice of λ governs the number of peaks that may be found. If set too high, peaks with low amplitude will be suppressed, and if set too low, peaks that originate from the noise will not be suppressed. However, due to the reweighting step, a too small λ will be compensated for, and therefore the algorithm is relatively robust to the choice of λ, as long as it is not set too large. Therefore, it is preferable to set λ to a small value. In these examples, we set λ equal to the tenth largest peak found in the periodogram. One could argue that we thereby limit the num-

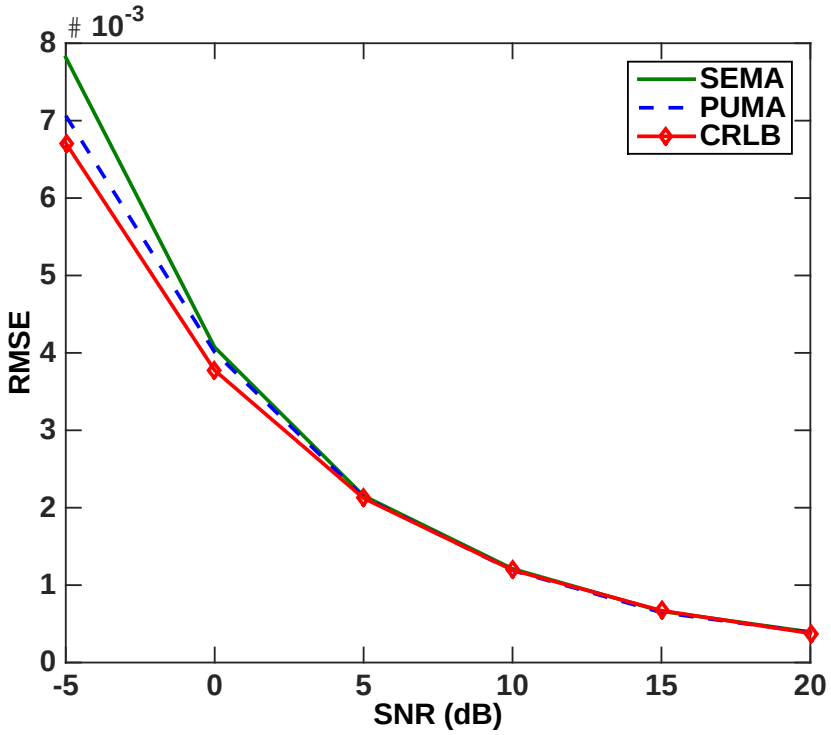

Figure 7: The average RMSE of $\beta_1^{(2)}$ and $\beta_2^{(2)}$ as a function of SNR.

ber of peaks that may be found, but that is easily avoided. If λ were set to equal the amplitude of the r:th largest peak and, when using the method, we found r peaks, one would run the algorithm a second time but with a somewhat smaller λ value. In this way, we make sure that we do not in fact limit the algorithm to a specified number of peaks. The test was performed using 250 Monte-Carlo simulations, for each value of the considered SNR. Figures 4-7 illustrate the total RMSE of all the unknown parameters. As can be seen from the figure, both the parametric PUMA, which has been allowed oracle model order information, and the proposed semi-parametric SEMA algorithms yield statistically efficient parameter estimates especially for larger larger SNR. Here, if the proposed algorithm did not manage to estimate the number of modes correctly, that estimate was then removed from the RMSE calculations for all methods. This happened two

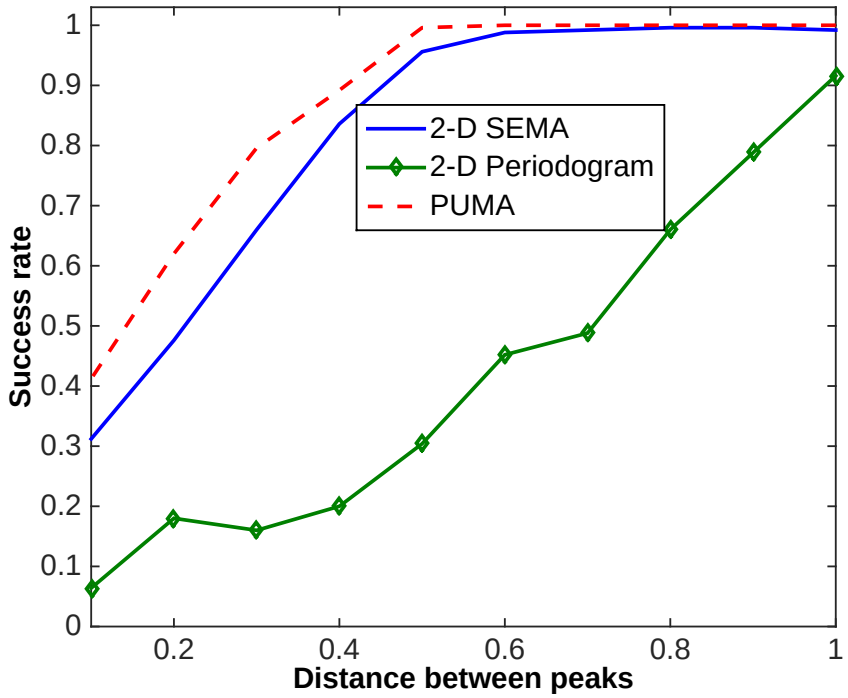

Figure 8: Ability to resolve two peaks as a function of the peak separation.

times out of 1500 Monte-Carlo simulations. The average computation times for 100 simulations on SNR level 20 was around 0.6 seconds for SEMA and 0.005 seconds for PUMA.

We proceed to examine the methods ability to resolve two closely spaced peaks. This was done by fixing the first mode at frequency $f_1 = (0.4, \quad 0.6)$, and letting the second mode gradually approach the first. The modes were initially separated by $1/N_1$ and $1/N_2$ in each frequency dimension, and the test was stopped when the modes were separated by $0.1/N_1$ and $0.1/N_2$. The data size for this example was again 33×31. The same SEMA settings as above were used. We also compare the estimates to that of a zero-padded 2-D periodogram, where 2^{13} zeros were padded in each dimension, but zoomed in on the correct frequencies (± 0.1 in each frequency). The damping parameters where fixed to 0.02 for

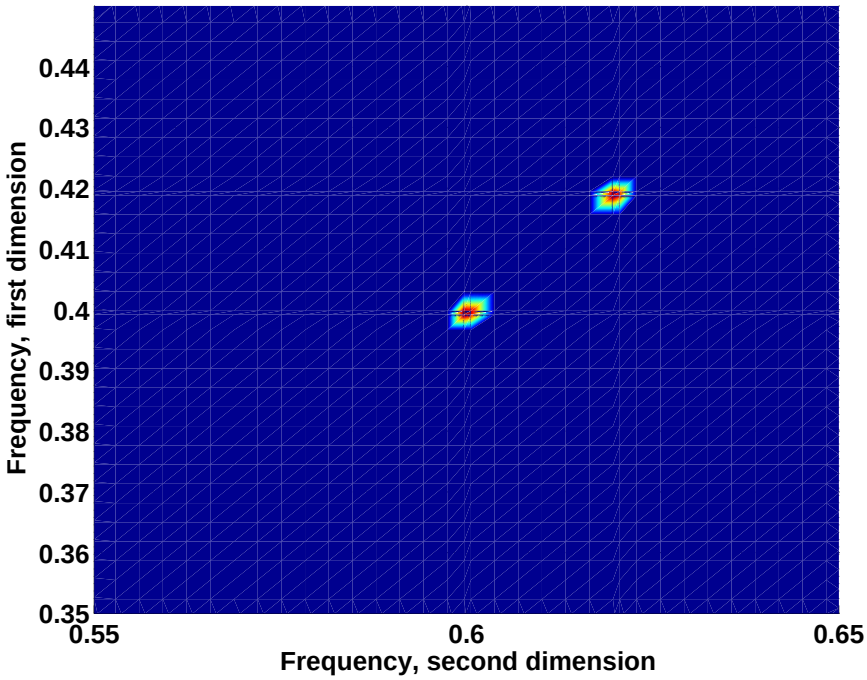

Figure 9: Resulting estimates using 2-D SEMA on two closely spaced modes.

all modes and dimensions, and the SNR was set to 10 dB. Furthermore, PUMA was again allowed complete knowledge of the number of peaks. To determine whether or not two peaks were resolved, we ensured that the method fulfilled at least two separation criteria: First, the peaks that were found had to be at least within a rectangle of size $1/N_1 \times 1/N_2$ from the correct frequencies; Secondly, the power of the valley between the peaks were allowed to be at most 90% of the average power of the peaks. If these two criteria were met, the modes were deemed to be resolved. The results are shown in Figure 8, where the x-axis should be interpreted as the distance divided by N_1, i.e., 0.1 means that the distance between the modes is $0.1/N_1$. As may be seen from the figure, the periodogram's ability to distinguish the two modes drastically decreases as the modes become closer. As may be expected. the PUMA method on the other hand manages to separate

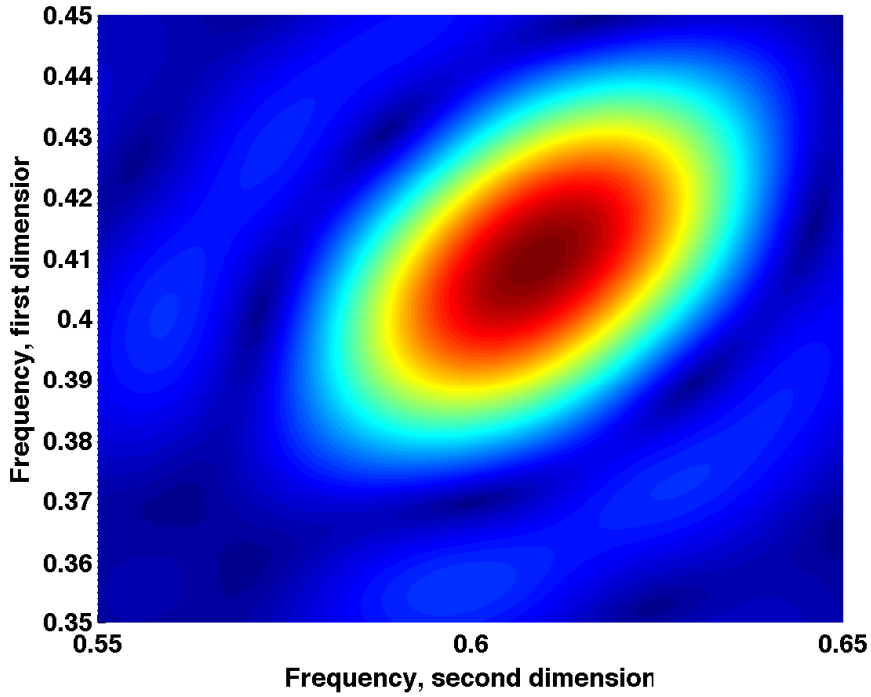

Figure 10: Resulting estimates using two dimensional periodogram on two closely spaced modes.

the modes very well until they are about 0.3 apart from each other. As can be seen from the figure, the SEMA method achieves about the same performance as PUMA until the distance is less than 0.4. It should be stressed that the PUMA estimator is given perfect prior knowledge about the number of modes, whereas the 2-D SEMA has no such prior information. As is clear from the figure, the SEMA estimate seems to be able to separate closely spaced modes almost as well as the parametric and statistically efficient PUMA estimator, without imposing any a priori model order information, as well as yielding far better performance than the periodogram estimate. A typical result is shown in Figures 9 and 10, where the peaks are separated by $0.5/N_1$. It clearly shows how SEMA manages to separate the two peaks, whereas the periodogram only shows one peak.

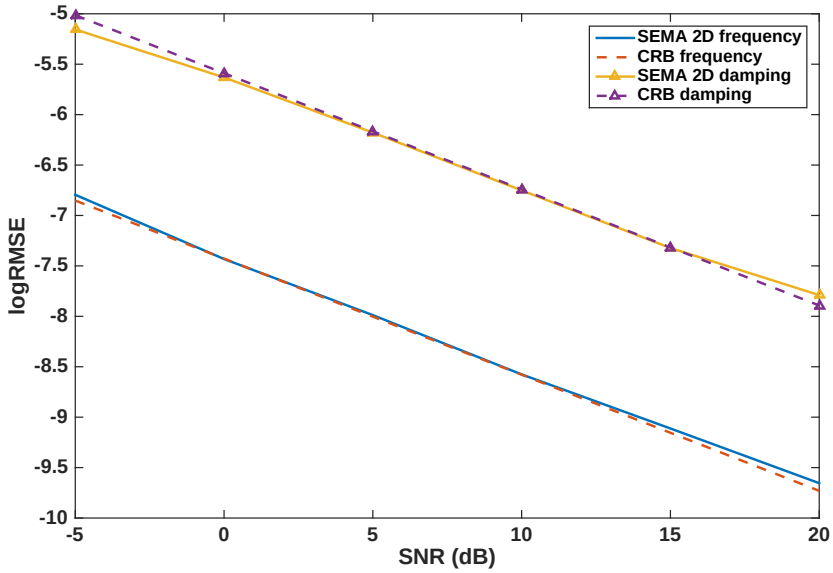

Figure 11: The RMSE for the frequency and damping estimates using SEMA for a non-unifomly sampled signal.

In the next example, we investigate how well SEMA works on non-uniformly sampled data. We made 100 Monte-Carlo simulations on a simulated NMR signal containing 33×31 sample points, where the second dimension was sampled uniformly and the first dimension was sampled in a non-uniformly manner, mimicking a typical high-dimensional NMR experiment. The frequency was randomly selected and separated at least $3/N_1$ from each other, whereas the β parameters were set to $\beta_1 = (0.01, 0.02)$ and $\beta_2 = (0.04, 0.03)$ throughout the simulation. Again, each mode was normalized in amplitude. In each dimension, 100 frequency grid points were used, and SEMA was allowed one iteration. Since PUMA does not work with non-uniformly sampled data, we instead applied an NLS search for the frequency and damping parameters in the mode estimation stage (Algorithm 2). Figure 11 shows the result where the frequency and damping parameters RMSE are shown together with the corresponding CRB.

Finally, to also illustrate the performance for higher dimensional data, we examine a 3-D data sets containing two unit amplitude damped modes at fre-

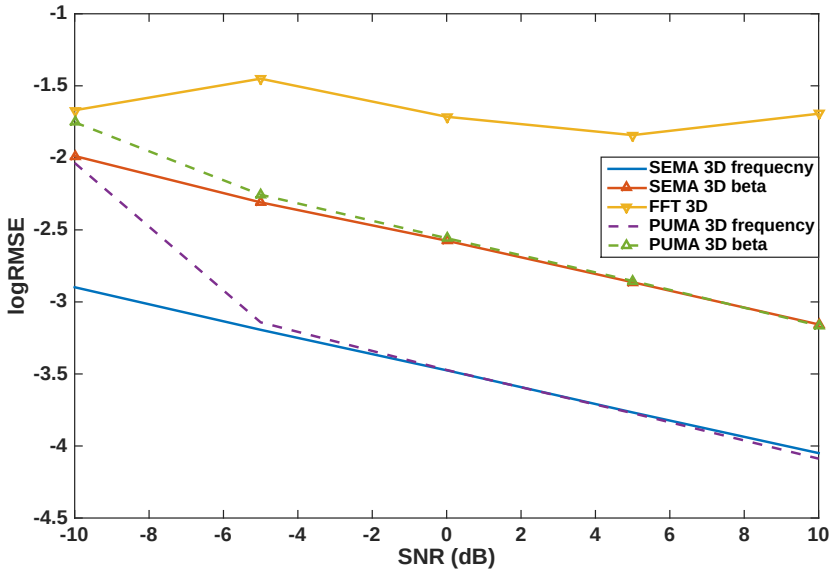

Figure 12: The log RMSE for the frequency estimates using SEMA and 3-D periodogram, together with the log RMSE for the damping estimates yielded from SEMA.

quencies drawn uniformly from $(0, 1)$, with damping parameters fixed to $\beta_1 = (0.01, 0.06)$, $\beta_2 = (0.02, 0.05)$, and $\beta_3 = (0.03, 0.04)$, and having $13 \times 13 \times 13$ samples. The modes were created so that they were separated at least by $1/N_1$ in all dimensions. The summed RMSE of the six frequency components was computed using 100 Monte-Carlo simulations for each considered SNR-level; ranging from -10 dB to 10 dB in steps of 5 dB. These estimates were compared to the N-dimensional PUMA estimator [10], and a 3-D periodogram zero-padded to 512 samples in each dimension. On our computer, it was not possible to allow for more zero-padding due to memory constraints. Furthermore, the estimates from the periodogram were selected as the two largest peaks in a cube of size $0.1 \times 0.1 \times 0.1$ around each of the true frequencies, thereby disallowing the periodogram to return any frequencies outside this area. The SEMA estimator were given an initial frequency grid of $15 \times 15 \times 15$ and allowed only one iteration. The user parameter λ was set to either 0.35 or to the mean of the all but the

ten largest peaks in the nonzero-padded periodogram, depending on which value was the smallest. The results can be seen in Figure 12, showing the log RMSE for the frequency estimates for the three methods, clearly showing the preferable performance of the SEMA algorithm as compared to the periodogram, and similar performance to the N-dimensional PUMA estimator, even though this has been allowed oracle model order knowledge. The figure also shows the log RMSE for the SEMA and PUMA damping estimates, obtained as a part of the procedure. We note that the used frequency resolution is not limiting the quality of the periodogram estimates via grid effects.

It is also worth noting that the evaluation time for the periodogram, implemented using Matlab's optimized FFT command, is only four times faster than using our proposed SEMA 3-D implementation, even though SEMA is implemented using standard Matlab code as well as estimating the damping parameters.

6 Conclusions

In this work, we have introduced a semi-parametric separable sparse model for N-dimensional damped sinusoidal signal components, forming a computationally efficient implementation exploiting the inherent structure of the resulting tensors, which allows us to treat the dictionary for each sampling dimension seperately. The proposed SEMA algorithms is found to yield highly accurate estimates of the frequency and damping coefficients of the signal modes, without imposing a priori knowledge on the number of modes present in the signal, a difficult for previously proposed parametric methods. To further reduce computational complexity, the proposed method reduces the 2-D dictionary into a sequence of 1-D dictionary learning problems, specifically exploiting the properties of the damping coefficients in a novel dictionary learning approach. The performance of the method is illustrated using 1-, 2-, and 3-D simulated data as compared to the (parametric) PUMA estimator, the Cramér-Rao lower bound, and a zero-padded periodogram estimate, as well as the corresponding non-parametric Capon and IAA based estimators, and a Lasso-based estimator, clearly illustrating the achievable performance gain.

7 Acknowledgment

The authors would like to thank the authors of [7], [10], and [40] for providing their implementation of the PUMA algorithm.

References

[1] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying Signals," in *39th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Florence, Italy, May 4-9 2014.

[2] S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, "High Resolution Sparse Estimation of Exponentially Decaying Two-dimensionalimensional Signals," in *22nd European Signal Processing Conference*, Lisbon, Portugal, 2014.

[3] J. Liu and X. Liu, "An Eigenvector-Based Approach for Multidimensional Frequency Estimation With Improved Identifiability," vol. 54, pp. 4543–4556, 2006.

[4] Y. Hua, "Estimating Two-Dimensional Frequencies by Matrix Enhancement and Matrix Pencil," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2267–2280, September 1992.

[5] J. Sacchini, W. Steedly, and R. Moses, "Two-dimensional Prony modeling and parameter estimation," *IEEE Transactions on Signal Processing*, vol. 41, no. 11, pp. 3127–3137, November 1993.

[6] S. Rouquette and M. Najim, "Estimation of Frequencies and Damping Factors by Two-Dimensional ESPRIT Type Methods," *IEEE Transactions on Signal Processing*, vol. 49, no. 49, pp. 237–245, January 2001.

[7] F. K. W. Chan, H. C. So, and W. Sun, "Subspace approach for two-dimensional parameter estimation of multiple damped sinusoids," *Signal Process.*, vol. 92, pp. 2172 – 2179, 2012.

[8] M. Haardt, F. Roemer, and G. Del Galdo, "Higher-Order SVD-Based Subspace Estimation to Improve the Parameter Estimation Accuracy in Multidimensional Harmonic Retrieval Problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3198–3213, July 2008.

[9] Y. Li, J. Razavilar, and K. J. R. Liu, "A High-Resolution Technique for Multidimensional NMR Spectroscopy," vol. 45, no. 1, pp. 78–86, 1998.

[10] W. Sun and H. C. So, "Accurate and Computationally Efficient Tensor-Based Subspace Approach for Multidimensional Harmonic Retrieval," vol. 60, no. 10, pp. 5077–5088, Oct. 2012.

[11] S. Sahnoun, E. H. Djermoune, and D. Brie, "Sparse Modal Estimation of 2-D NMR Signals," in *38th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Vancouver, Canada, May 26-31 2013.

[12] P. Stoica and T. Sundin, "Nonparametric NMR Spectroscopy," *J. Magn. Reson.*, vol. 152, pp. 57–69, 2001.

[13] E. Gudmundson, P. Stoica, J. Li, A. Jakobsson, M. D. Rowe, J. A. S. Smith, and J. Ling, "Spectral Estimation of Irregularly Sampled Exponentially Decaying Signals with Applications to RF Spectroscopy," *J. Magn. Reson.*, vol. 203, no. 1, pp. 167–176, March 2010.

[14] F. J. Frigo, J. A. Heinen, J. A. Hopkins, T. Niendorf, and B. J. Mock, "Using Peak-Enhanced 2D-Capon Analysis with Single Voxel Proton Magnetic Resonance Spectroscopy to Estimate T2* for Metabolites," in *Proc. of ISMRM*, 2004, vol. 12, p. 2437.

[15] G. O. Glentis and A. Jakobsson, "Computationally efficient damped Capon and APES spectral estimation," in *21st European Signal Processing Conference*, Marrakech, Morocco, Sept. 9-13 2013.

[16] I. F. Gorodnitsky and B. D. Rao, "Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm," vol. 45, no. 3, pp. 600–616, March 1997.

[17] 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.

[18] P. Stoica, Jian Li, and Hao He, "Spectral Analysis of Nonuniformly Sampled Data: A New Approach Versus the Periodogram," vol. 57, no. 3, pp. 843–858, March 2009.

[19] T. Yardibi, J. Li, P. Stoica, M. Xue, and A. B. Baggeroer, "Source Localization and Sensing: A Nonparametric Iterative Approach Based on Weighted Least Squares," vol. 46, no. 1, pp. 425–443, January 2010.

[20] X. Tan, W. Roberts, J. Li, and P. Stoica, "Sparse Learning via Iterative Minimization With Application to MIMO Radar Imaging," vol. 59, no. 3, pp. 1088–1101, March 2011.

[21] P. Stoica, P. Babu, and J. Li, "SPICE : a novel covariance-based sparse estimation method for array processing," vol. 59, no. 2, pp. 629 –638, Feb. 2011.

[22] P. Stoica and P. Babu, "SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation," *Signal Processing*, vol. 92, no. 7, pp. 1580–1590, July 2012.

[23] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," vol. 59, no. 5, pp. 2182 –2195, May 2011.

[24] P. Stoica and P. Babu, "Sparse Estimation of Spectral Lines: Grid Selection Problems and Their Solutions," vol. 60, no. 2, pp. 962–967, Feb. 2012.

[25] S. I. Adalbjörnsson and A. Jakobsson, "Sparse Estimation of Spectroscopic Signals," in *19th European Signal Processing Conference, EUSIPCO 2011*, Barcelona, Spain, 2011.

[26] S. Sahnoun, E. Djermoune, C. Soussen, and D. Brie, "Sparse multidimensional modal analysis using a multigrid dictionary refinement," *EURASIP J. Applied SP*, vol. 60, pp. 1–10, 2012.

[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] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

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

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

[31] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," vol. 14, no. 10, pp. 707–710, Oct. 2007.

[32] 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.

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

[34] 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.

[35] R. L. Bishop and S. I. Goldberg, *Tensor Analysis on Manifolds*, Dover Publications, Inc., New York, 1968.

[36] R. A. Horn and C. A. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, England, 1991.

[37] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, 4^{th} edition, 2013.

[38] C. Tadonki and B. Philippe, "Parallel Numerical Linear Algebra," chapter Parallel Multiplication of a Vector by a Kronecker Product of Matrices, pp. 71–89. Nova Science Publishers, Inc., Commack, NY, USA, 2001.

[39] J. Li and P. Stoica, "Efficient Mixed-Spectrum Estimation with Applications to Target Feature Extraction," vol. 44, no. 2, pp. 281–295, February 1996.

[40] H. C. So, F. Chanand W. H. Lau, and C. Chan, "An Efficient Approach for Two-Dimensional Parameter Estimation of a Single-Tone," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 1999–2009, April 2010.

[41] E. Gudmundson, Jun Ling, P. Stoica, Jian Li, and A. Jakobsson, "Spectral Estimation of Damped Sinusoids in the Case of Irregularly Sampled Data," in *Proceedings of the 9th International Symposium on Signals, Circuits and Systems (ISSCS 2009)*, Iasi, Romania, July 9-10 2009.

[42] A. Månsson, A. Jakobsson, and M. Akke, "Multidimensional Cramer-Rao Lower Bound for Non-Uniformly Sampled NMR Signals," in *22nd European Signal Processing Conference*, Lisbon, Sept. 1-5 2014.

C

Paper C

Sparse Semi-parametric Estimation of Harmonic Chirp Signals

Johan Swärd, Johan Brynolfsson, Andreas Jakobsson, and Maria Hansson-Sandsten

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

In this work, we present a method for estimating the parameters detailing an unknown number of linear, possibly harmonically related, chirp signals, using an iterative sparse reconstruction framework. The proposed method is initiated by a re-weighted group-sparsity approach, followed by an iterative relaxation-based refining step, to allow for high resolution estimates. Numerical simulations illustrate the achievable performance, offering a notable improvement as compared to other recent approaches. The resulting estimates are found to be statistically efficient, achieving the corresponding Cramér-Rao lower bound.

Key words: Harmonic chirps, multi-component, Block sparsity, Lasso, Cramér-Rao lower bound

1 Introduction

Many forms of everyday signals, ranging from radar and biomedical signals to seismic measurements and human speech, may be well modeled as signals with instantaneous frequencies (IF) that varies slowly over time [1]. Such signals are often modeled as linear chirps, i.e., periodic signals with an IF that changes linearly with time. Given the prevalence of such signals, much effort has gone into formulating efficient estimation algorithms of the start frequency and rate of development, and then, in particular, for signals only containing a single (complex-valued) chirp. One noteworthy such method is the phase unwrapping algorithm presented by Djuric and Kay [2]; further development of this method can be found in e.g. [3]. Other methods presented for single component estimation are, for example, based on Kalman filtering [4, 5], or sample covariance matrix estimates [6]. Similarly, in [7], the authors utilized the idea of single chirp modeling in detecting non-stationary phenomena in very noisy data. Recent work has to a larger extent focused on also identifying multi-component chirp signals, such as the maximum likelihood technique presented in [8], the fractional Fourier transform method [9–11], and the multitapered synchrosqueezed transform [12]. Others have used some Fourier based time-frequency estimate, e.g, the Wigner-Ville distribution, the reassigned spectrogram, or a Gabor dictionary, as a rough initial estimate, which may then be refined using image processing techniques to fit a linear chirp model [13–15]. The latter methods seem to render good estimates, although they typically require rather large data sets to do so. The reassignment method will yield perfect localization of the IF for each chirp component, given enough noise-free observations. Regrettably, it is quite sensitive to noise corruption [16]. Furthermore, in [17] a Lasso-based framework to estimate linear chirp signals was proposed that showed more robustness to noise as well as allowed for estimating an unknown number of unrelated linear chirps. Also, some efforts have been made to use a compressed sensing approach [18], where a dictionary containing a small number of chirps is formed and the final estimates are found using an FFT-based algorithm. The size of the dictionary was limited to the signal length, thus impairing the estimation abilities. Also, the method did not allow for any modeling of additional signal structure.

Often, the nonparametric methods have the advantage of computational efficiency, but generally also suffer from the poor resolution and high variance as is inherent to the spectrogram (see, e.g. [19]). The parametric methods on the other hand often have good performance and resolution, but generally require *a*

priori knowledge of the number of components in the signal. Furthermore, it is not uncommon that one also needs to have good initial estimates to be able to use such methods; otherwise, the algorithm might suffer from convergence problems.

Many naturally occurring signals show a harmonic structure, i.e., a fundamental frequency with a number of overtones that are integer multiples of the fundamental frequency. For such signals there are many proposed algorithms (see e.g. [20–22]). However, in many signals the signal structure suffers from inharmonicities, such that the spectral components are not exactly harmonic. Recently, two works have also examined extensions to the case of a single source harmonic chirp, containing a set of harmonically related chirps signals [23, 24]. These signals have lately started to attract interest due to their ability to model non-stationary harmonical signals, such as many forms of audio signals [23]. In these works, both a nonlinear least squares (NLS) [23] and a maximum likelihood solution [24] were examined. In this work, we extend upon and generalize the findings in [17], to account for an harmonic structure, where both the number of sources and the number of harmonic overtones for each source are unknowns, as well as allow for the case when some of the harmonics are missing. The algorithm requires very few samples to get an accurate estimate of the parameters, which allows the method to also model short segments of even highly non-linear chirp signals as being piecewise linear over each of the segments, yielding a quite accurate *local* signal representation. Furthermore, as long as the sampling times are known, the algorithm will also handle irregularly sampled data. Typically, most existing works rely on available *a priori* knowledge of the order of the models, although such details are in general unavailable, and are notoriously hard to estimate [21]. Recently, some efforts on alleviating these assumptions have been made for purely harmonic signals [22], wherein a block-sparse framework is utilized to form the estimates. The here presented work extends on these efforts, also allowing for inharmonic sources, using the ideas introduced in [23]. We demonstrate the performance of the proposed method using both real and simulated data, and compare the results with the corresponding Cramér-Rao lower bound (CRLB), which is also presented, as well as with competing algorithms. To improve on the computational complexity, we present an efficient implementation, utilizing the alternating direction method of multipliers (ADMM) framework (see, e.g. [25]).

In this paper, scalars will be denoted with lower case symbols, e.g. x, whereas vectors will be denoted with bold lower case, $\mathbf{x}$. Matrices will be denoted with bold upper case letter, $\mathbf{X}$. Furthermore, $(\cdot)^T$, $(\cdot)^H$, $\mathfrak{Re}$, and $\mathfrak{Im}$ will be used to

denote the transpose, the conjugate transpose, the real part, and the imaginary part, respectively.

The paper is structured as follows: In the next section, we introduce the signal model for harmonic chirp signals. Then, in section III, we derive the proposed algorithms and present some heuristics for setting the user parameters. In section IV, we present efficient implementations of the algorithms, whereas in section V, we illustrate the available performance of the introduced methods using numerical results. Finally, in section VI, we conclude upon our work.

2 Signal model

Consider

$$y(t) = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} \alpha_{k,l} e^{i2\pi\ell\varphi_k(t)} + e(t), \quad t = t_0, \ldots, t_{N-1} \tag{1}$$

where K and L_k denote the unknown number of fundamental chirps and the number of unknown harmonics for the kth component, respectively, whereas N denotes the number of available samples, t the sample times, which may be irregular, α_k the complex valued amplitude, $\varphi_k(t)$ the time dependent frequency function, and $e(t)$ an additive noise term, here assumed to be white, circularly symmetric, and Gaussian distributed. Furthermore, the chirp signal is assumed to be reasonable linear, at least under short time intervals, which allows $\varphi_k(t)$ to be modeled as

$$\varphi_k(t) = f_k^0 t + \frac{r_k}{2} t^2 \tag{2}$$

yielding the IF function

$$\varphi_k'(t) = f_k^0 + r_k t \tag{3}$$

where f_k^0 and r_k denote the starting frequency and the frequency rate, i.e., the frequency slope of the chirp, for chirp component k, respectively. The considered problem consists of estimating K, L_k, f_k^0, and r_k, as well as, in the process, also the phase, $\varphi_{k,\ell} \triangleq \angle\alpha_{k,\ell}$, and the magnitude, $a_{k,\ell} \triangleq |\alpha_{k,\ell}|$. Finally, we assume that $\min\{\ell\varphi_k'(t)\} \geq 0$ and $\max\{\ell\varphi_k'(t)\} \leq F_s, \forall(k,\ell)$, where F_s denotes the sampling frequency, in order to ensure that all frequencies in the signal are observable, i.e., fulfilling the Nyquist-Shannon sampling theorem.

3 Algorithm

In order to form an efficient algorithm for estimating the unknown parameters in (1), one may rewrite (1) as

$$\mathbf{y} = \tilde{\mathbf{D}}\tilde{\mathbf{a}} + \mathbf{e} \tag{4}$$

where

$$\mathbf{y} = \left[\begin{array}{ccc} y(t_0) & \dots & y(t_{N-1}) \end{array} \right]^T \tag{5}$$

$$\tilde{\mathbf{a}} = \left[\begin{array}{ccccc} \alpha_{1,1} & \dots & \alpha_{1,L_1} & \dots & \alpha_{K,L_K} \end{array} \right]^T \tag{6}$$

$$\tilde{\mathbf{D}} = \left[\begin{array}{ccccc} \mathbf{d}_{1,1} & \dots & \mathbf{d}_{1,L_1} & \dots & \mathbf{d}_{K,L_K} \end{array} \right] \tag{7}$$

$$\mathbf{d}_{k,\ell} = \left[\begin{array}{ccc} e^{i2\pi\ell\varphi_k(t_0)} & \dots & e^{i2\pi\ell\varphi_k(t_{N-1})} \end{array} \right]^T \tag{8}$$

and where $\mathbf{e}$ is formed in the same manner as $\mathbf{y}$. To allow for an unknown number of components, we expand the signal representation in (4) into one formed using a large dictionary containing $P \gg \sum_{k=1}^{K} L_k$ candidate chirps, such that

$$\mathbf{y} \approx \mathbf{D}\mathbf{a} \tag{9}$$

where $\mathbf{D}$ is an $N \times P$ dictionary matrix, and $\mathbf{a}$ the corresponding amplitudes, which thus mostly contains zeros, but with (at least) $\sum_{k=1}^{K} L_k$ non-zero elements. It is here assumed that P is selected sufficiently large so that the corresponding dictionary elements are close to the location of the true components and also spans the the relevant parameter space, e.g. ranging from 0 to F_s for the starting frequency parameter (see also [26, 27] for a related discussion). Solving (9) using ordinary least squares, if feasible, would yield a non-sparse solution, i.e., most of the indexes of $\mathbf{a}$ would be non-zero. Instead, we here impose the harmonic structure upon the solution by forcing it to choose between the different candidate chirp groups, while allowing for one or many of the overtones to be missing. To impose this structure, we form the minimization

$$\underset{\mathbf{x}}{\text{minimize}} \quad ||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 + \lambda||\mathbf{x}||_1 + \gamma \sum_{q=1}^{Q} ||\mathbf{x}[q]||_2 \tag{10}$$

where $\mathbf{x}[q]$ selects all elements in $\mathbf{x}$ corresponding to block q in $\mathbf{D}$, and Q denotes the number of blocks considered, where each block contains a fundamental chirp

and its overtones, i.e., for block q, $\mathbf{x}[q]$ denotes the elements of $\mathbf{x}$ that corresponds to

$$\left[\begin{array}{ccc} \mathbf{d}_{q,1} & \ldots & \mathbf{d}_{q,L_q} \end{array} \right] \tag{11}$$

in the dictionary. The first term in (10) measures the distance between the signal and the model, the second term enforces an overall sparsity between the available chirp candidates and thus limits the number of chirps that may be part of the solution. The third term in (10) acts as a sparsity enhancer for the number of harmonically related chirp groups that are allowed in the solution, thus promoting a solution that has fewer number of activated groups. Together, the two last terms in (10) promotes a solution that has few harmonically related chirp groups, and also allows for chirps within a group to be sparse. This optimization problem is convex as it is a sum of convex functions, and the solution may thus be found using standard interior-point methods, such as, e.g., SeDumi [28] and SDPT3 [29]. Furthermore, γ and λ are tuning parameters controlling the sparsity of the groups and the sparsity within the groups, respectively. It is worth noting that if setting $\gamma = 0$, one solves the problem of finding unrelated chirps in the signal. Even though P is finely spaced, the quality of the solution obtained from (10) will depend on the grid structure of $\mathbf{D}$, i.e., if the true components are not contained in the dictionary, the components that are the closest to the true chirps will be activated, ensuring that the corresponding indices in $\mathbf{x}$ will be non-zero. Therefore, the solution attained from (10) will be biased in accordance to the chosen grid structure of $\mathbf{D}$. To avoid this bias, the estimation procedure involves an additional step consisting of a nonlinear least squares (NLS) search to further increase the resolution. In order to do so, let the residual from (10) be formed as

$$\mathbf{r} = \mathbf{y} - \mathbf{D}\mathbf{x} \tag{12}$$

Then, each harmonic chirp component may be iteratively updated by first adding one component to the residual formed in (12), conducting a NLS search for the parameter estimates, initiated using the estimates found from (10), and then remove the found component using (12). When all components have been updated in this way, one may continue updating the residual with the newly refined estimates. The final estimates are found by iterating the entire refinement procedure a few times.

In the above algorithm, the user has to select a value for the parameters γ and λ. Of these, the value of γ penalizes the number of harmonic chirps allowed

in the solution, meanwhile the value of λ penalizes the overall number of chirps, thus allowing for sparsity within each harmonic chirp component. The values of γ and λ are commonly chosen through cross-validation [30], or by some data dependent heuristics. In the case of $\gamma = 0$, we herein suggest selecting

$$\lambda = \frac{||\mathbf{y}||_2^2}{2N} \tag{13}$$

which has empirically been shown to provide a reliable choice of λ, for the here considered data lengths. When both tuning parameters are active, the problem of setting good values becomes more complicated, since the two penalties interact. We have empirically found that, as long as λ is reasonably small, one may use (13) as a rule of thumb for also setting γ. To further increase the robustness to the choice of γ and λ, and to further enhance the sparsity, we propose a re-weighted approach, based on the technique introduced in [31]. In this approach, one solves the minimization iteratively, where, at every iteration, two weight matrices, $\mathbf{W}$ and $\mathbf{V}$, with weights $w_1, \ldots, w_P$ and $v_1, \ldots, v_Q$ on the diagonals and zeros elsewhere, are used. The diagonal elements in $\mathbf{W}$ and $\mathbf{V}$ are updated as

$$w_p^{(b)} = \frac{1}{|x_p^{(b-1)}| + \varepsilon}, \qquad p = 1, \ldots, P \tag{14}$$

$$v_q^{(b)} = \left(\frac{1}{||\mathbf{x}^{(b-1)}[q]||_2^2 + \varepsilon} \right)^{1/2}, \qquad q = 1, \ldots, Q \tag{15}$$

where the superscript b denotes the iteration number, and $\varepsilon > 0$ is a small offset parameter, which prevents the solution from diverging. At each iteration, one thus solves

$$\underset{\mathbf{x}}{\text{minimize}} \quad ||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2 + \lambda ||\mathbf{W}^{(b)}\mathbf{x}||_1 + \gamma \sum_{q=1}^{Q} v_q^{(b)} ||\mathbf{x}[q]||_2 \tag{16}$$

The resulting algorithm is outlined in Algorithm 1, where $\mathbf{D}(\cdot, k)$ and $\mathbf{x}(k)$ denote the kth column and the kth index of the matrix $\mathbf{D}$ and the vector $\mathbf{x}$, respectively. Furthermore, let $\hat{K}$ denote the number of non-zero elements in the solution from (10), and let the corresponding indices in $\mathbf{x}$ make up the index set $\mathbf{I}_{\hat{K}}$. Clearly, one must select an appropriate stopping criteria for the second loop in Algorithm 1. This may be done in various ways, such as when the parameter estimates does no

Algorithm 1 The HSMUCHES algorithm

Initiate $w_p = 1$, for $p = 1, \ldots, P$, and $v_q = 1$, for $q = 1, \ldots, Q$
for $b = 1, \ldots$ **do**
  Solve (16)
  Update (14) and (15)
end for
Compute (12)
for j = 1,... **do**
  for $k = 1, \ldots, \hat{K}$ **do**
    $\mathbf{z} = \mathbf{r} + \mathbf{D}(\cdot, \mathbf{I}_{\hat{K}}(k))^{(j)}\mathbf{x}(\mathbf{I}_{\hat{K}}(k))^{(j)}$
    Using $\mathbf{z}$, update $\mathbf{D}(\cdot, \mathbf{I}_{\hat{K}}(k))^{(j)}$ and $\mathbf{x}(\mathbf{I}_{\hat{K}}(k))^{(j)}$ via NLS
    Subtract the refined estimates from $\mathbf{z}$
  end for
end for

longer improve significantly in each iteration, or by setting a maximum number of iterations. Empirically, we found that 10 iterations where enough for convergence and through out this work, we used this as stopping criteria. It should be noted that the re-weighted approach introduces the tuning parameter ε. In this paper, we have set ε to be

$$\varepsilon = \frac{N}{||\mathbf{y}||_2^2} \tag{17}$$

which is in accordance with the discussion in [31], and which has been empirically shown to yield reliable estimates.

It should be noted that if $\gamma = 0$, the estimator does not assume any harmonic structure, and therefore constitutes solely a multi-chirp estimator; we term this the Sparse MUlticomponent Chirp EStimator (SMUCHES). In the case $\gamma > 0$, the estimator also allows for the possibility of harmonic chirp components; we term this the Harmonic Sparse MUlticomponent Chirp EStimator (HSMUCHES).

4 Efficient implementation

We proceed to examine efficient implementations of the proposed estimators using the ADMM framework. The discussion here is focused on the HSMUCHES

estimator, although the implementation also works for the SMUCHES algorithm by simply setting $\gamma = 0$. In general, an ADMM solves problems in the form

$$\begin{aligned} &\underset{\mathbf{x},\mathbf{z}}{\text{minimize}} \quad f(\mathbf{x}) + g(\mathbf{z}) \\ &\text{subject to} \quad \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = c \end{aligned} \tag{18}$$

In our case, $\mathbf{A} = \mathbf{I}$, $\mathbf{B} = -\mathbf{I}$, $c = 0$, $f(\mathbf{x}) = ||\mathbf{y} - \mathbf{D}\mathbf{x}||_2^2$, and $g(\mathbf{z}) = \lambda||\mathbf{z}||_1 + \gamma\sum_{q=1}^{Q}||\mathbf{z}[q]||_2$, where $\mathbf{I}$ denotes the identity matrix of size $N \times P$. The augmented Lagrangian for this minimization is formed as

$$L_\rho(\mathbf{x}, \mathbf{z}, \mathbf{u}) = f(\mathbf{x}) + g(\mathbf{z}) + \frac{\rho}{2}||\mathbf{x} - \mathbf{z} + \mathbf{u}||_2^2 \tag{19}$$

where $\mathbf{u}$ is the scaled dual variable, and ρ is the penalty parameter, penalizing the distance between $\mathbf{z}$ and $\mathbf{x}$. The ADMM finds the solution to (16) by iteratively solving (19) for each variable separately. The steps in the ADMM are

$$\mathbf{x}^{(k+1)} = \underset{\mathbf{x}}{\text{argmin}} \left(f(\mathbf{x}) + \frac{\rho}{2}||\mathbf{x} - \mathbf{z}^{(k)} + \mathbf{u}^{(k)}||_2^2 \right) \tag{20}$$

$$\mathbf{z}^{(k+1)} = \underset{\mathbf{z}}{\text{argmin}} \left(g(\mathbf{z}) + \frac{\rho}{2}||\mathbf{x}^{(k+1)} - \mathbf{z} + \mathbf{u}^{(k)}||_2^2 \right) \tag{21}$$

$$\mathbf{u}^{(k+1)} = \mathbf{x}^{(k+1)} - \mathbf{z}^{(k+1)} + \mathbf{u}^{(k)} \tag{22}$$

To find the solution to (20), one differentiates (19) with respect to $\mathbf{x}$ and put it equal to zero, yielding

$$\mathbf{x}^{(k+1)} = \left(\mathbf{D}^H\mathbf{D} + \rho\mathbf{I}\right)^{-1}\left(\mathbf{D}^H\mathbf{y} + \rho\left(\mathbf{z}^{(k)} - \mathbf{u}^{(k)}\right)\right) \tag{23}$$

To solve (21), one needs to take some further care as $g(\mathbf{z})$ is not differentiable at $\mathbf{z} = 0$. However, it can be shown (see e.g. [32]) that the solution to (21) is

$$\mathbf{z}^{(k+1)} = \boldsymbol{\mathcal{S}}\left(\mathbf{S}\left(\mathbf{x}^{(k+1)} + \mathbf{u}^{(k)}, \lambda/\rho\right), \gamma/\rho\right) \tag{24}$$

where $\mathbf{S}$ and $\boldsymbol{\mathcal{S}}$ are soft thresholds defined as

$$\mathbf{S}(\mathbf{x}, \varkappa) = \frac{x_j}{|x_j|}\max(|x_j| - \varkappa, 0) \tag{25}$$

$$\boldsymbol{\mathcal{S}}(\mathbf{x}, \varkappa) = \frac{\mathbf{x}[q]}{||\mathbf{x}[q]||_2}\max(||\mathbf{x}[q]||_2 - \varkappa, 0) \tag{26}$$

for $q = 1, \ldots, Q$, where $\mathbf{S}$ should be interpreted elementwise. Observing that $f(\mathbf{x})$ and $g(\mathbf{z})$ are closed, proper, and convex functions, and given $\rho > 0$, then, under some mild assumptions, if there is a solution to (16), then the algorithm will converge to this solution [33, 34]. Also, the choice of ρ will only effect the convergence rate, not whether or not the method will converge. Using this implementation, the computational complexity for SMUCHES is, for the Lasso part, about $N^3 + N^2P$. The computations in this part are dominated by (23), which only needs to be calculated once throughout the minimization. Furthermore, the computational complexity of the inverse is significantly decreased using the Woodbury matrix identity [35]. The NLS part of the proposed algorithm requires a computational complexity of about $\hat{K}NP$.

It may be noted that a dictionary similar to (7) was proposed in [18]; in this case, the dictionary was restricted to only contain N candidate chirps. As a result, the dictionary experienced low correlation between the columns, for which case the restricted isometry properties (RIP) will hold, suggesting that the signal may be recovered with high probability (see, e.g., [36]). The same result would hold for the dictionary in (7), if restricted in the same manner. However, to allow for high resolution estimates, the dictionary should, as discussed, be extended to contain many more chirp candidates, indicating that the dictionary columns will be highly correlated, thereby no longer satisfying the RIP. Fortunately, as is also shown in the next section, practical evidence indicate that even highly correlated dictionaries enjoy excellent signal recovery properties.

5 Numerical results

In order to evaluate the performance of the proposed algorithms, we examine their behavior on both real and simulated data, comparing them both to other alternative techniques, and to the CRLB (as derived in Appendix A). All the following root mean squared error (RMSE) curves are based on 1000 Monte Carlo simulations.

Initially, we examine a simulated uniformly sampled signal of length $N = 20$, consisting of two non-harmonic chirp components, as depicted in Figure 1, which is corrupted by white circularly symmetric Gaussian noise with a signal to noise ratio (SNR) of 10 dB, which is here defined as

$$\text{SNR} = 10\log_{10}\left(\frac{P_{\mathbf{y}}}{\sigma^2}\right) \tag{27}$$

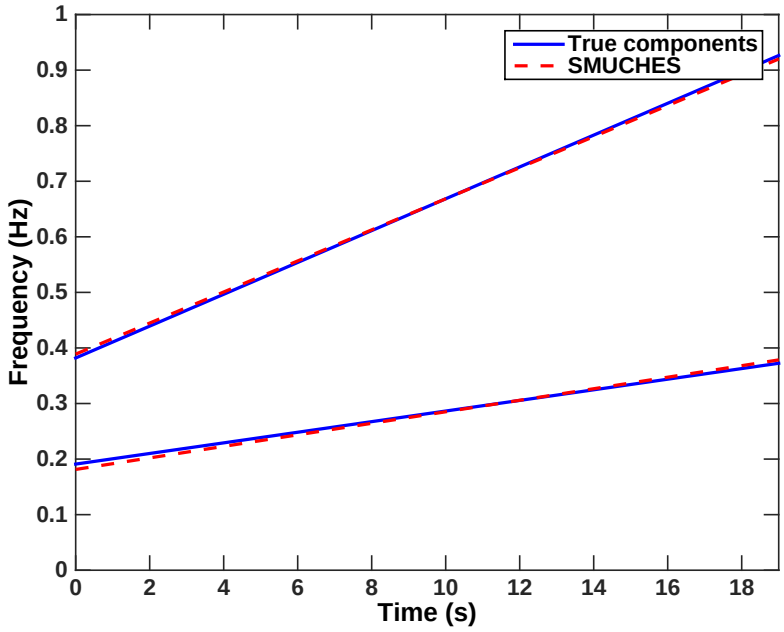

Figure 1: The figure shows the true (solid) and the estimated (dashed) IF.

where $P_{\mathbf{y}}$ denotes the power of the signal, and σ^2 the variance of the additive noise. The resulting estimates from the proposed SMUCHES method and for the reassigned spectrogram [16] are shown in Figures 1 and 2, respectively. As can be seen in Figure 2, the reassigned spectrogram finds the two chirp components, but the estimates are blurred, as well exhibiting jumps in the frequencies. On the other hand, as can be seen in Figure 1, the proposed method manages to find the chirp components without any such ambiguities.

We continue by showing how the proposed SMUCHES method may be used in tracking a non-linear chirp. In this example, we simulated an exponential chirp component defined as

$$\varphi(t) = \left(\frac{r^t - 1}{\log(r)}\right) f_0 \tag{28}$$

where f_0 and r are parameters determining the starting frequency and the exponential rate of change, respectively. The signal, containing $N = 105$ samples, was

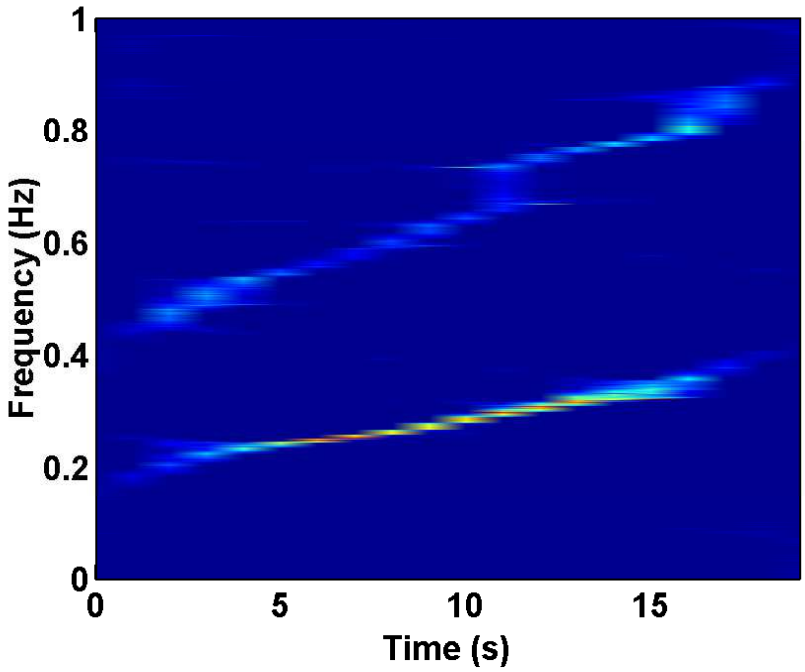

Figure 2: The figure shows the estimated time-frequency distribution of the chirp signals using the reassigned spectrogram.

divided in 7 equally sized sections, such that each segment may be reasonably well modeled as a linear chirp. The signal was corrupted by a white circularly symmetric Gaussian noise with SNR $= 20$ dB. The proposed algorithm was applied on each signal segment. The resulting chirp estimate is depicted in Figure 3, where it is clearly shown how the proposed method manages to estimate the evolving parameters of the non-linear chirp, showing that the local linear approximation is valid.

Next, we examine the estimation performance of the SMUCHES method as a function of SNR. In this example, the simulated signal contains only a single chirp component, with starting frequency $f^0 = 0.6/\pi$, frequency rate $r = 0.03/\pi$, amplitude $\alpha = 1$, and a uniformly distributed random phase $\varphi \in U[-\frac{1}{2}, \frac{1}{2})$, which was randomized for each simulation. The sample length is set to $N =$

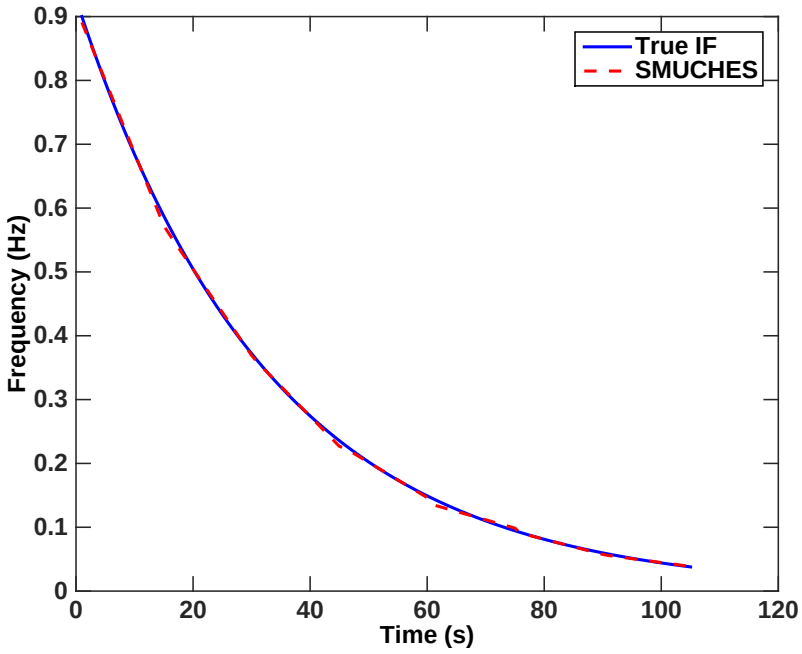

Figure 3: The estimated chirp in dashed lines as compared to the true chirp.

20. Figures 4 and 5 show the RMSE of the SMUCHES estimator, where λ and ε were selected using (13) and (17), as well as the discrete chirp Fourier transform algorithm (DCFT) [9], the algorithm presented by Djuric and Kay in [2], both being allowed oracle knowledge of the number of chirps in the signal, and the CRLB. It should be noted that the proposed methods do not assume any model order information, as they are estimating this as part of the optimization; clearly, this also implies that the method may estimate the wrong model orders. However, the proposed SMUCHES method only estimated the wrong number of components in 1 out of the 1000 simulations, and this at the SNR $=$ 5 dB level. For the other SNR levels, the order estimations were without any errors. To assert a fair comparison, the simulation where the proposed method estimated the wrong model order was removed from all methods, and is thus not included in the RMSE graphs. As is clear from Figures 4 and 5, the SMUCHES method,

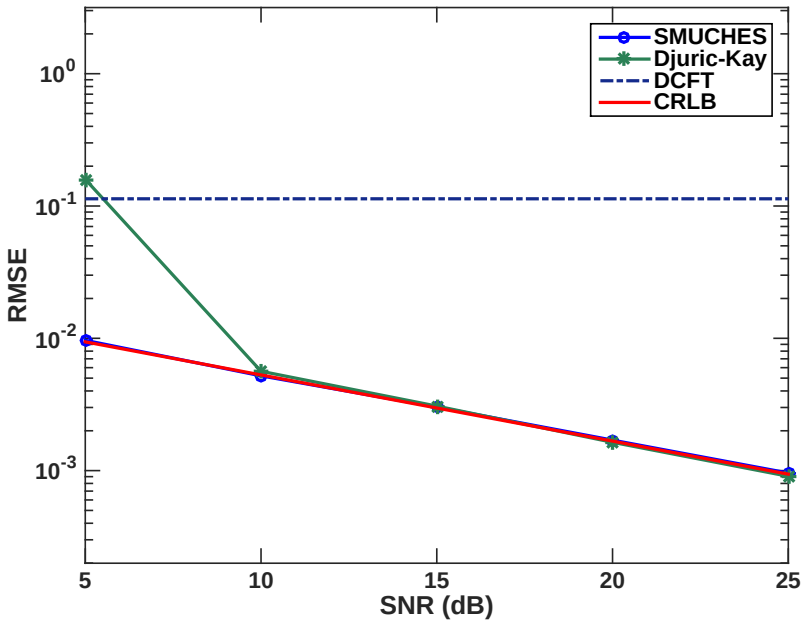

Figure 4: Performance of the proposed SMUCHES method, as compared with the Djuric-Kay method, the DCFT method, and the CRLB, when estimating the starting frequency of a single chirp.

without using any prior knowledge about the number of chirps, manages to attain the CRLB, as well as outperforming the Djuric-Kay algorithm, even though the latter has been allowed oracle model order information. Furthermore, it can be seen that the DCFT algorithm is stuck to its initial grid, which suggests why it does not manage to improve beyond a certain limit when the SNR increases. Examining the computational complexities, it was found that the Djuric-Kay and the DCFT algorithms (given oracle model orders) are notably faster to compute than the presented SMUCHES implementation, requiring on average (computed over 1000 simulations on a regular PC, for SNR $= 20$ dB) $2.3 \cdot 10^{-4}$, $5.1 \cdot 10^{-3}$, and $5.0 \cdot 10^{-1}$ seconds to execute, respectively.

We proceed by examining the performance on multicomponent chirp signals. Since the competing methods, which we previously compared with, cannot be

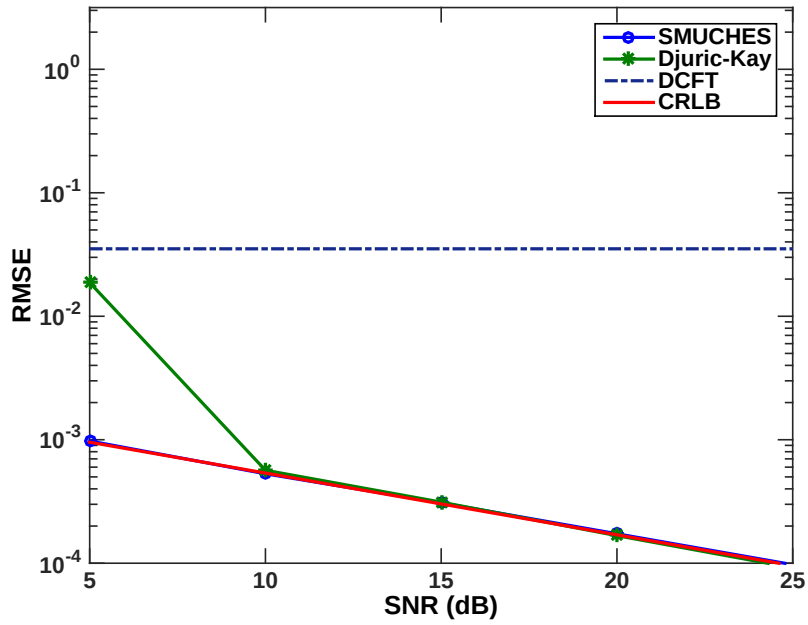

Figure 5: Performance of the proposed SMUCHES method, as compared with the the Djuric-Kay method, the DCFT method, and the CRLB, when estimating the frequency rate of a single chirp.

used on multicomponent data, we only show the results for the proposed method as compared to the corresponding CRLB. Figure 6 depicts the RMSE of the parameter estimations, as a function of SNR. The starting frequency of the chirps were $f_1^0 = 0.6/\pi$ and $f_2^0 = 1.2/\pi$, and the slope rates were $r_1 = 0.03/\pi$ and $r_2 = 0.09/\pi$. The amplitudes were set to unity and the phase were drawn as $\varphi \in U[-\frac{1}{2}, \frac{1}{2})$ at each simulation. Once again, λ and ε were chosen using (13) and (17). As one can note from Figure 6, the proposed method follows the CRLB for SNR levels greater or equal to 10 dB. In this case, the proposed method estimated the wrong model order 26 times out of the 1000 simulations, all for the SNR $= 5$ dB case, and not at all for higher SNRs. Again, these simulations were removed from the proposed method's RMSE, and the CRLB was adjusted correspondingly.

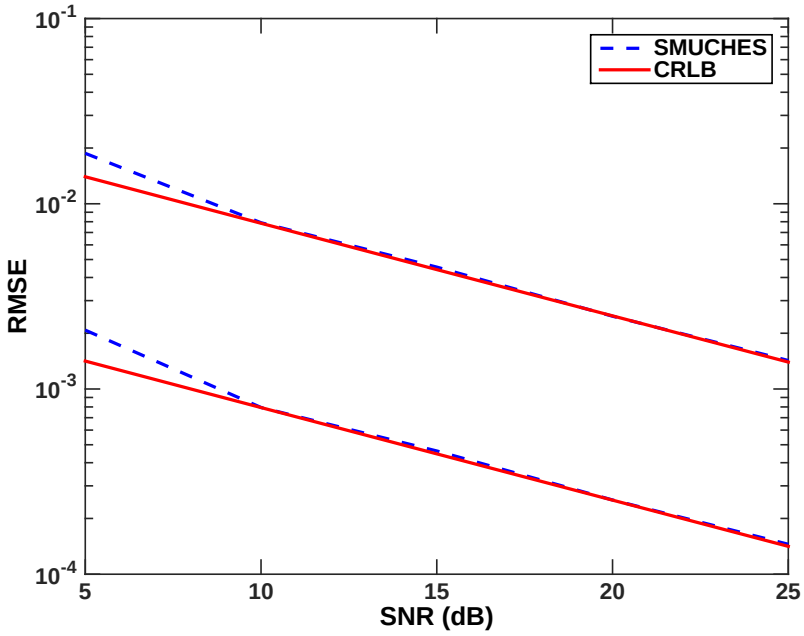

Figure 6: Performance of the proposed SMUCHES method when estimating the starting frequencies (top curves) and the frequency rates (bottom curves) of two non-crossing linear chirps, as compared to the CRLB.

Next, we examine the performance on irregularly sampled data constituting of 20 observations from a chirp signal with the same chirp components as in the previous example. The sampling times where drawn from a rectangular distribution in the range $(0, 20]$ and are depicted in Figure 7. The phase was drawn from $U[-\frac{1}{2}, \frac{1}{2})$ for each simulation. Figure 8 shows the resulting RMSE results. As for the earlier examples, for SNR greater than 5 dB, the proposed method attains the CRLB. The main difference to the uniform sampled case is that the resulting RMSE for SNR $= 5$ dB is worse. Also, the number of times the proposed method estimated the wrong model order increased to 51 times out of 1000, again, only for the SNR $= 5$ dB case. As before, for SNR greater than 5 dB, no errors in the model order estimation were made. Though slightly more sensitive to non-uniformly sampled data, it can be concluded that the proposed method is suitable

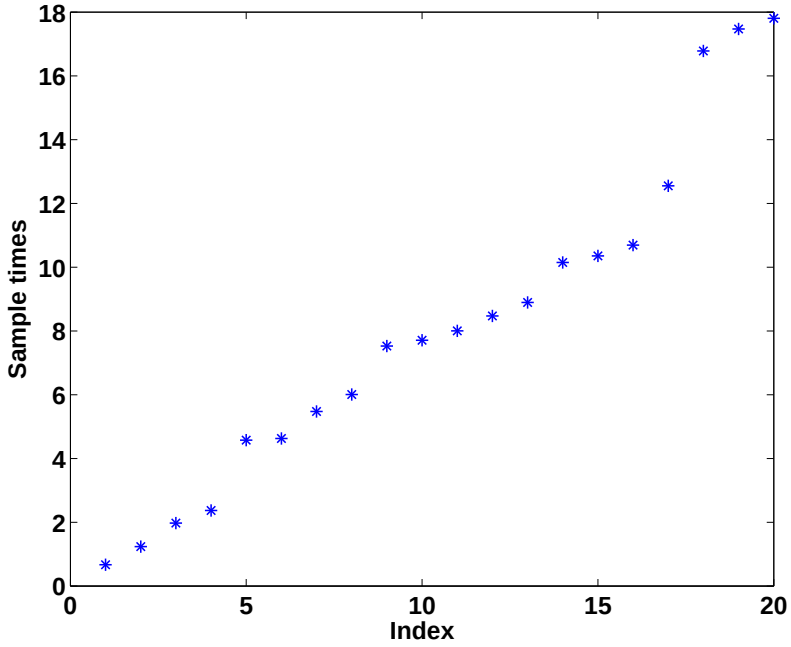

Figure 7: The distribution of the sample times.

to use also for non-uniformed sampled data.

We proceed to examine the performance on simulated harmonic data. The simulated chirp signal consist of one fundamental frequency and 3 overtones ($K = 1$ and $L_1 = 4$), each with unit amplitude and uniformly distributed random phase. The fundamental starting frequency was set to $f^0 = 0.2 * 3/\pi$ and the frequency slope to $r = -0.004 * 3/\pi$. The resulting RMSE are shown in Figures 9 and 10, as a function of SNR, when using $N = 20$ samples. The RMSEs for both the starting frequency and the frequency slope, are measured as mean value of the RMSE for each of the four components in the signal, i.e., for the fundamental frequency and the two overtones. Here, HSMUCHES estimated the wrong model order 54 times out of 1000 at SNR $= 5$ dB, 5 times out of 1000 at SNR $= 10$ dB, and made no mistakes at higher SNRs.

As SMUCHES does not take the harmonicity inherent in the signal in ac-

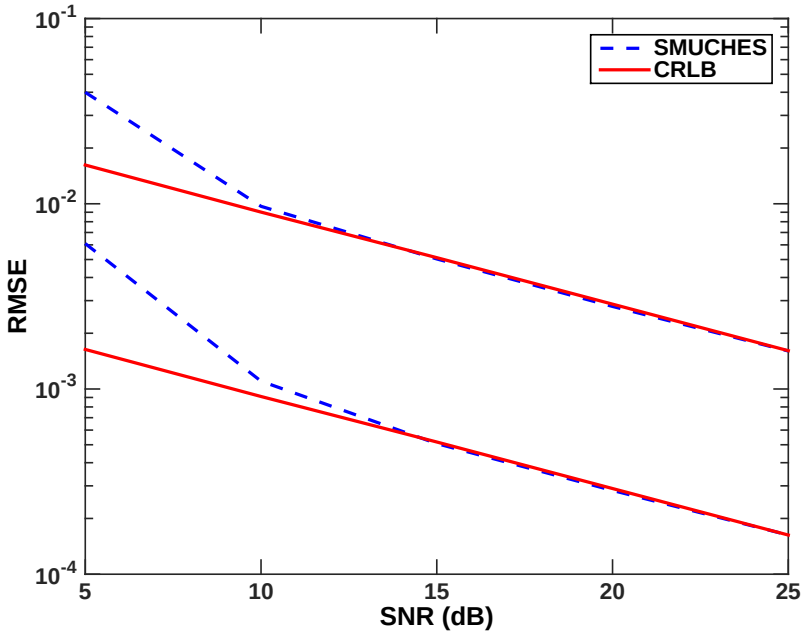

Figure 8: Performance of the proposed SMUCHES method when estimating the starting frequencies (top curves) frequency rates (bottom curves) of two non-crossing linear chirps for irregularly sampled data, as compared to the CRLB.

count, there are 18 parameters (model order, noise variance, and four parameters for each component) to estimate using only 20 samples, whereas HSMUCHES only has to estimate twelve parameters (model order, number of overtones, starting frequency, frequency slope, phase, noise variance, and four amplitudes). As a result, it can be expected that SMUCHES will make more order estimation mistakes than HSMUCHES, which was also found to be the case. Out of the 1000 simulations, SMUCHES made 906 model order errors at SNR $=$ 5 dB, 261 at SNR $=$ 10 dB, 21 at SNR $=$ 15 dB, 8 at SNR $=$ 20 dB, and 3 errors at SNR $=$ 25 dB. The tuning parameters for SMUCHES were selected using (13) and (17), and for HSMUCHES γ using (13), with $\lambda = 0$, and $\varepsilon = 10^{-4}$. Finally, we show the performance on real data, containing sounds from bats [37]. Many forms of audio sources, such as voiced speech and many forms of music,

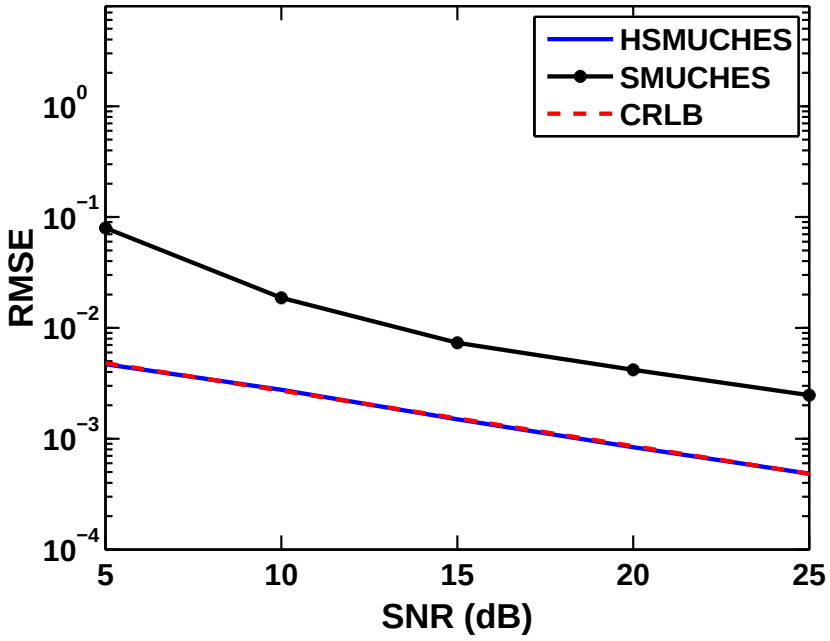

Figure 9: Performance of the proposed HSMUCHES methods applied to an harmonic chirp signal with one fundamental frequency and three overtones, as compared with the SMUCHES method and the CRLB, when estimating the starting frequencies.

may be well modelled as harmonic signals. Thus, it should be expected that the sound from a bat may contain a harmonic structure. The spectrogram of the bat signal is shown in Figure 11, suggesting that the signal contains one fundamental chirp with, at most, two overtones. Figure 12 shows the estimated harmonic structure when using HSMUCHES. Comparing the figures, it is clear that the HSMUCHES algorithm is well able to capture the changing frequencies in the harmonic signal, achieving a substantially better resolution than the spectrogram. As before, the tuning parameters for SMUCHES were selected using (13) and (17), and for HSMUCHES, γ was set to two times (13), $\lambda = 0.01$, and $\varepsilon = 10^{-4}$.

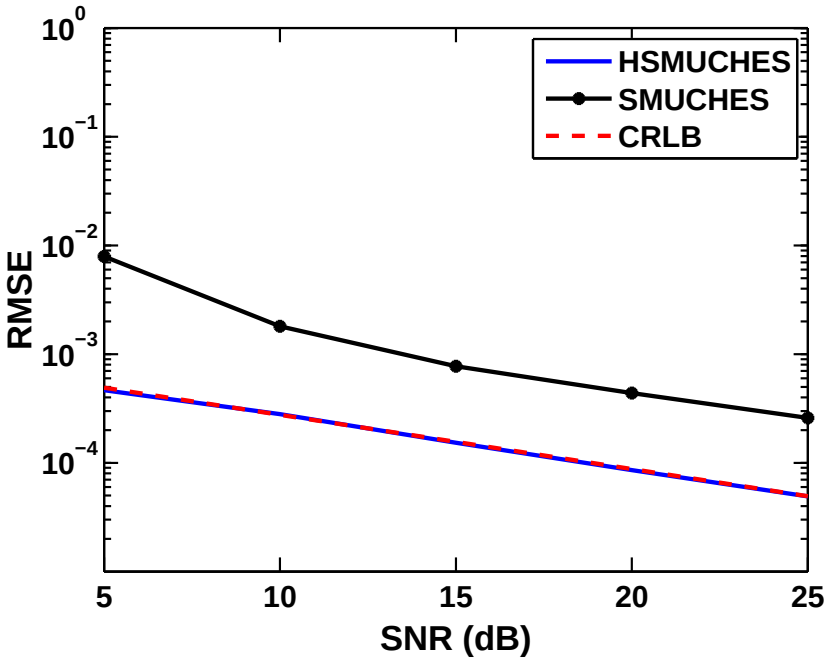

Figure 10: Performance of the proposed HSMUCHES methods applied to an harmonic chirp signal with one fundamental frequency and three overtones, as compared with the SMUCHES method and the CRLB, when estimating the frequency slopes.

6 Conclusion

In this paper, we have proposed two semi-parametric algorithms for estimating the parameters of an unknown number of chirp and harmonic chirp components in noisy data, respectively. The methods are shown to work well even for very short signals, and allow for both uniform and non-uniform sampled data. The methods are shown to attain the corresponding CRLB for both cases. Furthermore, it is shown in the paper that the methods can be also used to approximate non-linear chirps, by dividing the data into small sections, in which the non-linear chirps can be assumed to be reasonably linear. Numerical examples illustrate the preferable performance on both real and simulated signals.

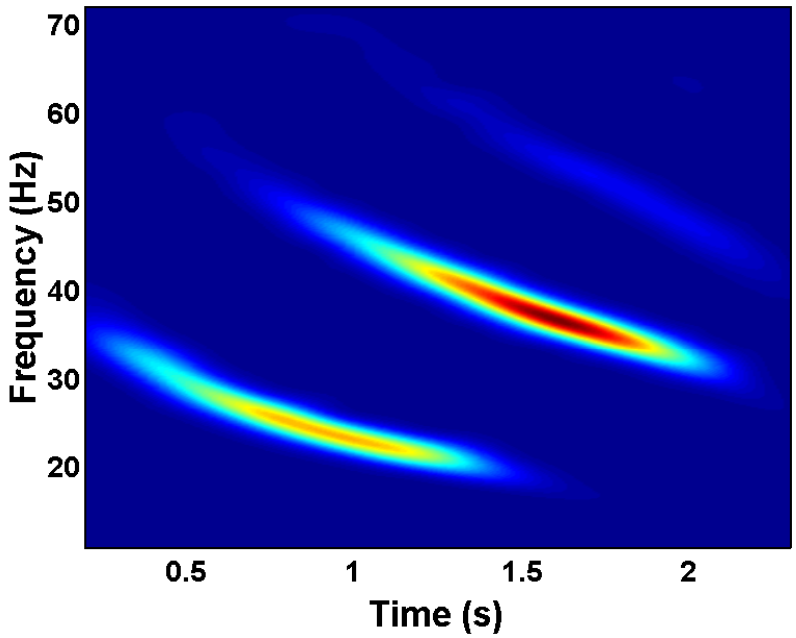

Figure 11: The figure shows the spectrogram of the bat chirp.

7 Acknowledgement

The author wishes to thank Curtis Condon, Ken White, and Al Feng of the Beckman Institute of the University of Illinois for the bat data and for permission to use it in this paper.

8 Cramér-Rao lower bound

The CRLB for a multi-component chirp signal has been derived in multiple papers, see e.g. [8]. Here, we derive the CRLB for the case of both regular frequencies and irregular sampling, as well as harmonic overtones. The Fisher information matrix (FIM) for any signal observed under complex valued additive white noise, with variance σ^2, can be set up block-wise as

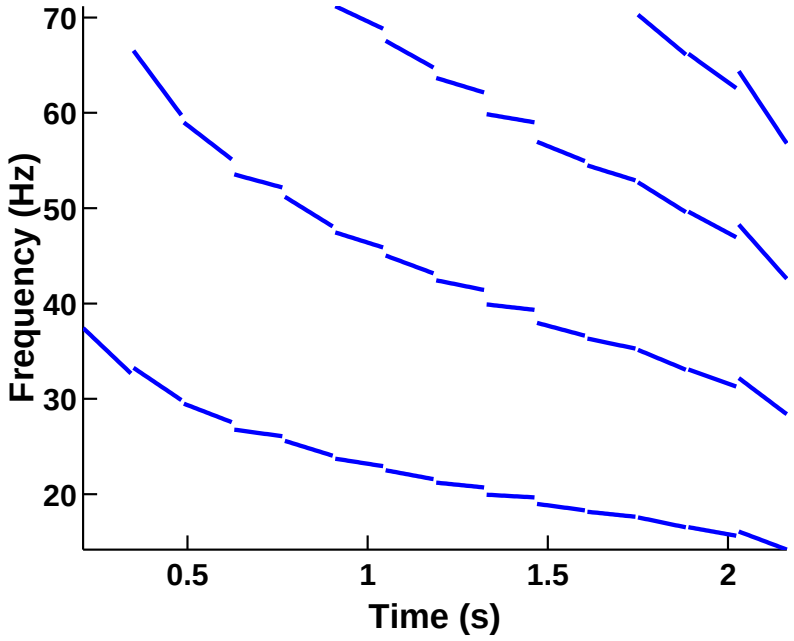

Figure 12: The figure shows the estimated time-frequency content of the bat signal using the proposed HSMUCHES algorithm.

$$J_{ij} = \frac{2}{\sigma^2} \sum_{n=0}^{N-1} \left(\frac{\partial \Re\mathfrak{e}\{y(t_n)\}}{\partial \vartheta_i} \frac{\partial \Re\mathfrak{e}\{y(t_n)\}}{\partial \vartheta_j} + \frac{\partial \Im\mathfrak{m}\{y(t_n)\}}{\partial \vartheta_i} \frac{\partial \Im\mathfrak{m}\{y(t_n)\}}{\partial \vartheta_j} \right) \tag{29}$$

where $\vartheta_k = [f_k^0\,,\ r_k\,,\ \varphi_{k,1}\,,\ \ldots\,,\ \varphi_{k,L_k}\,,\ \alpha_{k,1}\,,\ \ldots\,,\ \alpha_{k,L_k}]^T$, L_k is the number of harmonics, and $\alpha_{k,\ell}$ is the kth amplitude of the ℓth harmonic. Hence, the FIM will have $(K \times K)$ blocks, such that

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_{11} & \mathbf{J}_{12} & \cdots & \mathbf{J}_{1K} \\ \mathbf{J}_{21} & \mathbf{J}_{22} & \cdots & \mathbf{J}_{2K} \\ \cdots & \cdots & \ddots & \vdots \\ \mathbf{J}_{K1} & \mathbf{J}_{K2} & \cdots & \mathbf{J}_{KK} \end{bmatrix} \tag{30}$$

By denoting the Fisher information between the two parameters u and v as

$$\mathcal{I}(u,v) \triangleq \frac{\partial \Re\mathfrak{e}\{y(t_n)\}}{\partial u}\frac{\partial \Re\mathfrak{e}\{y(t_n)\}}{\partial v} + \frac{\partial \Im\mathfrak{m}\{y(t_n)\}}{\partial u}\frac{\partial \Im\mathfrak{m}\{y(t_n)\}}{\partial v} \tag{31}$$

each block in the FIM may be found as

$$\mathbf{J}_{kj} = \frac{2}{\sigma^2}\sum_{n=0}^{N-1}\begin{bmatrix} \mathcal{I}\left(\vartheta_k(1),\vartheta_j(1)\right) & \cdots & \mathcal{I}\left(\vartheta_k(1),\vartheta_j(M_j)\right) \\ \vdots & \ddots & \vdots \\ \mathcal{I}\left(\vartheta_k(M_k),\vartheta_j(1)\right) & \cdots & \mathcal{I}\left(\vartheta_k(M_k),\vartheta_j(M_j)\right) \end{bmatrix}$$

where $M_k = 2 + 2L_k$ denotes the number of parameters for the kth component. Defining

$$\Psi_{k,\ell}(t_n) \triangleq 2\pi\left(\ell\left(f_k^0 t_n + \frac{r_k}{t_n}t_n^2\right) + \varphi_k\right) \tag{32}$$

$$\Delta\Psi_{k,\ell,j,m}(t_n) \triangleq \Psi_{k,\ell}(t_n) - \Psi_{j,m}(t_n) \tag{33}$$

and each pairwise Fisher information is found as

$$\begin{aligned}
\mathcal{I}(f_k,f_j^0) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j}\alpha_{k,\ell}\alpha_{j,m}4\pi^2\ell m t_n^2\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(f_k^0,r_j) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j}\alpha_{k,\ell}\alpha_{j,m}2\pi^2\ell m t_n^3\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(f_k^0,\varphi_{j,m}) &= \sum_{\ell=1}^{L_k}\alpha_{k,\ell}\alpha_{j,m}4\pi^2\ell t_n\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(f_k^0,\alpha_{j,m}) &= \sum_{\ell=1}^{L_k}-\alpha_k 2\pi\ell t_n\sin\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(r_k,f_j^0) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j}\alpha_{k,\ell}\alpha_{j,m}2\pi^2\ell m t_n^3\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(r_k,r_j) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j}\alpha_{k,\ell}\alpha_{j,m}\pi^2\ell m t_n^4\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(r_k,\varphi_{j,m}) &= \sum_{\ell=1}^{L_k}\alpha_{k,\ell}\alpha_{j,m}2\pi^2\ell t_n^2\cos\Delta\Psi_{k,\ell,j,m}(t_n)\\
\mathcal{I}(r_k,\alpha_{j,m}) &= \sum_{\ell=1}^{L_k}-\alpha_{k,l}\pi\ell t_n^2\sin\Delta\Psi_{k,\ell,j,m}(t_n)
\end{aligned}$$

$$
\begin{aligned}
\mathcal{I}(\varphi_k, f_j^0) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j} \alpha_{k,\ell}\alpha_{j,m} 4\pi^2 m t_n \cos\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\varphi_k, r_j) &= \sum_{\ell=1}^{L_k}\sum_{m=1}^{L_j} \alpha_{k,\ell}\alpha_{j,m} 2\pi^2 m t_n^2 \cos\Delta\Psi_{k,j,\ell,m}(t_n) \\
\mathcal{I}(\varphi_{k,\ell}, \varphi_{j,m}) &= \alpha_{k,\ell}\alpha_{j,m} 4\pi^2 \cos\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\varphi_k, \alpha_{j,m}) &= \sum_{\ell=1}^{L_k} -\alpha_{k,\ell} 2\pi \sin\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\alpha_{k,\ell}, f_j^0) &= \sum_{m=1}^{L_j} \alpha_{j,m} 2\pi m t \sin\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\alpha_{k,\ell}, r_j) &= \sum_{m=1}^{L_j} \alpha_{j,m} \pi m t_n^2 \sin\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\alpha_{k,\ell}, \varphi_{j,m}) &= \alpha_{j,m} 2\pi \sin\Delta\Psi_{k,\ell,j,m}(t_n) \\
\mathcal{I}(\alpha_{k,\ell}, \alpha_{j,m}) &= \cos\Delta\Psi_{k,\ell,j,m}
\end{aligned}
$$

Finally, the CRLB is found as the inverse of the FIM.

References

[1] P. Flandrin, "Time-Frequency and Chirps," in *Wavelet Applications VIII*, 2001.

[2] P. Djurić and S. Kay, "Parameter Estimation of Chirp Signals," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 38, pp. 2118–2126, 1990.

[3] O. Besson, M. Ghogho, and A. Swami, "Parameter Estimation for Random Amplitude Chirp Signals," *IEEE Transactions on Signal Processing*, vol. 47, no. 12, pp. 3208–3219, 1999.

[4] I. Rusnak and L. Peled-Eitan, "New Approach to Estimation of Chirp Signal with Unknown Parameters," in *IEEE International Conference on Microwaves, Communications, Antennas and Electronics Systems*, Tel Aviv, Israel, Oct. 21-23 2013.

[5] J. Gal, A. Campeanu, and I. Nafornita, "The Estimation of Chirp Signals Parameters by an Extended Kalman Filtering Algorithm," in *10th International Symposium on Signals, Circuits and Systems*, Iasi, Romania, June 30-July 1 2011.

[6] B. Völcker and B. Ottersten, "Chirp Parameter Estimation from a Sample Covariance Matrix," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 603–612, March 2001.

[7] E. Candes, P. Charlton, and H. Helgason, "Detecting Highly Oscillatory Signals by Chirplet Path Pursuit," 2006, Publication: eprint arXiv:gr-qc/0604017.

[8] R. M. Liang and K. S. Arun, "Parameter Estimation for Superimposed Chirp Signals," in *5th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, San Francisco, USA, March 23-26 1992.

[9] X. Xia, "Discrete Chirp-Fourier Transform and Its Application to Chirp Rate Estimation," *IEEE Transactions on Signal Processing*, vol. 48, pp. 3122–3133, 2000.

[10] A. Brodzik, "On the Fourier Transform of Finite Chirps," *IEEE Signal Processing Letters*, vol. 13, no. 9, pp. 541–544, September 2006.

[11] D. Peacock and B. Santhanam, "Multicomponent Subspace Chirp Parameter Estimation Using Discrete Fractional Fourier Analysis," in *Proceedings of the IASTED International Conference Signal and Image Processing*, Dallas, USA, Dec. 14-16 2011.

[12] I. Daubechies, Y. Wang, and H. Wu, "ConceFT: Concentration of Frequency and Time via a multitapered synchrosqueezed transform," 2015, Publication: eprint arXiv:1507.05366 [math.ST].

[13] J. Xiao and P. Flandrin, "Multitaper Time-Frequency Reassignment for Nonstationary Spectrum Estimation and Chirp Enhancement," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2851–2860, June 2007.

[14] J. Guo, H. Zou, X. Yang, and G. Liu, "Parameter Estimation of Multicomponent Chirp Signals via Sparse Representation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp. 2261–2268, July 2011.

[15] B. Wang and J. Huang, "Instantaneous Frequency Estimation of Multi-Component Chirp Signals in Noisy Enviroments," *Journal of Marine Science and Applications*, vol. 6, no. 4, pp. 13–17, Dec 2007.

[16] F. Auger and P. Flandrin, "Improving the Readability of Time-Frequency and Time-Scale Representations by the Reassignment Method," *IEEE Transactions on Signal Processing*, vol. 43, pp. 1068–1089, 1995.

[17] J. Swärd, J. Brynolfsson, A. Jakobsson, and M. Hansson-Sandsten, "Sparse Semi-Parametric Chirp Estimation," in *Asilomar Conference on Signals, Systems and Computers*, Asilomar, USA, 2-5 Nov 2014.

[18] L. Applebaum, S. Howard, S. Searle, and R. Calderbank, "Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery,"

Applied and Computational Harmonic Analysis, vol. 26, no. 2, pp. 283–290, March 2009.

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

[20] I. Orović, S. Stanković, and A. Draganić, "Time-Frequency Analysis and Singular Value Decomposition Applied to the Highly Multicomponent Musical Signals," *Acta Acustica united with Acustica*, vol. 100, no. 1, pp. 93–101, 2014.

[21] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[22] 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.

[23] M. G. Christensen and J. R. Jensen, "Pitch Estimation for Non-Stationary Speech," in *48th Annual Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, USA, Nov. 2-5 2014.

[24] Y. Doweck, A. Amar, and I. Cohen, "Joint Model Order Selection and Parameter Estimation of Chirps with Harmonic Components," *IEEE Transactions on Signal Processing*, vol. PP, pp. 1, 2015.

[25] 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.

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

[27] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

[28] 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.

[29] 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.

[30] P. G. Bühlmann and S. van de Geer, *Statistics for High-Dimensional Data*, Springer Series in Statistics. Springer, 2011.

[31] 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.

[32] 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.

[33] 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.

[34] J. Eckstein and D.P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, April 1992.

[35] K. B. Petersen and M. S. Pedersen, "The Matrix Cookbook," http://matrixcookbook.com/.

[36] E. J. Candes and T. Tao, "Decoding by Linear Programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec 2005.

[37] R. G. Baraniuk and D. L. Jones, "A Signal Dependent Time Frequency Representation: Optimal Kernel Design," *IEEE Transactions on Signal Processing*, vol. 41, pp. 1589–1602, April 1993.

D

Paper D

Generalized Sparse Covariance-based Estimation

Johan Swärd[1], Stefan Ingi Adalbjörnsson[1], and Andreas Jakobsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*

Abstract

In this work, we generalize the recent sparse iterative covariance-based estimator (SPICE) by extending the problem formulation to allow for different norm constraints on the signal and noise parameters in the covariance model. The resulting extended SPICE algorithm offers the same benefits as the regular SPICE algorithm, including being hyper-parameter free, but the choice of norms allows further control of the sparsity in the resulting solution. We also show that the proposed extension is equivalent to solving a penalized regression problem, providing further insight into the differences between the extended and original SPICE formulations. The performance of the method is evaluated for different choices of norms, indicating the preferable performance of the extended formulation as compared to the original SPICE algorithm. Finally, we introduce two implementations of the proposed algorithm, one gridless formulating for the sinusoidal case, resulting in a semi-definite programming problem, and one grid-based, for which an efficient implementation is given.

Key words: Covariance fitting, sparse reconstruction, convex optimization

1 Introduction

Many problems in signal processing may be well described using a linear model, such that

$$\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{e} \tag{1}$$

where $\mathbf{y} \in \mathbb{C}^N$ is a vector of measurements, $\mathbf{B}$ a matrix of regressors, $\mathbf{x}$ the parameter vector, and $\mathbf{e}$ denotes an additive (complex-valued) noise term, typically assumed to have zero mean and covariance matrix $\mathbf{\Sigma}$. This model occurs in a wide range of applications, such as in, e.g., audio and speech processing [1,2] and spectroscopy [3–7].

Earlier works have primarily focused on *parametric* and *non-parametric* solutions to this problem. The latter kind of estimators typically do not assume any *a-priori* information about the signal, including assumptions on the model order or the signal structure. As a result, such techniques are more robust to uncertainties in the model assumptions that parametric solvers generally impose. However, this robustness also implies that non-parametric methods are, in general, not able to achieve the same level of performance as parametric approaches, given that the made model assumptions hold [8].

Recently, notable efforts have been made to combine these two approaches, developing so-called *semi-parametric* approaches, which typically only make some weak model structure assumptions, such that assuming that the solution is sparse, although restrain from making any stronger model order assumptions.

This is done by forming the dictionary, $\mathbf{B} \in \mathbb{C}^{N\times M}$, using $M \gg N$ signal candidates, whereof only a few are assumed present in the signal. This allows the problem to be reformulated as one of the subset of these M candidates best approximating the measured signal $\mathbf{y}$. This is typically done by enforcing sparsity on the vector $\mathbf{x}$, trading off model fit and the resulting level of sparsity.

In [9], this was done by introducing the Lasso optimization problem

$$\underset{\mathbf{x}}{\text{minimize}}\ \frac{1}{2}\,||\mathbf{y} - \mathbf{B}\mathbf{x}||_2^2 + \mu||\mathbf{x}||_1 \tag{2}$$

where the first term penalizes the ℓ_2-norm distance between the model and the signal, whereas the second term enforces sparsity upon the vector $\mathbf{x}$, with μ being a user parameter governing the trade-off between the two terms. During recent years, many other sparse techniques have been proposed (see, e.g., [10–15] and

the references therein). Many of these methods suffer from the drawback of requiring the selection of one or many user parameters, often being a non-trivial task. In some cases, the user parameters may be selected using physical aspects, or via some kind of rule of thumb (see, e.g., [16]). Other ideas include solving the problem for all different values of the parameter [15, 17], or to use some iterative process for aiding in the choice [10, 18, 19]. Another common way is to use cross-validation to find a suitable regularization parameter (see, e.g., [15]).

In [20], a novel sparse technique based on a covariance fitting criteria was proposed, avoiding the requirement of selecting any user parameters (see also [21–25]). The proposed minimization criteria was there formed as

$$\underset{\tilde{\mathbf{p}}\geq 0}{\text{minimize}} \; \left|\left|\mathbf{R}^{1/2}(\tilde{\mathbf{p}})\left(\mathbf{R}(\tilde{\mathbf{p}})-\mathbf{y}\mathbf{y}^{*}\right)\right|\right|_{F}^{2} \tag{3}$$

where $||\cdot||_F$ denotes the Frobenius norm, $(\cdot)^*$ the conjugate transpose, and where

$$\mathbf{R}(\tilde{\mathbf{p}}) = \mathbf{A}\mathbf{P}\mathbf{A}^{*} \tag{4}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{B} & \mathbf{I} \end{bmatrix} \tag{5}$$

$$\mathbf{p} = \begin{bmatrix} p_1 & \dots & p_M \end{bmatrix}^{T} \tag{6}$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 & \dots & \sigma_N \end{bmatrix}^{T} \tag{7}$$

$$\tilde{\mathbf{p}} = \begin{bmatrix} \mathbf{p}^{T} & \boldsymbol{\sigma}^{T} \end{bmatrix}^{T} \tag{8}$$

$$\mathbf{P} = \text{diag}\left(\tilde{\mathbf{p}}\right) \tag{9}$$

with $\mathbf{I}$ denoting the $N \times N$ identity matrix, $(\cdot)^T$ the transpose, σ_k the noise variance for sample k, and diag($\mathbf{z}$) the diagonal matrix with the vector $\mathbf{z}$ along its diagonal, and zeros elsewhere. It was further shown that solving (3) is equivalent with solving [20]

$$\underset{\tilde{\mathbf{p}}\geq 0}{\text{minimize}} \; \mathbf{y}^{*}\mathbf{R}^{-1}(\tilde{\mathbf{p}})\mathbf{y} + ||\tilde{\mathbf{W}}\tilde{\mathbf{p}}||_1 \tag{10}$$

where

$$\tilde{\mathbf{W}} = \text{diag}\left(\begin{bmatrix} w_1 & \dots & w_{M+N} \end{bmatrix}\right) \tag{11}$$

$$w_k = ||\mathbf{a}_k||_2^2/||\mathbf{y}||_2^2, \text{ for } k = 1, \dots, N+M \tag{12}$$

with $\mathbf{a}_k$ denoting the kth column of $\mathbf{A}$.

Clearly, both (2) and (10) minimize a signal fitting criteria. In the former case, this is done by minimizing the distance between the model and the data, whereas the latter measures the distance through the inverse of the (model) covariance matrix. Both methods also impose an ℓ_1 norm constraint, with the first one penalizing the parameters corresponding to the different candidates in the dictionary $\mathbf{B}$, whereas the second, the so-called SPICE formulation, penalizes both the parameters corresponding to $\mathbf{B}$ and the parameters corresponding to the noise.

In this work, we propose to generalize the SPICE formulation to allow for different penalties on $\mathbf{p}$ and $\boldsymbol{\sigma}$, as given in (6) and (7), respectively, for two different cases. The first case considers the situation when all noise variances, σ_k, are equal, whereas the second considers the case when they are allowed to differ. In the case of equal noise variances, we show that the choice of norm for the noise parameters corresponds to different choices of the regularizing parameter, μ. In the case when the noise variances are allowed to be different, the choices of norms are similarly shown to affect the sparsity level. This results in the fact that even if the different SPICE formulations are hyper-parameter free, one may interpret the choices of norms as the equivalence of selecting hyper-parameters dictating the sparseness of the solution, and that the original SPICE version is equivalent to one particular choice of norms. We also provide an efficient grid-based implementation of the proposed method, which, indirectly, allows for solving (weighted) square-root Lasso problems for a wide choice of regularizing parameters. Additionally, we state a semi-positive programming (SDP) problem that allows for solving the proposed SPICE extension, for the sinusoidal case, without the use of a grid search.

2 The $\{r, q\}$-SPICE formulation

It is worth noting that the second term in (10) penalizes the magnitude of each p_j and σ_k, thus promoting a sparse solution with only a few of the terms in $\tilde{\mathbf{p}}$ being non-zero. However, since the penalty does not distinguish between setting the different terms to zero, one may expect that some of the σ_k may be forced to be zero as a part of the minimization.

If this happens, the result will be solutions that are less sparse than desired. Intuitively, this may be understood by interpreting (10) to require that $\mathbf{R}$ is invertible. Thus, setting some σ_k to zero will cause the resulting covariance matrix,

R, to lose rank, unless some of the p_j are non-zero. This was also observed in [26], wherein a gridless formulation of SPICE was presented. For this formulation, it was shown that **R** had full rank with probability one, resulting in an overestimation of the model order. As a result, forcing any σ_k to zero will yield a less sparse **p**, thus increasing the estimated model order. This implies that, in the original SPICE formulation, σ_k and p_j are competing for the sparseness allowed in the solution of (10). In this work, we propose to treat the σ_k terms different from the rest of the p_j terms. A naive way of doing this could be to omit σ_k from the cost function of (10), but this would result in all the p_j terms being set to zeros, as σ_k may then take on any value which will make **R** full rank, and will thus make the p_j terms redundant. Clearly, the σ_k terms must instead be penalized to produce a meaningful solution to (1). This may be done in different ways, for instance using

$$\underset{\mathbf{p}\geq 0,\ \boldsymbol{\sigma}\geq 0}{\text{minimize}}\ \mathbf{y}^*\mathbf{R}^{-1}\mathbf{y} + ||\mathbf{W}\mathbf{p}||_r + ||\mathbf{W}_\sigma\boldsymbol{\sigma}||_q \tag{13}$$

where $r, q \geq 1$, such that

$$||\mathbf{W}\mathbf{p}||_r = \left[\sum_{k=1}^{M} w_k^r p_k^r\right]^{1/r} \tag{14}$$

$$||\mathbf{W}_\sigma\boldsymbol{\sigma}||_q = \left[\sum_{k=1}^{N} w_{M+k}^q \sigma_k^q\right]^{1/q} \tag{15}$$

$$\mathbf{W} = \text{diag}\left(\begin{bmatrix} w_1 & \dots & w_M \end{bmatrix}\right) \tag{16}$$

$$\mathbf{W}_\sigma = \text{diag}\left(\begin{bmatrix} w_{M+1} & \dots & w_{M+N} \end{bmatrix}\right) \tag{17}$$

Thus, using $r = 1$ and $q = 1$ yields the original SPICE formulation. More general regularization functions could also be used. Furthermore, one could use an approach reminiscent of the one presented in [27], considering also the case when all $0 < r, q < 1$, resulting in a concave penalty term. However, in this work, we restricted our attention to the $\{r, q\}$-norm case, using $r \geq 1$ and $q \geq 1$, terming the result the $\{r, q\}$-SPICE formulation.

3 Linking {r,q}-SPICE to penalized regression

To examine the implications of introducing the r- and q-norms in the SPICE formulation, we examine the connection between $\{r, q\}$-SPICE and a penalized

regression problem, such as the Lasso expression in (2). In doing so, we follow the derivation in [23, 24], distinguishing between the case when each σ_k is allowed to have a distinct value, and the case when all σ_k are equal. To do so, we recall the following lemma (see also [24]):

Theorem 3.1. Let

$$\mathbf{P} = \text{diag}\left(\begin{bmatrix} p_1 & \dots & p_M \end{bmatrix}\right) \tag{18}$$

and

$$\mathbf{\Sigma} = \text{diag}\left(\begin{bmatrix} \sigma_1 & \dots & \sigma_N \end{bmatrix}\right) \tag{19}$$

Then,

$$\begin{aligned} \mathbf{y}^*\mathbf{R}^{-1}\mathbf{y} = \underset{\mathbf{x}}{\text{minimize}}\ & (\mathbf{y}-\mathbf{B}\mathbf{x})^*\mathbf{\Sigma}^{-1}(\mathbf{y}-\mathbf{B}\mathbf{x}) \\ & + \sum_{k=1}^{M} |x_k|^2/p_k \end{aligned} \tag{20}$$

with the minimum occurring at

$$\hat{\mathbf{x}} = \mathbf{\Sigma}\mathbf{B}^*\mathbf{R}^{-1}\mathbf{y} \tag{21}$$

□

3.1 Varying noise variance

Using Lemma 1, one may express (13) as

$$\begin{aligned} \underset{\mathbf{x},\mathbf{p},\sigma}{\text{minimize}}\ & \sum_{k=1}^{N} |y_k - \mathbf{b}_k^*\mathbf{x}|^2/\sigma_k + \sum_{k=1}^{M} |x_k|^2/p_k \\ & + \left(\sum_{k=1}^{M} w_k^r p_k^r\right)^{1/r} + \left(\sum_{k=1}^{N} w_{M+k}^q \sigma_k^q\right)^{1/q} \end{aligned} \tag{22}$$

Solving (22) for p_j yields

$$p_j = w_k^{-\frac{r}{r+1}} |x_k|^{\frac{2}{r+1}} ||\mathbf{W}^{1/2}\mathbf{x}||^{\frac{r-1}{r+1}}_{\frac{2r}{r+1}} \tag{23}$$

Differentiating the function to be minimized in (22) with respect to σ_k and setting it to zero yields

$$-\frac{|y_k - \mathbf{b}_k^* \mathbf{x}|^2}{\sigma_k^2} + \frac{w_{M+k}^q \sigma_k^{q-1}}{||\mathbf{W}_\sigma \boldsymbol{\sigma}||_q^{q-1}} = 0 \tag{24}$$

Summing over k on both sides and simplifying, one arrives at

$$||\mathbf{W}_\sigma \boldsymbol{\sigma}||_q = ||\mathbf{W}_\sigma^{1/2} \mathbf{r}||_{\frac{2q}{q+1}} \tag{25}$$

Inserting (25) into (24) yields

$$\sigma_k = w_{M+k}^{-\frac{q}{q+1}} |r_k|^{\frac{2}{q+1}} \left|\left|\mathbf{W}_\sigma^{1/2} \mathbf{r}\right|\right|_{\frac{2q}{q+1}}^{\frac{q-1}{q+1}} \tag{26}$$

Finally, inserting (23) and (26) into (22) yields

$$\underset{x}{\text{minimize}} \ \left|\left|\mathbf{W}_\sigma^{1/2} (\mathbf{y} - \mathbf{B}\mathbf{x})\right|\right|_{\frac{2q}{q+1}} + \left|\left|\mathbf{W}^{1/2} \mathbf{x}\right|\right|_{\frac{2r}{r+1}} \tag{27}$$

As may be noted from the resulting expression, using $q = 1$ yields the least absolute deviations (LAD) estimate, whereas using $q = \infty$ yields the (unscaled) square-root Lasso. The implications of this is discussed further below. Regardless of the choice of q, the corresponding problem in (13) will still be scale invariant. This may be seen by following the example in [24], scaling each p_k and σ_k with a constant c and do the same for the cost function in (13), defining

$$\begin{aligned} g(\mathbf{p}, \boldsymbol{\sigma}) &\triangleq c\mathbf{y}^* \left(\mathbf{A}c\mathbf{P}\mathbf{A}^*\right)^{-1} \mathbf{y} \\ &\quad + c \left[\sum_{k=1}^{M} w_k^r c^r p_k^r\right]^{1/r} + c \left[\sum_{k=M+1}^{N+M} w_k^q c^q p_k^q\right]^{1/q} \\ &= \mathbf{y}^* \left(\mathbf{A}\mathbf{P}\mathbf{A}^*\right)^{-1} \mathbf{y} + c^2 \left[\sum_{k=1}^{M} w_k^r p_k^r\right]^{1/r} + \\ &\quad c^2 \left[\sum_{k=M+1}^{N+M} w_k^q p_k^q\right]^{1/q} \end{aligned} \tag{28}$$

Let $f(\mathbf{p}, \boldsymbol{\sigma})$ denote the cost function in (13). Then, one may use Lemma 2 in [24] to conclude that if

$$\{\hat{\mathbf{p}}, \hat{\boldsymbol{\sigma}}\} = \arg\min_{\mathbf{p},\boldsymbol{\sigma}} g(\mathbf{p}, \boldsymbol{\sigma}) \tag{29}$$

and

$$\{\hat{\bar{\mathbf{p}}}, \hat{\bar{\boldsymbol{\sigma}}}\} = \arg\min_{\bar{\mathbf{p}},\bar{\boldsymbol{\sigma}}} f(\bar{\mathbf{p}}, \bar{\boldsymbol{\sigma}}) \tag{30}$$

then

$$\hat{\bar{\mathbf{p}}} = c\hat{\mathbf{p}} \tag{31}$$

where $c > 0$, which is true in the here examined case as well. The observed scale invariance implies that one may view the $\{r, q\}$-SPICE method as being hyper-parameter free in the same sense as the original SPICE algorithm is. Furthermore, it may be noted that when converting the p_k to x_k, using (21), any scaling will disappear.

3.2 Uniform noise variance

If, similar to [23, 24], one instead assumes that all the noise terms have equal variance, treating the case when$\sigma_k = \sigma, \ \forall k$, one may observe interesting connection to the Lasso. Under these assumptions, it has been shown that the SPICE problem is connected to the (weighted) square-root Lasso problem [23, 24], i.e.,

$$\underset{\mathbf{x}}{\text{minimize}} \ ||\mathbf{y} - \mathbf{B}\mathbf{x}||_2 + \mu||\mathbf{W}^{1/2}\mathbf{x}||_1 \tag{32}$$

where $\mu = N^{-1/2}$ yields the SPICE estimator. Following the derivation in Section 3.1, together with the assumption that all the noise terms have equal variance, yields $\mu = N^{-1/2q}$ for the $\{r, q\}$-SPICE formulation, implying the equivalent formulation

$$\underset{\mathbf{x}}{\text{minimize}} \ ||\mathbf{y} - \mathbf{B}\mathbf{x}||_2 + \mu||\mathbf{W}^{1/2}\mathbf{x}||_{\frac{2r}{r+1}} \tag{33}$$

As a result, the choice of q corresponds to selecting the weight that governs the trade-off between the model fitting term and the regularization of the parameters, whereas the choice of r decides which norm will be used in the regularization

of the parameters. Thus, using $r = 1$ means that increasing q corresponds to increasing the sparsity in the (weighted) square-root Lasso; this implies that if the signal at hand is assumed to be sparse, solving $\{r, q, \}$-SPICE with $q > 1$ will yield preferable estimates. Furthermore, setting $r \to \infty$ yields a ridge regression problem, with q governing the amount of regularization. We note that it might be preferable to solve (33) using the $\{r, q\}$-SPICE formulation, rather than solving (33) directly.

4 Efficient implementation

As will be argued later, for sparse problems, the most interesting setting for $\{r, q\}$-SPICE is when $r = 1$, since, according to (33), this will yield an ℓ_1 regularization. To this end, we will in this section derive an efficient implementation for this case. In [20], an efficient implementation of SPICE was introduced. To derive the steps of this algorithm, it was noted that the original SPICE minimization in (10) could also be expressed as

$$\underset{\{p_k \geq 0\}_{k=1}^{M},\, \{\sigma_k \geq 0\}_{k=1}^{N}}{\text{minimize}} \quad \mathbf{y}^* \mathbf{R}^{-1} \mathbf{y} \text{ subject to} \tag{34}$$

Furthermore, it was noted that one could further rewrite the objective in (34) by considering the optimization problem

$$\underset{\mathbf{Q}}{\text{minimize}}\ \mathbf{y}^* \mathbf{Q}^* \mathbf{P}^{-1} \mathbf{Q} \mathbf{y} \text{ subject to } \mathbf{Q}^* \mathbf{A} = \mathbf{I} \tag{35}$$

which has the solution $\mathbf{Q}_0 = \mathbf{P}\mathbf{A}^* \mathbf{R}^{-1}$. By defining

$$\beta = \mathbf{Q}\mathbf{y} \tag{36}$$

one may rewrite (34) as

$$\underset{\{p_k \geq 0\}_{k=1}^{M},\, \{\sigma_k \geq 0\}_{k=1}^{N}}{\text{minimize}} \quad \sum_{k=1}^{M+N} \frac{|\beta_k|^2}{p_k} \text{ subject to } \sum_{k=1}^{M} w_k p_k + \sum_{k=1}^{N} w_k \sigma_k = 1 \tag{37}$$

The estimates may then be found by iteratively updating $\mathbf{R}$ and solving for p_k in (37). For $\{r, q\}$-SPICE, with $r = 1$, when assuming different values for the σ_k, the same update for $\mathbf{R}$ may be used, but instead of (37), one needs to solve

$$\underset{\{p_k\geq 0\}_{k=1}^{M},\{\sigma_k\geq 0\}_{k=1}^{N}}{\text{minimize}} \sum_{k=1}^{M}\frac{|\beta_k|^2}{p_k}+\sum_{k=1}^{N}\frac{|\beta_{M+k}|^2}{\sigma_k}$$
$$\text{subject to } \sum_{k=1}^{M+N} w_k p_k+\left(\sum_{k=1}^{N} w_{M+k}^q \sigma_k^q\right)^{1/q}=1 \tag{38}$$

From the Karush-Kuhn-Tucker (KKT) conditions [28], it follows that

$$-\frac{|\beta_k|^2}{p_k^2}+\lambda w_k=0, \text{ for } k=1,\ldots,M \tag{39}$$
$$-\frac{|\beta_{M+k}|^2}{\sigma_k^2}+\lambda\sigma_k^q w_{M+k}^q\left(\sum_{k=1}^{N} w_{M+k}^q\sigma_k^{q-1}\right)^{1/q}=0 \tag{40}$$

where λ denotes the dual variable, for $k=1,\ldots,M$, together with the constraint in (37). Solving these equation for each p_k and σ_k yields

$$p_k=\frac{|\beta_k|}{\sqrt{w_k}\lambda^{1/2}} \tag{41}$$
$$\sigma_\ell=\frac{|\beta_{M+\ell}|^{\frac{2}{q+1}}||\mathbf{W}_\sigma^{1/2}\boldsymbol{\beta}_\sigma||_{\frac{2q}{q+1}}^{\frac{q-1}{q+1}}}{w_{M+\ell}^{\frac{q}{q+1}}\lambda^{1/2}} \tag{42}$$
$$\lambda=\left(||\mathbf{W}^{1/2}\boldsymbol{\beta}||_1+||\mathbf{W}_\sigma^{1/2}\boldsymbol{\beta}_\sigma||_{\frac{2q}{q+1}}\right)^2 \tag{43}$$

for $k=1,\ldots,M$ and $\ell=1,\ldots,N$, where

$$\boldsymbol{\beta}=\left[\begin{array}{ccc}\beta_1 & \ldots & \beta_M\end{array}\right]^T \tag{44}$$
$$\boldsymbol{\beta}_\sigma=\left[\begin{array}{ccc}\beta_{M+1} & \ldots & \beta_{M+N}\end{array}\right]^T \tag{45}$$

This allows for the formulation of an efficient implementation by iteratively forming $\mathbf{R}$ from (4), β_k from (36), and p_k and σ_k from (41) and (42), respectively. Since $\{1,q\}$-SPICE allows for a more sparse solution than the original SPICE, one may speed up the computations further by removing the zero valued p_k when forming $\mathbf{R}$ and β_k.

Algorithm 1 The $\{r, q\}$-SPICE estimator with $r = 1$

1: Initiate $p_k^{(0)} = |\mathbf{b}_k^* \mathbf{y}|^2 / ||\mathbf{b}_k||^4$, for $k = 1, \ldots, M$, $\sigma_k^{(0)} = |y_k|$, for $k = 1, \ldots, N$, and set $i = 1$
2: **while** the termination criteria is not fulfilled **do**
3: Let $\mathbf{R}^{(i)} = \mathbf{A}\mathbf{P}^{(i)}\mathbf{A}^*$
4: Form λ from (43)
5: Update $p_k^{(i)}$ from (41), for each $k = 1, \ldots, M$
6: Update $\sigma_k^{(i)}$ from (42), for each $k = 1, \ldots, N$
7: Set $i = i + 1$
8: **end while**

Algorithm 2 The $\{r, q\}$-SPICE estimator for equal σ_k with $r = 1$.

1: Initiate $p_k^{(0)} = |\mathbf{b}_k^* \mathbf{y}|^2 / ||\mathbf{b}_k||^4$, for $k = 1, \ldots, M$,
$\sigma^{(0)} = \sqrt{\frac{1}{N-1} \sum_{k=1}^{N} \left(y_k - \bar{y}\right)^2}$, for $k = 1, \ldots, N$, and set $i = 1$
2: **while** the termination criteria is not fulfilled **do**
3: Let $\mathbf{R}^{(i)} = \mathbf{A}\mathbf{P}^{(i)}\mathbf{A}^*$
4: Form λ from (48)
5: Update $p_k^{(i)}$ from (46), for each $k = 1, \ldots, M$
6: Update $\sigma_k^{(i)}$ from (47), for each $k = 1, \ldots, N$
7: Set $i = i + 1$
8: **end while**

When instead assuming that $\sigma_k = \sigma, \ \forall k$, one obtains the steps

$$p_k = \frac{|\beta_k|}{\sqrt{w_k}\lambda^{1/2}} \tag{46}$$

$$\sigma = \frac{||\beta_M||_2}{N^{1/2q}\lambda^{1/2}} \tag{47}$$

$$\lambda = \left(||\mathbf{W}^{1/2}\beta||_1 + ||N^{1/(2q)}\beta_\sigma||_2\right)^2 \tag{48}$$

for $k = 1, \ldots, M$. Algorithms 1 and 2 summarize the $\{1, q\}$-SPICE implementations for the two settings, with $\bar{y}$ denoting the mean value of the vector $\mathbf{y}$. Similar to the previous case, since using $q > 1$ will enforce more sparsity than $q = 1$, one may utilize this added sparsity in the implementation of the algorithm. Since most of the elements in $\mathbf{p}$ will be zero, one may form $\mathbf{R}^{-1}$ by only considering

the columns and rows of $\mathbf{A}$ and $\mathbf{A}^*$ corresponding to the non-zero entries in $\mathbf{p}$. Let $\hat{K}^{(i)}$ be the number of non-zero entries in $\mathbf{p}^{(i)}$ at iteration i. Then, if $\hat{K} < N$, one may use the Woodbury matrix identity to efficiently calculate the inverse of $\mathbf{R}$ (see, e.g., [29]).

The termination criterias in Algorithms 1 and 2 can take on many forms. In this work, we have chosen to terminate the algorithms when the percentage of change in $\mathbf{p}$ and $\boldsymbol{\sigma}$ between two consecutive iterations falls below a certain level, say in the range $[10^{-9}, 10^{-3}]$.

Note that the algorithm described in Algorithm 2 solves a (weighted) square-root Lasso problem, where the different choices of q corresponds to different levels of sparsity, i.e., different values of μ in (32). If one is interested in solving a (weighted) square-root Lasso with $\mu = \mu_0$, then one may instead solve the $\{r, q\}$-SPICE with $q = -\frac{1}{2 \ln \mu_0}$, as long as $q > 1$, and with $r = 1$. Thus, the algorithm in Algorithm 2 presents an attractive and efficient way of solving the (weighted) square-root Lasso problem, for a large range of different μ.

To give an idea of the running time of the proposed algorithm as compared with a standard SDP solver (see, e.g., [30, 31]), the algorithms were tested on a problem with $M = 10000$, $N = 1000$, and with $q = 5$, and $r = 1$, where the data vector, $\mathbf{y}$, contained 3 sinusoids, using a standard PC (2.6 Ghz Intel Core i7, 16 GB RAM). The corresponding run times were roughly 4 seconds for the Matlab implementation in Algorithm 2 and 4132 seconds for the SDP Matlab solver[1].

5 Off-grid solution

Many forms of estimation problems are solved by evaluating over a grid of the parameters of interest. However, such a solution may cause concerns when the sought solution falls outside the grid or may be found in between grid points. A common solution to this problem is to increase the grid size to thereby minimize the distance from the closest grid point to the true parameter value (see, e.g., [32, 33]). However, such a solution might cause the columns of the extended dictionary to be highly correlated, thereby decreasing the performance of the method (we instead refer the interested reader to other works treating this issue, e.g., [33–36] and the references therein). In [26] and [37], an off-grid solution to

[1] Our implementation of $\{r, q\}$-SPICE will be made available on the authors' web-pages upon publication.

the original SPICE version was presented for the sinusoidal case. In this section, we similarly provide one possible version of off-grid estimation for the proposed $\{r,q\}$-SPICE method for a signal containing superimposed sinusoids. In order to do so, it may initially be noted that one may separate $\mathbf{R}$ into two different matrices, such that

$$\mathbf{R} = \mathbf{B}^*\text{diag}\left(\mathbf{p}\right)\mathbf{B} + \text{diag}\left(\boldsymbol{\sigma}\right) \triangleq \mathbf{T}(\mathbf{u}) + \text{diag}\left(\boldsymbol{\sigma}\right) \tag{49}$$

where $\mathbf{T}(\mathbf{u})$ is a Toeplitz matrix with $\mathbf{u}$ forming the first column of $\mathbf{T}(\mathbf{u})$. Thus, (13) may be expressed as (see also [26, 37])

$$\begin{aligned}
&\underset{\mathbf{u},\boldsymbol{\sigma},x}{\text{minimize}}\ ||\mathbf{y}||_2^2 x + ||\text{diag}(\mathbf{T}(\mathbf{u}))||_r + ||\mathbf{W}_\sigma\boldsymbol{\sigma}||_q \\
&\text{subject to}\ \begin{bmatrix} x & \mathbf{y}^* \\ \mathbf{y} & \mathbf{T}(\mathbf{u}) + \text{diag}\left(\boldsymbol{\sigma}\right) \end{bmatrix} \geq 0 \\
&\qquad\qquad \mathbf{T}(\mathbf{u}) \geq 0 \\
&\qquad\qquad \mathbf{T}(\mathbf{u}) - \mathbf{T}(\mathbf{u})^* = 0 \\
&\qquad\qquad \boldsymbol{\sigma} \geq 0
\end{aligned} \tag{50}$$

and under the additional constraint that $\mathbf{T}(\mathbf{u})$ is a Toeplitz matrix. The optimization problem in (50) is convex, and may be solved using, e.g., a publicly available SDP solver, such as the one presented in [30, 31]. The final off-grid estimates may then be found using the celebrated Vandermonde decomposition in combination with, for instance, Prony's method (see [8, 38] for further details on such an approach).

6 Numerical examples

Using the interpretation provided by the reformulation in Section 3, it is clear that the choice of r will decide what kind of regularization that will be used. Thus, choosing $r = 1$ will yield an ℓ_1 norm and letting $r \to \infty$ will result in the ℓ_2 norm. In this paper, we consider sparse problems, and will therefore mainly confine our attention to the case where $r = 1$, since this will yield the most sparse convex regularizer, namely ℓ_1.

From the discussion in Section 2, one may expect that SPICE will set some of the elements in $\boldsymbol{\sigma}$ to zero, since the sparsity enforcing term in (10) also applies to these parameters. Figure 1 shows the estimated $\mathbf{p}$ and $\boldsymbol{\sigma}$ for the SPICE and the

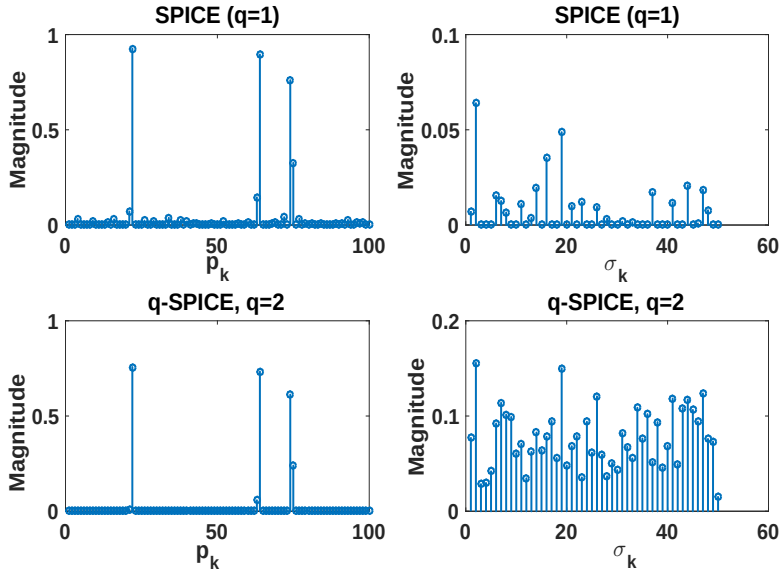

Figure 1: The resulting estimates of $\mathbf{p}$ and σ from the SPICE and the q-SPICE estimator (q=2). Note that q-SPICE is sparser in $\mathbf{p}$, whereas SPICE is sparser in σ. In this example r is set to $r = 1$.

$\{r, q\}$-SPICE estimators, when applied to a linear signal formed using (1) with three non-zero components. As expected, using $r = 1$, $\{r, q\}$-SPICE offers a sparser $\mathbf{p}$ vector as compared to SPICE, whereas the solution is more sparse in $\boldsymbol{\sigma}$ for SPICE. As a result, the sparsity constraints on the σ_k terms in $\{r, q\}$-SPICE are thus relaxed and are instead subjected to a bounding of their power in the q-norm, thus allowing for more sparsity in $\mathbf{p}$.

We will proceed by showing the difference in performance for different values of r and q, to provide an example on how the different choices of these norms affect the estimates. We investigate two properties of the estimators, namely the resulting root-mean-squared error (RMSE) of the frequency estimates, defined as

$$\text{RMSE} \triangleq \sqrt{\frac{1}{P}\sum_{k=1}^{P} |\hat{\vartheta}_k - \vartheta_k|^2} \tag{51}$$

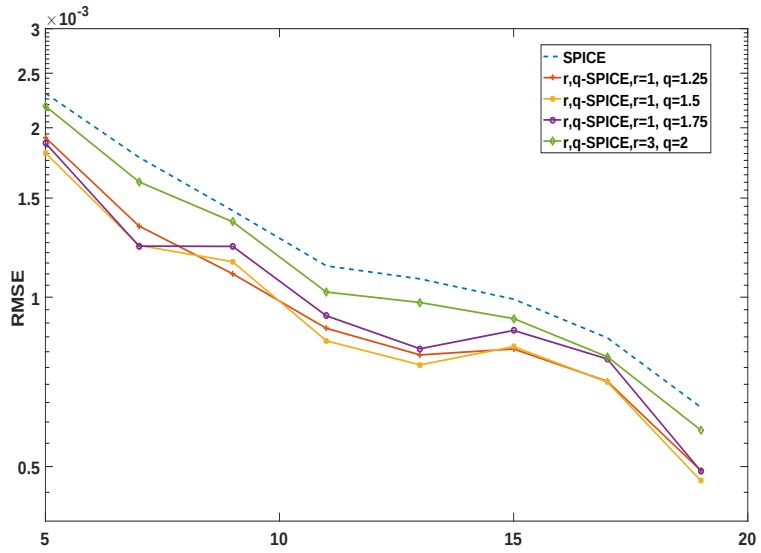

Figure 2: The RMSE of the frequency estimates, as a function of SNR for $\{r, q\}$-SPICE and SPICE.

where ϑ_k is the true frequency of the kth component, whereas $\hat{\vartheta}_k$ is the formed estimate, and the ability to correctly estimate the model order. The signal was $N = 50$ samples long and contained 4 sinusoids with unit magnitude and random phase. The simulation was done using 100 Monte-Carlo simulations for each SNR-level, where the signal-to-noise ratio (SNR) is defined as

$$\text{SNR} = 10 \log_{10} \left(\frac{P_y}{P_\sigma} \right) \tag{52}$$

with P_y denoting the power of the true signal and P_σ the power of the noise. The noise used was circular white Gaussian noise, and the noise terms were allowed to differ. The solution was obtained by solving (50) for all settings except for the original SPICE, where the estimates were obtained from solving the problem formulated in [37]. In Figure 2, the resulting RMSEs are shown for different values of r and q, as a function of the SNR. To make the figures readable, 11 outliers were removed for SPICE and for the $r = 3, q = 2$ case for $\{r, q\}$-SPICE

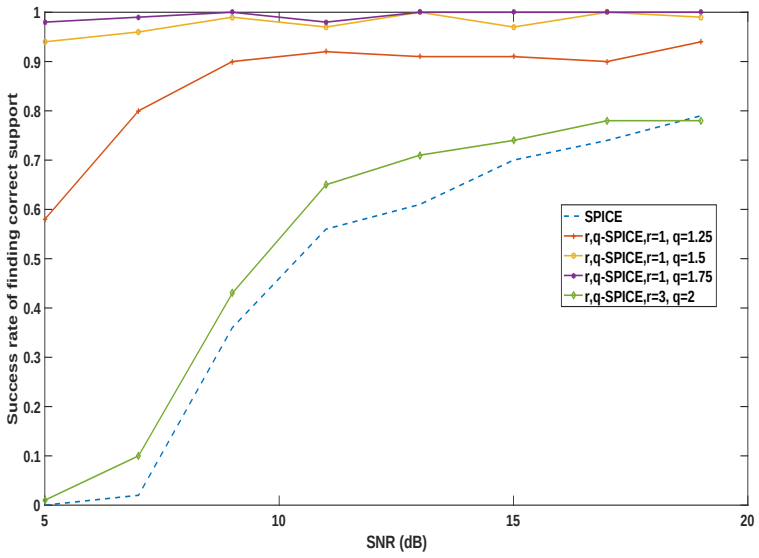

Figure 3: The probability of finding the correct model order of the signal as a function of SNR for $\{r, q\}$-SPICE and SPICE.

each, whereas only 2, 5, and 5 outliers were removed for the case where $q = 1.25$, $q = 1.5$, and $q = 1.75$, respectively. Furthermore, to remove the noise peaks that appear when using small values of q, all peaks smaller than 20 % of the largest found peak were removed. Note, however, that this is not necessary for the case where q is larger. As is clear from the figure, the RMSE is decreased as the sparsity level is increased, with the $\{r, q\}$-SPICE versions outperforming the original SPICE. This is also true for the resulting model order estimation, which is shown in Figure 3. As may be expected, when increasing q the sparsity is increased and the spurious peaks are removed, but as q is further increased, the true peaks start to disappear. In this setting, it seems to be beneficial to set the norms around $q = 1.5$ and $r = 1$. From these results, we conclude that the generalized version of SPICE allows for better estimation of parameter values, as well as model order. As was expected, using $r > 1$ was not beneficial when confronted with a sparse signal, and we will therefore, in the succeeding example, restrict our attention to

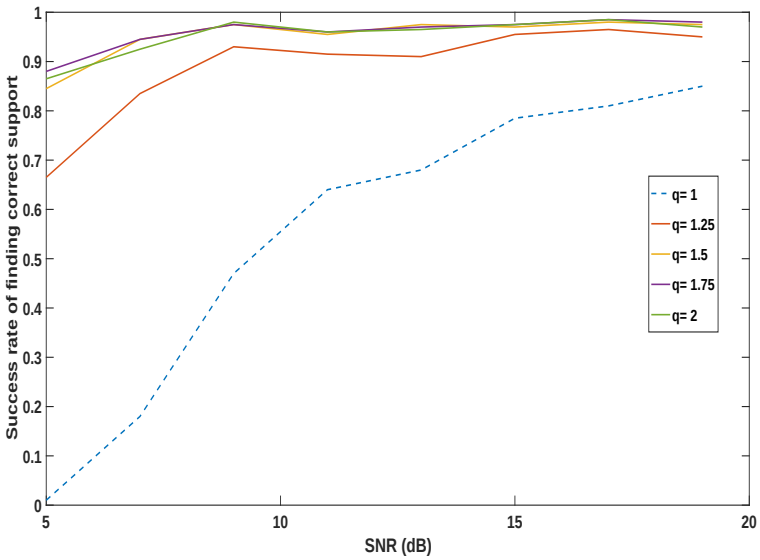

Figure 4: The probability of finding the correct support of the signal as a function of q and SNR. Here, all the σ_k are assumed to be equal. In this example, $r = 1$.

the case where $r = 1$, referring to the method as q-SPICE. However, it should be stressed that for certain situations, it might be preferable to use $r > 1$, e.g., in situations when otherwise considering to use ridge regression; we will further examine this aspect in future works.

Arguably, the most important property of a sparse estimator is the ability to return the true support of the signal, as well as yielding reasonable amplitude estimates for this support. However, it seems inevitable that when including a sparsity enforcing penalty, one also introduced a (downwards) bias on the magnitude of the amplitudes. Fortunately, this problem is often easy to overcome by simply re-estimating the amplitudes using, e.g., least squares, once the true support is known. Accordingly, we will in this section focus on the methods ability of finding the true support of the signal. To this end, 200 Monte-Carlo simulation for each SNR level are formed. In each simulation, $N = 50$ samples of a signal containing three sinusoids, each with unit magnitude, and phase uniformly drawn

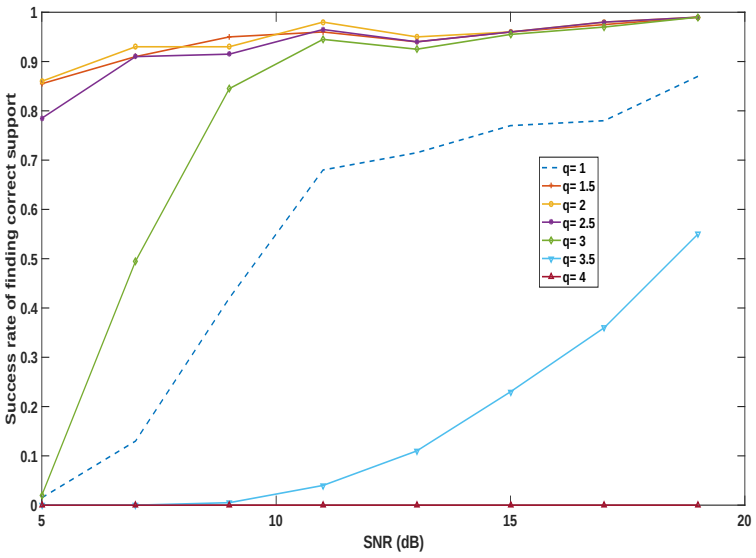

Figure 5: The probability of finding the correct support of the signal as a function of q and SNR. Here, all the σ_k are assumed to be equal. Here, $r = 1$.

from $(0, 2\pi]$, was created. The normalized frequencies were uniformly selected, but were at least $1/2N$ apart. The dictionary contained $M = 1000$ candidate sinusoids, selected on a uniform frequency grid from $(0, 1]$. The estimated support was selected to be the elements of the vector $\mathbf{x}$ that had a corresponding absolute value of at least 20% of the largest estimated value in $\mathbf{x}$. This was done to allow for comparison with the less sparse q-SPICE versions, for cases with small q value (most notably $q = 1$). It may be noted that for values of q that are large, this is not necessary. The support was deemed correctly estimated if the estimated frequencies were at most two grid points away from the true frequencies.

Figure 4 shows the results of applying q-SPICE, for different values of q, assuming that all the σ_k are the same, with $q = 1$ yielding the SPICE estimate. As is clear from the figure, the results improve with increasing q values. From the discussion in Section 3.2, we note that this corresponds to increasing the value of μ in (32), thus increasing the sparsity in the estimates. Thus, one could assume

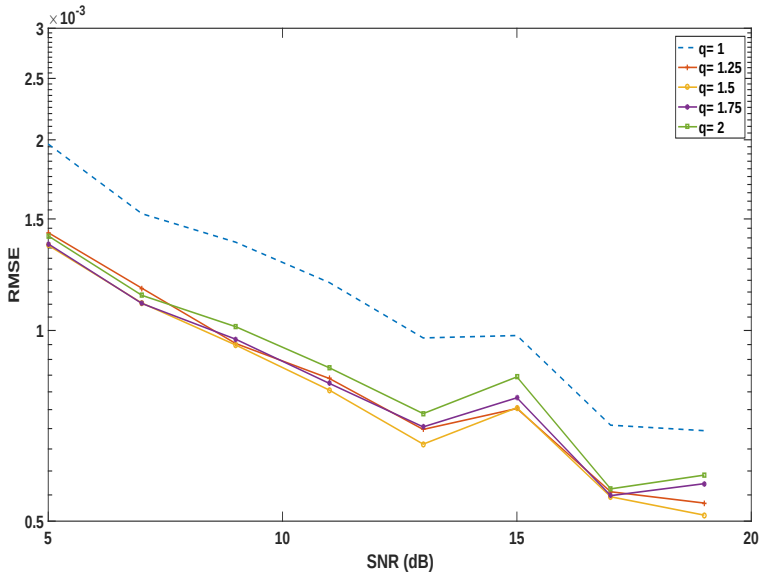

Figure 6: The RMSE of the frequency estimates, as a function of q and SNR. Here, all the σ_k are assumed to be equal. In this example, $r = 1$.

that when further increasing q, the estimate of the support should decline. In Figure 5, this behavior can be seen, where now q-SPICE is evaluated over a range of larger q values. It is also apparent from the figure that the best value for q is for this signal somewhere around $q = 2$, which corresponds to using $\mu \approx 0.38$ in (33). Next, we investigate the precision for different values of q, by using the RMSE of the frequency estimates. Figure 6 shows the resulting RMSE of the frequency estimates, for the three largest values of $\mathbf{x}$. As can be seen in the figure, the RMSE is clearly improving as q is increased, corresponding to sparser solutions. For smaller values of q, the results are not very sparse, and large spurious noise peaks can be found. To improve readability, seven, two, and three outliers were removed from the cases $q = 1$, $q = 1.25$, and $q = 1.5$, respectively. If q is increased too much this will, of course, make the solution too sparse, thus risking setting non-noise peaks to zero. This can also be seen in Figure 5, where for about $q = 3$, the probability of retrieving the true support of the signal starts to decline,

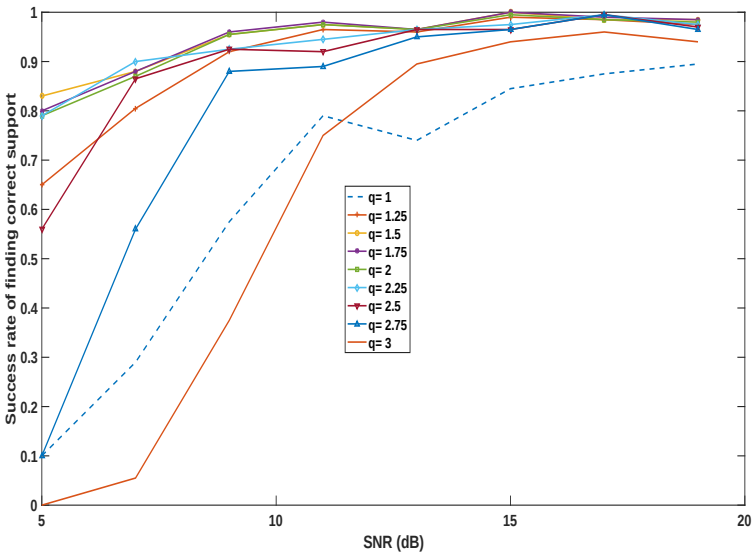

Figure 7: The probability of finding the correct support of the signal as a function of q and SNR. Here, $r = 1$.

and at $q > 3.5$, the solution is too sparse.

We proceed by considering the case when the σ_k parameters are allowed to take on different values, using the same set-up as above. Figures 7 and 8 show the probability of estimating the correct support of the signal and the RMSE of the three largest frequency estimates, respectively. Again, in the interest of readability, three outliers were removed from $q = 1$, six outliers from $q = 1.25$, and three outliers for $q = 1.5$. As previously noted, it is clear from the figures that q governs the sparsity enforced on the solution. From the figures, one may also see that for this setup, it is advantageous to choose q in the interval $q = [1.25, 2.25]$. To demonstrate the differences in the solutions obtained from using different values of q, we show a typical simulation result for four different values of q, namely $q = 1, 1.5, 2$, and 2.5, for the settings above, with SNR $= 5$ dB. Figure 9 shows the results, where it may again be noted that the sparsity level increases with q.

Finally, we provide a numerical example showing the results from solving the

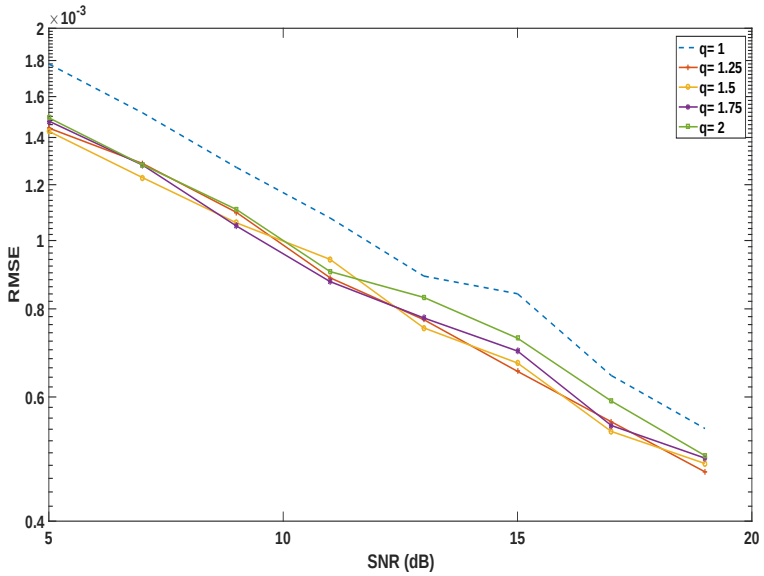

Figure 8: The RMSE of the frequency estimates, as a function of q and SNR. Here, $r = 1$.

$\{r, q\}$-SPICE using (50), with $r = 1$ and $q = 1.75$, and for the case where each noise variance are allowed to differ across the samples. In this scenario, we evaluated the gridless version of $\{r, q\}$-SPICE, given in (50), and the gridless version of SPICE, given in [37], together with the grid-based $\{r, q\}$-SPICE, given a frequency grid of $M = 500$ grid points. In each of the 100 Monte-Carlo simulations, the $N = 50$ samples long signal contained four sinusoids, each with random phase, with two peaks having magnitude 4, one peak magnitude 2, and the last one unit magnitude. The frequencies were selected not to be closer than $1/2N$ from each other and were randomly selected in each simulation from the interval $(0, 1]$. Both gridless versions were computed using the SDP-solver in CVX [30, 31]. Figure 10 and 11 show the resulting RMSE and probability of finding the correct support as functions of the SNR level, respectively. As seen in the figures, the two versions of the q-SPICE outperforms the gridless version of SPICE. It is worth noting that in this scenario, only SPICE had the benefit of

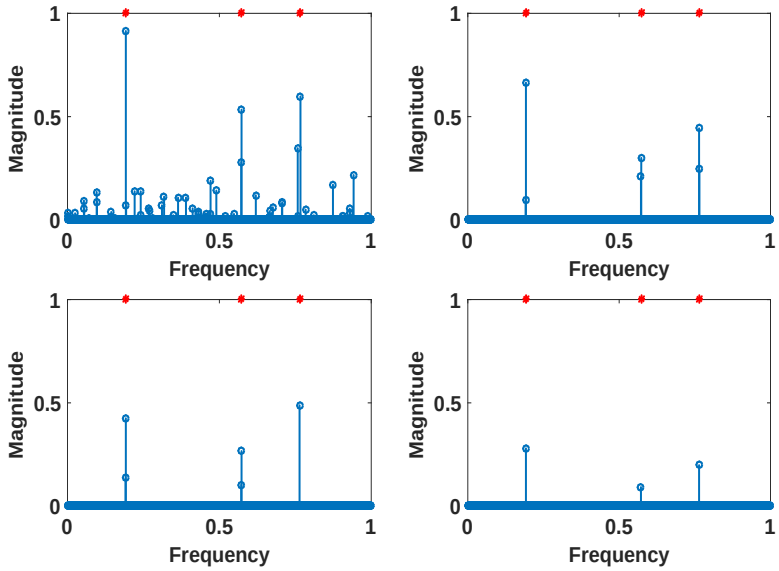

Figure 9: A typical result from q-SPICE for different values of q. Top left: $q = 1$, top right $q = 1.5$, bottom left $q = 2$, and bottom right $q = 2.5$. The red stars indicate the position and size of the true sinusoids. In this example, $r = 1$.

removing the smallest peaks. Furthermore, the model order was deemed correct if the method found the true number of peaks, thus there were no limitation on how close an estimated frequency had to be the true value. If the model order was too high, the four largest peaks were selected to compute the RMSE, whereas if the model order was too low, these estimates were omitted from the RMSE evaluations.

Furthermore, one may see that the gridless version of q-SPICE is slightly better than the gridded version. However, this slight improvement from using the gridless q-SPICE version may not be worth the extra computation time; the gridless version took on average 9.4 seconds to execute, whereas the gridded version only took 0.5 seconds. However, it is worth recalling that other works on gridless solutions implicate that faster implementations are available (see, e.g., [39]), and these improvements in implementation can likely also be applied to the gridless

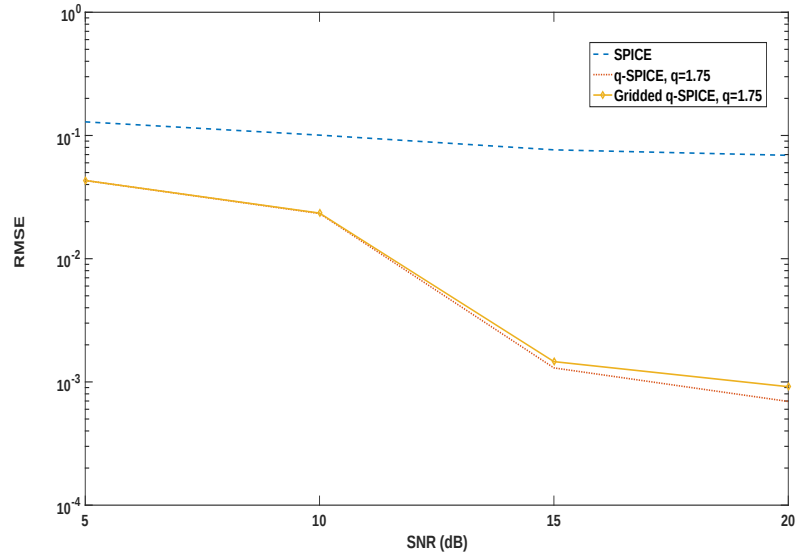

Figure 10: The RMSE of the frequency estimates, as defined in (51), as a function of SNR for the gridless versions of q-SPICE and SPICE, together with the gridded version of q-SPICE.

q-SPICE.

7 Conclusion

In this paper, we introduced a generalization of the SPICE method, in which we allow for a trade-off between the penalties for the model, using a q-norm, and the noise parameters, using an r-norm. We show that for larger values of q, one achieves a higher level of sparsity and better performance for recovering the support of the signal. Furthermore, we show that the proposed method is equivalent to a penalized regression formulation, with the $\frac{2q}{q+1}$ norm on the model fit, for the case when we let the noise variance vary across all samples. In the case where the noise variance is assumed to be equal for all samples, it is shown that the proposed method is equal to the (weighted) square-root Lasso, where the regular-

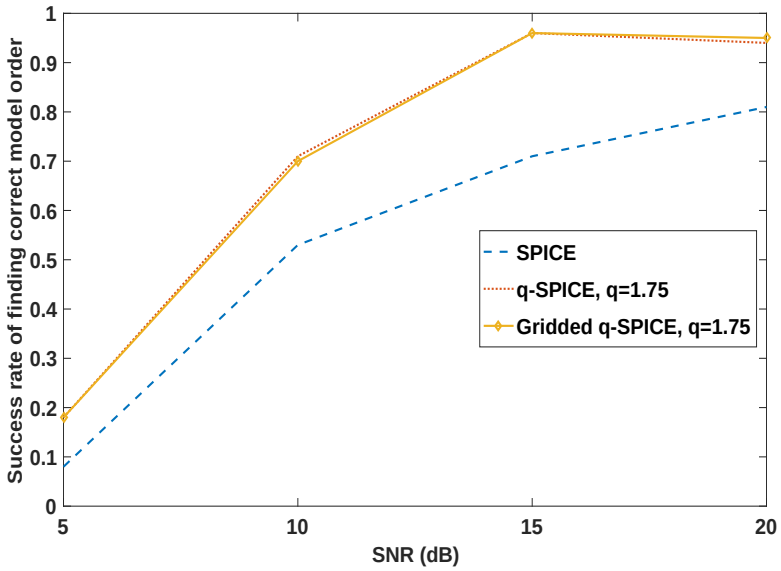

Figure 11: The probability of finding the correct model order of the signal as a function of SNR for the gridless versions of q-SPICE and SPICE, together with the gridded version of q-SPICE.

ization parameter has a one-to-one correspondence to the choice of q for a given problem. Furthermore, we provide a fast and efficient implementation for both the case when $r = 1$ and the noise variances are equal for all samples, and where they are allowed to differ. As a result of the shown equivalence, the presented implementation offers an attractive alternative for solving $\frac{2q}{q+1}$-norm problems, and, perhaps more interesting, (weighted) square-root Lasso problems for different regularization parameters. We also present a gridless version of $\{r, q\}$-SPICE for the sinusoidal signals, which is on the form of an SDP problem. Numerical result show the preferred performance of the $\{r, q\}$-SPICE as compared to the original SPICE method, both for gridded and for gridless versions for the estimator.

References

[1] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[2] 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.

[3] Y. Li, J. Razavilar, and K. J. R. Liu, "A High-Resolution Technique for Multidimensional NMR Spectroscopy," *IEEE Trans. Biomed. Eng.*, vol. 45, no. 1, pp. 78–86, 1998.

[4] W. Sun and H. C. So, "Accurate and Computationally Efficient Tensor-Based Subspace Approach for Multidimensional Harmonic Retrieval," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5077–5088, Oct. 2012.

[5] S. D. Somasundaram, A. Jakobsson, J. A. S. Smith, and K. Althoefer, "Exploiting Spin Echo Decay in the Detection of Nuclear Quadrupole Resonance Signals," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 925–933, April 2007.

[6] Y. Tan, S. L. Tantum, and L. M. Collins, "Cramér-Rao Lower Bound for Estimating Quadrupole Resonance Signals in Non-Gaussian Noise," *IEEE Signal Process. Lett.*, vol. 11, no. 5, pp. 490–493, May 2004.

[7] 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.

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

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

[10] 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.

[11] 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.

[12] J. Fang, J. Li, Y. Shen, H. Li, and S. Li, "Super-Resolution Compressed Sensing: An Iterative Reweighted Algorithm for Joint Parameter Learning and Sparse Signal Recovery," *IEEE Signal Process. Lett.*, vol. 21, no. 6, pp. 761–765, 2014.

[13] 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.

[14] X. Tan, W. Roberts, J. Li, and P. Stoica, "Sparse Learning via Iterative Minimization With Application to MIMO Radar Imaging," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1088–1101, March 2011.

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

[16] J. Swärd, J. Brynolfsson, A. Jakobsson, and M. Hansson-Sandsten, "Sparse Semi-Parametric Estimation of Harmonic Chirp Signals," *IEEE Trans. Signal Process.*, vol. 64, no. 7, pp. 1798–1807, April 2016.

[17] 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.

[18] D. Wipf and S. Nagarajan, "Iterative Reweighted ℓ_1 and ℓ_2 Methods for Finding Sparse Solutions," *IEEE J. Sel. Topics in Signal Processing*, vol. 4, pp. 317–329, 2010.

[19] 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.

[20] 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, no. 1, pp. 35–47, Jan 2011.

[21] P. Stoica, P. Babu, and J. Li, "SPICE : a novel covariance-based sparse estimation method for array processing," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 629 –638, Feb. 2011.

[22] P. Stoica and P. Babu, "SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation," *Signal Processing*, vol. 92, no. 7, pp. 1580–1590, July 2012.

[23] C. R. Rojas, D. Katselis, and H. Hjalmarsson, "A Note on the SPICE Method," *IEEE Trans. Signal Process.*, vol. 61, no. 18, pp. 4545–4551, Sept. 2013.

[24] 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.

[25] D. Zachariah and P. Stoica, "Online Hyperparameter-Free Sparse Estimation Method," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3348–3359, July 2015.

[26] Z. Yang and L. Xie, "On Gridless Sparse Methods for Line Spectral Estimation From Complete and Incomplete Data," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3139–3153, June 2015.

[27] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 707–710, Oct. 2007.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[29] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, 4^{th} edition, 2013.

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

[31] 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.

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

[33] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

[34] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed Sensing Off the Grid," *IEEE Trans. Inform. Theory*, vol. 59, no. 11, pp. 7465–4790, Nov 2013.

[35] B. N. Bhaskar, G. Tang, and B. Recht, "Atomic Norm Denoising with Applications to Line Spectral Estimation," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5987 – 5999, July 2013.

[36] Z. Yang, L. Xie, and C. Zhang, "Off-Grid Direction of Arrival Estimation Using Sparse Bayesian Inference," *IEEE Trans. Signal Process.*, vol. 61, no. 1, pp. 38 – 43, October 2012.

[37] P. Stoica, G. Tang, Z. Yang, and D. Zachariah, "Gridless Compressed-Sensing Methods for Frequency Estimation: Points of Tangency and Links to Basics," in *22nd European Signal Processing Conference*, Lisbon, Portugal, 2014.

[38] T. Blu, P. L. Dragotti, M. Vetterli, P. Marziliano, and L. Coulot, "Sparse Sampling of Signal Innovations," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 31 – 40, March 2008.

[39] Z. Yang and L. Xie, "Enhancing Sparsity and Resolution via Reweighted Atomic Norm Minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 4, pp. 995–1006, Feb 2016.


E

Paper E

Online Estimation of Multiple Harmonic Signals

Filip Elvander, Johan Swärd, and Andreas Jakobsson

Centre for Mathematical Sciences, Lund University, Lund, Sweden

Abstract

In this paper, we propose a time-recursive multi-pitch estimation algorithm using a sparse reconstruction framework, assuming that only a few pitches from a large set of candidates are active at each time instant. The proposed algorithm does not require any training data, and instead utilizes a sparse recursive least squares formulation augmented by an adaptive penalty term specifically designed to enforce a pitch structure on the solution. The amplitudes of the active pitches are also recursively updated, allowing for a smooth and more accurate representation. When evaluated on a set of ten music pieces, the proposed method is shown to outperform other general purpose multi-pitch estimators in either accuracy or computational speed, although not being able to yield performance as good as the state-of-the art methods, which are being optimally tuned and specifically trained on the present instruments. However, the method is able to outperform such a technique when used without optimal tuning, or when applied to instruments not included in the training data.

Key words: Adaptive signal processing, dictionary learning, group sparsity, multi-pitch estimation, sparse recursive least squares

1 Introduction

The problem of estimating the fundamental frequency, or pitch, arises in a variety of fields, such as in speech and audio processing, non-destructive testing, and biomedical modeling (see, e.g., [1–6], and the references therein). In such applications, the measured signal may often result from several partly simultaneous sources, meaning that both the number of pitches, and the number of overtones of each such pitch, may be expected to vary over the signal. Such would be the case, for instance, in most forms of audio signals. The resulting multi-pitch estimation problem is in general difficult, with one of the most notorious issues being the so-called sub-octave problem, i.e., distinguishing between pitches whose fundamental frequencies are related by powers of two. Both non-parametric, such as methods based on autocorrelation (see, e.g., [7] and references therein), and parametric multi-pitch estimators (see, e.g., [2]) have been suggested, where the latter are often more robust to the sub-octave problem, but rely heavily on accurate *a priori* model order information of both the number of pitches present and the number of harmonic overtones for each pitch.

Regrettably, the need for accurate model order information is a significant drawback, as such information is typically difficult to obtain and may vary rapidly over the signal. In order to alleviate this, several sparse reconstruction algorithms tailored for multi-pitch estimation have recently been proposed, allowing for estimators that do not require explicit knowledge of the number of sources or their harmonics; for example, in [8], the so-called PEBS estimator was introduced, exploiting the block-sparse structure of the pitch signal. This estimator was then further developed in [9], such that the likelihood of erroneously selecting a sub-octave in place of the true pitch was lowered, while also introducing a self-regularization technique for selecting the penalty parameters. Both these estimators form implicit model order decisions based on one or more tuning parameters that dictate the relative weight of various penalties. As shown in the above cited works, the resulting estimators are able to allow for (rapidly) varying model orders, without significant loss of performance. Earlier works based on sparse representations of signals also include works such as [10], which considers atomic decomposition of audio signals in both the time and the frequency domains.

There have also been methods proposed for multi-pitch estimation and tracking that are source specific, i.e., tailored specifically to sources, e.g., musical instruments, that are known to be present in the signal. In [11], the authors perform multi-pitch estimation on music mixtures by, via a probabilistic framework,

matching the signal to a pre-learned dictionary of spectral basis vectors that correspond to instruments known to be present in the signal. A similar source specific idea was used in [12], where pitch estimation was performed by matching the signal to spectral templates learned from individual piano keys. Other methods specifically designed to handle multi-pitch estimation for pianos include [13–15]. Another field of research is designing multi-pitch estimators based on a two-matrix factorisation of the short-time Fourier transform, i.e., a non-negative matrix factorization (see, e.g., [16–18]). The method has also been used in the sparse reconstruction framework, for instance to learn atoms in order to decompose the signal [19]. A common assumption is also that of spectral smoothness within each pitch, which may also be exploited in order to improve the estimation performance (see, e.g., [13, 17, 20, 21]).

In many audio processing applications, pitch tracking is of great interest and despite being a problem that has been studied for a long time, it still attracts a lot of attention. Over the years, there have been many different approaches for tracking pitches; some of the more recent include particle filters [22], neural networks [23], and Bayesian filtering [24]. Many of these methods require *a priori* model order information, and/or are limited to the single pitch case. The sparse pitch estimators in [8], [9] are robust to these model assumptions, and allow for multiple pitches. However, these estimators process each data frame separately, treating each as an isolated and stationary measurement, without exploiting the information obtained from earlier data frames when forming the estimates. To allow for such correlation over time, the PEBS estimator introduced in [8] was recently extended to exploit the previous pitch estimates, as well as the power distribution of the following frame, when processing the current data frame [25]. In this work, we extend on this effort, but instead propose a fully time-recursive problem formulation using the sparse recursive least squares (RLS) estimator. The resulting estimator does not only allow for more stable pitch estimates as compared to earlier sparse multi-pitch estimators, as more information is used at each time-point, but also decreases the computational burden of each update, as new estimates are formed by updating already available ones.

On the other hand, sparse adaptive filtering is a field attracting steadily increasing attention, with, for instance, the sparse RLS algorithm being explored for adaptive filtering in, e.g., [26–28]. Other related studies include [29], wherein the authors use a projection approach to solve a recursive LASSO-type problem, and [30], which introduced an online recursive method allowing for an underly-

ing dynamical signal model and the use of sparsity-inducing penalties. Recursive algorithms designed for group-sparse systems have also been introduced, such as the ones presented in [31–33], but to the best of our knowledge, no such technique has so-far been applied to the multi-pitch estimation problem. This is the problem we strive to address in this paper. It should be noted that the here presented work differs from many other multi-pitch estimators in that it only exploits the assumption that the signal of interest is generated by a harmonic sinusoidal model. Recently, quite a few methods for multi-pitch estimation adhering to the machine learning paradigm have been proposed (see, e.g., [34], [35]). In these methods, a model is trained on labeled signals, such as, e.g., notes played by individual music instruments, extracting features from the training data that are then used for classification in the estimation stage. As opposed to this, the method presented here is not dependent on being trained on any dataset prior to the estimation.

Our earlier efforts on multi-pitch estimation based on sparse modeling, such as the PEBS [8] and PEBSI-Lite [9] algorithms, have focused on frame-based multi-pitch estimation techniques, with PEBS introducing the use of block sparsity to form the pitch estimates, and PEBSI-Lite refining these ideas and introducing a self-regularization technique to select the required user parameters. In this work, we build on the insights from these algorithms, and expand these ideas by introducing a method that allows for a sample-by-sample updating, in the form of an RLS-like sparse estimator, thereby allowing the estimates to also exploit information available in earlier data samples. The sub-octave problems experienced by PEBS and later alleviated by PEBSI-Lite, with the use of a total-variation penalty enforcing spectral smoothness, is here addressed using an adaptively re-weighted block penalty. Furthermore, we introduce a signal-adaptive updating scheme for the dictionary frequency atoms that allows the proposed method to, e.g., track frequency modulated signals, and alleviates grid mismatches otherwise commonly experienced by dictionary based methods.

The remainder of this paper is organized as follows; in the next section, we introduce the multi-pitch signal model and its corresponding dictionary formulation. Then, in Section 3, we introduce the group sparse RLS formulation for multi-pitch estimation, followed by a scheme for decreasing the bias of the harmonic amplitude estimates in Section 4. Section 5 presents a discussion about various algorithmic considerations. Section 6 contains numerical examples illustrating the performance of the proposed estimator on various audio signals.

Finally, Section 7 concludes upon the work.

1.1 Notation

In this work, we use lower case non-bold letters such as x to denote scalars and lower case boldface letter such as $\mathbf{x}$ to denote vectors. Upper case bold face letters such as $\mathbf{X}$ are used for matrices. We let diag $(\mathbf{x})$ denote a diagonal matrix formed with the vector $\mathbf{x}$ along its diagonal. Sets are denoted using upper case calligraphic letters such as $\mathcal{A}$. If $\mathcal{A}$ and $\mathcal{B}$ are sets of integers, then $\mathbf{x}_{\mathcal{A}}$ denotes the sub-vector of $\mathbf{x}$ indexed by $\mathcal{A}$. For matrices, $\mathbf{X}_{\mathcal{A},\mathcal{B}}$ denotes the matrix constructed using the rows indexed by $\mathcal{A}$ and columns indexed by $\mathcal{B}$. We use the shorthand $\mathbf{X}_{\mathcal{A}}$ to denote $\mathbf{X}_{\mathcal{A},\mathcal{A}}$. Furthermore, $[\bar{\cdot}]$, $[\cdot]^H$, and $[\cdot]^T$ denotes complex conjugation, conjugate transpose, and transpose, respectively. Also, $|\mathcal{A}|$ is the cardinality of the set $\mathcal{A}$, and $|\mathbf{x}|$ denotes the number of elements in the vector $\mathbf{x}$, unless otherwise stated. Finally, we for vectors $\mathbf{x} \in \mathbb{C}^n$ let $\|\mathbf{x}\|_\ell$ denote the ℓ-norm, defined as

$$\|\mathbf{x}\|_\ell = \left(\sum_{j=1}^{n} |x_j|^\ell \right)^{1/\ell} \tag{1}$$

and use $i = \sqrt{-1}$.

2 Signal model

Consider a measured signal[1], $y(t)$, that is generated according to the model $y(t) = x(t) + e(t)$, where

$$x(t) = \sum_{k=1}^{K(t)} \sum_{\ell=1}^{L_k(t)} w_{k,\ell}(t) \mathrm{e}^{i2\pi f_k(t)\ell t} \tag{2}$$

with $K(t)$ denoting the number of pitches at time t, with fundamental frequencies $f_k(t)$, having $L_k(t)$ harmonics, $w_{k,\ell}(t)$ the complex-valued amplitude of the ℓth harmonic of the kth pitch, and where $e(t)$ denotes a broad-band additive noise. It should be stressed that the number of pitches, as well as their fundamental frequencies, and the number of harmonics for each source, may vary over time.

[1] For notational and computational simplicity, we here consider the discrete-time analytic signal of any real-valued measured signal.

It is worth noting that we here assume a harmonic signal, such as detailed in (2); however, as shown in the numerical section, the proposed method does also work well for somewhat inharmonic signals, such as, e.g., those resulting from a piano.

We here attempt to approximate the measured signal using a sparse representation in an over-complete harmonic basis, see, e.g., [36]. Specifically, as in [8], [9], the signal sources are approximated using a sparse modeling framework containing P candidate pitches, each allowed to have up to $L_{\max}$ harmonics, such that

$$x(t) \approx \sum_{p=1}^{P} \sum_{\ell=1}^{L_{\max}} w_{p,\ell}(t) \mathrm{e}^{i2\pi f_p(t)\ell t} \tag{3}$$

where the dictionary is selected large enough so that (at least) $K(t)$ candidate pitches, $f_p(t)$, reasonably well approximate the true pitch frequencies (see also, e.g., [37], [38]), i.e., such that $P \gg \max_t K(t)$ and $L_{\max} \gg \max_{t,k} L_k(t)$. It should be noted that as the signal is assumed to contain relatively few pitches at each time instance, the resulting amplitude vector will be sparse, although with a harmonic structure reflecting the overtones of the pitches. Furthermore, it may be noted that the frequency grid-points, $f_p(t)$, are allowed to vary with time, which will here be implemented using an adaptive dictionary learning scheme. Using this framework, the pitches present in the signal at time t may be implicitly estimated by identifying the non-zero amplitude coefficients, $w_{p,\ell}(t)$.

3 Group-sparse RLS for pitches

Exploiting the structure of the signal, we introduce the group-sparse adaptive filter, $\mathbf{w}(t)$, which at time t is divided into P groups according to

$$\mathbf{w}(t) = \left[\begin{array}{ccc} \mathbf{w}_1^T(t) & \ldots & \mathbf{w}_P^T(t) \end{array} \right]^T \tag{4}$$

$$\mathbf{w}_p(t) = \left[\begin{array}{ccc} w_{p,1}(t) & \ldots & w_{p,L_{\max}}(t) \end{array} \right]^T \tag{5}$$

implying that, ideally, only $K(t)$ sub-vectors $\mathbf{w}_p(t)$ will be non-zeros at time t. In order to achieve this, the filter is formed as

$$\hat{\mathbf{w}}(t) = \arg\min_{\mathbf{w}} g_t(\mathbf{w}) + h_t(\mathbf{w}) \tag{6}$$

where $\hat{\mathbf{w}}(t)$ denotes the solution of (6), $g_t(\mathbf{w})$ the regular RLS criterion, (see, e.g., [39]), formed as

$$g_t(\mathbf{w}) = \frac{1}{2}\sum_{\tau=1}^{t} \lambda^{t-\tau}\left|y(\tau) - \mathbf{w}^T\mathbf{a}(\tau)\right|^2 \tag{7}$$

and $h_t(\mathbf{w})$ a sparsity inducing penalty function. Note that a similar adaptive filter formulation for estimating sparse data structures was introduced in [27]. However, whereas [27] considered sparse signals, we in this work expand this approach to also consider block sparsity, and specifically the pitch structure. As a result, the dictionary is here formed as

$$\mathbf{a}(t) = \left[\begin{array}{ccc} \mathbf{a}_1^T(t) & \ldots & \mathbf{a}_P^T(t) \end{array}\right]^T \tag{8}$$

$$\mathbf{a}_p(t) = \left[\begin{array}{ccc} \mathrm{e}^{i2\pi f_p(t)t} & \ldots & \mathrm{e}^{i2\pi f_p(t)L_{\max}t} \end{array}\right]^T \tag{9}$$

and $\lambda \in (0,1)$ being a user-determined forgetting factor. The choice of the forgetting factor λ will reflect assumptions on the variability of the spectral content of the signal, with λ close to 1 implying an almost stationary signal, whereas a smaller value will allow for a quicker adaption to changes in the spectral content. The sparsity inducing function, $h_t(\mathbf{w})$, should be selected as to encourage a pitch-structure in the solution; in [9], which considered multi-pitch estimation on isolated time frames, this function, which then was not a function of time, was selected as

$$h(\mathbf{w}) = \gamma_1\|\mathbf{w}\|_1 + \gamma_2\sum_{p=1}^{P}\left\|\mathbf{F}\mathbf{w}_{\mathcal{G}_p}\right\|_1 \tag{10}$$

where $\mathbf{F}$ is the first difference matrix and $\mathcal{G}_p$ is the set of indices corresponding to the harmonics of the candidate pitch p. The second term of this penalty function is the ℓ_1-norm of the differences between consecutive harmonics and acts as a total variation penalty on the spectral envelope of each pitch. Often referred to as the sparse fused LASSO [40], this penalty was in [9] used to promote solutions with spectral smoothness in each pitch, although requiring some additional refinements to achieve this. To allow for a fast implementation, we will here instead consider the time-varying penalty function

$$h_t(\mathbf{w}) = \gamma_1(t)\|\mathbf{w}\|_1 + \sum_{p=1}^{P}\gamma_{2,p}(t)\left\|\mathbf{w}_{\mathcal{G}_p}\right\|_2 \tag{11}$$

where $\gamma_1(t)$ and $\gamma_{2,p}(t)$ are non-negative regularization parameters. This penalty, often called the sparse group LASSO [41] when combined with a squared ℓ_2-norm model fit term, is reminiscent of the one used in the PEBS method introduced in [8], and belongs to the class of methods utilizing mixed norms for sparse signal estimation (see, e.g., [42]). The second term of this penalty function, the pitch-wise ℓ_2-norm, has a group-sparsifying effect, encouraging solutions where active harmonics are grouped together into a few number of pitches. As the frequency content of different pitches may be quite similar due to overlapping, or close to overlapping, harmonics, the group penalty thus prevents erroneous activation of isolated harmonics, while still allowing the different groups to retain harmonics shared by different sources (see also [8], [9]). In the case of overlapping harmonics in the signal, i.e., the presence of two pitches which share at least one harmonic, the ℓ_2-norm will favor solutions of the optimization problem (6) in which the powers of these harmonics are shared among the two pitches. The precise level of sharing is decided by the relative powers of the unique harmonics of each pitch so that the pitch having unique harmonics with more power will also be assigned a larger share of the power corresponding to the overlapping harmonics. In the case of the the two pitches having unique harmonics with equal combined power, the power of the overlapping harmonics will also be shared equally. However, when, as in [8], using fixed penalty parameters $\gamma_1(t)$ and $\gamma_{2,p}(t)$, the resulting estimate has been shown to be prone to mistaking a pitch for its sub-octave (see also [9]). In order to discourage this type of erroneous solutions, we will herein introduce a way of adaptively choosing the group sparsity parameter, $\gamma_{2,p}(t)$, as further discussed below.

We note that $g_t(\mathbf{w})$, as defined in (7), may be expressed in matrix form as

$$g_t(\mathbf{w}) = \frac{1}{2}\left\|\mathbf{\Lambda}_{1:t}^{1/2}\mathbf{y}_{1:t} - \mathbf{\Lambda}_{1:t}^{1/2}\mathbf{A}_{1:t}\mathbf{w}\right\|_2^2 \tag{12}$$

where

$$\mathbf{y}_{\tau:t} = \begin{bmatrix} y(\tau) & \dots & y(t) \end{bmatrix}^T \tag{13}$$

$$\mathbf{A}_{\tau:t} = \begin{bmatrix} \mathbf{a}(\tau) & \dots & \mathbf{a}(t) \end{bmatrix}^T \tag{14}$$

and with $\mathbf{\Lambda}_{1:t} = \operatorname{diag}\left(\begin{bmatrix} \lambda^{t-1} & \lambda^{t-2} & \dots & 1 \end{bmatrix}\right)$. To simplify notation, define

$$\mathbf{R}(t) \triangleq \mathbf{A}_{1:t}^H \mathbf{\Lambda}_{1:t}\mathbf{A}_{1:t} \tag{15}$$

$$\mathbf{r}(t) \triangleq \mathbf{A}_{1:t}^H \mathbf{\Lambda}_{1:t} \mathbf{y}_{1:t} \,. \tag{16}$$

With these definitions, the minimization in (6) may be formed using proximal gradient iterations, (see, e.g., [43]), such that the jth iteration may be expressed as

$$\hat{\mathbf{w}}^{(j+1)}(t) = \arg\min_{\mathbf{w}} \frac{1}{2s(t)} \left\| \boldsymbol{\nu}^{(j)} - \mathbf{w} \right\|_2^2 + h_t(\mathbf{w}) \tag{17}$$

where

$$\boldsymbol{\nu}^{(j)} = \hat{\mathbf{w}}^{(j)}(t) + s(t) \left[\mathbf{r}(t) - \mathbf{R}(t) \hat{\mathbf{w}}^{(j)}(t) \right] \tag{18}$$

with $s(t)$ denoting the step-size. We note that this update is reminiscent of the one presented in [27], which considers the problem of ℓ_1-regularized recursive least squares, although it should be noted that the ℓ_1-norm for complex vectors in [27] is defined to be the sum of the absolute values of the real and imaginary parts separately, whereas we here use the more common definition, as given by (1). In [27], the authors motivate their minimization algorithm by casting it as an EM-algorithm using reasoning from [44], as well as some further assumptions about properties of the signal. By studying the zero sub-differential equations for (17), it can be shown that the closed form solution for each group p can be computed separately as (see, e.g., equations (54)-(55) and (32)-(38) in [8]; for further details, see also [41])

$$\tilde{\boldsymbol{\nu}}_{\mathcal{G}_p}^{(j)} = S_1 \left(\boldsymbol{\nu}_{\mathcal{G}_p}^{(j)}, s(t)\gamma_1(t) \right) \tag{19}$$

$$\hat{\mathbf{w}}_{\mathcal{G}_p}^{(j+1)}(t) = S_2 \left(\tilde{\boldsymbol{\nu}}_{\mathcal{G}_p}^{(j)}, s(t)\gamma_{2,p}(t) \right) \tag{20}$$

where $S_1(\cdot)$ and $S_2(\cdot)$ are the soft thresholding operators corresponding to the ℓ_1- and ℓ_2-norms, respectively, i.e.,

$$S_1(\mathbf{z}, \alpha) = \frac{\max\left(|\mathbf{z}| - \alpha, 0\right)}{\max\left(|\mathbf{z}| - \alpha, 0\right) + \alpha} \odot \mathbf{z} \tag{21}$$

$$S_2(\mathbf{z}, \alpha) = \frac{\max\left(\|\mathbf{z}\|_2 - \alpha, 0\right)}{\max\left(\|\mathbf{z}\|_2 - \alpha, 0\right) + \alpha} \mathbf{z} \tag{22}$$

where, in (21), $|\mathbf{z}|$ denotes the vector obtained by taking the absolute value of each element of the vector $\mathbf{z}$, the max function operates element-wise on the vector $\mathbf{z}$,

and $\odot$ denotes element-wise multiplication. Furthermore, as $\mathbf{R}(t)$ and $\mathbf{r}(t)$ can be expressed as

$$\mathbf{R}(t) = \sum_{\tau=1}^{t} \lambda^{t-\tau} \mathbf{a}(\tau)\mathbf{a}^H(\tau) \tag{23}$$

$$\mathbf{r}(t) = \sum_{\tau=1}^{t} \lambda^{t-\tau} y(\tau)\bar{\mathbf{a}}(\tau) \tag{24}$$

these entities can be updated according to

$$\mathbf{R}(t) = \lambda\mathbf{R}(t-1) + \mathbf{a}(t)\mathbf{a}^H(t) \tag{25}$$

$$\mathbf{r}(t) = \lambda\mathbf{r}(t-1) + y(t)\bar{\mathbf{a}}(t)\,, \tag{26}$$

when new samples become available. Here, $\bar{(\cdot)}$ denotes complex conjugation.

4 Refined amplitude estimates

In general, the sparsity promoting penalty function $h_t(\mathbf{w})$ will introduce a downward bias on the magnitude of the amplitude estimates formed by (6). However, as the support of $\hat{\mathbf{w}}(t)$ will reflect the fundamental frequencies present in the signal, we can refine the amplitude estimates by minimizing a least squares criterion. As this problem only considers amplitudes of harmonics of pitches that are believed to be in the signal, we do not need to use any sparsity inducing penalties and can therefore avoid the magnitude bias. This will be analogous to estimating the amplitudes of each harmonic using recursive least squares assuming that the support of the filter is known. To this end, let

$$\mathcal{S}(t) = \bigcup_{p\in\mathcal{A}(t)} \mathcal{G}_p \tag{27}$$

$$\mathcal{A}(t) = \left\{ p \mid \left\| \hat{\mathbf{w}}_{\mathcal{G}_p}(t) \right\|_2 > 0 \right\}\,, \tag{28}$$

i.e., $\mathcal{A}(t)$ is the set of active pitches determined by the sparse filter $\hat{\mathbf{w}}(t)$, at time t, and $\mathcal{S}(t)$ is the index set corresponding to the harmonics of these pitches. Let $\breve{\mathbf{w}}(t)$ denote the refined amplitude estimates at time t. Given $\hat{\mathbf{w}}(t)$, and thereby $\mathcal{S}(t)$, we update this filter according to

$$\breve{\mathbf{w}}_k(t) = 0\,, k \notin \mathcal{A}(t) \tag{29}$$

$$\breve{\mathbf{w}}_{\mathcal{S}(t)}(t) = \underset{\mathbf{w}\in\mathbb{C}^{|\mathcal{S}(t)|}}{\arg\min}\, \mathbf{w}^H\mathbf{R}_{\mathcal{S}(t)}\mathbf{w} - \mathbf{w}^H\mathbf{r}_{\mathcal{S}(t)} - \mathbf{r}_{\mathcal{S}(t)}^H\mathbf{w} + \xi\|\mathbf{w} - \breve{\mathbf{w}}_{\mathcal{S}(t)}(t-1)\|_2^2 \tag{30}$$

where $\mathbf{R}_{\mathcal{S}(t)}(t)$ is the $|\mathcal{S}(t)| \times |\mathcal{S}(t)|$ matrix constructed by the rows and columns of $\mathbf{R}(t)$ indexed by $\mathcal{S}(t)$ and $\mathbf{r}_{\mathcal{S}(t)}(t)$ is the $|\mathcal{S}(t)|$ dimensional vector constructed by the elements of $\mathbf{r}(t)$, indexed by $\mathcal{S}(t)$. The second term of (30) is a proximal term that will promote a smooth trajectory for the magnitude of the filter coefficients, where the parameter $\xi > 0$ controls the smoothness. This type of smoothness-promoting penalty has earlier been used, for instance, to enforce temporal continuity in NMF applications [45]. To avoid inverting large matrices, we split the solving of (30) into $\mathcal{A}(t)$ problems of size $L_{\max}$ using a cyclic coordinate descent scheme (see also, e.g., [26]). To this end, define the index sets

$$\mathcal{Q}_p = \mathcal{S}(t) \setminus \mathcal{G}_p \ , p \in \mathcal{A}(t) \ , \tag{31}$$

i.e., the indices corresponding to harmonics that are not part of pitch p. Considering only terms in the cost function in (30) that depend on harmonics of the pth pitch, we can form an update of the corresponding filter coefficients according to

$$\breve{\mathbf{w}}_{\mathcal{G}_p}(t) = \underset{\mathbf{w}\in\mathbb{C}^{L_{\max}}}{\arg\min}\, \mathbf{w}^H\mathbf{R}_{\mathcal{G}_p}\mathbf{w} - \mathbf{w}^H\mathbf{r}^{(p)} - \mathbf{r}^{(p)H}\mathbf{w} + \xi\left\|\mathbf{w} - \breve{\mathbf{w}}_{\mathcal{G}_p}(t-1)\right\|_2^2 \tag{32}$$

where

$$\mathbf{r}^{(p)} = \mathbf{r}_{\mathcal{G}_p} - \mathbf{R}_{\mathcal{G}_p,\mathcal{Q}_p}\tilde{\mathbf{w}}_{\mathcal{Q}_p} \ . \tag{33}$$

The vector $\tilde{\mathbf{w}}_{\mathcal{Q}_p} \in \mathbb{C}^{|\mathcal{Q}_p|}$ contains the (partially updated) filter coefficients that correspond to other pitches than p, i.e.,

$$\tilde{\mathbf{w}}_{\mathcal{G}_q} = \begin{cases} \breve{\mathbf{w}}_{\mathcal{G}_q}(t) & \text{if updated} \\ \breve{\mathbf{w}}_{\mathcal{G}_q}(t-1) & \text{if not updated} \end{cases} \tag{34}$$

for $q \neq p$. By setting the gradient of (32) with respect to $\mathbf{w}$ to zero, we find the update of $\breve{\mathbf{w}}_{\mathcal{G}_p}(t)$ to be

$$\breve{\mathbf{w}}_{\mathcal{G}_p}(t) = \left(\mathbf{R}_{\mathcal{G}_p} + \xi\mathbf{I}\right)^{-1}\left(\mathbf{r}^{(p)} + \xi\breve{\mathbf{w}}_{\mathcal{G}_p}(t-1)\right) \ . \tag{35}$$

5 Algorithmic considerations

We proceed to examine some implementation aspects of the presented algorithm, first discussing the appropriate choice of the penalty parameters, then possible computational speed-ups, as well as ways of adaptively updating the used pitch dictionary.

5.1 Parameter choices

In order to discourage solutions containing erroneous sub-octaves, we here propose to update the group penalty parameter, in iteration j of the filter update (17), as

$$\gamma_{2,p}(t) = \gamma_2(t) \max\left(1, \frac{1}{\left|\hat{w}_{p,1}^{j-1}(t)\right| + \varepsilon}\right) \tag{36}$$

where $\left|\hat{w}_{p,1}^{j-1}(t)\right|$ is the estimated amplitude of the first harmonic of group p, obtained in iteration $j-1$, with $\varepsilon \ll 1$ being a user-specified parameter selected to avoid a division by zero. In this paper, we use $\varepsilon = 10^{-5}$. As sub-octaves will typically have missing first harmonics, such a choice will encourage shifting power from the sub-octave to the proper pitch. Similar types of re-weighted penalties have earlier been used to enhance sparsity in the estimated signal (see, e.g., [46], [47]). Studies using many different kinds of pitch signals indicate that the overall performance of the algorithm is relatively insensitive to the choice of the parameter $s(t)$, which may typically be selected in the range $s(t) \in \left[10^{-5}, 10^{-3}\right]$. Here, we use $s(t) = 10^{-4}$. The choice of the penalty parameters $\gamma_1(t)$ and $\gamma_2(t)$ can be made using inner-products between the dictionary and the signal. Letting Δ denote the time-lag, define

$$\eta(t,\mu) = \mu\left\|\mathbf{\Lambda}_{1:\Delta}\mathbf{A}_{t-\Delta:t}^{H}\mathbf{y}_{t-\Delta:t}\right\|_{\infty} \tag{37}$$

where $\mu \in (0,1)$. A good rule of thumb is choosing $\gamma_1(t)$ in the neighborhood of (37) with $\mu = 0.1$, whereas a corresponding reasonable value for $\gamma_2(t)$ is $\mu = 1$. Empirically, the performance of the algorithm has been seen to be robust to variations of these choices of μ. This method emulates choosing the values of the penalty parameters based on the correlation between the signal and the dictionary in a finite window. Here, the window length, Δ, is determined by the forgetting

factor, λ, and by how much correlation one is willing to lose as a result from the truncation. For example, selecting

$$\Delta = \frac{\log(0.01)}{\log \lambda} \tag{38}$$

will yield a window such that the excluded samples will contribute to less than 0.01 of the correlation. It should be noted that for smoothly varying signals, $\gamma_1(t)$ and $\gamma_2(t)$ only need to be updated infrequently.

5.2 Iteration speed-up

As the signal is assumed to have a sparse representation in the dictionary $\mathbf{a}(t)$, one may expect updates of the coefficients of many groups, here indexed by q, to result in zero amplitude estimates. As such groups do not contribute to the pitch estimates, these groups would preferably be excluded from the updates in (17)-(18). If assuming the support of $\mathbf{w}(t)$ to be constant for all t, one could thus sequentially discard such groups from the updating step, and thereby decrease computation time. However, as generally pitches may disappear and then re-appear, as well as drift in frequency over time, we will here only exclude the groups q from the updating steps temporarily. That is, if at time τ, we have $\left\|\hat{\mathbf{w}}_{\mathcal{G}_q}\right\|_2 < \tilde{\varepsilon}$, where $\tilde{\varepsilon} \ll 1$, the group q is considered not to be present in the signal and is therefore excluded from the updating steps for a waiting period, T. After that period, it is again included in the updates, allowing it to again appear in the signal. Defining the set $\mathcal{U}$, indexing the groups that are considered active, the group q is adaptively included and excluded from $\mathcal{U}$ depending on the size of $\left\|\hat{\mathbf{w}}_{\mathcal{G}_q}\right\|_2$. If the signal can be assumed to have slowly varying spectral content, meaning that the support of $\mathbf{w}(t)$ is also varying slowly, the waiting period T may be chosen to be quite long, as to improve the computational efficiency. In general, choosing T as to correspond to a few milliseconds allows for a speed-up of the algorithm while at the same time enabling it to track the time evolution of $\mathbf{w}(t)$.

5.3 Dictionary learning

In general, a signal's pitch frequencies may vary over time, for instance, due to vibrato. Applying the filter updating scheme using fixed grid-points will therefore result in rapidly changing support of the filter or energy leakage between adjacent blocks of the filter, here indexed by p. In order to overcome this problem, and to

Algorithm 1 The PEARLS algorithm

Initialise $\hat{\mathbf{w}}(0) \leftarrow \mathbf{0}$, $\mathbf{R}(0) \leftarrow \mathbf{0}$, $\mathbf{r}(0) \leftarrow \mathbf{0}$
$t \leftarrow 1$
repeat {Recursive update scheme}
  $\mathbf{R}(t) \leftarrow \lambda \mathbf{R}(t-1) + \mathbf{a}(t)\mathbf{a}^H(t)$
  $\mathbf{r}(t) \leftarrow \lambda \mathbf{r}(t-1) + y(t)\bar{\mathbf{a}}(t)$
  $j \leftarrow 0$
  $\hat{\mathbf{w}}^{(j)}(t) \leftarrow \hat{\mathbf{w}}(t-1)$
  repeat {Proximal gradient update}
    $\boldsymbol{\nu}^{(j)} \leftarrow \hat{\mathbf{w}}^{(j)}(t) + s(t)\left[\mathbf{r}(t) - \mathbf{R}(t)\hat{\mathbf{w}}^{(j)}(t)\right]$
    $\hat{\mathbf{w}}^{(j+1)}(t) \leftarrow \arg\min_{\mathbf{w}} \frac{1}{2s(t)}\left\|\boldsymbol{\nu}^{(j)} - \mathbf{w}\right\|_2^2 + h_t(\mathbf{w})$
    $j \leftarrow j + 1$
  until convergence
  $\hat{\mathbf{w}}(t) \leftarrow \hat{\mathbf{w}}^{(j)}(t)$
  Determine $\mathcal{A}(t)$ and $\mathcal{S}(t)$
  $\breve{\mathbf{w}}_k(t) \leftarrow 0$, $k \notin \mathcal{A}(t)$
  $\breve{\mathbf{w}}_{\mathcal{S}(t)}(t) = \arg\min_{\mathbf{w}\in\mathbb{C}^{|\mathcal{S}(t)|}} \mathbf{w}^H\mathbf{R}_{\mathcal{S}(t)}\mathbf{w} - \mathbf{w}^H\mathbf{r}_{\mathcal{S}(t)} - \mathbf{r}_{\mathcal{S}(t)}^H\mathbf{w} + \xi\|\mathbf{w} - \breve{\mathbf{w}}_{\mathcal{S}(t)}(t-1)\|_2^2$
  Update active set $\mathcal{U}$
  if $t \in \mathcal{T}$ **then**
    Update dictionary
  end if
  $t \leftarrow t + 1$
until end of signal

allow for smooth tracking of pitches over time, we propose a scheme for adaptively updating the dictionary of candidate pitches. This adaptive adjustment scheme also allows for the use of a grid with coarser resolution than would otherwise be possible. Let $\mathcal{T} = \{\tau_k\}_k$ be the set of time points in which the dictionary is updated. As only groups $\hat{\mathbf{w}}_{\mathcal{G}_p}(\tau_k)$ with non-zero power are considered to be present in the signal, one only has to adjust the fundamental frequencies of these. Assuming that the current estimate of such a candidate pitch frequency is $f_p(\tau_{k-1})$, one only needs to consider adjusting it on the interval $f_p(\tau_{k-1}) \pm \frac{1}{2}\delta_{f,k}(t)$, where $\delta_{f,k}(t)$ denotes the current grid-point spacing. The update can be formed using

the approximate non-linear least squares method in [48], [2], where, instead of $L_{\max}$, one uses the harmonic order corresponding to the non-zero components of $\hat{\mathbf{w}}_{\mathcal{G}_p}(\tau_k)$. This refined estimate is obtained by first forming the residual, and adding back the current group of harmonics, whereafter the approximate non-linear least squares method is applied to update the frequencies. The adjusted frequency $f_p(\tau_k)$ is then used to update the dictionary on the time interval $\left[\tau_k,\, \tau_{k+1}\right)$. After updating the dictionary, the filter coefficient estimates will, due to the recursive nature of the method, be partly based on the old dictionary and partly on the updated one. It is thus very likely that after the dictionary update the phase component of the two filter coefficient parts will differ. To avoid this, we instead incorporate the phase into the dictionary, thus obtaining a filter coefficient with zero phase. This is accomplished by estimating the phases at the same time as the frequencies are updated in the dictionary updating step. Each estimated phase is then multiplied with the corresponding column of the dictionary, thus including the phases into the dictionary. This update corresponds to changing (8) and (9) to

$$\mathbf{a}(t,\boldsymbol{\varphi}) = \left[\begin{array}{cccc} \mathbf{a}_1^T(t,\boldsymbol{\varphi}_1) & \ldots & \mathbf{a}_P^T(t,\boldsymbol{\varphi}_P) \end{array}\right]^T \tag{39}$$

$$\mathbf{a}_p(t,\boldsymbol{\varphi}_p) = \left[\begin{array}{cccc} \mathrm{e}^{i2\pi f_p(t)t + i\pi\varphi_{p_1}} & \ldots & \mathrm{e}^{i2\pi f_p(t)L_{\max}t + i\pi\varphi_{p_{L_{\max}}}} \end{array}\right]^T \tag{40}$$

where

$$\boldsymbol{\varphi} = \left[\begin{array}{ccc} \boldsymbol{\varphi}_1^T & \cdots & \boldsymbol{\varphi}_P^T \end{array}\right]^T \tag{41}$$

$$\boldsymbol{\varphi}_p = \left[\begin{array}{ccc} \varphi_{p_1}^T & \cdots & \varphi_{p_{L_{\max}}} \end{array}\right]^T \tag{42}$$

with φ_{p_ℓ} denoting the phase of the ℓth harmonic of the pth pitch. With this formulation the phases are incorporated into the dictionary, thus rendering the amplitudes real valued.

Together with the discussed algorithmic considerations, the presented time-recursive multi-pitch estimator is detailed in Algorithm 1. The algorithm is termed the Pitch Estimation using dictionary-Adaptive Recursive Least Squares (PEARLS) method[2].

[2]An implementation in MATLAB may be found at http://www.maths.lu.se/staff/andreas-jakobsson/publications/

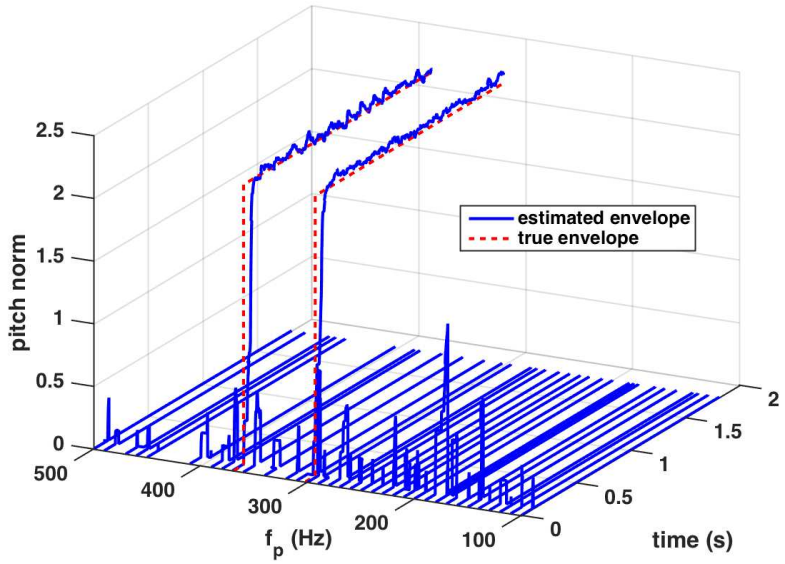

Figure 1: Pitch frequency and pitch norm estimates, i.e., estimates of $f_p(t)$ and $\left|\left|\breve{\mathbf{w}}_{\mathcal{G}_p}(t)\right|\right|_2$ as produced by PEARLS when applied to a simulated two-pitch signal with fundamental frequencies 302 and 369 Hz, respectively, deviating from the original dictionary grid points by 2 and 1 Hz respectively.

6 Numerical results

In this section, we evaluate the performance of the proposed PEARLS algorithm using both simulated signals and real audio recordings.

6.1 Simulated signals

To demonstrate the effect of the smoothing parameter, ξ, as well as the ability of PEARLS to smoothly track the amplitudes of pitches, we first consider an illustrative example with a two-pitch signal. Figure 1 shows the time evolution of the pitch frequency and pitch norm estimates, i.e., estimates of $f_p(t)$ and $\left|\left|\breve{\mathbf{w}}_{\mathcal{G}_p}(t)\right|\right|_2$, as produced by PEARLS when applied to a two-pitch signal with fundamental frequencies 302 and 369 Hz, respectively, where both pitches are constituted

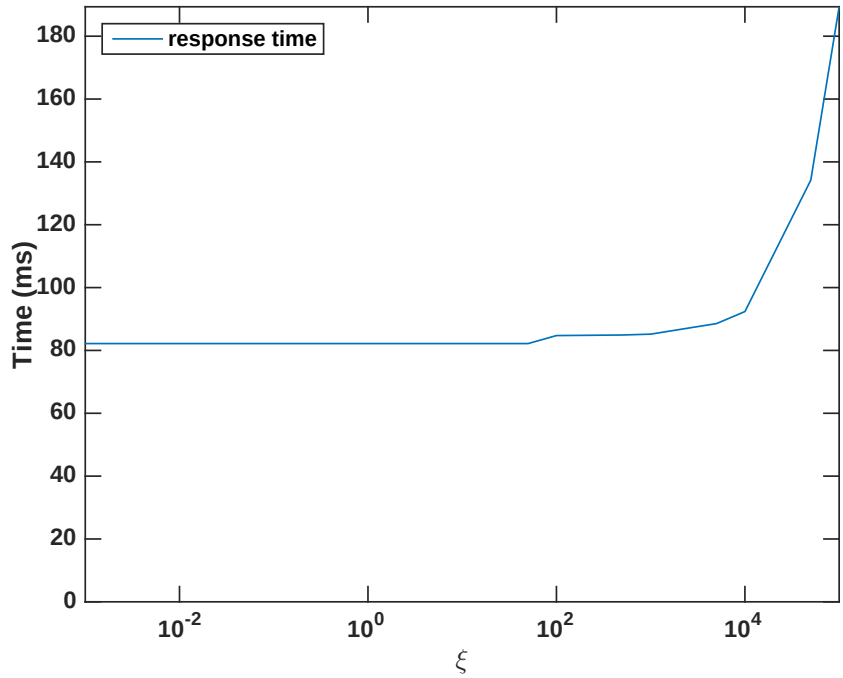

Figure 2: Response time for different values of the smoothing parameter ξ.

by 5 harmonics each. Both pitches enter the signal after 90 ms, reaching their maximum amplitudes momentarily and keeping them for the rest of the signal duration. The signal was sampled at 11 kHz. The settings for PEARLS was $L_{\max} = 10$, $\lambda = 0.995$, and the smoothing parameter was $\xi = 10^4$. The original pitch frequency grid was chosen so that the true pitch frequencies deviated from the closest grid points by 2 and 1 Hz, respectively. As can be seen from the figure, the estimate initially, before the pitch signals appear, contains several spurious pitch estimates, but then quickly finds the pitch signals when these appear in the data. At this point, the spurious peaks are suppressed and the estimates are seen to well follow the true pitch envelopes. It is worth noting that both the response time and the steady state variance of the estimates will be influenced by the choice of the smoothing parameter, ξ. Figures 2 and 3 illustrate this effect by considering the response time, defined as the time required for the PEARLS amplitude estimate to reach 95% of its peak value, and the steady state amplitude variance,

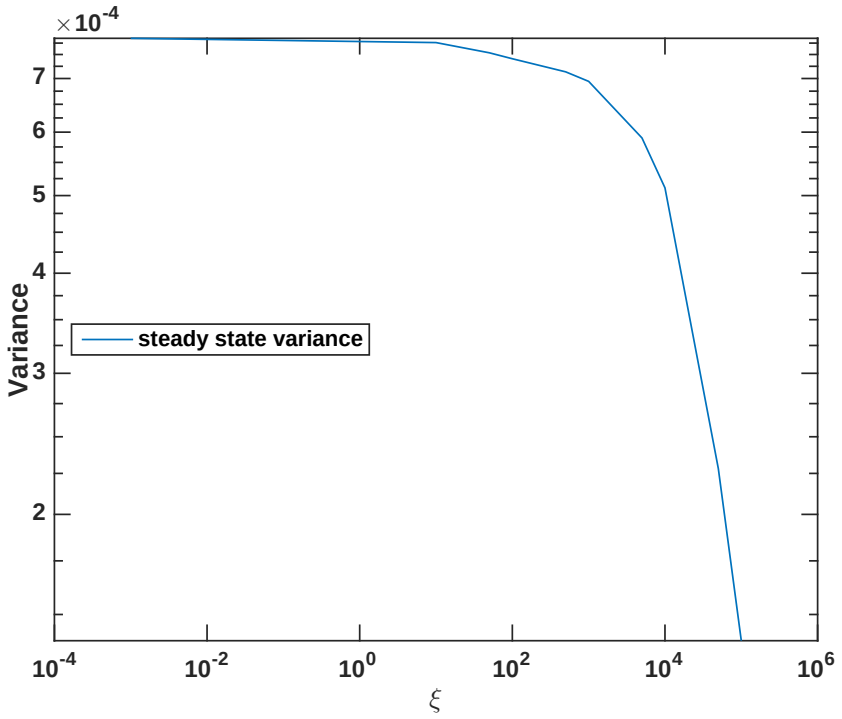

Figure 3: Steady state variance of the pitch norm estimate for different values of the smoothing parameter ξ.

respectively. The signal considered is the same as in Figure 1. As can be seen from the figures, a higher value of ξ implies a longer response time for PEARLS, while at the same time promoting a more smooth pitch norm trajectory, just as could be expected.

The PEARLS algorithm is not restricted to form estimates of stationary pitches; it is also able to cope with amplitude and frequency modulated signals. In Figure 4, PEARLS has been applied to a two-pitch signal with fundamental frequencies that oscillate according to sine waves with frequencies 2 and 3 Hz on the intervals 327 ± 2 Hz and 394 ± 3 Hz, respectively. Also, the pitch norms are not constant, but are amplitude modulated according to a Hamming window. As can be seen, PEARLS is able to track the two pitches smoothly both in frequency and in pitch norm. Here, the pitches consisted of 5 and 7 harmonics, respect-

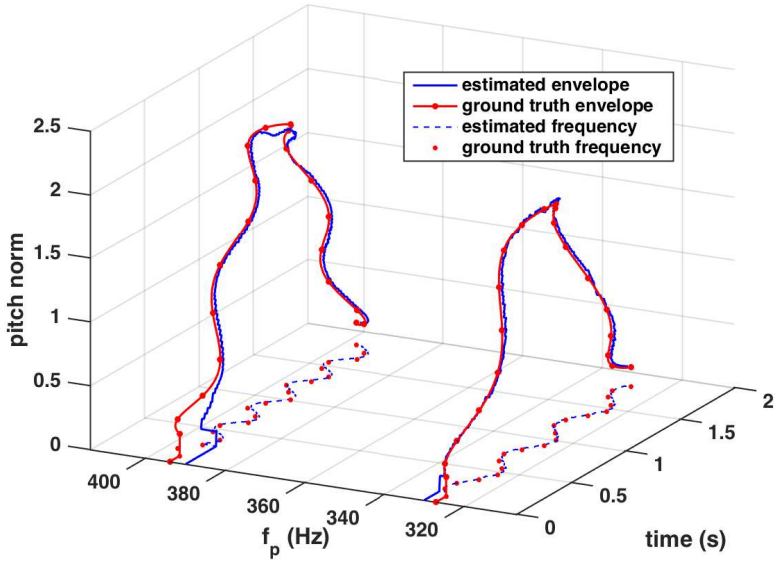

Figure 4: Pitch frequency and pitch norm estimates, i.e., estimates of $f_p(t)$ and $\left|\left|\breve{\mathbf{w}}_{\mathcal{G}_p}(t)\right|\right|_2$, as produced by PEARLS when applied to a simulated two-pitch signal with fundamental frequencies that oscillate according to sine waves.

ively. The signal was sampled at 11 kHz, with PEARLS using the same settings as above. As comparison, Figure 5 presents a corresponding plot for the multi-pitch estimator ESACF [7], using recommended settings. As ESACF only estimates pitch frequencies, pitch norm estimates have been obtained using least squares, assuming known harmonic orders. ESACF is a frame based estimator and the signal was therefore here subdivided into 30ms windows. As can be seen, the ESACF estimates deviate from the true pitch frequencies, causing the amplitude estimates to degrade. Figure 6 demonstrates the usefulness of using the dictionary learning procedure. In this figure, PEARLS is again applied to the signal with two frequency modulated pitches, but this time the dictionary learning scheme is excluded from Algorithm 1. As can be seen in the figure, PEARLS is still able to estimate the frequency content, as well as the pitch norms, but the tracking is now performed by different elements of $\breve{\mathbf{w}}(t)$, as the frequency modulation causes

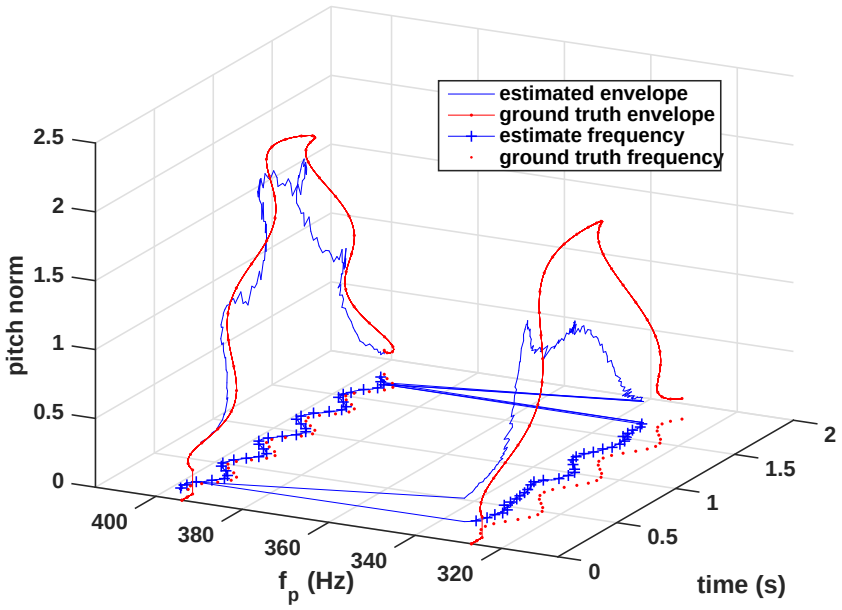

Figure 5: Pitch frequency, i.e., estimates of $f_p(t)$, as produced by ESACF when applied to a simulated two-pitch signal with fundamental frequencies that oscillate according to sine waves. The pitch norms, i.e., $\left|\left|\breve{\mathbf{w}}_{\mathcal{G}_p}(t)\right|\right|_2$, have been estimated by applying least squares to the ESACF pitch frequency estimates using oracle harmonic orders.

the different candidate pitches to become activated and then deactivated, with the activation-deactivation cycles following the periods of the frequency modulation. Also, there is some power-sharing between adjacent pitch groups of $\breve{\mathbf{w}}(t)$ at time points where the frequency modulating sinusoids change sign. In contrast, the dictionary learning scheme allows for a much smoother tracking as the movable dictionary elements counters the activation-deactivation phenomenon, which can be observed in Figure 4.

6.2 Real audio

We proceed to evaluate the performance of PEARLS on the Bach10 dataset [49]. This dataset consists of ten excerpts from chorals composed by J. S. Bach, and

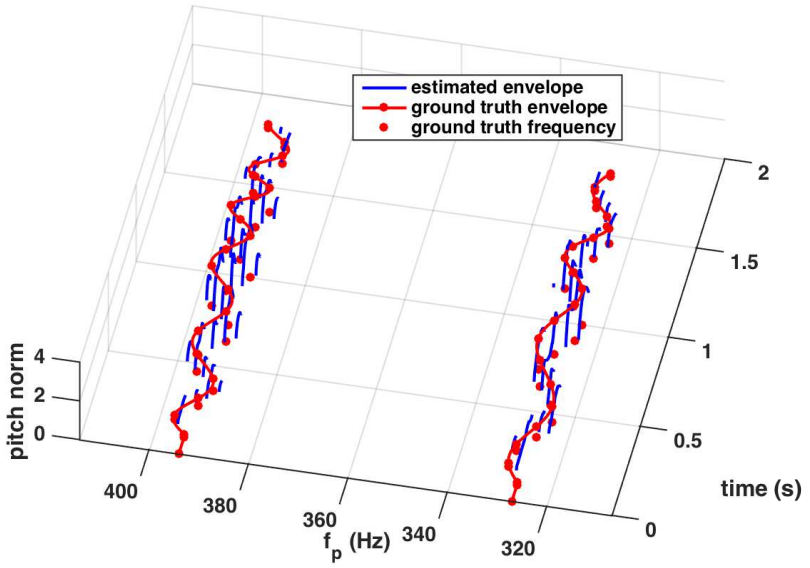

Figure 6: Pitch frequency and pitch norm estimates, i.e., estimates of $f_p(t)$ and $\left|\left|\breve{\mathbf{w}}_{\mathcal{G}_p}(t)\right|\right|_2$, as produced by PEARLS when applied to a simulated two-pitch signal with fundamental frequencies that oscillate according to sine waves. Here, the dictionary learning scheme is excluded from Algorithm 1.

have been arranged to be performed by an ensemble consisting of a violin, a clarinet, a saxophone, and a bassoon, with each excerpt being 25-42 seconds long. The algorithm settings for PEARLS were $\lambda = 0.985$, $\xi = 10^3$, $L_{\max} = 6$, and the dictionary was updated every 10 ms using 45 ms of past signal samples. Each music piece, originally sampled at 44.1 kHz, was down-sampled to 11.025 kHz. The PEARLS estimates were compared to ground truth values with a time-resolution of one reference point every 30 ms. The ground truth fundamental frequencies were obtained by applying the single-pitch estimator YIN [50] to each separate channel with manual correction of obvious errors. The results are presented in Table 1, presenting values of the performance measures *Accuracy*, *Precision*, and *Recall*, as defined in [51]. As in [51], an estimated fundamental frequency is associated with a ground truth fundamental frequency if it lies within a quarter-tone,

	PEARLS	PEBSI-Lite	BW15	ESACF
Accuracy	0.437	0.449	0.515	0.269
Precision	0.683	0.631	0.684	0.471
Recall	0.548	0.609	0.675	0.386

Table 1: Performance measures for the PEARLS, PEBSI-Lite, BW15, and ESACF algorithms, when evaluated on the Bach10 dataset.

or 3%, of the ground truth fundamental frequency. For comparison, Table 1 also includes corresponding performance measures for the PEBSI-Lite [9] and ESACF algorithms. The values for PEBSI-Lite and ESACF were originally presented in [9], and the settings for these algorithms are the same as is presented there. Also presented in Table 1 are performance measures obtained when applying the method presented in [35], hereafter referred to as BW15, after the authors and year of publication, to the same dataset. Being trained on databases of music instrument, this method uses probabilistic latent component analysis to produce pitch estimates and is specifically tailored to estimate pitches in music signals. The frequency resolution of the obtained estimates corresponds to that of the Western chromatic scale, i.e., to the keys of the piano.

As can be seen, PEARLS clearly outperforms ESACF and performs on par with PEBSI-Lite when considering these measures, although it should be stressed that PEARLS has significantly lower computational complexity than PEBSI-Lite. The BW15 methods performs better than the other presented methods, including PEARLS, for this dataset. This is as the performance of the BW15 estimate was formed when using an *a posteriori* thresholding of the obtained estimate, optimally selecting the threshold level as to maximize the performance measures; this in order to illustrate the best possible performance achievable for BW15. However, several other choices of possible threshold levels resulted in BW15 performing worse than both PEARLS and PEBSI-Lite. Furthermore, the BW15 estimator is sensitive to mismatches between the examined signal and the training dataset used to construct its priors. This is illustrated by applying the BW15 and PEARLS estimators to a signal consisting of two (harmonic) trumpet notes and two (inharmonic) piano notes. The trumpets are playing the notes A4 and D♭5, corresponding to the fundamental frequencies 440 and 554.37 Hz, whereas the pianos are playing the notes E4 and G♯4, corresponding to the fundamental frequencies

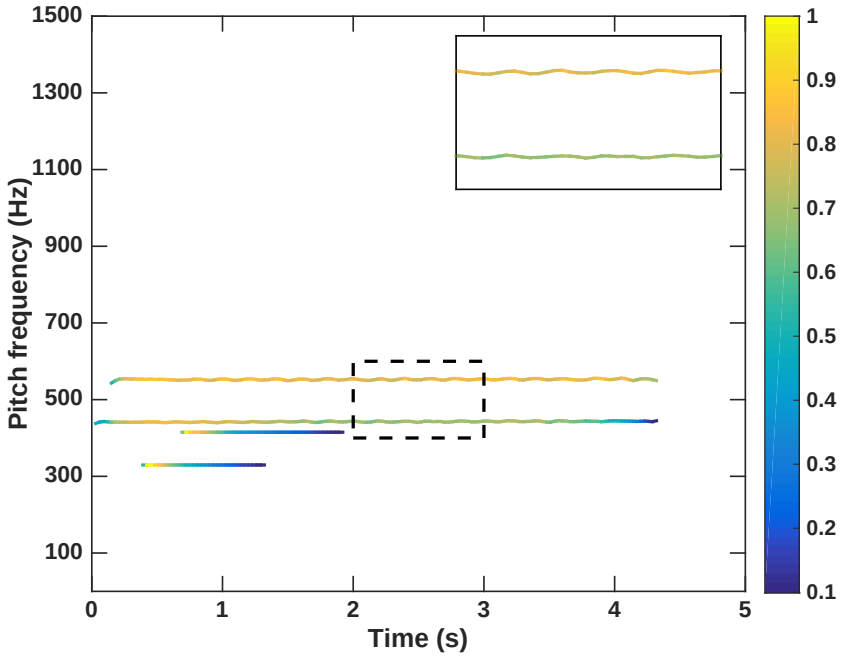

Figure 7: Ground truth for a signal consisting of two trumpets and two pianos. The amplitude of each pitch, i.e., the pitch norm, is illustrated by the color of each track. The amplitudes have been normalized so that the maximal amplitude is 1.

329.65 and 415.3 Hz. The signal was sampled at 11.025 kHz. The ground truth pitches can be seen in Figure 7. Here, the amplitude, i.e., the pitch norm, of each pitch is illustrated by the color of each track. The amplitude has been normalized so that the maximum amplitude is equal to one. The corresponding estimates produced by PEARLS (using the same settings as for the Bach10 dataset) and BW15 are presented in Figures 8 and 9, respectively.

As can be seen from Figure 8, PEARLS is able to correctly identify both the trumpet and the piano pitches, despite the pianos being inharmonic and thereby differing from the assumed signal model, as given in (2). Note that PEARLS is also able to smoothly track the frequency modulation caused by that trumpets are playing with vibrato, which can be more clearly seen from the zoomed-in portions of Figures 7 and 8. In contrast, as seen in Figure 9, BW15 is able to

correctly identify the piano pitches (note that pianos were included in the training dataset used by the authors of [35]), but instead of identifying the sinusoidal content corresponding to the trumpets (which are not in the training dataset) as originating from only two pitches, several of the individual harmonics are instead being assigned individual pitches.

It may be noted that the method does not accurately represent the vibratos; this as the estimates of BW15 are restricted to correspond to the keys of the piano. It should further be noted that the pitches indicated as being the most significant by BW15 are not those corresponding to the true fundamental frequencies, but instead higher order harmonics. This problem is arguably due to the mismatch between the content of the signal and the database used to train the method. Thus, for this example, it is not possible to recover the true pitches by thresholding the solution of BW15, as the thresholding would eliminate true pitch candidates before getting rid of the erroneous ones. Although the estimates produced by BW15 could arguably be improved by extending its training data to also include trumpets, this example illustrates that basing estimation on exploiting the features of a signal model, as PEARLS does, can be beneficial in terms of the generality of the estimator, even in the face of slight deviations from the assumed signal model, which in this case takes the form of inharmonicity for the pianos. It can be noted that an interesting future development would be to combine the benefits from training a hidden Markov model, as is done in BW15, with the more robust approach in PEARLS.

Another recent method that would be of interest to consider in this respect would be the one presented in [21], which also exhibits some conceptual similarities with the herein presented algorithm. Notably, the sparsifying role played by the ℓ_1-norm herein is in [21] formed by instead determining the significant spectral peaks using an estimate of the noise floor. The pitch selection, herein formed using the group-wise ℓ_2-norm, is in [21] made by matching spectral content with that of components in a large training data set, which is also used to measure the power concentration for low-order harmonics, as well as a synchronicity measure. The relative weighting of these components is selected using training data. Using a greedy approach, the method in [21] then iteratively adds candidate pitches to the estimate; the power allocation between pitches that have overlapping harmonics is resolved using an interpolation scheme utilizing the power of harmonics unique to each candidate pitch. In contrast, the number of active pitches is herein decided by the optimal point of (6), where candidate pitches not contained in the signal

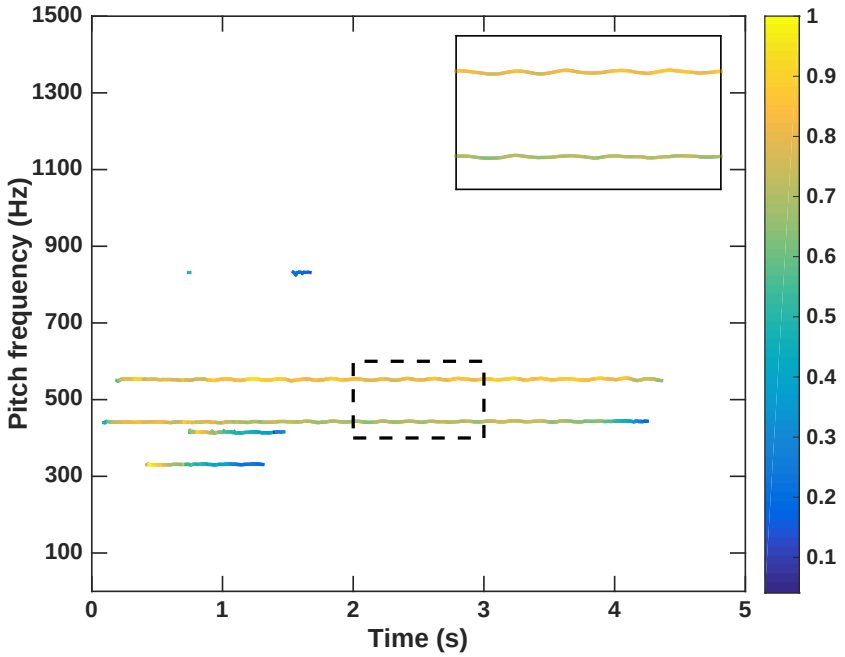

Figure 8: Estimates produced by PEARLS when applied to a signal with two trumpets as well as two pianos. The amplitude of each pitch, i.e., the pitch norm, is illustrated by the color of each track. The amplitudes have been normalized so that the maximal amplitude is 1.

should be assigned zero power. It can also be noted that the optimization problem presented here does not favor spectral smoothness; rather, the ℓ_2-norm will favor collecting as much power as possible into a few candidate pitches. The power of overlapping harmonics will therefore tend to be allocated to pitches with more prominent unique harmonics. Using a MATLAB implementation of PEARLS on a 2.68 GHz PC, the average running time for the Bach pieces was 20 minutes. The Bach pieces were on average 33 seconds long[3]. For PEBSI-Lite, the average running time was 54 minutes, with the signal being divided into non-overlapping frames of length 30 ms.

[3]We note that the current implementation has not exploited that the filter updating step (17) can be done for all P candidate pitches in parallel. Similarly, the computations for PEBSI-Lite can also be parallelized, as each time frame can be processed in isolation.

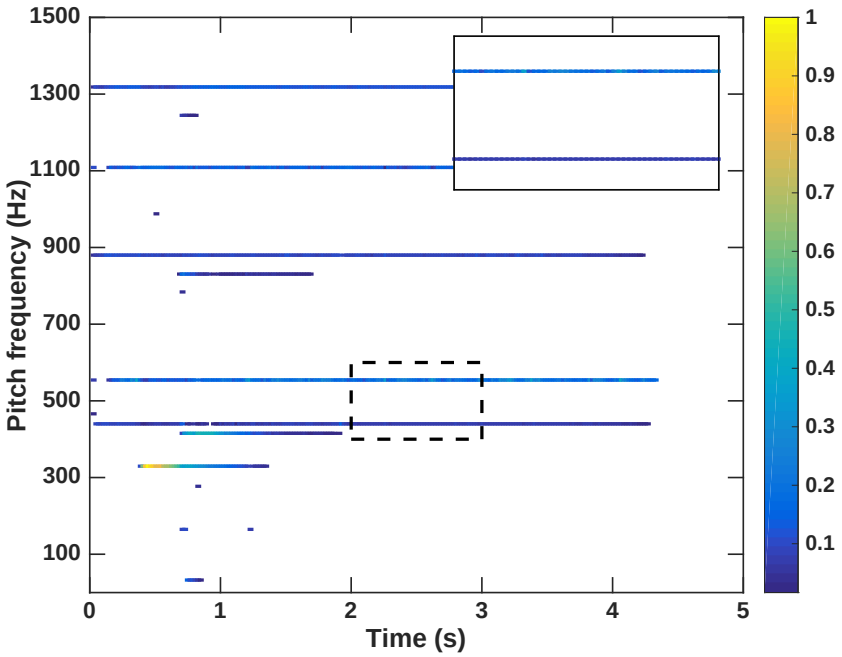

Figure 9: Estimates produced by BW15 when applied to a signal with two trumpets as well as two pianos. The magnitudes of the estimates are illustrated by the color of the pitch tracks. The magnitudes have been normalised so that the maximal magnitude is 1.

As an illustration of the performance of PEARLS on the Bach10 dataset, Figures 10 and 11 present the estimated fundamental frequencies obtained using ESACF and PEARLS, respectively, for the piece *Ach, Gott und Herr*, as compared to the ground truth for each instrument. Here, in order to make a fair comparison of the computational complexities of the estimators, the ESACF estimate was computed on windows of length 30 ms, where two consecutive windows overlapped in all but one sample. Although ESACF can arguably be applied to windows with smaller overlap, this setup meant that ESACF would produce pitch tracks with the same time resolution as PEARLS. This resulted in an average running time of 11 minutes per music piece, that is, about half that of PEARLS. As can be seen from the figures, PEARLS is considerably better at tracking the instruments than ESACF. In Figure 12, the corresponding results for BW15 are shown. The figure

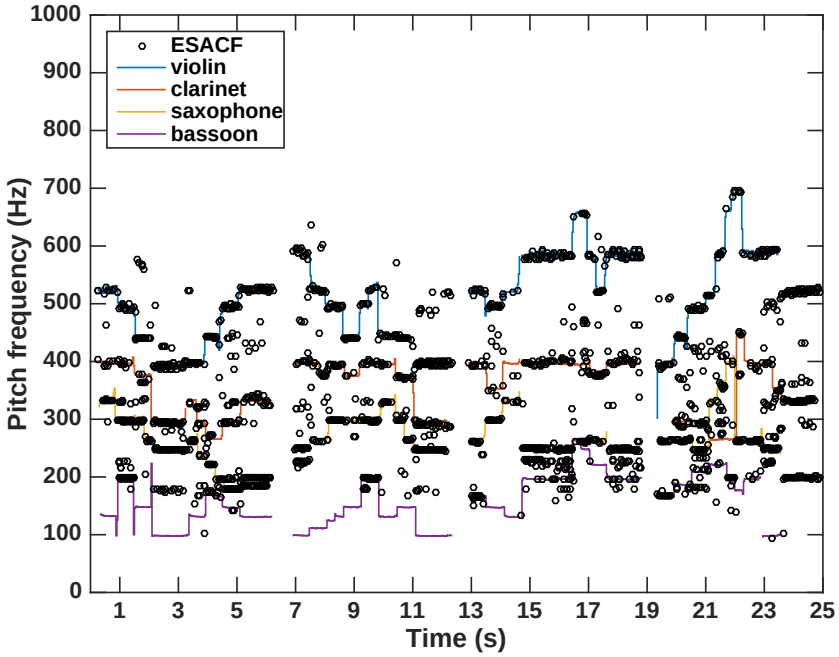

Figure 10: Pitch tracks produced by ESACF when applied to a 25 seconds excerpt of J. S. Bach's *Ach, Gott und Herr* performed by a violin, a clarinet, a saxophone, and a bassoon.

has been truncated at 1000 Hz to simplify inspection, although pitch estimates with fundamental frequencies higher than 1000 Hz did occur repeatedly. From the figure, it is clear that BW15 is better able to track the bassoon (which is included in the method's training data) than either PEARLS or ESACF. It can also be noted that the discrete nature of the BW15 estimator prevents it from tracking smaller frequency variations, such as vibratos.

7 Conclusions

In this work, we have presented a time-recursive multi-pitch estimation algorithm, based on a both sparse and group-sparse reconstruction technique. The method has been shown to be able to accurately track multiple pitches over time, in fundamental frequency as well as in amplitude, without requiring prior knowledge

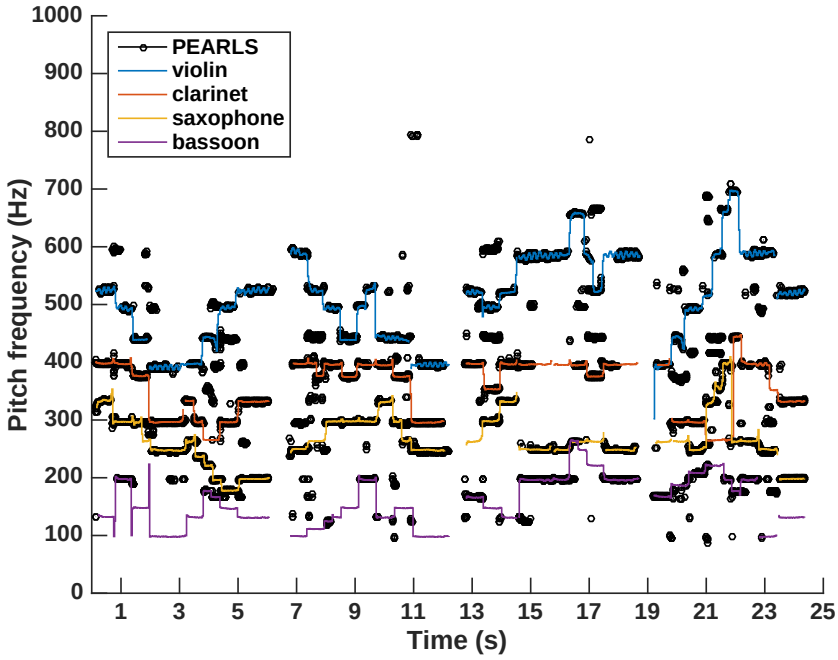

Figure 11: Pitch tracks produced by PEARLS when applied to a 25 seconds excerpt of J. S. Bach's *Ach, Gott und Herr* performed by a violin, a clarinet, a saxophone, and a bassoon.

of the number of pitches nor the number of harmonics present in the signal. Furthermore, we have presented a scheme for adaptively changing the signal dictionary, thereby providing robustness against grid mismatch, as well as allowing for smooth tracking of frequency modulated signals. We have shown that the proposed method yields accurate results when applied to real data, outperforming other general purpose multi-pitch estimators in either estimation accuracy and/or computational speed. The method has further been shown to be robust to deviations from the assumed signal model, although it is not able to yield performance as good as that achievable by a state-of-the art method being optimally tuned and specifically trained on the present instruments. However, the method is able to outperform such a technique when used without optimal tuning, or when applied to instruments not included in the training data.

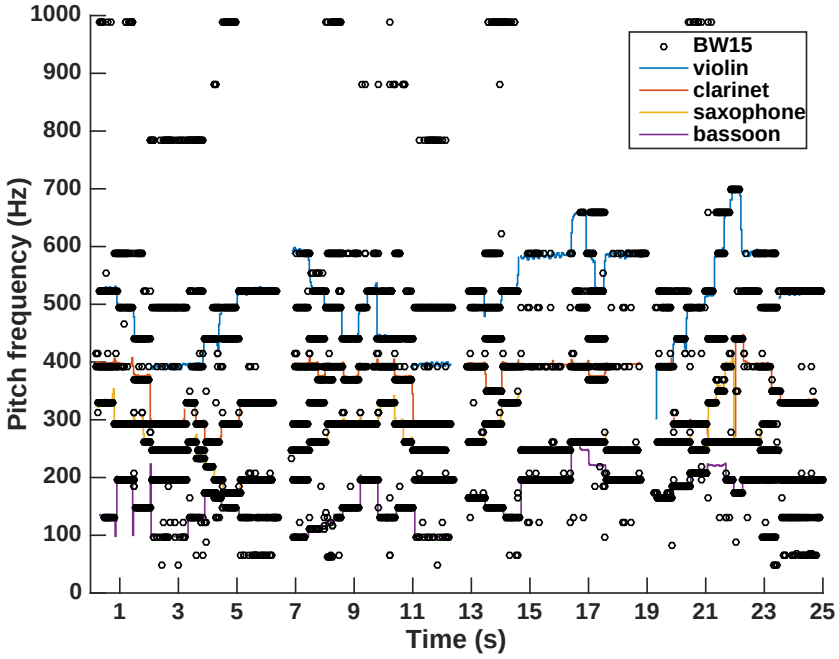

Figure 12: Pitch tracks produced by BW15 when applied to a 25 seconds excerpt of J. S. Bach's *Ach, Gott und Herr* performed by a violin, a clarinet, a saxophone, and a bassoon.

References

[1] A. Klapuri and M. Davy, *Signal Processing Methods for Music Transcription*, Springer, 2006.

[2] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[3] M. Müller, *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*, Springer International Publishing, 2015.

[4] R. B. Randall, *Vibration-Based Condition Monitoring: Industrial, Aerospace and Automotive Applications*, John Wiley & Sons, Chichester, UK, 2011.

[5] M. A. Little, P. E. McSharry, E. J. Hunter, J. Spielman, and L. O. Ramig, "Suitability of Dysphonia Measurements for Telemonitoring of Parkinson's disease," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1015–102, April 2009.

[6] P. Somervuo, A. Harma, and S. Fagerlund, "Parametric Representations of Bird Sounds for Automatic Species Recognition," vol. 14, no. 6, pp. 2252–2263, November 2006.

[7] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 708–716, 2000.

[8] 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.

[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] 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.

[11] 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.

[12] A. Dessein, A. Cont, and G. Lemaitre, "Real-Time Polyphonic Music Transcription With Non-Negative Matrix Factorisation and Beta-Divergence," in *Proceedings of the 11th International Society for Music Information Retrieval Conference*, Utrecht, NL, August 2010, pp. 489–494.

[13] V. Emiya, R. Badeau, and B. David, "Multipitch Estimation of Piano Sounds Using a New Probabilistic Spectral Smoothness Principle," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 18, no. 6, pp. 1643–1654, August 2010.

[14] 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.

[15] A. Cogliati, Z. Duan, and B. Wohlberg, "Piano Music Transcription with Fast Convolutional Sparse Coding," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing*, Boston, MA, Sept 2015, pp. 1–6.

[16] 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.

[17] E. Vincent, N. Bertin, and R. Badeau, "Adaptive Harmonic Spectral Decomposition for Multiple Pitch Estimation," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 18, no. 3, pp. 528–537, March 2010.

[18] 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.

[19] 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.

[20] 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.

[21] C. Yeh, A. Roebel, and X. Rodet, "Multiple Fundamental Frequency Estimation and Polyphony Inference of Polyphonic Music Signals," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 18, no. 6, pp. 1116–1126, August 2010.

[22] G. Zhang and S. Godsill, "Tracking Pitch Period Using Particle Filters," in *IEEE Workhop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, October 2013.

[23] K. Han and D. Wang, "Neural Networks For Supervised Pitch Tracking in Noise," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014.

[24] H. Hajimolahoseini, R. Amirfattahi, S. Gazor, and H. Soltanian-Zadeh, "Robust Estimation and Tracking of Pitch Period Using an Efficient Bayesian Filter," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1219–1229, July 2016.

[25] 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.

[26] 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.

[27] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The Sparse RLS Algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4013–4025, August 2010.

[28] N. Vaswani and J. Zhan, "Recursive Rrecovery of Sparse Signal Sequences From Compressive Measurements: A Reveiew," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3523–3549, July 2016.

[29] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online Sparse System Identification and Signal Reconstruction Using Projections Onto Weighted ℓ_1 Balls," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 936–952, March 2011.

[30] E. C. Hall and R. M. Willett, "Online Convex Optimization in Dynamic Environments," *IEEE J. Sel. Topics in Signal Processing*, vol. 9, no. 4, pp. 647–662, June 2015.

[31] Y. Chen and A. O. Hero, "Recursive $\ell_{1,\infty}$ Group Lasso," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3978–3987, Aug. 2012.

[32] E. Eksioglu, "Group sparse RLS algorithms," *International Journal of Adaptive Control and Signal Processing*, vol. 28, pp. 1398–1412, 2014.

[33] S. Jiang and Y. Gu, "Block-Sparsity-Induced Adaptive Filter for Multi-Clustering System Identification," *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5318–5330, October 2015.

[34] B. Fuentes, R. Badeau, and G. Richard, "Harmonic Adaptive Latent Component Analysis of Audio and Application to Music Transcription," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 21, no. 9, pp. 1854–1866, September 2013.

[35] E. Benetos and T. Weyde, "An Efficient Temporally-Constrained Probabilistic Model for Multiple-Instrument Music Transcription," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, Malaga, Spain, October 2015.

[36] 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.

[37] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

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

[39] S. Haykin, *Adaptive Filter Theory (4th edition)*, Prentice Hall, Inc., Englewood Cliffs, N.J., 2002.

[40] 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.

[41] 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.

[42] M. Kowalski, "Sparse Regression Using Mixed Norms," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 303 – 324, 2009.

[43] P. Combettes and V. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM Jour. Multiscale Modeling & Simulation*, vol. 4, pp. 1168–1200, 2005.

[44] M. A. T. Figueiredo and R. D. Nowak, "An EM Algorithm for Wavelet-Based Image Restoration," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 906–916, August 2003.

[45] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 15, no. 3, pp. 1066–1074, March 2007.

[46] 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.

[47] 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.

[48] 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.

[49] Z. Duan and B. Pardo, "Bach10 dataset," http://music.cs.northwestern.edu/data/Bach10.html, Accessed December 2015.

[50] 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.

[51] 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.

F

Paper F

Off-grid Fundamental Frequency Estimation

Johan Swärd[1], Hongbin Li[2], and Andreas Jakobsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2] *Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, New Jersey, USA*

Abstract

In this paper, we propose an off-grid method for estimating an unknown number of fundamental frequencies. Starting with a conventional dictionary matrix, containing sets of candidate fundamental frequencies and their corresponding harmonics, a non-convex log-sum cost function is formed such that it imposes the harmonic structure and treats every fundamental frequency in the dictionary as a parameter. The cost function is then iteratively decreased by minimizing a surrogate function, and, in each iteration, the fundamental frequencies are refined, whereas redundant parameters are omitted from the dictionary. The proposed method is tested on both real and simulated data, showing its preferred performance as compared to other state-of-the-art multi-pitch estimators.

Key words: Group sparsity, multi-pitch estimation, dictionary learning, off-grid estimation

1 Introduction

In areas such as audio, biomedicine, and mechanics, the estimation of fundamental frequencies is often of central importance. In particular, the multi-pitch problem is challenging, as one needs to determine not only the number of fundamental frequencies, but also the number of harmonics related to each fundamental frequency. This problem has historically been addressed by utilizing

various forms of model order estimators, or by simply assuming the model order is already known *a-priori* [1–4]. Early pitch estimation methods relied on covariance-based methods as the ones presented in [5, 6]. Later, filterbank- and subspace-based methods were introduced and MUSIC-like methods were widely used [7–12]. Recent contributions include, e.g., [13], where the computational speed is in focus, and [14] where the problem is to estimate the fundamental frequencies in noisy environments when multiple people speak at the same time. In [15], the Pitch Estimation using ℓ_2 norm and Block Sparsity (PEBS) algorithm was presented, where the fundamental frequency estimation problem was instead solved by using a (block-)sparsity approach, thereby combining the model order estimation with the overall estimation of the fundamental frequencies and their harmonics. Based on the promising performance of the initial PEBS algorithm, several improvements have been suggested, including focusing on the choice of hyperparameters [16], time-updating [17], and computation complexity [18]. The results presented in these works illustrate the benefits of using a sparse framework for solving multi-pitch estimation problems.

Sparse reconstruction methods are used in a vast number of areas and has been intensively studied (see, e.g., [19–24]). As in the case of PEBS, the resulting sparse problems have often been expressed using dictionary matrices, containing a large quantity of possible signal candidates, with the assumption that only a small subset of these candidates is needed to approximate the signal well. These candidates are often selected on a pre-defined grid that spans the parameter space of interest. Recently, some concerns have been raised as to how this grid-based selection of candidates affects the performance. In [25], it was shown that since the grid and the true parameters are unlikely to coincide, this may cause the estimation to deteriorate. If one, in an effort to circumvent this, increases the number of grid points to decrease the distance between the grid and the true parameters, the dictionary matrix will become increasingly coherent, i.e., the columns of the dictionary matrix become correlated, which may in turn degrade the performance, and increase the computational complexity of the algorithm. To counter these drawbacks, it has recently been suggested that one may instead solve the sparse problem without applying a grid, using so-called gridless methods. One noticeable example of this is the use of the atomic norm [26–30], where the sparse problem is instead formulated as a convex semi-definite program (SDP). The use of the atomic norm can be seen as solving the sparse problem using an infinite grid, but without the problem of a resulting coherent dictionary matrix.

Unfortunately, the atomic norm formulation does not easily allow for imposing general data structures to the cost function, and, typically, any additional model constraints will fundamentally change the problem formulation. This is in contrast to the grid-based approaches, where such model structures could easily be accounted for by adding different constraints to the cost function.

In this paper, we aim to combine the benefits of the off-grid methods with the use of a cost function that easily allows for adding structure to the signal of interest. To this effect, we will expand on the PEBS formulation and introduce a method for solving problems involving group sparsity with sparse groups based on the super-resolution iterative reweighted (SURE-IR) method [31]. We then proceed to adress both the computational complexity issue as well as the appropriate choice of hyperparameters for the introduced estimator. Using both simulated and real audio data, we illustrate the preferable performance of the introduced estimator, comparing to several earlier alternative formulations. For the real data case, we test the proposed method using the Bach10 data set, containing 10 musical pieces composed by Johann Sebastian Bach, showing that the proposed method achieves similar performance as state-of-the-art music transcription methods, although without the need of *any* training data, as is typically utilized by such methods.

It should be noted that the proposed method is not limited to audio problems, although this is here the main focus. Indeed, due to the possibility of adding new constraints to the cost function, the technique may likely be extended to find use in other related fields, such as studies of mechanical vibrations (see, e.g., [32–35]).

2 Signal model and earlier work

Consider the multi-pitch signal model[1]

$$y(n) = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} \alpha_{k,\ell} e^{2i\pi f_k \ell t_n} + \varepsilon(n) \tag{1}$$

where f_k denotes the kth fundamental frequency (also denoted pitch), $\alpha_{k,\ell}$ the complex amplitude corresponding to the ℓth overtone of the kth fundamental frequency, t_n, for $n = 1, \ldots, N$, the nth time point, and $\varepsilon(n)$ any non-tonal

[1] For notational and computational simplicity, we here consider the discrete-time analytic signal of the (real-valued) measured signal.

audio or noise component, here, for simplicity, being modelled as a complex-valued white Gaussian noise (see also [36]). Often, the problem of interest is that of estimating f_k for $k = 1, \ldots, K$. If this set is known, as well as the number of overtones for each pitch, L_k, the corresponding amplitudes of the overtones may be formed, for instance, using least squares (LS).

Typically, it is non-trivial to determine the required model orders; for simplicity, we will initially consider the problem of only estimating K sinusoids in noise. This corresponds to the case where $L_k = 1$ for all k. To form an efficient estimator, one may then include the model order estimation into the estimation of the frequencies, for instance by forming the sparse optimization problem (see also [37])

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

where $\mathbf{A}$ is a dictionary matrix, $\mathbf{z}$ a vector containing the complex amplitudes, λ is a hyperparameter that controls the amount of sparsity in the solution, and

$$\mathbf{y} = \begin{bmatrix} y(1) & \ldots & y(n) \end{bmatrix}^T \tag{3}$$

Usually, the dictionary, $\mathbf{A}$, is an $N \times M$ matrix containing $M \gg N$ signal candidates (in this case sinusoids).

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \ldots & \mathbf{a}_M \end{bmatrix} \tag{4}$$

where $\mathbf{a}_k = \begin{bmatrix} \mathrm{e}^{2i\pi f_k t_1} & \ldots & \mathrm{e}^{2i\pi f_k t_N} \end{bmatrix}^T$.

The first part of (2) is thus a data-fitting term, whereas the second term is a sparsity enhancing term, penalizing the magnitude of $\mathbf{z}$, thus promoting a sparse solution, containing only a few signal candidates. This methodology is widely used in signal processing and has been popular for many years (see, e.g., [19]). However, it has in recent times been argued that using a pre-defined grid may cause the estimation to deteriorate, mainly because of the fact that the true parameter value will typically not exactly coincide with any of the grid points. Trying to increase the grid size, in an effort to minimize the distance from the grid points to the true values, may further harm the estimation as the dictionary matrix then becomes more coherent. To address this issue, a gridless method based on the use of the atomic norm was proposed in [27]. Instead of solving a problem based on a

dictionary matrix, the authors proposed the gridless formulation (for the noiseless case)

$$\begin{aligned} &\underset{x,\mathbf{u}}{\text{minimize}} \ \frac{1}{2}(x+u_1) \\ &\text{subject to} \ \begin{bmatrix} T(\mathbf{u}) & \mathbf{y}^H \\ \mathbf{y} & x \end{bmatrix} \geq 0 \end{aligned} \tag{5}$$

where $T(\mathbf{u})$ forms a Hermitian Toeplitz matrix with the vector $\mathbf{u}$ on its first row, and where u_1 denotes the first element in $\mathbf{u}$. The corresponding frequencies are then obtained using a Vandermonde decomposition of $T(\mathbf{u}^*)$, where $\mathbf{u}^*$ denotes the value of $\mathbf{u}$ at the solution of (5). The atomic norm enjoys many benefits (for a more detailed discussion on the topic, see, e.g., [26–30]), but it is generally hard to generalize the method to accommodate for other model restrictions, such as block sparsity or, e.g., spectral smoothness [38]. Furthermore, if needed, it is not clear how one may impose any of the assumed signal structures when retrieving the frequency estimates using, e.g., the Vandermonde decomposition. As an alternative, another gridless approach was suggested in [31], which was based on the formulation of a non-convex optimization problem. The proposed problem utilized a logarithmic penalty to enforce sparsity, such that

$$\underset{\mathbf{z},\boldsymbol{\vartheta}}{\text{minimize}} \ ||\mathbf{y} - \mathbf{A}(\boldsymbol{\vartheta})\mathbf{z}||_2^2 + \lambda \sum_{m=1}^{M} \log\left(|z_m|^2 + \eta\right) \tag{6}$$

where $\eta > 0$ is a parameter ensuring that the function is not evaluated at zero, and z_m denotes the mth element of $\mathbf{z}$. It should be noted that the dictionary matrix is now parameterized over the parameter vector $\boldsymbol{\vartheta}$, containing the sought fundamental frequencies. Thus, instead of using a fixed grid, the grid points are selected as to minimize the cost function in (6). Using a logarithmic penalty will enhance the sparsity, but, at the same time, render the problem non-convex. To solve the problem, a majorization-minimization (MM) approach was proposed in [31] and the optimization problem was reformulated using a surrogate function, thus yielding a simplified version of the original problem. This allows the problem to be solved in closed-form for the amplitudes, as a function of $\boldsymbol{\vartheta}$, such that

$$\mathbf{z}^*(\boldsymbol{\vartheta}) = \left(\mathbf{A}^H(\boldsymbol{\vartheta})\mathbf{A}(\boldsymbol{\vartheta}) + \lambda \mathbf{D}^{(i)}\right)^{-1} \mathbf{A}^H(\boldsymbol{\vartheta})\mathbf{y} \tag{7}$$

where

$$\mathbf{D}^{(i)} = \text{diag}\left(\frac{1}{|z_1^{(i)}|^2 + \eta}, \ldots, \frac{1}{|z_M^{(i)}|^2 + \eta}\right) \tag{8}$$

with $z_m^{(i)}$ denoting the mth element of $\mathbf{z}$ at iteration i. Using this closed-form solution, the frequencies may then be found using a gradient descent method. The resulting algorithm starts with an initial grid and then iteratively refines the grid points to find the correct solution. This results in a dynamic grid, where the redundant grid points are removed, and the grid points closest to the true solution are refined. The initial grid may here be much coarser than the grid needed to solve (2) with a classic grid-based solution. In the following, we will extend on the SURE-IR algorithm to allow for the incorporation of block penalties, as well as sparsity within each block, showing how the resulting technique may be used to solve the multi-pitch problem.

3 Proposed method

To take the harmonic structure in (1) into consideration and generalize the above discussed SURE-IR algorithm, we need to reformulate the problem so that it allows for a closed form solution similar to (7). In order to do so, let $\mathbf{A}(\boldsymbol{\vartheta})$ denote the $N \times M$ dictionary matrix with

$$\mathbf{A}(\boldsymbol{\vartheta}) = \left[\begin{array}{ccc} \mathbf{A}_1(\vartheta_1) & \ldots & \mathbf{A}_G(\vartheta_G) \end{array} \right] \tag{9}$$

$$\mathbf{A}_g(\vartheta_g) = \left[\begin{array}{cccc} \mathbf{a}(\vartheta_g) & \mathbf{a}(2\vartheta_g) & \ldots & \mathbf{a}(L_g\vartheta_g) \end{array} \right] \tag{10}$$

$$\mathbf{a}(\ell\vartheta_g) = \left[\begin{array}{ccc} \mathrm{e}^{i2\pi\ell\vartheta_g t_1} & \ldots & \mathrm{e}^{i2\pi\ell\vartheta_g t_N} \end{array} \right]^T / \sqrt{(N)} \tag{11}$$

where ϑ_g denotes the fundamental frequency for the gth pitch-group, for $g = 1, \ldots, G$, with G denoting the number of considered groups, and $M = \sum_{g=1}^{G} L_g$, i.e., the total number of frequencies considered in the initial grid. Note that by dividing with $\sqrt{N}$, the columns of the matrix $\mathbf{A}(\boldsymbol{\vartheta})$ are normalized. Using the logarithmic penalty for a group penalty, and at the same time allowing for sparsity within the groups, one may consider the cost function

$$\underset{\mathbf{z},\boldsymbol{\vartheta}}{\text{minimize}}\ \lambda \sum_{g=1}^{G} \sum_{\ell=1}^{L_g} \log\left(|z_{g,\ell}|^2 + \eta\right) +$$

$$\mu \sum_{g=1}^{G} \log\left(||\mathbf{z}_g||_2^2 + \eta\right)/L_g + ||\mathbf{y} - \mathbf{A}(\vartheta)\mathbf{z}||_2^2 \tag{12}$$

where μ and λ are hyperparameters that govern the group sparsity and the overall sparsity, respectively, $\eta > 0$ are constants ensuring that the functions are not evaluated over zero, and where $\mathbf{z}_g$ denotes the amplitudes related to group g in $\mathbf{A}$. As expected, the problem in (12) is not convex and difficult to solve. To allow for a closed form solution for $\mathbf{z}$, the second term in (12) is rewritten as

$$\sum_{g=1}^{G} \log(||\mathbf{z}_g||_2^2 + \eta)/L_g = \sum_{g=1}^{G} \log(||\mathbf{F}_g\mathbf{z}||_2^2 + \eta)/L_g \tag{13}$$

where $\mathbf{F}_g$ is a diagonal matrix with ones on the diagonal corresponding to group g, and zeros elsewhere. To solve (12), we then follow the same approach as in [31] and use an MM approach. To do so, a surrogate function, $Q(\mathbf{z}|\mathbf{z}^{(i)})$, which is much simpler than the original function, is devised such that it coincides with the original function at the current point $\mathbf{z}^{(i)}$, and is greater than or equal to the original function everywhere else. It can be shown that minimizing (or even just decreasing) $Q(\mathbf{z}, \mathbf{z}^{(i)})$ then yields a non-increasing updating step in the original function, thus yielding a method of minimizing the more complex function, using simpler functions. An appropriate surrogate function to (12) may be selected as

$$\begin{aligned}\psi_1(\mathbf{z}|\mathbf{z}^{(i)}) = \sum_{g=1}^{G} L_g^{-1} \Bigg(& \frac{||\mathbf{F}_g\mathbf{z}||_2^2 + \eta}{||\mathbf{F}_g\mathbf{z}^{(i)}||_2^2 + \eta} \\ & + \log(||\mathbf{F}_g\mathbf{z}^{(i)}||_2^2 + \eta) - 1 \Bigg)\end{aligned} \tag{14}$$

for the second term in (12) and

$$\psi_2(\mathbf{z}|\mathbf{z}^{(i)}) = \sum_{g=1}^{G} \sum_{\ell=1}^{L_g} \left(\frac{|z_{g,\ell}|^2 + \eta}{|z_{g,\ell}^{(i)}|^2 + \eta} + \log(|z_{g,\ell}^{(i)}|^2 + \eta) - 1 \right) \tag{15}$$

for the first term, thus yielding

$$Q(\mathbf{z}|\mathbf{z}^{(i)}) = \mu\psi_1(\mathbf{z}|\mathbf{z}^{(i)}) + \lambda\psi_2(\mathbf{z}|\mathbf{z}^{(i)})$$

Removing terms that are independent of $\mathbf{z}$ and ϑ, the surrogate cost function may be re-written as

$$\underset{\mathbf{z},\vartheta}{\text{minimize}}\ S(\mathbf{z}, \vartheta|\mathbf{z}^{(i)}) \tag{16}$$

where

$$S(\mathbf{z}, \boldsymbol{\vartheta}|\mathbf{z}^{(i)}) = \lambda \mathbf{z}^H \mathbf{D}_0^{(i)} \mathbf{z} + \mu \sum_{g=1}^{G} \mathbf{z}^H \mathbf{F}_g^H D_g^{(i)} \mathbf{F}_g \mathbf{z}/L_g + ||\mathbf{A}(\boldsymbol{\vartheta})\mathbf{z} - \mathbf{y}||_2^2 \tag{17}$$

with

$$\mathbf{D}_0^{(i)} = \text{diag}\left(\frac{1}{|z_1^{(i)}|^2 + \eta}, \ldots, \frac{1}{|z_M^{(i)}|^2 + \eta}\right) \tag{18}$$

$$D_g^{(i)} = \frac{1}{||\mathbf{F}_g \mathbf{z}^{(i)}||_2^2 + \eta}, \quad \text{for } g = 1, \ldots, G \tag{19}$$

Furthermore, let

$$\mathbf{H}^{(i)} = \sum_{g=1}^{G} \mathbf{F}_g^H D_g^{(i)} \mathbf{F}_g / L_g \tag{20}$$

Differentiating $S(\mathbf{z}, \boldsymbol{\vartheta}|\mathbf{z}^{(i)})$ with respect to $\mathbf{z}$, setting it equal to zero, yields

$$\frac{\partial S(\mathbf{z}, \boldsymbol{\vartheta}|\mathbf{z}^{(i)})}{\partial \mathbf{z}} = 0 \Leftrightarrow \tag{21}$$

$$\mathbf{z}(\boldsymbol{\vartheta})^* = \left(\lambda \mathbf{D}_0^{(i)} + \mu \mathbf{H}^{(i)} + \mathbf{A}(\boldsymbol{\vartheta})^H \mathbf{A}(\boldsymbol{\vartheta})\right)^{-1} \mathbf{A}(\boldsymbol{\vartheta})^H \mathbf{y} \tag{22}$$

Using (22), one may then find the $\boldsymbol{\vartheta}$ that minimizes (16) by searching for the best $\boldsymbol{\vartheta}$ using, e.g., a steepest descent method, by substituting (22) in (16), yielding

$$\underset{\boldsymbol{\vartheta}}{\text{minimize}}\ S(\mathbf{z}^*, \boldsymbol{\vartheta}|\mathbf{z}^{(i)}) = - \mathbf{y}^H \mathbf{A}(\boldsymbol{\vartheta}) \left(\lambda \mathbf{D}_0^{(i)} + \mu \mathbf{H}^{(i)} + \mathbf{A}(\boldsymbol{\vartheta})^H \mathbf{A}(\boldsymbol{\vartheta})\right)^{-1} \mathbf{A}(\boldsymbol{\vartheta})^H \mathbf{y} \tag{23}$$

Following the reasoning in [31], one may show that the original cost function will be non-increasing when one decreases the surrogate function, thus showing that $\Gamma(\boldsymbol{\vartheta}^{(i+1)}, \mathbf{z}^{(i+1)}) \leq \Gamma(\boldsymbol{\vartheta}^{(i)}, \mathbf{z}^{(i)})$. This proof has been presented in [31] for the problem in (6); the corresponding proof for the here considered case follows directly, and is, in the interest of brevity, thus omitted.

Interestingly, the minimization problem in (23) is very similar to the one in [31]; the difference lies in the introduction of $\mu \mathbf{H}^{(i)}$, which weights the different $z_{g,\ell}$ accordingly to the power of the group they belong to. This indicates how

easy it is to extend the SURE-IR algorithm and allow for the modeling of other structures in the signal. For instance, one may consider adding a logarithmic version of the total variation penalty to (12), which would then simply add another term in (23). This suggests that the SURE-IR approach, in contrast to, e.g., atomic norm, can allow for adding and subtracting different penalties and may thus easily be extended to cover also other model structures.

For the gradient based search, one needs to compute the gradient of $S(\mathbf{z}^*, \boldsymbol{\vartheta}|\mathbf{z}^{(i)})$ with respect to $\boldsymbol{\vartheta}$. The gradient for the single sinusoid case was presented in [31] and the reader is referred to that paper for the details. However, we note that, in contrast to the single sinusoidal case, the derivative of one fundamental frequency, $\partial\mathbf{A}(\boldsymbol{\vartheta})/\partial\boldsymbol{\vartheta}$, is in the examined case operating on all the elements of that pitch group; the derivative will thus be a matrix instead of a vector for the here considered case. Thus the direction, d_g, for which the frequency for the pitch group g is moving is

$$d_g = -\mathbf{y}^H \left(\mathbf{T}_1 + \mathbf{A}\boldsymbol{\mathcal{G}}\mathbf{A}^H + \mathbf{T}_1^H \right) \mathbf{y} \tag{24}$$

where $\mathfrak{Re}(\cdot)$ and $\mathrm{Tr}(\cdot)$ denote the real part and the trace, respectively, and where

$$\mathbf{T}_1 = \frac{\partial \mathbf{A}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} \mathbf{T}_2 \mathbf{A}(\boldsymbol{\vartheta})^H \tag{25}$$

with

$$\mathbf{T}_2 = \left(\lambda \mathbf{D}_0 + \mu \mathbf{H} + \mathbf{A}(\boldsymbol{\vartheta})^H \mathbf{A}(\boldsymbol{\vartheta}) \right)^{-1} \tag{26}$$

and

$$\boldsymbol{\mathcal{G}} = -\mathbf{T}_2 \left(\frac{\partial \mathbf{A}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}}^H \mathbf{A}(\boldsymbol{\vartheta}) + \mathbf{A}(\boldsymbol{\vartheta})^H \frac{\partial \mathbf{A}(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} \right) \mathbf{T}_2 \tag{27}$$

When forming the gradient step, each harmonic is then multiplied with its corresponding harmonic order, i.e., ℓ. Thus, the updating becomes

$$\vartheta_g^{(i+1)} = \vartheta_g^{(i)} - \alpha d_g \tag{28}$$

where α denotes the step length.

The algorithm starts by first selecting a grid of fundamental frequencies, and then adding the harmonics, thus forming a grid containing G fundamental frequencies and M total grid points (thereby including both the fundamental frequencies and their respective harmonics). In pitch estimation, one has to pay particular attention to the so-called halfling problem [15, 16]. This problem stems

from the fact that the frequencies corresponding to $\{f_0, 2f_0, \ldots, L_0 f_0\}$ are also present in the group corresponding to $f_0/2$. This ambiguity results in that the algorithms often prefer to choose the lower fundamental frequency. A common solution to this problem is to include a total variation penalty, which can easily be included in the proposed method. However, we opt to overcome this problem by, similarly to [17], instead penalize the amplitudes in each group with the power of the group's fundamental frequency. Thereby, if the amplitude of the candidate fundamental frequency is zero, the other amplitudes in that group will be heavily penalized; thus, if there is any competition between f_0 and $f_0/2$ candidates, the method is more likely to choose the higher fundamental frequency. This penalty is not necessary after the algorithm has found some initial estimates of the groups, and may be removed after a couple of iterations, which will, in the same way as decreasing μ and λ (see next section), result in an improved estimate.

4 Implementational aspects

The implementation of the proposed algorithm relies on three steps in each iteration: solving (23) using a gradient-based minimization, evaluating $\mathbf{z}$ for the new value of $\boldsymbol{\vartheta}$ using (22), and removing redundant grid points. The last step is implemented to reduce the computational complexity by decreasing the size of the matrices $\mathbf{A}(\boldsymbol{\vartheta})$, $\mathbf{D}_0$, and $\mathbf{H}$. To speed up the calculations, one may start pruning the dictionary after merely a few iterations. This is done by removing all the groups and all the individual frequencies in case their magnitudes are below a certain predefined limit, say τ, which we in this paper has selected to be $\tau = 0.05$.

Appropriately setting hyperparameters such as μ and λ is often a difficult problem. In this work, we take a practical stance to this problem. First, we observe that if the true $\boldsymbol{\vartheta}$ were known, one would solve (23) with $\mu = \lambda = 0$. Thus, we should expect the method to improve if we gradually decreased λ and μ. To this end, we begin setting λ as in [31]. Then, after the first pruning step, we decrease λ by half each iteration, thereby gradually improving the estimates. Similar to the method introduced in [31], the extended algorithm will also decrease η in each iteration. The choice of μ is more critical. A too small value of μ will result in too many groups being involved in the solution, and a too large value will suppress true groups and often result in the method breaking down. If one is not able to find a suitable value of μ, one may first run the algorithm by setting a large μ; if the method breaks down, i.e., yields an empty set, the problem is

Algorithm 1 The BSURE-IR estimator

1: Define an M element grid, $\boldsymbol{\vartheta}$, over the considered fundamental frequencies, and let $\lambda = \lambda_0$, $\mu = \mu_0$, $\tau = \tau_0$, $i = 1$, $\mathbf{z}^{(0)} = \mathbf{0_{MG}}$, and $\mathbf{z}^{(1)} = \mathbf{1_{MG}}$.
2: **while** $i < 2$ or $||\mathbf{z}^{(i)} - \mathbf{z}^{(i-1)}||_2 > \tau$ **do**
3: Form $\mathbf{H}^{(i)}$ from (18), (19), and (20).
4: Update $\mathbf{z}(\boldsymbol{\vartheta})^{(i)}$ from (22).
5: Update $\boldsymbol{\vartheta}^{(i)}$ by solving (23) using (28).
6: Decrease λ and μ, prune the dictionary and remove all columns of $\mathbf{A}(\boldsymbol{\vartheta})$ corresponding to elements in $\mathbf{z}$ that are $|z_{g,\ell}| < 0.05$ and $||\mathbf{z}_g||_2 < 0.05$.
7: Set $i = i + 1$
8: If $||\mathbf{z}||_0 = 0$, then set $\mu = \mu_0/2$ and restart the iterations with $i = 1$.
9: **end while**

simply resolved using a smaller value of μ. As noted above, it may be beneficial to continue to decrease the value of μ, which can be efficiently computed by warm-starting the algorithm for each decrease of μ. This approach to selecting a good value of μ is possible since with the pruning step, the computational complexity is low. As shown in the numerical section, the proposed method is notably faster than the SURE-IR algorithm when using a dictionary with the same number of frequencies. This is primarily due to the fact that even though the number of grid points are the same, the proposed method only has the fundamental frequencies as variables; thus, when calculating the gradient, and pruning the dictionary, these steps become more efficient.

We coin the presented method the block super-resolution iteratively reweighted (BSURE-IR). Algorithm 1 summaries the proposed method, wherein $\mathbf{0_{MG}}$ and $\mathbf{1_{MG}}$ denote an $MG \times 1$ long vector of zeros and ones, respectively, and τ a predefined stopping criteria.

5 Numerical examples

In this section, we investigate the performance of the proposed method and compare the results to other competing methods. Throughout this section, we will evaluate the methods' ability to correctly estimate the frequencies by measuring

the root-mean-squared-error (RMSE), defined as

$$\text{RMSE}(\hat{\boldsymbol{\vartheta}}) = \sqrt{\frac{1}{\sum_{k=1}^{K} L_k} \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} (\vartheta_{k,\ell} - \hat{\vartheta}_{k,\ell})^2} \tag{29}$$

where $\vartheta_{k,\ell}$ denotes the true parameter value, $\hat{\vartheta}_{k,\ell}$ the estimated value, and $\hat{\boldsymbol{\vartheta}}$ the vector of parameters that are estimated. In the following, we compare the methods' RMSE as a function either of the length of the signal, N, or the signal-to-noise-ratio (SNR), defined as

$$\text{SNR} = 10 \log \left(\frac{P}{\sigma^2} \right) \tag{30}$$

where P is the power of the noiseless signal and σ^2 the variance of the noise. For each SNR level or signal length, the presented results are found using 100 Monte-Carlo simulations. In the first example, an $N = 30$ long uniformly sampled signal with a single pitch was considered. The fundamental frequency was uniformly drawn between $[1/7, 1/3)$ for each Monte-Carlo simulation and the number of harmonics were selected as $\left\lfloor \frac{1}{f_0} \right\rfloor$ for each fundamental frequency, f_0, with $\lfloor \cdot \rfloor$ denoting the floor operator. Four algorithms were considered; BSURE-IR, SURE-IR [31], ANLS [9], and the PEBS algorithm [15]. The BSURE-IR method was allowed an initial grid of 15 elements over the fundamental frequencies, ranging from $[0.1, 0.3]$, and the number of harmonics selected as $\left\lfloor \frac{1}{f_0} \right\rfloor$, for each considered fundamental frequency, f_0, thus yielding a dictionary containing a total of 77 spectral lines. The initial value of μ was set to 100. The SURE-IR algorithm was also allowed a dictionary containing 77 elements, although these being unstructured. The ANLS was allowed 2^8 grid points and was given the same range over the fundamental frequency as BSURE-IR, as well as perfect model order knowledge. The PEBS algorithm was given prior information about where the fundamental frequency was positioned, given as a range of ± 0.02 around the true value. In this range, PEBS was given 1000 grid points and the initial user parameters were set to 5 and 30 for the parameter governing the ℓ_1 and the ℓ_2 norms, respectively. Furthermore, for the PEBS algorithm, only the largest peak was selected from the estimates, thus not requiring the algorithm to make a correct model order, thereby avoiding the problem of wrongly setting the hyperparameters. This was not true for the other methods, where each wrong model order estimate was recorded. The resulting RMSE may be seen in Figure 1, where it can

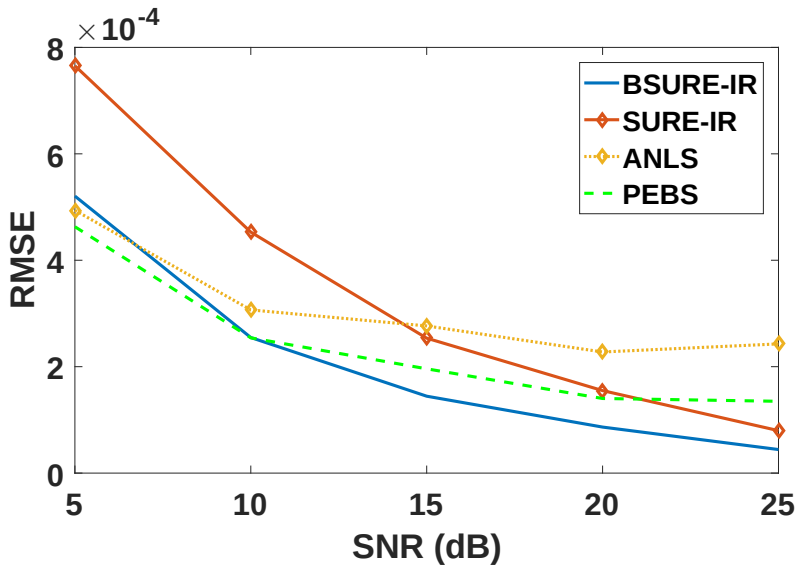

Figure 1: The RMSE of the frequency estimates, as defined in (20), as a function of SNR, for uniformly sampled data.

be seen that the proposed method outperforms the other methods for SNR-levels of 10 dB and above. Interestingly, it can be seen that the grid-based methods have similar performance to the BSURE-IR for low SNR levels, whereas the two off-grid methods excel for higher SNR levels; even SURE-IR, which does not take the harmonic structure in consideration, actually outperforms the two grid-based methods that actively exploits the harmonic structure. In this setting, the BSURE-IR method failed to correctly estimate the model order 6 times for the lowest SNR level, but managed to correctly do so for the other SNRs. The average run-times for the methods were 3.0 seconds for BSURE-IR, 10.5 seconds for SURE-IR, 0.1 seconds for ANLS, and 4.7 for PEBS.

Proceeding, we investigate how the performance is affected by non-uniformly sampled data. This scenario is not as common for audio samples, but is so in many other areas. As ANLS does not allow for this case, the algorithm is omitted from comparison. Using the same settings as before, but now with non-uniform sampled data with length $N = 30$ sampled from 60 measurements, the RMSE was measured for the methods. Figure 2 shows the result. As expected,

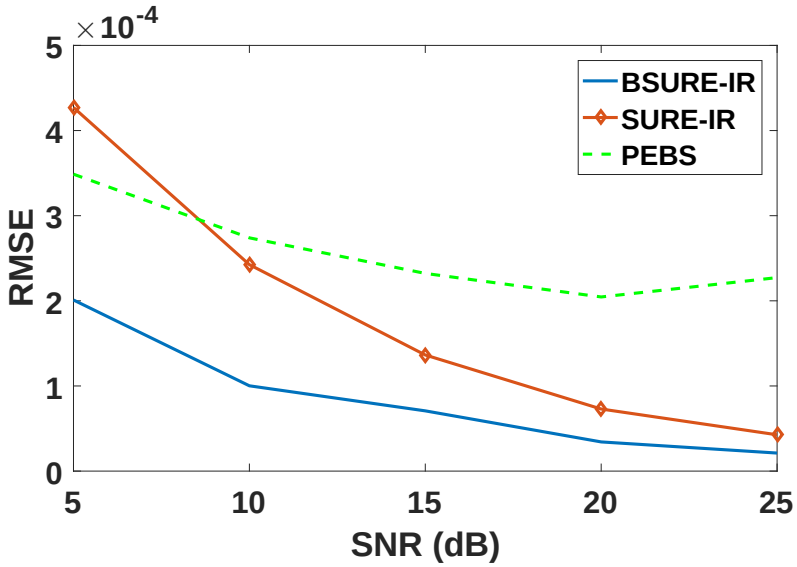

Figure 2: The RMSE of the frequency estimates as a function of SNR for non-uniformly sampled data.

BSURE-IR again outperforms the competing methods. Again, comparing SURE-IR with PEBS, the latter seems to benefit from exploiting the harmonic structure for lower SNR levels. However, when the SNR level reaches 10 dB, the unstructured SURE-IR again outperforms the PEBS algorithm. Here, BSURE-IR failed to determine the correct model order 6 times for SNR 5 dB, but estimated it correctly in the other cases. The run times in this setting were 2.5 seconds for BSURE-IR, 11.5 seconds for SURE-IR, and 18.5 seconds for PEBS.

In the third example, we investigate the performance as function of the length of the signal. Figure 3 shows the results when using the same settings as before, but with N ranging from 20 to 300 and with SNR fixed at 15 dB. Once again it may be seen that the purposed method outperforms the competing methods. In this scenario, we had to remove 86 outliers for PEBS to make the figure readable; 55 outliers for $N = 20$, 29 for $N = 25$, and 2 for $N = 30$. BSURE-IR estimated the wrong model order five times, once for $N = 20$, $N = 100$, and $N = 300$, and twice for $N = 200$. The run times for the considered algorithms were 2.3 seconds for BSURE-IR, 8.6 seconds for SURE-IR, and 14.2 seconds for PEBS.

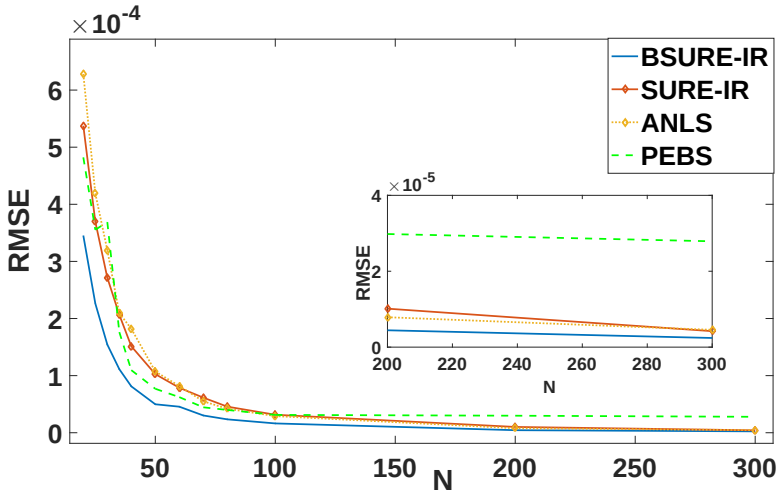

Figure 3: The RMSE of the frequency estimates as a function of the data length, N.

In the fourth example, we look at the case were the signal contains multiple pitches. Here, we consider a signal with length $N = 30$, non-uniformly sampled and with two fundamental frequencies set at $0.15\pi/3$ and $0.26\pi/3$. Figure 4 shows the resulting RMSE for all frequencies in both pitches. For the case when the SNR level is 5 dB, BSURE-IR seems to have problem to get the model order correct, and 41 times the estimated order model was incorrect. This only happened 8 times for the other SNR levels. For PEBS, 42 outliers were removed to make the figure more readable. If disregarding the 5 dB case, one can see that the BSURE-IR method outperforms the PEBS algorithm for the multi-pitch case. Note that, again, PEBS is given *K a priori* and is also zoomed in around the correct fundamental frequencies. Also, PEBS are now allowed 1000 grid points for each fundamental frequency. The run times for this examples are 5.2 seconds for BSURE-IR and 84.3 seconds for PEBS. The increase in run time for PEBS is mainly due to the increase in grid size.

In the final example, we evaluate the performance of the methods on the Bach10 dataset [39]. The data set contains ten excerpts from chorals that were composed by Johann Sebastian Bach. The instruments playing in the pieces are a violin, a clarinet, a saxophone, and a bassoon, and the set contains many se-

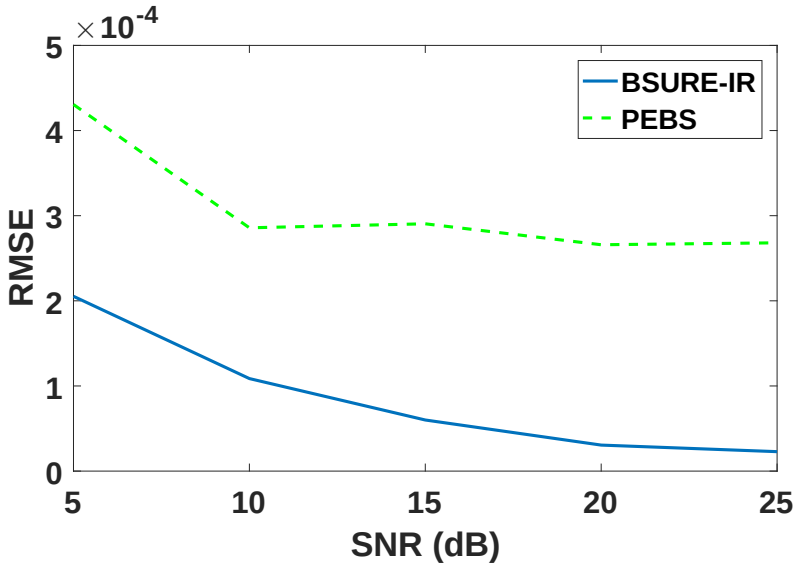

Figure 4: The RMSE of the frequency estimates of a multi-pitch signal containing two pitches for non-uniformly sampled data.

quences where the overtones overlaps. The resulting estimates are compared to ground truth fundamental frequencies, obtained by applying the single pitch estimator YIN [40] to each separate channel. Obvious errors in the ground truth were corrected for manually. Each excerpt is about 25-42 seconds long. Table 1 presents the performance measures accuracy, precision, and recall, as defined in [41]. In Table 1, the performance of the BSURE-IR estimator is compared to four other multi-pitch estimators, namely PEARLS [17], PEBS [15], PEBSI-Lite [16], and ESACF [6], as well as a state-of-the-art music transcription method [42], here denoted BW15 (after the surnames of the authors and the year of publication). For BSURE-IR, the starting value of μ was set to 1 and the number of initial fundamental frequency grid-points 30, and the maximum allowed L was set to 4. PEARLS is a time-recursive multi-pitch estimator, with a dictionary learning scheme that resembles a gridless method, but uses a different cost function, and ESACF is a auto-correlation based multi-pitch estimator. The BW15 method is a music transcription algorithm that uses a probabilistic latent component analysis to produce pitch estimates that are trained on databases of music instruments.

Method	Accuracy	Precision	Recall	Pre-trained
BSURE-IR	0.47	0.71	0.58	No
PEARLS	0.44	0.68	0.54	No
PEBS	0.39	0.56	0.51	No
PEBSI-Lite	0.45	0.63	0.61	No
BW15	0.52	0.68	0.68	Yes
ESACF	0.27	0.47	0.39	No

Table 1: Performance measures for the BSURE-IR, PEARLS, PEBS, PEBSI-Lite, BW15, and ESACF algorithms, when evaluated on the Bach10 dataset.

We choose to include this method into the comparison to show the performance of a state-of-the-art method that is pre-trained and specifically tailored for music transcription, which is not the case for the other discussed methods. The settings and results from ESACF and PEBSI-Lite were taken from [16] and for PEARLS and BW15, the setting and results were from [17]. The PEBS settings and results were obtained from [18]. Figure 5 shows the resulting BSURE-IR estimates of the fundamental frequencies from an excerpt of J. S. Bach's *Ach, Gott und Herr* performed by a violin, a bassoon, a clarinet, and a saxophone. As can be seen from the figure, BSURE-IR manages to capture most of the fundamental frequencies without too many false positives. Furthermore, from Table 1, one may see that the BSURE-IR method scores higher on both accuracy and precision as compared to the other multi-pitch estimators, and has somewhat even score for recall. Not surprisingly, BW15 attains a higher score than BSURE-IR, except for precision, where BSURE-IR attains a slightly higher score. However, it should be stressed that BW15 has been trained on the instruments included in the Bach10 data set, whereas BSURE-IR has not. We note that, as for a future research topic, it would be interesting to try to combine the probabilistic approach of BW15 and the more robust BSURE-IR signal model approach.

6 Conclusions

In this paper, we present a novel off-grid multi-pitch estimator. By parameterizing the dictionary containing the candidate pitches and solving a non-convex

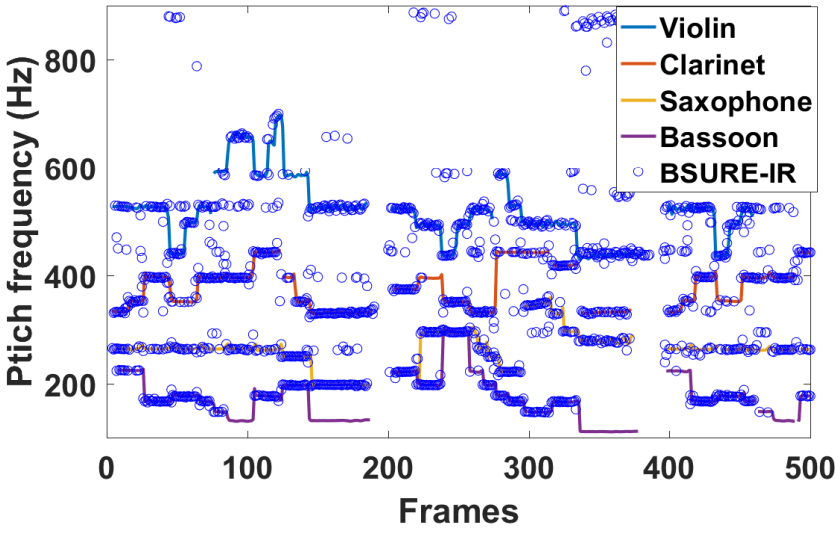

Figure 5: The resulting estimation of the fundamental frequencies (pitches) of the Bach10 data set.

optimization problem using a majorization-minimization approach, an iterative method is derived. In each iteration, the dictionary is pruned which allows for a decreased computational complexity. The method is evaluated on both simulated and real data. In the real data case, the proposed method is shown to yield similar performance as a specialized music transcription algorithm that is pre-trained on the instruments present in the signal. Furthermore, the proposed method is benchmarked against other popular multi-pitch estimates, showing the preferred performance of the proposed method.

References

[1] H. Akaike, "A New Look at Statistical Model Identification," *IEEE Trans. Autom. Control*, vol. 19, pp. 716–723, 1974.

[2] J. Rissanen, "Modeling by Shortest Data Description," *Automatica*, vol. 14, pp. 465–471, 1978.

[3] G. Schwarz, "Estimating the Dimension of a Model," *Ann. Stat.*, vol. 6, pp. 461–464, 1978.

[4] 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.

[5] L. Rabiner, "On the use of autocorrelation analysis for pitch detection," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 25, no. 1, pp. 24–33, 1977.

[6] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 708–716, 2000.

[7] M. G. Christensen, J. .L Højvang, A. Jakobsson, and S. H. Jensen, "Joint fundamental frequency and order estimation using optimal filtering," *EURASIP Journal on Advances in Signal Processing*, vol. 13, pp. 1–18, 2011.

[8] M. G. Christensen, J. H. Jensen, A. Jakobsson, and S. H. Jensen, "On optimal filter designs for fundamental frequency estimation," *IEEE Signal Process. Lett.*, vol. 15, pp. 745–748, 2008.

[9] 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.

[10] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan & Claypool, San Rafael, Calif., 2009.

[11] K. W. Chan and H. C. So, "Accurate frequency estimation for real harmonic sinusoids," *IEEE Signal Process. Lett.*, vol. 11, no. 7, pp. 609–612, 2004.

[12] 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.

[13] J. K. Nielsen, T. L. Jensen, J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Fast fundamental frequency estimation: Making a ststatistical efficient estimator computationally efficient," *Elsevier Signal Processing*, vol. 135, pp. 188–197, Jan 2017.

[14] J. Zeremdini, M. A. B. Messaoud, and A. Bouzid, "Multiple comb filters and autocorrelation of the multi-scale product for multi-pitch estimation," *Elsevier Signal Processing*, vol. 120, pp. 45–53, May 2017.

[15] 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.

[16] 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.

[17] F. Elvander, J. Swärd, and A. Jakobsson, "Online Estimation of Multiple Harmonic Signals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 2, pp. 273–284, February 2017.

[18] S. Lei, F. Elvander, J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Computationally Efficient Multi-Pitch Estimation Using Sparsity," in *11th IMA internation Conference on Mathematics in Signal Processing*, Birmingham, England, Dec 2016.

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

[20] 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.

[21] 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.

[22] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 707–710, Oct. 2007.

[23] Jean-Jacques Fuchs, "On sparse representation in arbitrary redundant bases," *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 1341–1344, June 2004.

[24] J. A. Tropp, *Topics in Sparse Approximation*, Computational and applied mathematics, 2004.

[25] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

[26] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The Convex Geometry of Linear Inverse Problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, Dec 2012.

[27] B. N. Bhaskar, G. Tang, and B. Recht, "Atomic Norm Denoising with Applications to Line Spectral Estimation," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5987 – 5999, July 2013.

[28] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed Sensing Off the Grid," *IEEE Trans. Inform. Theory*, vol. 59, no. 11, pp. 7465–4790, Nov 2013.

[29] Z. Yang and L. Xie, "On Gridless Sparse Methods for Line Spectral Estimation From Complete and Incomplete Data," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3139–3153, June 2015.

[30] Z. Yang and L. Xie, "Enhancing Sparsity and Resolution via Reweighted Atomic Norm Minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 4, pp. 995–1006, Feb 2016.

[31] 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.

[32] C. S. Sunnersjö, "Rolling Bearing Vibrations - The Effects of Geometrical Imperfections and Wear," *Journal of Sound and Vibration*, vol. 98, no. 4, pp. 455–474, Feb 1985.

[33] A. M. Ahmadi, D. Petersen, and C. Howard, "A nonlinear dynamic vibration model of defective bearings - the importance of modelling the finite size of rolling elements," *Mechanical Systems and Signal Processing*, vol. 52-53, pp. 309–326, Feb 2015.

[34] W. Liu, Y. Zhang, Z. J. Feng, J. S. Zhao, and D. Wang, "A study on waviness induced vibration of ball bearings based on signal coherence theory," *Journal of Sound and Vibration*, vol. 333, no. 23, pp. 6107–6120, Nov 2014.

[35] A. Cubillo, S. Perinpanayagam, and M. Esperon-Miguez, "A review of physics-based models in prognostics: Application to gears and bearings of rotating machinery," *Advances in Mechanical Engineering*, vol. 8, no. 8, pp. 1–21, Aug 2016.

[36] P. C. Hansen and S. H. Jensen, "Subspace-Based Noise Reduction for Speech Signals via Diagonal and Triangular Matrix Decompositions: Survey and Analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–24, 2007.

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

[38] S. I. Adalbjörnsson, J. Swärd, and A. Jakobsson, "Enhancing smoothness in amplitude modulated sparse signals," in *11th IMA International Conference on Mathematics in Signal Processing*, Birmingham, England, Dec 2016.

[39] Z. Duan and B. Pardo, "Bach10 dataset," http://music.cs.northwestern.edu/data/Bach10.html, Accessed December 2015.

[40] 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.

[41] 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.

[42] E. Benetos and T. Weyde, "An Efficient Temporally-Constrained Probabilistic Model for Multiple-Instrument Music Transcription," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, Malaga, Spain, October 2015.

G

Paper G

Estimating Sparse Signals Using Integrated Wideband Dictionaries

Maksim Butsenko[1], Johan Swärd[2], and Andreas Jakobsson[2]
[1] *Thomas Johann Seebeck Dept. of Electronics, Tallinn University of Technology, Tallinn, Estonia*
[2] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*

Abstract

In this paper, we introduce a wideband dictionary framework for estimating sparse signals. By formulating integrated dictionary elements spanning bands of the considered parameter space, one may efficiently find and discard large parts of the parameter space not active in the signal. After each iteration, the zero-valued parts of the dictionary may be discarded to allow a refined dictionary to be formed around the active elements, resulting in a zoomed dictionary to be used in the following iterations. Implementing this scheme allows for more accurate estimates, at a much lower computational cost, as compared to directly forming a larger dictionary spanning the whole parameter space or performing a zooming procedure using standard dictionary elements. Different from traditional dictionaries, the wideband dictionary allows for the use of dictionaries with fewer elements than the number of available samples without loss of resolution. The technique may be used on both one- and multi-dimensional signals, and may be exploited to refine several traditional sparse estimators, here illustrated with the Lasso and the SPICE estimators. Numerical examples illustrate the improved performance.

Key words: Sparse signal reconstruction, dictionary learning, convex optimization

1 Introduction

A wide range of common applications yield signals that may be well approximated using a sparse reconstruction framework, and the area has as a result attracted notable interest in the recent literature (see, e.g., [1–3] and the references therein). Much of this work has focused on formulating convex algorithms that exploit different sparsity inducing penalties, thereby encouraging solutions that are well represented using only a few elements from some (typically known) dictionary matrix, $\mathbf{D}$. If the dictionary is appropriately chosen, even very limited measurements can be shown to allow for an accurate signal reconstruction [4,5]. Recently, increasing attention has been given to signals that are best represented using a continuous parameter space. In such cases, the discretization of the parameter space that is typically used to approximate the true parameters will not represent the noise-free signal exactly, resulting in solutions that are less sparse than desired. This problem has been examined in, e.g., [6–8], wherein discretization recommendations and new bounds of the reconstruction guarantees were presented, taking possible grid mismatches into consideration. Typically, this results in the use of large and over-complete dictionaries, which, although quite efficient, often violate the assumptions required to allow for a perfect recovery guarantee.

As an alternative, one may formulate the reconstruction problem using a continuous dictionary, such as in, e.g., [9–11]. This kind of formulations typically use an atomic norm penalty, as introduced in [12], which allows for a way to determine the most suitable convex penalty to recover the signal, even over a continuous parameter space. These solutions often offer an accurate signal reconstruction, but also require the solving of large and computationally rather cumbersome optimization problems, thereby limiting the size of the considered problems.

In this work, we examine an alternative way of approaching the problem, proposing the use of wideband dictionary elements, such that the dictionary is formed over B subsets of the continuous parameter space. In the estimation procedure, the activated subsets are retained and refined, whereas non-activated sets are discarded from the further optimization. This screening procedure may be broken down into two steps. The first step is to remove the parts of the parameter space not active in the signal, whereafter, in the second step, a smaller dictionary is formed covering only the parts of the parameter space that were active in the first step. This smaller dictionary may then again be expanded with candidates close to the activated elements, thereby yielding a zoomed dictionary in these regions. The process may then be repeated to further refine the estimates as desired.

Without loss of generality, the proposed principle is here illustrated on the problem of estimating the frequencies of K complex-valued M-dimensional sinusoid corrupted by white circularly symmetric Gaussian noise. The one-dimensional case of this is a classical estimation problem, originally expressed using a sparse reconstruction framework in [13], and having since attracting notable attention (see, e.g., [14–17]). Here, using the classical formulation, the resulting sinusoidal dictionary will allow for a K-sparse representation of frequencies on the grid, whereas the grid mismatch of any off-grid components will typically yield solutions with more than K components. Extending the dictionary to use a finely spaced dictionary, as suggested in, e.g., [8], will yield the desired solution, although at the cost of an increased complexity. In this work, we instead proceed to divide the spectrum into B (continuous) frequency bands, each band possibly containing multiple spectral lines. This allows for an initial coarse estimation of the signal frequencies, without the risk of missing any off-grid components. Due to the iterative refining of the dictionary, closely spaced components are successfully separated as the dictionary is refined; as the wideband elements span the full band, no power is off-grid, avoiding the problem of a non-sparse solution due to dictionary mismatch.

Other screening methods that decrease the dictionary size have been proposed. For instance, in [18–23], methods for finding the elements in the dictionary that corresponds to zero-valued elements in the sparse vector were proposed. Based on the inner product between the large dictionary and the signal, a rule was formed for deeming whether or not a dictionary element was present in the signal or not. Although these methods show a substantial decrease in computational complexity, one still has to form the inner product between the likely large dictionary and the signal. To alleviate this, one may instead use the here proposed wideband dictionary elements, thereby discarding large parts of the parameter space. Since the wideband dictionary is magnitudes smaller than the full dictionary required to achieve the reconstruction, the computational complexity is significantly reduced.

The proposed principle is not limited to methods that use discretization of the parameter space; it may also be used when solving the reconstruction problem using gridless methods, such as the methods in [9–11]. It has been shown that if the reconstruction problem allows for any prior knowledge about the location of the frequencies, e.g., the frequencies are located within a certain region of the spectrum, one may use this information to improve the estimates [24]. The

proposed method may also be used to attain such prior information, and thus improving the overall estimates as a result.

To illustrate the performance of the proposed dictionary, we make use of two different sinusoidal estimators, namely the Lasso [25] and the SPICE estimators [26, 27]; the first finding the estimate by solving a penalized regression problem, whereas the latter instead solves a covariance fitting problem.

The remainder of this paper is organized as follows: in the next section, the problem of estimating an M-dimensional sinusoidal signal is introduced, followed, in Section III, by the introduction of the proposed wideband dictionary. In Section IV, a discussion about the computational complexity reduction allowed by the proposed wideband dictionary is given, and, in Section V, the performance of the proposed wideband dictionary is illustrated by numerical examples. Finally, in Section VI, we conclude on our work.

2 Problem statement

To illustrate the wideband dictionary framework consider the problem of estimating the K frequencies $f_k^{(m)}$, for $k = 1, \dots, K$ and $m = 1, \dots M$, of an M-dimensional signal $y_{n_1,\dots,n_M}$, with

$$y_{n_1,\dots,n_M} = \sum_{k=1}^{K} \beta_k \mathrm{e}^{2i\pi f_k^{(1)} t_{n_1}^{(1)} + \cdots + 2i\pi f_k^{(M)} t_{n_M}^{(M)}} + \varepsilon_{n_1,\dots,n_M} \tag{1}$$

for $n_m = 1, \dots, N_m$, and where K denotes the (unknown) number of sinusoids in the signal. Furthermore, let β_k and $f_k^{(m)}$ denote the complex amplitude and frequency of the kth frequency and mth dimension, respectively, $t_{n_m}^{(m)}$ the n_mth sample time in the mth dimension, and $\varepsilon_{n_1,\dots,n_M}$ an additive noise observed at time $t_{n_1}, \dots, t_{n_M}$. The signal model in (1) may be equivalently described by an M-dimensional (M-D) tensor

$$\mathcal{Y} = \sum_{k=1}^{K} \beta_k \tilde{\mathbf{d}}_{(k)}^{(1)} \circ \tilde{\mathbf{d}}_{(k)}^{(2)} \cdots \circ \tilde{\mathbf{d}}_{(k)}^{(M)} + \mathcal{E} \tag{2}$$

where $\circ$ denotes the outer product, and

$$\tilde{\mathbf{d}}_{(k)}^{(m)} = \begin{bmatrix} \mathrm{e}^{2i\pi f_k^{(m)} t_1^{(m)}} & \dots & \mathrm{e}^{2i\pi f_k^{(m)} t_{N_m}^{(m)}} \end{bmatrix}^T \tag{3}$$

To determine the parameters of the model in (1) or (2), as well as the model order, we proceed by creating a dictionary containing a set of signal candidates, each representing a sinusoid with a unique frequency. By measuring the distance between the signal candidates and the measured signal, and by promoting a sparse solution, one may find a small set of candidates that best approximates the signal. To this end, we form dictionary elements on the form

$$\mathbf{d}_{(k)}^{(m)} = \left[\begin{array}{ccc} \mathrm{e}^{2i\pi f_p^{(m)} t_1^{(m)}} & \dots & \mathrm{e}^{2i\pi f_p^{(m)} t_{N_m}^{(m)}} \end{array} \right]^T \tag{4}$$

for $p = 1, \dots, P_M$, where $P_M \gg K$ denotes the number of candidates in dimension m. Here, the dictionary is assumed to be fine enough so that the unknown sinusoidal component will (reasonably well) coincide with K dictionary elements. Often, it is more convenient to work with a vectorized version of the tensor. Let $\mathbf{y} = \mathrm{vec}(\mathcal{Y})$, where $\mathrm{vec}(\cdot)$ stacks the tensor into a vector. One may then re-write (2) as

$$\mathbf{y} = \left(\mathbf{D}^{(M)} \otimes \mathbf{D}^{(M-1)} \otimes \cdots \otimes \mathbf{D}^{(1)} \right) \beta \tag{5}$$

where $\otimes$ denotes the Kronecker product, suggesting that one may find both the unknown parameters and the model order by forming the Lasso problem (see, e.g., [13, 25])

$$\min_{\beta} \frac{1}{2} ||\mathbf{y} - \mathbf{D}\beta||_2^2 + \lambda ||\beta||_1 \tag{6}$$

where $\mathbf{D} = \left(\mathbf{D}^{(M)} \otimes \mathbf{D}^{(M-1)} \otimes \cdots \otimes \mathbf{D}^{(1)} \right)$ and $\|\cdot\|_q$ denotes the q-norm. The penalty on the 1-norm of β will ensure that the found solution will be sparse, with λ denoting a user parameter governing the desired sparsity level of the solution. The frequencies, as well as their order, are then found as the non-zero elements in β.

As shown in [8], the number of dictionary elements, P, typically has to be large to allow for an accurate determination of the correct parameters. This means that for multi-dimensional signals, the dictionary quickly becomes inhibitory large. Thus, it is often not feasible in practice to directly compute the solution of (6) using a dictionary constructed from such finely space candidates. As an alternative, one may use a zooming procedure, where one first employs an initial coarse dictionary, $\mathbf{D}_1$, to determine the parameter regions of interest, and then

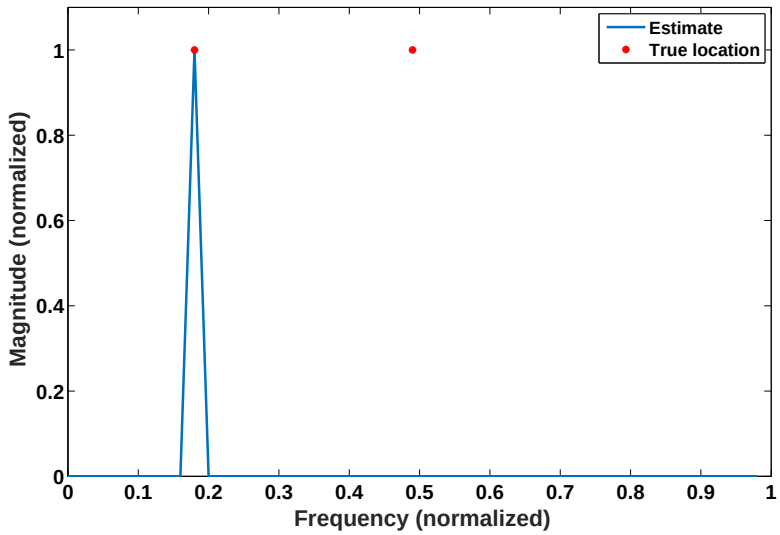

Figure 1: The inner-product of a dictionary containing $P = 50$ (narrowband) candidate frequency elements and the noise-free signal, with $N = 100$.

employ a fine dictionary, $\mathbf{D}_2$, centered around the initially found candidates (see, e.g., [28, 29] for similar approaches). This allows for a computationally efficient solution of the optimization problem in (6), but suffers from the problem of possibly missing off-grid components far from the initial coarse frequency grid. This is illustrated in Figure 1 for a 1-D signal, where the inner-product between the dictionary and the signal is depicted together with the location of the true peaks. In this noise-free example, we used $N = 100$ samples and $P = 50$ dictionary elements, with one of the frequencies being situated in between two adjacent grid points in the dictionary. As seen in the figure, the coarse initial estimate fails to detect the presence of the second signal component, which is thereby discarded as a possibility in the following refined estimate. Increasing the number of candidate frequencies will result in the side-lobes of the dictionary elements decreasing the gap between the frequency grid points, making the inner-product between the dictionary and the signal larger for components that lie in between two candidate frequencies. However, doing so will increase the computational complexity correspondingly, begging the question if one may retain a low number of candidate

frequencies, while still reducing the likelihood of missing any off-grid components. This is the problem we shall examine in the following.

3 Integrated wideband dictionaries

We note that the above problem results from the dictionary being formed over a set of single-component candidates, thereby increasing the risk of neglecting the off-grid components. In order to avoid this, we here propose a wideband dictionary framework, such that each of the dictionary elements is instead formed over a range of such single-component candidates. This is done by letting the dictionary elements be formed over an integrated range of the parameter(s) of interest, in this case being the frequencies of the candidate sinusoids. For a multi-dimensional sinusoidal dictionary, the resulting B integrated wideband elements should thus be formed as

$$\begin{aligned} &a_{b^{(1)},\dots,b^{(M)}}(t^{(1)},\dots,t^{(M)}) = \\ &\int_{f_{b^{(1)}}}^{f_{b^{(1)}+1}} \cdots \int_{f_{b^{(M)}}}^{f_{b^{(M)}+1}} \\ &\mathrm{e}^{2i\pi(f^{(1)}t^{(1)}+\cdots+f^{(M)}t^{(M)})}\, df^{(1)} \dots df^{(M)} \end{aligned} \tag{7}$$

for $t^{(m)} = 1,\dots,N_m$ for all $m = 1,\dots,M$, where $f_b^{(m)}$ and $f_{b+1}^{(m)}$ are the two frequencies bounding the frequency band, for $b = 1,\dots,B$, for the mth dimension. The resulting elements are then gathered into the dictionary, $\mathbf{B}$, where each column contains a specific wideband of the M-D parameter space for all time samples, where each element is formed as the solution from (7), such that, in this case,

$$\begin{aligned} &a_{b^{(1)},\dots,b^{(M)}}(t^{(1)},\dots,t^{(M)}) = \\ &\prod_{m=1}^{M} \frac{\mathrm{e}^{2i\pi f_{b^{(m)}+1}t^{(m)}} - \mathrm{e}^{2i\pi f_{b^{(m)}}t^{(m)}}}{2i\pi t^{(m)}} \end{aligned} \tag{8}$$

Note that (8) corresponds to the M-D inverse Fourier transform of 1, i.e., it is the M-D inverse Fourier transform of an M-D section in the frequency domain with unit amplitude. For the 1-D case, this simplifies to

$$\begin{cases} 1, & \text{for } f_a \le f \le f_b \\ 0, & \text{otherwise} \end{cases} \quad \xrightarrow{\mathcal{F}^{-1}} \quad \begin{cases} \frac{\mathrm{e}^{2i\pi f_b t} - \mathrm{e}^{2i\pi f_a t}}{2i\pi t} \\ 0 \end{cases} \tag{9}$$

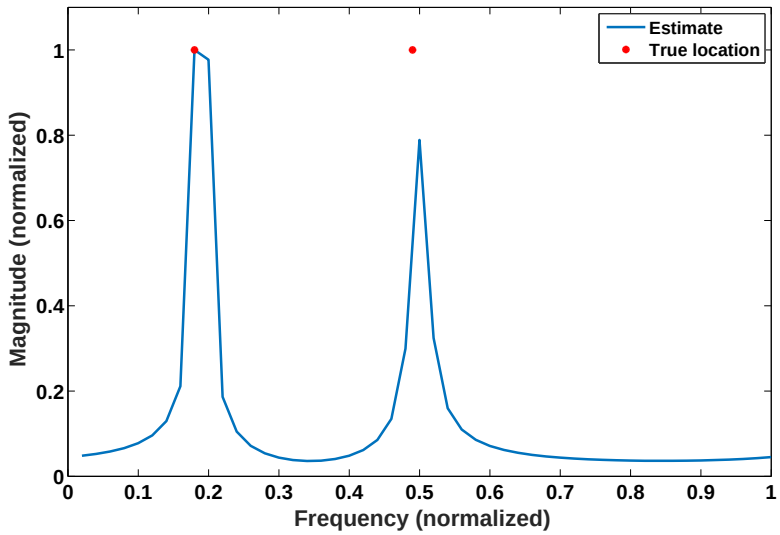

Figure 2: The inner-product of a dictionary containing $B = 50$ (wideband) candidate frequency elements and the noise-free signal, with $N = 100$.

The inner-product between the proposed dictionary, $\mathbf{B}$, and the earlier 1-D signal is shown in Figure 2, using the same number of dictionary elements as in Figure 1, clearly indicating that the proposed dictionary is able to locate the off-grid frequency. This is due the wideband nature of the proposed dictionary, which thus has less power concentrated at the grid points, but covers a wider range of frequencies, not reducing to zero, or close to zero, anywhere within the band (as is the case for the narrowband dictionary elements). As a result, using the wideband dictionary elements, it is possible to use a smaller dictionary, thereby reducing the computational complexity, without increasing the risk of missing components in the signal. To further show this, 1000 Monte-Carlo simulations were conducted for each considered signal to noise ratio (SNR) level. In each simulation, we considered a signal containing two sinusoids, where the frequencies were randomly selected on $(0, 1]$ with a spacing of at least $2/N$, with $N = 100$ denoting the signal length. The sinusoids had the magnitudes 4 and 5, with a randomly selected phase between $(0, 2\pi]$. Two dictionaries were given, one containing ordinary sinusoids and one containing the proposed wideband components, both contain-

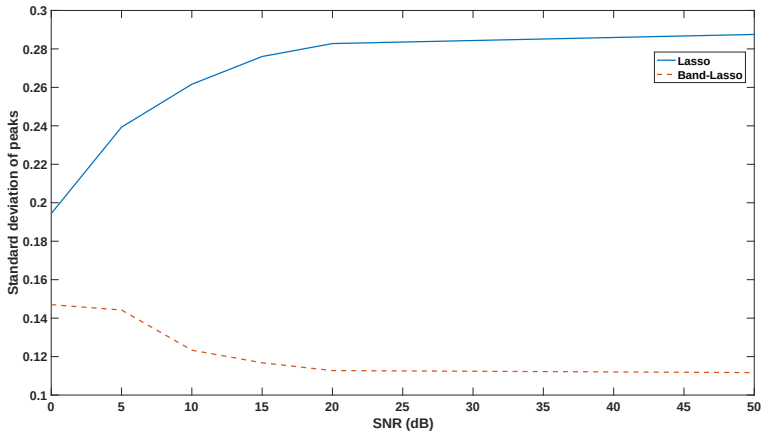

Figure 3: The standard deviation of the peaks as a function of SNR.

ing $P = B = 50$ elements. For each dictionary, the inner-products with the signal where computed, where the amplitudes were normalized so that the largest estimated peak had unit magnitude. Figure 3 shows the variance of the smallest peak for different SNR-levels. As is clear from the figure, the variance of the peaks are much lower for the banded case. The reason why the sinusoidal dictionary results in a larger variance is due to the fact that the main lobe is much thinner in this case than in the banded counterpart. This means that when the sinusoids happen to have frequencies that do not overlap with the main lobe of the dictionary, the power in the inner-product will be small. This will not only make such components harder to detect, but will also make it more difficult to determine a suitable regularizing hyperparameter, λ. When P decreases below N, the gaps between the frequency candidates in the single-component dictionary become so large that if one of the sinusoids in the signal has its frequency values between two adjacent grid points, the likelihood that this sinusoid lie in the null-space of the dictionary increases. This problem is avoided with the wideband dictionary as it is more likely to eliminate any gaps.

This property is depicted in Figure 4, where the success rate of finding the true support is displayed as a function of the number of samples, N, and the number of bands in the dictionary, B, for different number of sinusoids in the signal, K. The estimation was done for a noise-free signal by solving (6), using

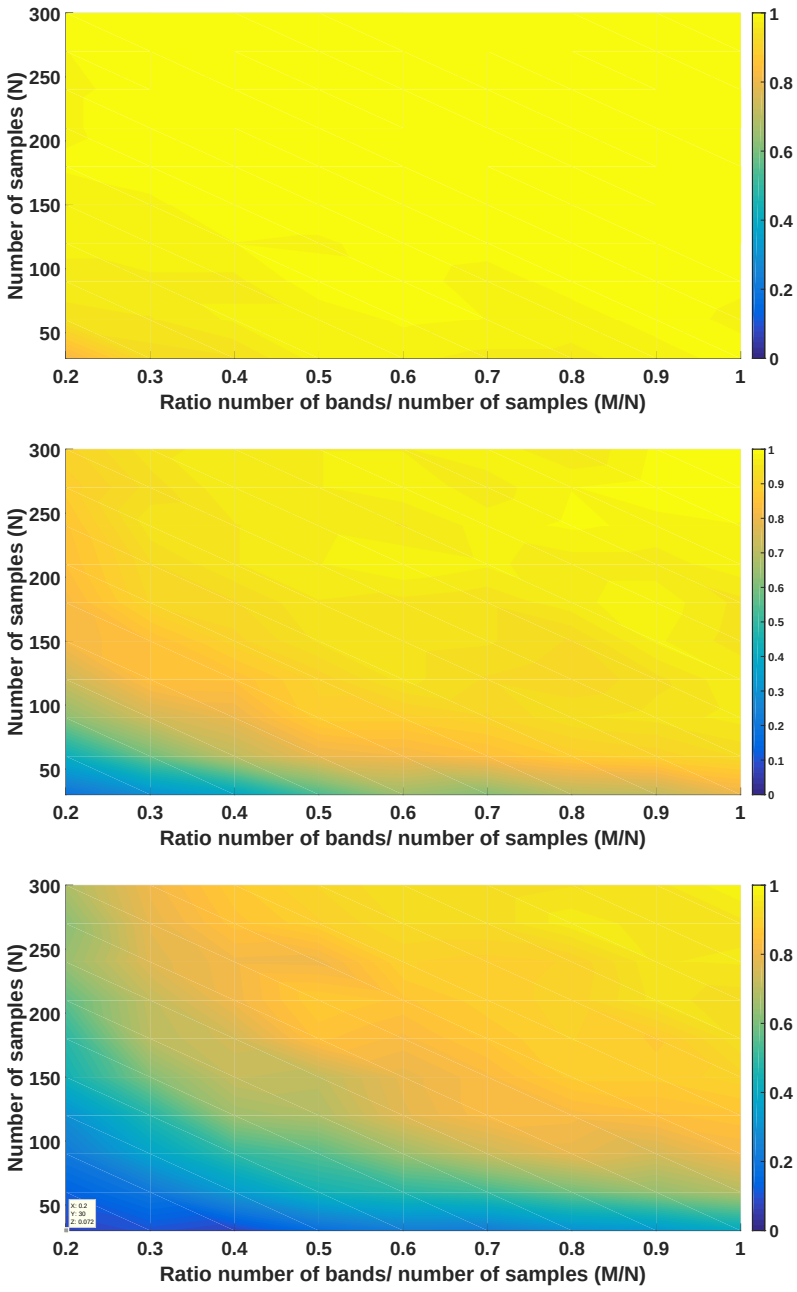

Figure 4: The success rate of finding the true support as a function of the number of samples (y-axis) and the ratio between the number of bands in the dictionary and the number of samples (x-axis), for different values of K. Top, $K = 3$, middle, $K = 7$, and bottom, $K = 13$.

wideband dictionaries and letting

$$\lambda = 0.3 \max_{i=1,\ldots,B} |\mathbf{d}_i^H \mathbf{y}| \tag{10}$$

where $\mathbf{d}_i$ denotes the ith column of $\mathbf{D}$. In the top figure, the signal contains three sinusoids, and it is clearly the case that the banded dictionary is enable to retrieve the true support for all setting of N and M/N, except for the case when $N = 30$ and $M/N < 7$. In the middle and bottom figures, where $K = 7$ and $K = 11$, respectively, it is shown that when the number of sinusoids in the signal increases, a larger number of samples is needed to allow for a successful reconstruction, which is reasonable, as one needs more information to be able to correctly estimate more parameters. However, the banded dictionary is able to retrieve the true support as long as the number of samples is big enough and the ratio M/N is not too small. It is further clear from the figures, that the banded dictionary actually retrieves the true support even though $M < N$. Examining the part of the signal that is unexplained by the support, one may note that no bands outside the true support were activated. Thus, only the bands that either were part of the true support, or that were adjacent to a band included in the true support, were activated. The reason why some of the adjacent bands were activated is that when the true frequency is very close to the left (or right) limit of the band, it will also activate the adjacent bands.

The proposed approach is not the only way to form a wideband dictionary. For example, one could populate the dictionary using discrete prolate spheroid sequences (DPSS) [30]. For an integer Q and with real-valued $0 < W < \frac{1}{2}$, the DPSS are a set of Q discrete-time sequences for which the amplitude spectrum is band-limited. The most interesting property of the DPSS for our discussion is the fact that the energy spectrum of the dictionary elements are highly concentrated in the range $[-W, W]$, suggesting that the DPSS could be a suitable basis for the candidates in a wideband dictionary, where the candidates are formed such that each covers a $1/B$-th part of the spectrum. In the numerical section below, we examine how the use of DPSS candidates compare to the integrated wideband candidates in (8).

4 Complexity analysis

To illustrate the computational benefits of using the wideband dictionary as compared to forming the full dictionary, we proceed with our example of determining

K M-D sinusoids by solving (6) using the popular ADMM algorithm [31]. In order to do so, we first transform the problem into a vector form reminiscent to (5), and split the variable β into two variables, here denoted $\mathbf{x}$ and $\mathbf{z}$, after which the optimization problem may be reformulated as

$$\underset{\mathbf{x},\mathbf{z}}{\text{minimize}} \ \frac{1}{2}||\mathbf{y}-\mathbf{A}\mathbf{x}||_2^2+\lambda||\mathbf{z}||_1 \quad \text{subj. to} \quad \mathbf{x}=\mathbf{z} \tag{11}$$

having the (scaled) augmented Lagrangian

$$\frac{1}{2}||\mathbf{y}-\mathbf{A}\mathbf{x}||_2^2+\lambda||\mathbf{z}||_1+\frac{\rho}{2}||\mathbf{x}-\mathbf{z}+\mathbf{u}||_2^2 \tag{12}$$

where $\mathbf{u}$ is the scaled dual variable and ρ is the step length (see [31] for a detailed discussion on the ADMM). The minimization is thus formed by iteratively solving (12) for $\mathbf{x}$ and $\mathbf{z}$, as well as updating the scaled dual variable $\mathbf{u}$. This is done by finding the (sub-)gradient for $\mathbf{x}$ and $\mathbf{z}$ of the augmented Lagrangian, and setting it to zero, fixing the other variables to their latest values. The steps for the jth iteration are thus

$$\mathbf{x}^{(j+1)}=\left(\mathbf{A}^H\mathbf{A}+\rho\mathbf{I}\right)^{-1}\left(\mathbf{A}^H\mathbf{y}+\mathbf{z}^{(j)}-\mathbf{u}^{(j)}\right) \tag{13}$$

$$\mathbf{z}^{(j+1)}=S(\mathbf{x}^{(j+1)}+\mathbf{u}^{(j)},\lambda/\rho) \tag{14}$$

$$\mathbf{u}^{(j+1)}=\mathbf{u}^{(j)}+\mathbf{x}^{(j+1)}-\mathbf{z}^{(j+1)} \tag{15}$$

where $(\cdot)^H$ denotes the Hermitian transpose, $(\cdot)^{(j)}$ the jth iteration, and $S(\mathbf{v},\varkappa)$ is the soft threshold operator, defined as

$$S(\mathbf{v},\varkappa)=\frac{\max\left(|\mathbf{v}|-\varkappa,0\right)}{\max\left(|\mathbf{v}|-\varkappa,0\right)+\varkappa}\odot\mathbf{v} \tag{16}$$

where $\odot$ denotes the element-wise multiplication for any vector $\mathbf{v}$ and scalar $\varkappa$.

The computationally most demanding part of the resulting ADMM implementation is to form the inverse in (13) and to calculate $\mathbf{A}^H\mathbf{y}$. These steps are often done by QR factorizing the inverse in (13) prior to the iteration, so that this part is only calculated once. After this, the QR factors are used when forming the inner product. To give a simple example on the difference between the two types of dictionaries, we exclude any further computational speed-ups and show the difference on brute force computations of the above ADMM. This is done

to give an idea on the effect $P < N$ has on the computational complexity. The total computational cost for the step in (13) depends on the size of the matrix $\mathbf{A}$. Let $N = \prod_{m=1}^{M} N_m$ and $P = \prod_{m=1}^{M} P_m$, then $\mathbf{A}$ is a $N \times P$ matrix. If $P < N$, computing the inverse will cost approximately P^3 operations, plus an additional P^2N operations to form the Gram matrix $\mathbf{A}^H\mathbf{A}$. Furthermore, to compute $\mathbf{A}^H\mathbf{y}$ requires PN operations, and the final step to compute $\mathbf{x}$ costs P^2 operations. If instead $P > N$, one may make use of the Woodbury matrix identity [32], allowing the inverse to be formed using N^3+3PN^2 operations, whereafter one has to compute $\mathbf{A}^H\mathbf{y}$ and the final matrix-vector multiplication, together costing $PN + P^2$ operations. In total, the x-step will have the cost of roughly $P^3+(N+1)P^2+NP$, if $P < N$, or $N^3 + 3PN^2 + PN + P^2$, if $N < P$.

Since using the banded dictionary allows for a smaller dictionary, one may calculate the computational benefit of using the integrated dictionary as compared to just using an ordinary dictionary with large P. Consider using only a single-stage narrowband dictionary, $\mathbf{D}_1$, with $P > N$ dictionary elements. This requires $C_1 = N^3 + 3PN^2 + P^2 + PN$ operations if using the above ADMM solution, with the dictionary $\mathbf{D}_1$ in the place of $\mathbf{A}$ in (13)-(15). If, on the other hand, one uses a multiple-stage wideband dictionary with N dictionary elements in the initial coarse dictionary, $\mathbf{B}_1$ (which is more than required, but simplifies the calculations), the cost of forming the first stage (coarse) minimization is $C_2 = 2(N^3 + N^2)$. By taking the difference, i.e., forming

$$\begin{aligned} R = C_1 - C_2 = N^3 + 3PN^2 + P^2 + \\ + PN - 2(N^3 + N^2) \end{aligned}$$

one obtains the available computational resources, R, that are left for the dictionaries of the zoomed-in stages, without increasing the overall computational cost above that of the narrowband dictionary solution. Let $\mathbf{B}_z$ denote the zoomed-in dictionary with ηN number of bands, where $0 < \eta < 1$ denotes the ratio between the number of available bands in the dictionary and the number of samples. Then, one may deduce the grid size for each $\mathbf{B}_z$ that is allowed without increasing the overall computational complexity as compared to using the narrowband dictionary by solving

$$R = KI_z\left((\eta N)^3 + (N+1)(\eta N)^2 + \eta N^2\right)$$

where I_z denotes the number of zooming steps and K the number of sinusoids in the signal. To illustrate the resulting difference, consider the following settings:

$P = 1000$, $N = 100$, $K = 5$, and $\eta = 2/3$. To only use half the resources that are needed to solve the full narrowband problem, one may, using the wideband dictionary, use 4 stages of zooming, resulting in a grid spacing of roughly 10^{-9}, as compared to 10^{-3} for the narrowband dictionary. One may of course also use a zooming procedure when using the narrowband dictionaries, although this would increase the risk of missing any off-grid component. This means that the smallest number of dictionary elements, for the narrowband dictionary to avoid missing any off-grid components, is $P = N$, and thus the wideband dictionary would need only at most η^2 of the computational resources needed for the ordinary dictionary, at each zooming stage.

It is worth stressing that the wideband dictionary framework introduced here is not limited to the Lasso-style minimizations such as the one examined in (6). There are many other popular methods that could be implemented using this approach. As an example of how the wideband dictionary can be applied for other typical sparse estimation algorithms, consider the SPICE algorithm [14,27], formed as the solution to

$$\underset{\mathbf{p},\boldsymbol{\sigma}\geq 0}{\text{minimize}}\ \mathbf{y}^*\mathbf{R}^{-1}\mathbf{y} + ||\mathbf{p}||_1 + ||\boldsymbol{\sigma}||_1 \tag{17}$$

where

$$\mathbf{R}(\mathbf{p}) = \mathbf{A}\mathbf{P}\mathbf{A}^* \tag{18}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{B} & \mathbf{I} \end{bmatrix} \tag{19}$$

$$\mathbf{p} = \begin{bmatrix} p_1 & \dots & p_M \end{bmatrix}^T \tag{20}$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 & \dots & \sigma_N \end{bmatrix}^T \tag{21}$$

$$\tilde{\mathbf{p}} = \begin{bmatrix} \mathbf{p}^T & \boldsymbol{\sigma}^T \end{bmatrix}^T \tag{22}$$

$$\mathbf{P} = \text{diag}\,(\tilde{\mathbf{p}}) \tag{23}$$

Alternatively, one may consider the more general $\{r, q\}$-SPICE formulation[1] [33, 34]

$$\underset{\mathbf{p}\geq 0}{\text{minimize}}\ \mathbf{y}^*\mathbf{R}^{-1}\mathbf{y} + ||\mathbf{p}||_r + ||\boldsymbol{\sigma}||_q \tag{24}$$

[1] In this formulation, we assume that the columns of the dictionaries are normalized to have unit norm.

Using the wideband dictionary over **B** in (17) or (24) will allow for much smaller dictionaries as opposed to using ordinary sinusoidal dictionaries. Many other sparse reconstruction techniques may be extended similarly. Generally, the wideband dictionary may be used either as an energy detector which finds the parts of the spectrum that have most energy, or in a zooming procedure similar to the one described above.

5 Numerical examples

In this section, we proceed to examine the performance of the proposed method, initially illustrating that the use of a two-stage wideband estimator will have the same estimation quality as when using the ordinary (one-stage) narrowband Lasso estimator.

5.1 One-dimensional data

We initially considered a signal consisting of $N = 75$ samples containing $K = 3$ (complex-valued) sinusoids corrupted by a zero-mean white Gaussian noise with SNR $= 10$ dB. In each simulation, the sinusoidal frequencies are drawn from a uniform distribution, over $[0, 1)$, with all amplitudes having unit magnitude and phases drawn from a uniform distribution over $[0, 2\pi)$. The performance is then computed using three different dictionaries, namely the (ordinary) narrowband dictionary, **D**, with $P = 1000$ and $P = 75$ elements, respectively, and the proposed wideband dictionary, **B**, using $B_1 = 75$ elements, followed by a second-stage narrowband dictionary using $B_2 = 25$ elements per active band. For each dictionary, we evaluate the performance for varying values of the user parameter α using $\lambda = \alpha\lambda_{max}$, where $\lambda_{max} = \max_i |\mathbf{x}_i^H \mathbf{y}_i|$ is the smallest tuning parameter value for which all coefficients in the solution are zero [19]. Here, $\mathbf{x}_i$ denotes either the ith column of the **D** dictionary or the ith column of the **B** dictionary. Each estimated result is then compared to the ground truth, counting the number of correctly estimated and underestimated model orders. The results are shown in Figure 5. As can be seen from the figure, the best results are achieved when $\alpha \leq 0.65$, in which case the proposed wideband dictionary, using $B_1 = 75$ bands, followed by a second stage narrowband dictionary, with $B_2 = 25$ for each activated band, have similar performance to the narrowband dictionary using $P = 1000$ dictionary elements.

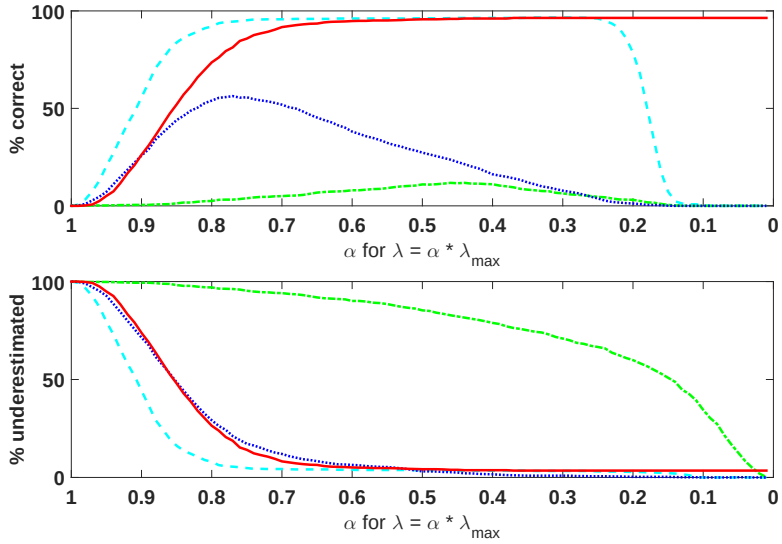

Figure 5: The probability of (top) correctly estimating and (bottom) underestimating the number of spectral lines, for the (single-stage) narrowband dictionary, using $P = 1000$ elements (cyan, dashed) and $P = 75$ elements (green, dot-dashed), and for the initial wideband dictionary, using $B_1 = 75$ elements (blue, dotted), and the (two-stage) wideband dictionary, using $B_1 = 75$ elements, together with $B_2 = 25$ elements per activated bands in the refining dictionary (red, solid).

Proceeding, we asses the mean-square error (MSE) for the two-stage dictionary, showing the MSE as a function of SNR for the first-stage wideband dictionary, $\mathbf{B}_1$, and second-stage wideband refining dictionary, $\mathbf{B}_2$. Here, and in the following, we consider situations where the number of elements in the dictionary is less than number of samples. As was described before, this is a situation where the performance of narrowband dictionaries can deteriorate seriously. For this experiment, we considered a signal with $N = 300$ samples containing $K = 2$ (complex-valued) sinusoids, being corrupted by different levels of zero-mean white Gaussian noise with SNR in the range $[5, 20]$ dB. Figure 6 shows the resulting MSE for the Lasso estimator for the estimates with correctly estimated model order; for runs with the correct model order estimation we also removed

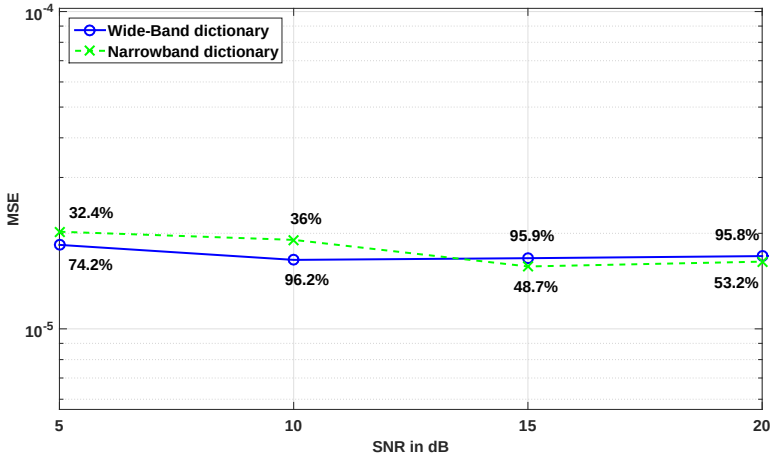

Figure 6: Mean-square error curves for different SNR levels for the single-stage narrowband dictionary, using $P = 100$, as compared to the two-stage dictionary, using $B_1 = 20$ integrated wideband elements in the first stage, followed by $B_2 = 5$ wideband elements in the second stage. The percentage of correct model order estimation (excluding outliers) is shown as a percentage on top of the corresponding MSE value.

outliers from the final MSE calculation. We consider an estimate as an outlier if $|f - \hat{f}| > \Delta f$, where Δf was defined as two times the possible resolution, where possible resolution is defined as $1/P$ for the narrowband dictionary and $1/(B_1 \cdot B_2)$ for the wideband dictionary. Figure 7 shows the MSE for the same experiment done using the SPICE estimator. The number of outliers removed for the Lasso estimator was: 4, 0, 0, 0 for the wideband dictionary and 7, 16, 10 and 11 for the narrowband dictionary (corresponding to SNRs of 5, 10, 15, and 20 dB). The number of outliers removed for the SPICE estimator was; 17, 1, 1, 0 for the wideband dictionary and 52, 80, 117, and 103 for the narrowband dictionary. As can be seen from the figures, the two-stage dictionary using a wideband dictionary using $B_1 = 20$ bands, followed by a refining dictionary using $B_2 = 5$ wideband elements, achieves the same performance as the single-stage narrowband dictionary using $P = 100$ elements in terms of resolution. However, the narrow-band dictionary will for this case fail to reliably restore the signal with

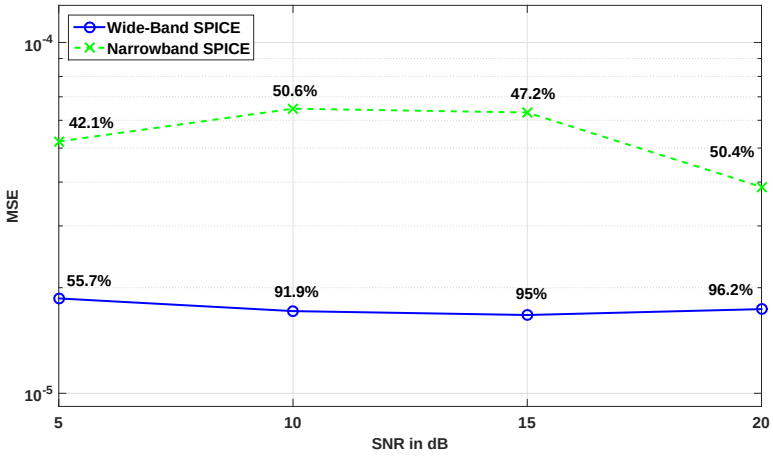

Figure 7: Mean-square error curves for different SNR levels for the single-stage narrowband dictionary, using $P = 100$, as compared to the two-stage dictionary, using $B_1 = 20$ integrated wideband elements in the first stage, followed by $B_2 = 5$ wideband elements in the second stage. The percentage of correct model order estimation (excluding outliers) is shown as a percentage on top of the corresponding MSE value.

reconstruction success rates of merely $30 - 50\%$.

Table 1 shows the corresponding complexity cost of some of the different settings in the numerical section. Note that to simplify the comparison this is the complexity of solving the ADMM without utilizing any structures of the dictionary matrices.

Next, we consider non-uniformly sampled data with $N = 400$ samples, for $K = 2$ sinusoids. For this experiment, we also added a third estimation step for the iterative wideband dictionary. After initial estimation with $B_1 = 10$ wideband dictionary elements, we zoom into the active bands with $B_2 = 10$ dictionary elements per active band, and then once again with $B_3 = 5$ dictionary elements. In spite of the three stage zooming, the method requires considerably less computational operations as compared to using a corresponding narrowband dictionary, but results in better performance both in terms of resolution and model-order accuracy. The resulting MSEs are shown in Figure 8. All results are computed

Settings	Complexity ratio
$D = 1000, N = 200, K = 3$	1
$B_1 = 20, B_2 = 5$	897
$B_1 = 20, B_2 = 40$	27
$D = 1000$,$N = 400, K = 3$	26
$B_1 = 10, B_2 = 10, B_3 = 5$	1000

Table 1: Complexity reduction compared to using the full dictionary and the distance between the final grid for different settings. Here, D indicates the narrowband dictionary, whereas B_1, B_2 indicates the two-stage dictionary using B_1 wideband elements in the first stage, followed by B_2 wideband elements in the second-stage.

using 1000 Monte-Carlo simulations.

5.2 Two-dimensional data

In this subsection, we present results on a 2-D data set. In this example, each dimension is sampled uniformly with $N = 100$ samples. We compare a narrowband dictionary with $P = 49$ elements per dimension with the wideband dictionary using $B_1 = 7$ bands per dimension in the first step and a wideband dictionary with $B_2 = 7$ elements per active band in a second (zooming) step. Here, we use two separate wideband dictionaries, the first, $\mathbf{B}$, using integrated dictionary elements as defined in (7), and the second, $\mathbf{B}_{DPSS}$, which contains elements based on DPSS. For the DPSS-based dictionary, we used a sequence length of $Q = 100$ and $W = 1/2.1$. Using $W < 1/2.1$ results in dictionary elements which concentrate energy in a more narrow band and are therefore not suitable for the dictionary with $B_1 = B_2 = 7$ elements. We considered a signal containing $K = 2$ (complex-valued) sinusoids per dimension, with the signal being corrupted by a zero-mean white Gaussian noise. In each simulation, the sinusoidal frequencies are drawn from a uniform distribution, over $[0, 1)$, with all the amplitudes having unit magnitude. The two dictionaries are compared against each other based on the MSE performance in the same manner as in the previous subsection, with the MSE being calculated as the average value for both dimensions if the model order estimate for the iteration was correct. Outliers are removed before the MSE calculation. The number of outliers removed was: 6, 15, 25, 30 for the $\mathbf{B}_{DPSS}$ dictionary and 20, 22, 17 and 24 for the narrowband dictionary

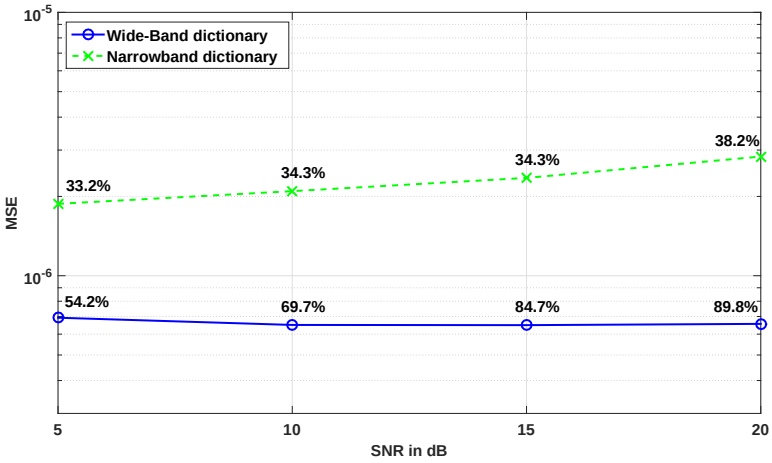

Figure 8: Signal estimation for non-uniform sampling: mean-square error curves for different SNR levels for the single-stage narrowband dictionary, using $P = 200$ elements, as compared to the three-stage dictionary, using $B_1 = 10$ integrated wideband elements in the first stage, followed by $B_2 = 10$ and $B_3 = 5$ wideband dictionaries in the second stage and third stage per active band detected in the previous stage. The correct model order estimations are shown in percentage above each point.

(corresponding to SNRs of 5, 10, 15, and 20 dB). The wideband dictionary $\mathbf{B}$ did not result in any outliers. The percentages of correct model order estimates are shown for each SNR value. Figure 9 shows the resulting MSE curves. It can be seen that the wideband dictionary with integrated sinusoids outperforms the DPSS-based wideband dictionary both in terms of MSE and model-order accuracy. Comparing to using the narrowband dictionary, it can be seen that both wideband dictionaries outperform it both in terms of MSE and model-order estimation. Also in this example, the wideband dictionaries provide a considerable reduction in computational complexity as well as a robustness in terms of estimating off-grid components. All results are computed using 100 Monte-Carlo simulations.

Using the same setup as described above we also evaluated the performance of the proposed approach when the number of sinusoids to detect is higher. Again,

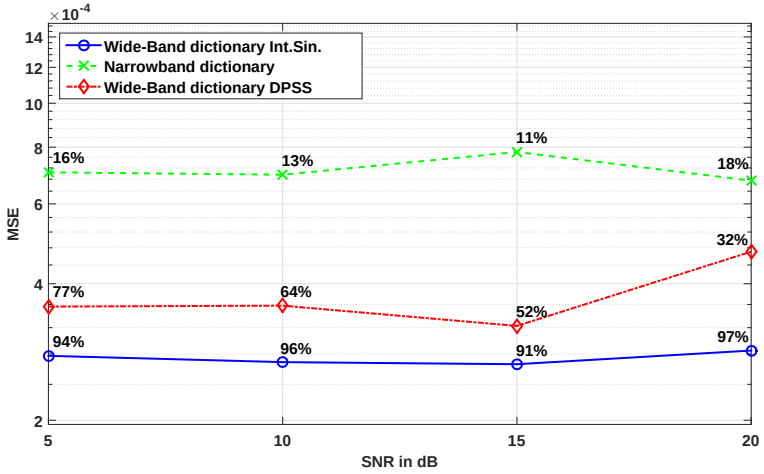

Figure 9: Signal estimation in two dimensions: mean-square error curves for different SNR levels for the single-stage narrowband dictionary $\mathbf{D}$, using $P = 100$ per dimension, as compared to the two-stage dictionaries (DPSS based and integrated sinusoids based), using $B_1 = 7$ wideband elements in the first stage, followed by $B_2 = 7$ wideband elements in the second stage (per active band).

we considered the ordinary narrowband dictionary, $\mathbf{D}$, and the wideband dictionary, $\mathbf{B}$, from the previous experiment. We calculated the percentage of correct model order estimation for signals with $K = 4, 6, 8$, and 10 (complex-valued) sinusoids. The results were computed using 100 Monte-Carlo simulations; the correct model order estimation percentages for different SNR levels are shown in Figure 10. The best regularization parameters λ for solving the Lasso for each case were found beforehand with the grid-search method. For this, we selected the range of parameter $\alpha \in [0.05, 0.7]$ with the step-size 0.05 and ran 100 Monte-Carlo simulations for each model order and then picked the best parameter for the selected model order based on model order accuracy. For the two-step wideband dictionary, a grid-search was done for the set of α parameter for the both stages. It can be clearly seen that for situations where the number of elements in the dictionary is lower than the number of samples, the narrow-band dictionary fails to produce any meaningful results.

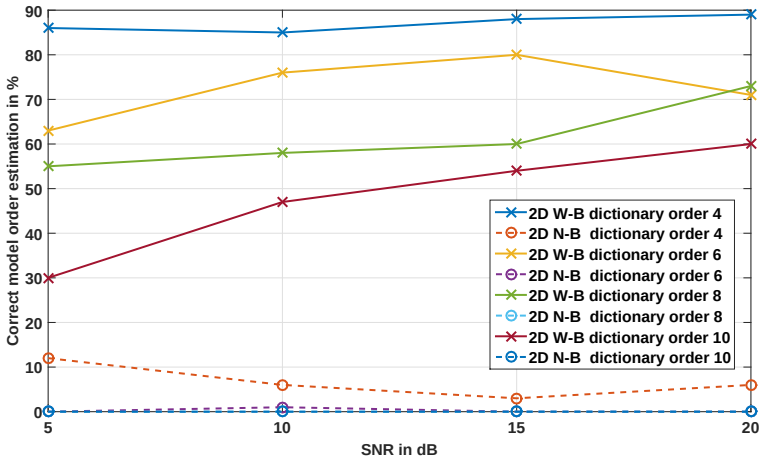

Figure 10: Percentage of correct model order esimations for different number of sinusoids and for different SNR levels for wideband dictionary (W-B) and narrowband dictionary (N-B).

6 Conclusion

In this paper, we have introduced a wideband dictionary framework, allowing for a computationally efficient reconstruction of sparse signals. Wideband dictionary elements are formed as spanning bands of the considered parameter space. In the first stage, one may typically use a coarse grid using the integrated wideband dictionary locating the bands of interest, whereafter non-active parts of the parameter space are discarded. In the next stage, a refining dictionary can be used to more precisely determine the parameters of interest on the active bands from the previous step, allowing for an iterative zooming procedure. The technique is illustrated for the problem of estimating multidimensional sinusoids corrupted by Gaussian noise, showing that the same accuracy can be achieved, although at a computationally substantially lower cost and with much less risk of missing any off-grid components. The proposed framework is here illustrated for the Lasso and SPICE estimators, but other sparse reconstruction techniques may be extended similarly.

References

[1] M. Unser and P. Tafti, *An introduction to sparse stochastic processes*, Cambridge University Press, 2013.

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

[3] E. J. Candès and M. B. Wakin, "An Introduction To Compressive Sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, March 2008.

[4] 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.

[5] D.L. Donoho, "Compressed Sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289–1306, 2006.

[6] M. A. Herman and T. Strohmer, "Genral Deviants: An Analysis of Perturbations in Compressed Sensing," *IEEE J. Sel. Topics in Signal Processing*, vol. 4, no. 2, pp. 342–349, April 2010.

[7] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to Basis Mismatch in Compressed Sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182 –2195, May 2011.

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

[9] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed Sensing Off the Grid," *IEEE Trans. Inform. Theory*, vol. 59, no. 11, pp. 7465–4790, Nov 2013.

[10] Y. Chi and Y. Chen, "Compressive Two-Dimensional Harmonic Retrieval via Atomic Norm Minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1030–1042, Feb 2015.

[11] Z. Yang and L. Xie, "Enhancing Sparsity and Resolution via Reweighted Atomic Norm Minimization," *IEEE Trans. Signal Process.*, vol. 64, no. 4, pp. 995–1006, Feb 2016.

[12] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The Convex Geometry of Linear Inverse Problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, Dec 2012.

[13] 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.

[14] 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, no. 1, pp. 35–47, Jan 2011.

[15] P. Stoica and P. Babu, "SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation," *Signal Processing*, vol. 92, no. 7, pp. 1580–1590, July 2012.

[16] 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.

[17] 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.

[18] L. E. Ghaoui, V. Viallon, and T. Rabbani, "Safe Feature Elimination for the LASSO and Sparse Supervised Learning Problems," 2011, Publication: eprint arXiv:1009.4219v2.

[19] R. Tibshirani, J. Bienand, J. Friedman, T. Hastieand N. Simon, J. Taylor, and R. J. Tibshirani, "Strong rules for discarding predictors in lasso-type problems," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 74, no. 2, pp. 245–266, 2012.

[20] Z. J. Xiang, Y. Wang, and P. J. Ramadge, "Screening Tests for Lasso Problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, 2016.

[21] A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval, "A Dynamic Screening Principle for the Lasso," in *Proceedings of the 22nd European Signal Processing Conference*, Lisbon, Portugal, 1-5 September 2014.

[22] O. Fercoq, A. Gramfort, and J. Salmon, "Mind the Duality Gap: Safe Rules for the Lasso," 2015, Publication: eprint arXiv:1505.03410v3.

[23] J. Liu, Z. Zhao, J. Wang, and J. Ye, "Safe Screening With Variational Inequalities and Its Application to LASSO," 2014, Publication: eprint arXiv:1307.7577v3.

[24] Z. Yang and L. Xie, "Frequency-Selective Vandermonde Decomposition of Toeplitz Matrices With Applications," 2016, Publication: eprint arXiv:1605.02431.

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

[26] P. Stoica, P. Babu, and J. Li, "SPICE : a novel covariance-based sparse estimation method for array processing," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 629 –638, Feb. 2011.

[27] 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.

[28] S. Sahnoun, E. H. Djermoune, and D. Brie, "Sparse Modal Estimation of 2-D NMR Signals," in *38th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Vancouver, Canada, May 26-31 2013.

[29] 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.

[30] D. Slepian, "Prolate Spheroidal Wave Functions, Fourier Analysis, and Uncertainty - V: the Discrete Case," *The Bell System Technical Journal*, vol. 57, no. 5, pp. 1371–1430, May-June 1978.

[31] 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.

[32] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, 4^{th} edition, 2013.

[33] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "A Generalization of the Sparse Iterative Covariance-based Estimator," in *42nd IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, New Orleans, USA, March, 5-9 2017.

[34] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Generalized Sparse Covariance-based Estimation," *Elsevier Signal Processing*, 2017, Accepted for publication.

H

Paper H

Multi-dimensional Grid-less Estimation of Saturated Signals

Filip Elvande[1], Johan Swärd[1], and Andreas Jakobsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*

Abstract

This work proposes a multidimensional frequency and amplitude estimator tailored for noise corrupted signals that have been clipped. Formulated as a sparse reconstruction problem, the proposed algorithm estimates the signal parameters by solving an atomic norm minimization problem. The estimator also exploits the waveform information provided by the clipped samples, incorporated in the form of linear constraints that have been augmented by slack variables as to provide robustness to noise. Numerical examples indicate that the algorithm offers preferable performance as compared to methods not exploiting the saturated samples.

Key words: Atomic norm, de-clipping, gridless reconstruction

1 Introduction

Many forms of practical measurements suffer from clipping, for instance due to limitations in the dynamic span of the analog-to-digital (AD) converter, possibly necessitated by needs of resolution, or by additive interference offsetting the signal unexpectedly. In such cases, the measured signal is occasionally saturated at its minimum or maximum values, typically requiring these samples to be treated as missing. One may attempt to reconstruct such samples using various forms of interpolation or by using estimators of the relevant signal information that allow for missing samples (see, e.g., [1–4]). There have also been methods proposed for using gain masks in the sampling stage as to mitigate the effects of clipping [5], as

well as post-processing methods for countering the harmonic distortion induced by clipping [6].

More recently, several reconstruction schemes exploiting an assumed signal sparsity have been proposed. In [7], the authors extend the concept of image inpainting (see, e.g., [8]) to audio signals in order to reconstruct the clipped samples. In [9], the authors utilize a compressed sensing formulation, as well as exploit features of the human auditory system, in order to increase the perceived signal quality. Other approaches include iterative hard thresholding [10], greedy methods [11], smooth regularization [12], social sparsity exploiting temporal dependence [13], and non-negative matrix factorization [14], whereas theoretical recovery guarantees have been studied in [15]. The related field of estimation and reconstruction of 1-bit signals is also attracting interest (see, e.g., [16, 17]). Such signals only retain the sign of the sampled analog waveform, which can be seen as an extreme form of clipping. The problem of signal reconstruction of more generally quantized measurements has been explored in [18].

In this work, we propose an algorithm that exploits the assumed *a priori* structure of the signals of interest. This structure may, for instance, be that the signal can be well modelled as a sum of decaying sinusoids, as is common in areas such as spectroscopy, or by some other well structured signal. By formulating an estimator of the unknown parameters detailing the assumed signal structure, taking into account both the available and the saturated samples, we propose a sparse reconstruction algorithm that is able to exploit the information available in the saturated samples, while still being robust to the presence of additive noise. Robustness against noise is achieved by not enforcing hard clipping constraints, i.e., the proposed estimator does not constrain the reconstructed waveform to saturate at precisely the same samples as the observed signal, as this would make the estimator vulnerable to amplitude bias. Instead, the clipping information is taken into account by adding linear constraints, relaxed using slack variables, allowing also the noise to cause saturation.

Assuming that the measured signal consists of relatively few signal components, the algorithm may be constructed as a sparse reconstruction problem using a signal dictionary formed using the assumed signal waveforms, taking into account the saturation information of the clipped samples. In order to allow the signal of interest to be formed over a continuous parameter space, we express the resulting optimization as an atomic norm minimization. The atomic norm has previously been successfully exploited to develop estimators allowing for off-grid

components (see, e.g., [19–21]). Here, we propose a similar formulation to exploit the structure of the assumed signal, while incorporating information of the saturated samples. We note that an approach reminiscent of ours was recently proposed in [22] for line spectrum estimation from 1-bit samples, although that work considered only noise-free signals. In audio application the signal may often be well modeled as sum of harmonically related sinusoids. As noted above, clipped samples often occurs in audio applications. In such cases, it is reasonable to instead exploit the expected harmonic structure of speech or tonal music. This may be done by extending the here proposed idea using the atomic norm framework developed in [23].

2 Proposed estimator

In this section we present the proposed estimator. We begin by initially presenting the one-dimensional (1-D) version for real-valued sinusoidal data, and then generalize the formulation to allow for both complex and multidimensional data.

2.1 One dimensional case

To illustrate the proposed algorithm, we assume that the signal of interest, $\mathbf{y}$, consists of N samples of a sum of K real-valued sinusoids corrupted by an additive Gaussian noise, such that

$$\mathbf{y} = \mathbf{A}\mathbf{d} + \mathbf{e} \tag{1}$$

where $\mathbf{d} \in \mathbb{R}^{K\times 1}$ denotes the amplitude vector, $\mathbf{e}$ the additive noise, and

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \dots & \mathbf{a}_K \end{bmatrix}$$
$$\mathbf{a}_k = \begin{bmatrix} \cos(2\pi f_k t_1 + \varphi_k) & \dots & \cos(2\pi f_k t_N + \varphi_k) \end{bmatrix}^T$$

with f_k and φ_k denoting the kth frequency and phase, respectively. Furthermore, let Ω^-, Ω^+, and Ω denote the indices of $\mathbf{y}$ that are clipped from below, from above, and all the non-clipped indices of $\mathbf{y}$, respectively. In order to reconstruct the signal of interest successfully, one needs to estimate the signal parameters, here the frequencies and amplitudes, as well as the model order, K, all which are assumed to be unknown. The typical way of dealing with the clipped samples in $\mathbf{y}$ is to treat these as missing data points, and simply omit them from the

measurement vector. The unknown parameters, and the model order, are then estimated using a technique that allows for missing samples, such as, e.g., [24].

It is well known that dictionary techniques using a predefined grid suffers when the true parameters are not on the grid. To alleviate this problem, and also account for the missing samples, we here make use of an atomic-norm formulation. Defining an atom set as $\mathcal{A} = \{\mathbf{a}(f,\varphi) : f \in [0,1],\ \varphi \in [0,2\pi)\}$ and an atom as $[\mathbf{a}(f,\varphi)]_t = \cos(2\pi f t + \varphi)$, a signal containing a sum over K sinusoids can be expressed as

$$\mathbf{y}^* = \sum_{k=1}^{K} d_k \mathbf{a}(f_k, \varphi_k) \tag{2}$$

The atomic norm is defined as

$$\begin{aligned} ||\mathbf{y}||_{\mathcal{A}} &= \inf\{t > 0 :\ \mathbf{y} \in t\ \mathrm{conv}(\mathcal{A})\} \\ &= \inf_{d_k \geq 0, \varphi_k \in [0,2\pi), f_k \in [0,1]} \left\{ \sum_k d_k :\ \mathbf{y} = \sum_k d_k \mathbf{a}(f_k, \varphi_k) \right\} \end{aligned}$$

where $\mathrm{conv}(\mathcal{A})$ denotes the convex hull of $\mathcal{A}$. This formulation can be interpreted as finding the sparsest linear combination of atoms that constitutes the signal. In [20], it was shown that the atomic norm may be expressed equivalently as a (computationally tractable) semidefinite program (SDP) on the form

$$\begin{aligned} \underset{x,\mathbf{z},\mathbf{u}}{\text{minimize}} \quad & x + u_1 + \frac{1}{2}\|\mathbf{y}_\Omega - \mathbf{z}_\Omega\|_2^2 \\ \text{subject to} \quad & \begin{bmatrix} x & \mathbf{z}^H \\ \mathbf{z} & \mathbf{T}(\mathbf{u}) \end{bmatrix} \succeq 0 \\ & \mathbf{T}(\mathbf{u}) \in \mathbb{T}^{N \times N} \end{aligned} \tag{3}$$

where $\mathbf{y}_\Omega$ and $\mathbf{z}_\Omega$ denote the measured signal and the signal model over the non-saturated samples, respectively, whereas $\mathbb{T}$ denotes the set of all $N \times N$ symmetric Toeplitz matrices, with $\mathbf{T}(\mathbf{u})$ denoting the Toeplitz matrix with $\mathbf{u}$ on its first column. Since the problem in (3) is an SDP, it is also convex, and may as a result be computed using solvers, such as, e.g., CVX [25]. The third term in (3) penalizes the difference between the observed samples for the measured signal and the optimization variable, $\mathbf{z}$, corresponding to the noise-free, non-clipped signal. Solving this optimization problem will yield a signal, $\mathbf{z}$, where the missing values

have been estimated, a scalar, x, corresponding to the sum of the absolute values of the amplitudes, and the vector $\mathbf{u}$ that forms the Toeplitz matrix $\mathbf{T}(\mathbf{u})$, from which, using, e.g., a Vandermonde decomposition, the resulting frequency estimates may be found. This approach has been shown to be very efficient in both retrieving the missing samples, as well as estimating the frequencies [20, 26]. However, it should be noted that the approach treats the clipped samples as missing, and is thus wasteful in the sense that the information that the measured signal is above (or below) the clipping limit is not incorporated in the optimization problem.

To alleviate this, we proceed to extend the minimization to also incorporate this information in the saturated samples. Clearly, since a clipped sample may not always indicate that the true wave form should be clipped, this should be taken into consideration when forming the optimization problem. This discrepancy appears when the true wave form is inside the measurable region, but the noise pushes the sample over (under) the saturation limit. To incorporate this effect, we introduce the variables $\boldsymbol{\varepsilon}^+$ and $\boldsymbol{\varepsilon}^-$. These capture the discrepancy between the observed and the true signal waveform for the samples saturated due to the additive noise, and should preferably be as small as possible to reduce the influence of this problem. Incorporating both changes, the minimization may be expressed as

$$\begin{aligned} \underset{x,\mathbf{z},\boldsymbol{\varepsilon},\mathbf{u}}{\text{minimize}} \quad & \mu(x+u_1) + \lambda\|\boldsymbol{\varepsilon}\|_1 + \frac{1}{2}\|\mathbf{y}_\Omega - \mathbf{z}_\Omega\|_2^2 \\ \text{subject to} \quad & \begin{bmatrix} x & \mathbf{z}^H \\ \mathbf{z} & \mathbf{T}(\mathbf{u}) \end{bmatrix} \succeq 0 \\ & \mathbf{T}(\mathbf{u}) \in \mathbb{T}^{N\times N} \\ & \mathbf{z}_{\Omega^+} + \boldsymbol{\varepsilon}^+ \geq \gamma \\ & \mathbf{z}_{\Omega^-} + \boldsymbol{\varepsilon}^- \leq -\gamma \end{aligned} \tag{4}$$

where

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \boldsymbol{\varepsilon}^+ & \boldsymbol{\varepsilon}^- \end{bmatrix} \tag{5}$$

and with μ and λ denoting user parameters governing the denoising and the regularization of the $\boldsymbol{\varepsilon}$, respectively.

As before, the resulting frequency estimates are then found by using, e.g., a Vandermonde decomposition on $\mathbf{T}(\mathbf{u})$. Although estimates of the amplitudes, d_k,

can also be obtained from such a Vandermonde decomposition, these estimates will be biased towards zero due to the regularization parameter μ. In order to refine the amplitude estimates, we therefore propose to additionally solve

$$\begin{aligned} \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\varepsilon}}{\text{minimize}} \quad & \frac{1}{2}\left\|\mathbf{y}_\Omega - \mathbf{Z}_\Omega(\hat{\mathbf{f}})\mathbf{r}\right\|_2^2 + \lambda\|\boldsymbol{\varepsilon}\|_1 \\ \text{subject to} \quad & \mathbf{Z}_{\Omega^+}(\hat{\mathbf{f}})\mathbf{r} + \boldsymbol{\varepsilon}^+ \geq \gamma \\ & \mathbf{Z}_{\Omega^-}(\hat{\mathbf{f}})\mathbf{r} + \boldsymbol{\varepsilon}^- \leq -\gamma \end{aligned} \tag{6}$$

where

$$\mathbf{Z}(\hat{\mathbf{f}}) = \begin{bmatrix} \mathbf{c}_1 & \dots & \mathbf{c}_K & \mathbf{s}_1 & \dots & \mathbf{s}_K \end{bmatrix} \tag{7}$$

$$\mathbf{c}_k = \left[\cos(2\pi\hat{f}_k t_1) \ \dots \ \cos(2\pi\hat{f}_k t_N)\right]^T \tag{8}$$

$$\mathbf{s}_k = \left[\sin(2\pi\hat{f}_k t_1) \ \dots \ \sin(2\pi\hat{f}_k t_N)\right]^T \tag{9}$$

$$\mathbf{r} = \begin{bmatrix} \boldsymbol{\alpha}^T & \boldsymbol{\beta}^T \end{bmatrix}^T . \tag{10}$$

Here, $\hat{\mathbf{f}}$ denotes the vector of frequency estimates obtained from the Vandermonde decomposition of $\mathbf{T}(\mathbf{u})$, and $\mathbf{Z}(\hat{\mathbf{f}})$ is the dictionary matrix of cosine and sine atoms corresponding to these frequencies. Thus, the resulting optimization is a least squares (LS) problem, constrained to satisfy the clipping conditions of the observed signal, where the slack variable $\boldsymbol{\varepsilon}$ is again exploited to provide robustness against noise. Using trigonometric identities, each amplitude estimate, $\hat{d}_k$, is then constructed from the minimizing vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ as

$$\hat{d}_k = \sqrt{\alpha_k^2 + \beta_k^2} \, . \tag{11}$$

2.2 *D*-dimensional case

In this section, we generalize the proposed estimator to allow for complex-valued data as well as expand the optimization problem to also be able to deal with *D*-dimensional data. We begin by defining what we in this paper mean by complex clipping. Let γ denote the clipping level. The real and the imaginary parts of the signal are typically treated separately, resulting in the following definition of complex clipping:

Definition 2.1. Clipping of complex-valued data.
Sample n in a complex signal $\mathbf{y}^{\text{unclipped}}$ is subjected to clipping if either

(i) $|\mathfrak{Im}\{\mathbf{y}_n^{\text{unclipped}}\}| > \gamma$

and will assume the value $\mathfrak{Im}\{\mathbf{y}_n\} = \gamma \text{sign}(\mathfrak{Im}\{\mathbf{y}_n^{\text{unclipped}}\})$, and/or

(ii) $|\mathfrak{Re}\{\mathbf{y}_n^{\text{unclipped}}\}| > \gamma$

and will assume the value $\mathfrak{Re}\{\mathbf{y}_n\} = \gamma \text{sign}(\mathfrak{Re}\{\mathbf{y}_n^{\text{unclipped}}\})$ where $\mathfrak{Re}$ and $\mathfrak{Im}$ denote the real and the imaginary part, respectively. □

It is worth noting that Definition 2.1 allows the real part of a sample to be correctly recorded, whereas the imaginary part is clipped, or vice versa. It also allows both the real- and the imaginary parts of the sample to be clipped, as well as being below γ in both dimensions.

In [27], the atomic norm framework was expanded to allow for two-dimensional data, and this was further generalized in [28] for the multidimensional case, where it also was shown that the Vandermonde decomposition that is used in the one-dimensional case to retrieve the frequency estimates has a multidimensional counterpart, and may thus be used for frequency retrieval for multidimensional data. We now present the D-dimensional version of the proposed estimation algorithm for clipped complex data.

Let $\boldsymbol{\mathcal{Y}}$ be the $N_1 \times N_2 \times \cdots \times N_D$ data tensor and let $\mathbf{y}$ be the vectorized version of $\boldsymbol{\mathcal{Y}}$ with size $N \times 1$, where $N = \prod_{n=1}^{N} N_n$. We here define the vectorization as being operated on the mode-1 matricization, or unfolding (see also [29]). Thus, in the two-dimensional case, the vectorization reduces to stacking the columns of the 2-D data matrix. The order of the vectorization is not important as long as it is consistent. The atomic norm minimization problem taking the clipping information into account may then be formulated as

$$
\begin{aligned}
\underset{x,\mathbf{z},\mathbf{T},\boldsymbol{\varepsilon}}{\text{minimize}} \quad & \mu\left(x + \operatorname{tr}\{\mathbf{T}\}\right) + \lambda\|\boldsymbol{\varepsilon}\|_1 \\
& + \frac{1}{2}\|\mathfrak{Re}(\mathbf{y}_{\Omega_{\mathfrak{Re}}}) - \mathfrak{Re}(\mathbf{z}_{\Omega_{\mathfrak{Re}}})\|_2^2 \\
& + \frac{1}{2}\|\mathfrak{Im}(\mathbf{y}_{\Omega_{\mathfrak{Im}}}) - \mathfrak{Im}(\mathbf{z}_{\Omega_{\mathfrak{Im}}})\|_2^2 \\
\text{subject to} \quad & \begin{bmatrix} x & \mathbf{z}^H \\ \mathbf{z} & \mathbf{T} \end{bmatrix} \succeq 0 \\
& \mathfrak{Re}(\mathbf{z}_{\Omega^+_{\mathfrak{Re}}}) + \boldsymbol{\varepsilon}^+_{\mathfrak{Re}} \geq \gamma \\
& \mathfrak{Im}(\mathbf{z}_{\Omega^+_{\mathfrak{Im}}}) + \boldsymbol{\varepsilon}^+_{\mathfrak{Im}} \geq \gamma \\
& \mathfrak{Re}(\mathbf{z}_{\Omega^-_{\mathfrak{Re}}}) + \boldsymbol{\varepsilon}^-_{\mathfrak{Re}} \geq -\gamma \\
& \mathfrak{Im}(\mathbf{z}_{\Omega^-_{\mathfrak{Im}}}) + \boldsymbol{\varepsilon}^-_{\mathfrak{Im}} \geq -\gamma
\end{aligned}
\tag{12}
$$

where $\boldsymbol{\varepsilon} = \begin{bmatrix} \boldsymbol{\varepsilon}^+_{\mathfrak{Re}} & \boldsymbol{\varepsilon}^+_{\mathfrak{Im}} & \boldsymbol{\varepsilon}^-_{\mathfrak{Re}} & \boldsymbol{\varepsilon}^-_{\mathfrak{Im}} \end{bmatrix}$, with $\Omega_{\mathfrak{Re}}$ and $\Omega_{\mathfrak{Im}}$ denote the subset of the elements corresponding to the samples in $\mathbf{y}$ that have not been clipped in their real and imaginary parts, respectively. The notation $\Omega^+_{\mathfrak{Re}}$ and $\Omega^-_{\mathfrak{Re}}$ denote for the subset of elements corresponding to the samples in $\mathbf{y}$ that have their real part clipped with positive sign and negative sign, respectively, and similar for $\Omega^+_{\mathfrak{Im}}$ and $\Omega^-_{\mathfrak{Im}}$. As before, $\boldsymbol{\varepsilon}$ acts as a slack-variable, allowing the clipping of the real and imaginary parts to be considered caused by the noise and not the true waveform. Furthermore, $\mathbf{T}$ is a D-level Toeplitz matrix (see [28] for a detailed definition). In the two-dimensional case, the 2-level $N_1N_2 \times N_1N_2$ Toeplitz matrix becomes

$$
\mathbf{T} = \begin{bmatrix} \mathbf{T}_0 & \mathbf{T}_{-1} & \cdots & \mathbf{T}_{-(N_1-1)} \\ \mathbf{T}_1 & \mathbf{T}_0 & \cdots & \mathbf{T}_{-(N_1-2)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{T}_{N_1-1} & \mathbf{T}_{N_1-2} & \cdots & \mathbf{T}_0 \end{bmatrix}
\tag{13}
$$

where each

$$
\mathbf{T}_{n_1} = \begin{bmatrix} z_{n_1,0} & z_{n_1,-1} & \cdots & z_{n_1,-(N_2-1)} \\ z_{n_1,1} & z_{n_1,0} & \cdots & z_{n_1,-(N_2-2)} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n_1,N_2-1} & z_{n_1,N_2-2} & \cdots & z_{n_1,0} \end{bmatrix}
\tag{14}
$$

for $n_1 = -(N_1 - 1), \ldots, N_1 - 1$, is an $N_2 \times N_2$ Toeplitz matrix.

Similar to (4), the first term in (12) minimizes $\mathbf{z}^H \mathbf{T}^{-1} \mathbf{z}$ and controls the size of $\mathbf{T}$. The second term regularizes the slack variables $\boldsymbol{\varepsilon}$ using the one norm. This corresponds to letting only a few of the elements in $\boldsymbol{\varepsilon}$ to be active in the solution. The third and fourth term bounds the variable $\mathbf{z}$ to be close to the noisy signal, in a two norm sense, corresponding to a data fitting term. Note that the two last terms corresponds to the proposed denoising term in [26]. Having obtained estimates of the set of frequencies from the optimal $\mathbf{T}$ estimate, one may then estimate the amplitudes by solving a LS problem analogous to that in (6), namely

$$\begin{aligned}
\underset{\mathbf{d},\boldsymbol{\varepsilon}}{\text{minimize}} \quad & \lambda \|\boldsymbol{\varepsilon}\|_1 + \frac{1}{2} \left\| \mathfrak{Re}(\mathbf{y}_{\Omega_{\mathfrak{Re}}}) - \mathfrak{Re}\left(\mathbf{Z}_{\Omega_{\mathfrak{Re}}}(\hat{\mathbf{f}})\mathbf{d} \right) \right\|_2^2 \\
& + \frac{1}{2} \left\| \mathfrak{Im}(\mathbf{y}_{\Omega_{\mathfrak{Im}}}) - \mathfrak{Im}\left(\mathbf{Z}_{\Omega_{\mathfrak{Im}}}(\hat{\mathbf{f}})\mathbf{d} \right) \right\|_2^2 \\
\text{subject to} \quad & \mathfrak{Re}\left(\mathbf{Z}_{\Omega^+_{\mathfrak{Re}}}(\hat{\mathbf{f}})\mathbf{d} \right) + \boldsymbol{\varepsilon}^+_{\mathfrak{Re}} \geq \gamma \\
& \mathfrak{Im}\left(\mathbf{Z}_{\Omega^+_{\mathfrak{Im}}}(\hat{\mathbf{f}})\mathbf{d} \right) + \boldsymbol{\varepsilon}^+_{\mathfrak{Im}} \geq \gamma \\
& \mathfrak{Re}\left(\mathbf{Z}_{\Omega^-_{\mathfrak{Re}}}(\hat{\mathbf{f}})\mathbf{d} \right) + \boldsymbol{\varepsilon}^-_{\mathfrak{Re}} \geq -\gamma \\
& \mathfrak{Im}\left(\mathbf{Z}_{\Omega^-_{\mathfrak{Im}}}(\hat{\mathbf{f}})\mathbf{d} \right) + \boldsymbol{\varepsilon}^-_{\mathfrak{Im}} \geq -\gamma
\end{aligned} \tag{15}$$

where $\mathbf{d}$ is the vector of amplitudes of the K D-dimensional sinusoids and $\mathbf{Z}(\hat{\mathbf{f}})$ is the $N \times K$ matrix defined as

$$\mathbf{Z}(\hat{\mathbf{f}}) = \mathbf{A}^{(d)}(\hat{\mathbf{f}}) \otimes \ldots \otimes \mathbf{A}^{(1)}(\hat{\mathbf{f}}) \tag{16}$$

where $\otimes$ denotes the Kroenecker product and

$$\mathbf{A}^{(d)}(\hat{\mathbf{f}}) = \left[\begin{array}{ccc} \mathbf{a}_1^{(d)}(\hat{\mathbf{f}}) & \ldots & \mathbf{a}_K^{(d)}(\hat{\mathbf{f}}) \end{array} \right] \tag{17}$$

$$\mathbf{a}_k^{(d)}(\hat{\mathbf{f}}) = \left[\begin{array}{ccc} \mathrm{e}^{(2i\pi \hat{f}_k^{(d)} t_1^{(d)})} & \ldots & \mathrm{e}^{(2i\pi \hat{f}_k^{(d)} t_N^{(d)})} \end{array} \right]^T \tag{18}$$

3 Numerical evaluation

We proceed to examine the performance of the proposed algorithm, initially striving to estimate the frequencies and amplitudes of a clipped 1-D signal consisting of $K = 2$ sinusoids using (4) and (6), respectively. This is done by forming 500

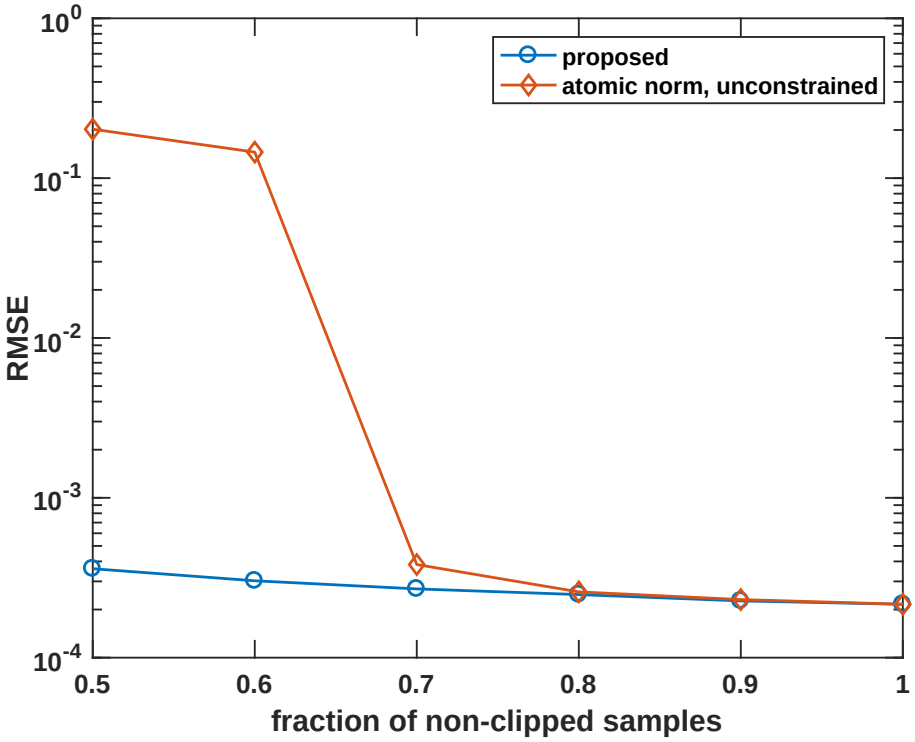

Figure 1: RMSE for the frequency estimates produced by the constrained and unconstrained estimators, as a function of the fraction of unclipped data.

Monte Carlo (MC) simulations of $N = 100$ samples, where in each simulation the frequencies, f_1 and f_2, are drawn uniformly on the intervals $[0.08, 0.1]$ and $[0.11, 0.13]$, respectively. Furthermore, the amplitudes and phases are drawn uniformly on $[0.8, 1.2]$ and $[0, 2\pi)$, respectively. We examine the performance for two cases, the first varying the number of clipped samples while keeping the signal to noise ratio (SNR) fixed at 15 dB, where SNR is defined as

$$\text{SNR} = 10 \log_{10} \left(\frac{P_y}{\sigma^2} \right) \tag{19}$$

with P_y denoting the power of the true signal. In the second, the SNR is instead varying for the case of 30% clipped samples. The performance is measured using the sum of the root mean squared error (RMSE) for the frequencies, f_1 and f_2,

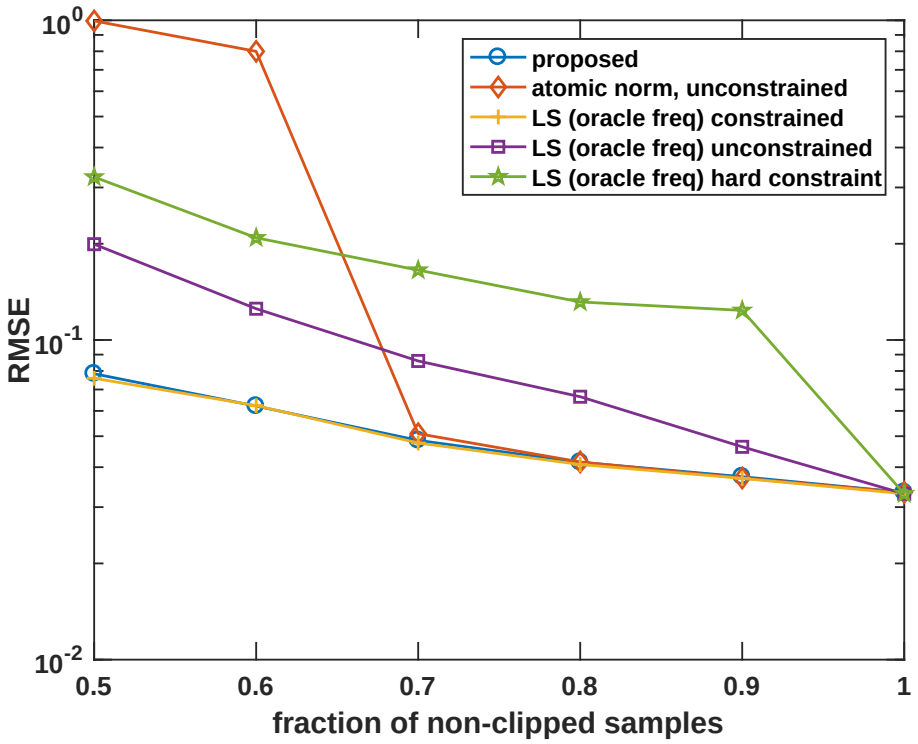

Figure 2: RMSE for the amplitude estimates produced by the constrained and unconstrained estimators, as well as the LS estimators, as a function of the fraction of unclipped data.

as well as for the amplitudes, d_1 and d_2, where the RMSE for each component is defined as

$$\text{RMSE} = \sqrt{\frac{1}{P}\sum_{k=1}^{P} |\hat{\vartheta}_k - \vartheta_k|^2} \tag{20}$$

where ϑ is the true parameter, $\hat{\vartheta}_k$ is the kth MC estimate of that parameter, and P is the number of MC simulations. For all simulation settings, we use $\mu = 1$ and $\lambda = 1$. As comparison, we also include the performance of the atomic norm minimization which only considers the unclipped samples, i.e., where the estimates are obtained by solving (4) without including the constraints, or equivalently,

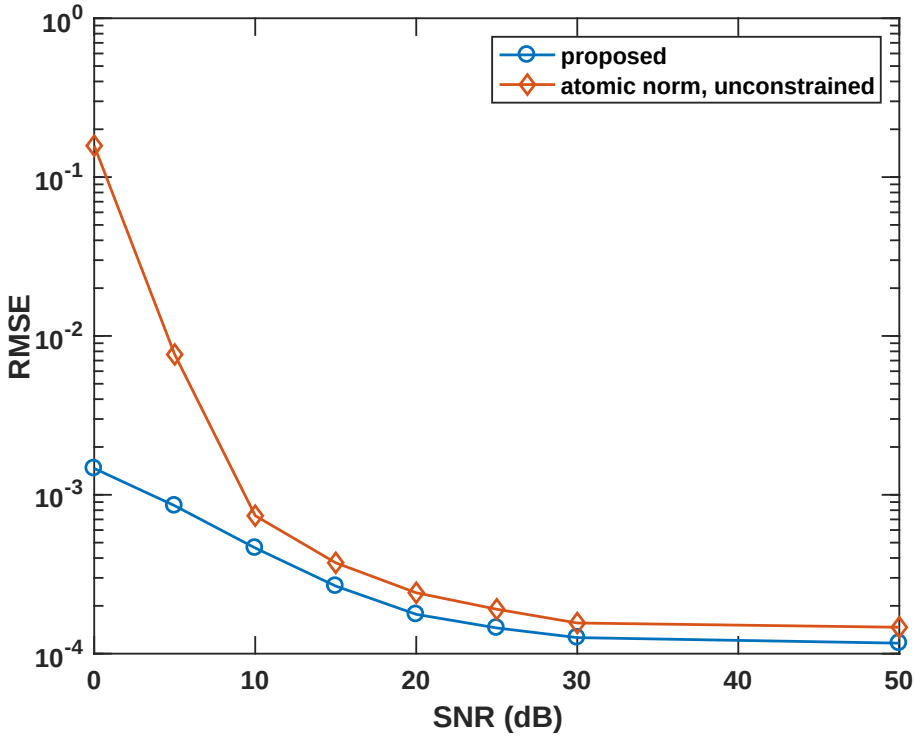

Figure 3: RMSE for the frequency estimates produced by the constrained and unconstrained estimators, as a function of the SNR.

by setting $\lambda = 0$. For the obtained set of frequency estimates, the amplitudes are then estimated using (6). Also, for the amplitude estimates, we include comparisons with three least squares (LS) estimators that have been given oracle knowledge of the frequencies. The first of these LS estimators estimates $\mathbf{d}$ by solving (6) using the ground truth frequencies. The second considers only the unclipped samples, i.e., solves (6) using $\lambda = 0$. Lastly, the third estimators uses hard clipping constraint, i.e., $\boldsymbol{\varepsilon} = \mathbf{0}$, or equivalently, (6) is solved using $\lambda = +\infty$.

For the scenarios considering varying fractions of non-clipped samples, the RMSE for the frequency and amplitude estimates are presented in Figures 1 and 2, respectively. As can be seen from Figure 1, the proposed estimator is robust to the occurrence of clipped samples, and produces estimates whose accuracy is

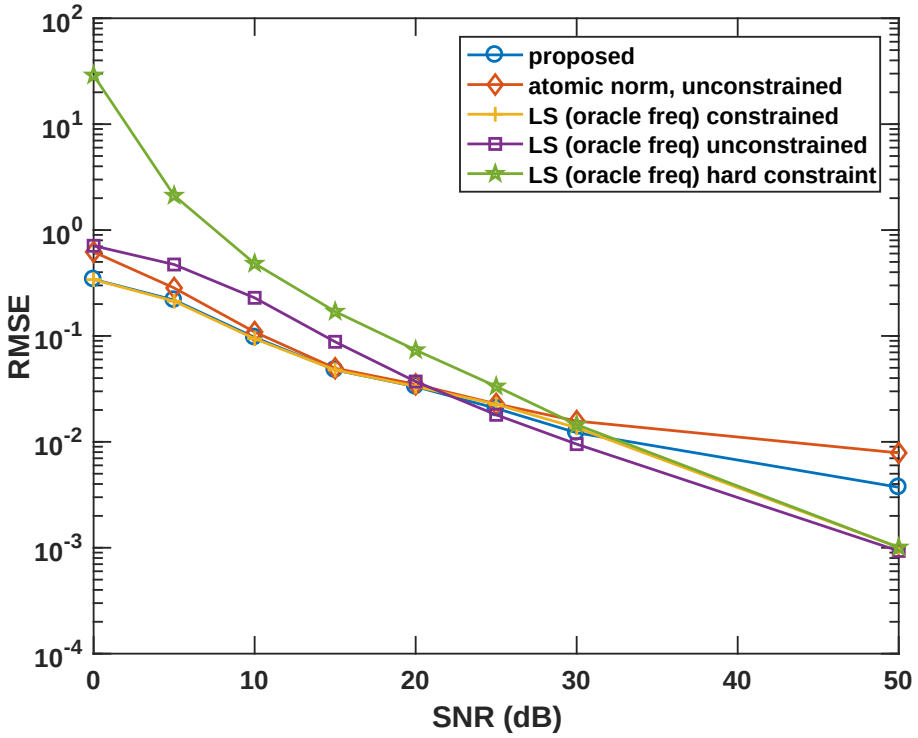

Figure 4: RMSE for the amplitude estimates produced by the constrained and unconstrained estimators, as well as the least squares estimators, as a function of the fraction of unclipped data.

close to unaffected by the fraction of clipped samples. By comparison, the alternative estimator that only considers the non-clipped samples breaks down as the fraction of non-clipped samples decreases. As can be seen in Figure 2, the robustness of the proposed estimator then translates into improved amplitude estimates. In the figure, it can be seen that the three estimators utilizing (6) perform the best, as they can salvage information contained in the clipped samples; the LS estimator operating on only non-clipped samples suffers from the smaller samples size, whereas the constrained LS estimator using no slack variables suffers from a positive amplitude bias induced by the corrupting noise component, **e**.

Similar conclusions may be drawn from Figures 3 and 4, showing the RMSE

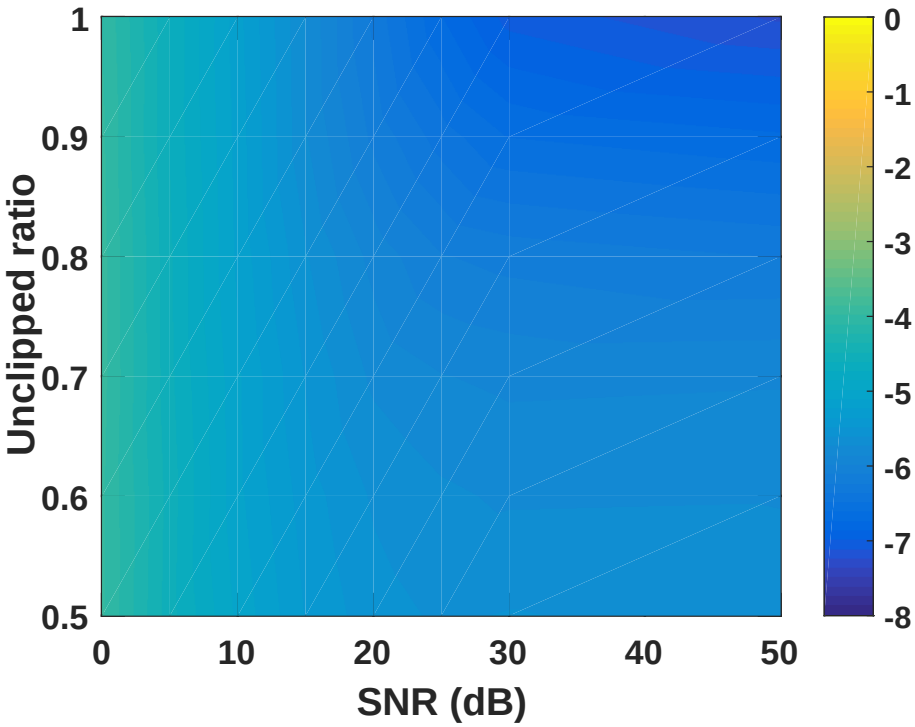

Figure 5: RMSE for the frequency estimates produced by the proposed multidimensional estimator, as a function of the fraction of unclipped data and the SNR level.

for the frequency and amplitude estimates for the scenario with varying SNR. Also in this case, the proposed estimator displays greater robustness, and is less sensitive to noise than the estimators utilizing only the non-clipped samples. It may be noted from the figure that the RMSEs of the two estimators do not converge as the SNR increases; the proposed estimator consistently outperforms the estimator using only the non-clipped samples. Interestingly, for the highest SNR considered, i.e., 50 dB, the three LS estimators have identical performance. This is to be expected, as such a low noise setting renders the slack variable $\boldsymbol{\epsilon}$ superfluous as the constraints are satisfied by the uncorrupted waveform themselves. Also, as can be seen from the figure, for SNRs 25 and 30 dB, the LS estimator considering only non-clipped samples actually performs better than the estimators using

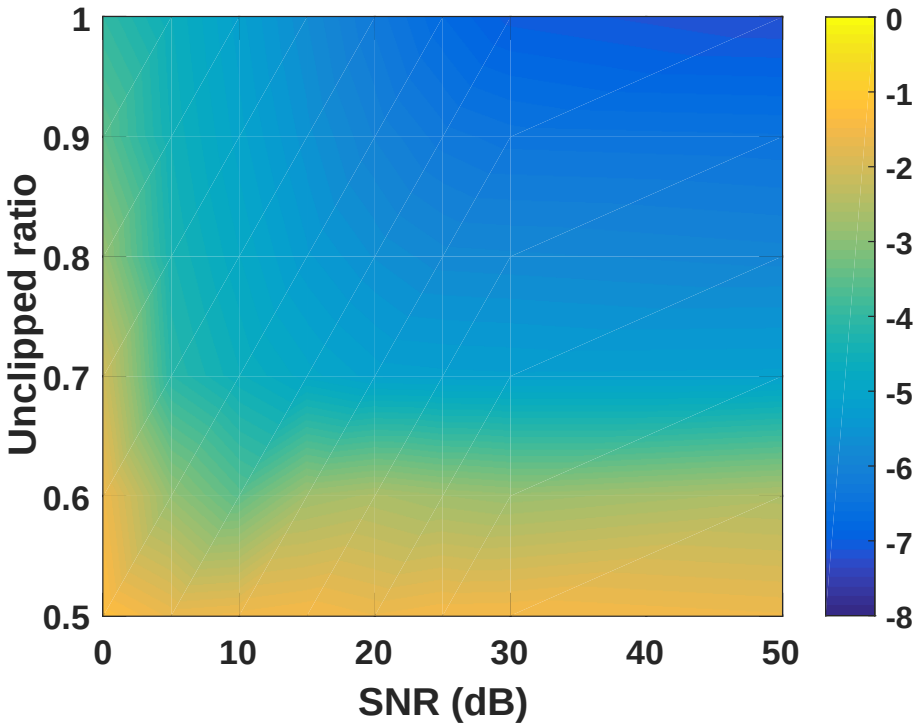

Figure 6: RMSE for the frequency estimates produced by the unconstrained 2-D atomic norm estimator, as a function of the fraction of unclipped data and the SNR level.

(6), which is probably due to the slack variable $\boldsymbol{\varepsilon}$ introducing degrees of freedom that are not beneficial in such high, but not extreme, SNR settings. Furthermore, one may note that the atomic norm-based estimators perform worse than the LS estimators for the highest SNR settings, as it also have to estimate the frequencies.

We proceed by showing the performance of the estimator for multidimensional complex data using (12). All tests were done using 2-D data with size $N_1 = N_2 = 8$, containing two 2-D sinusoids with random phase, frequency, and magnitude. The data was on the form

$$\mathbf{y} = \mathbf{A}^{(2)} \otimes \mathbf{A}^{(1)} \boldsymbol{\alpha} + \mathbf{e} \tag{21}$$

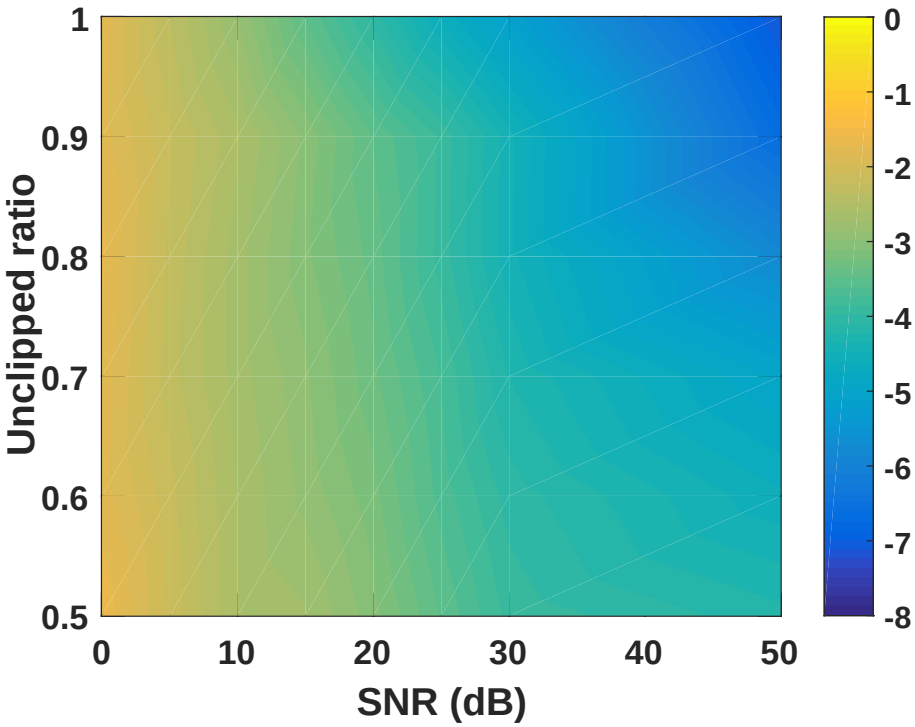

Figure 7: RMSE for the amplitude estimates produced by the proposed multidimensional estimator as a function of the fraction of unclipped data and the SNR level.

where $\boldsymbol{\alpha}$ denotes the $K \times 1$ complex vector corresponding to the amplitudes, $\mathbf{e}$ the additive noise, $\otimes$ the Kronecker product, and the superscript $(\cdot)^{(d)}$ denotes the dimension d. The matrix $\mathbf{A}^{(d)}$, for $d = 1, 2$, are constructed as

$$\mathbf{A}^{(d)} = \left[\begin{array}{ccc} \mathbf{a}_1^{(d)} & \dots & \mathbf{a}_K^{(d)} \end{array} \right] \tag{22}$$

$$\mathbf{a}_k^{(d)} = \left[\begin{array}{ccc} \mathrm{e}^{2i\pi f_k^{(d)} t_1^{(d)}} & \dots & \mathrm{e}^{2i\pi f_k^{(d)} t_N^{(d)}} \end{array} \right]^T \tag{23}$$

The phase was sampled from $[0, 2\pi)$, whereas the frequencies were selected uniformly from $[0, 1]$ but separated $1/N_d$ in each dimension. The magnitudes where uniformly selected between $[0.8, 1.2]$. In the first example, we investigate the RMSE on the frequency estimation of the proposed method compared with the

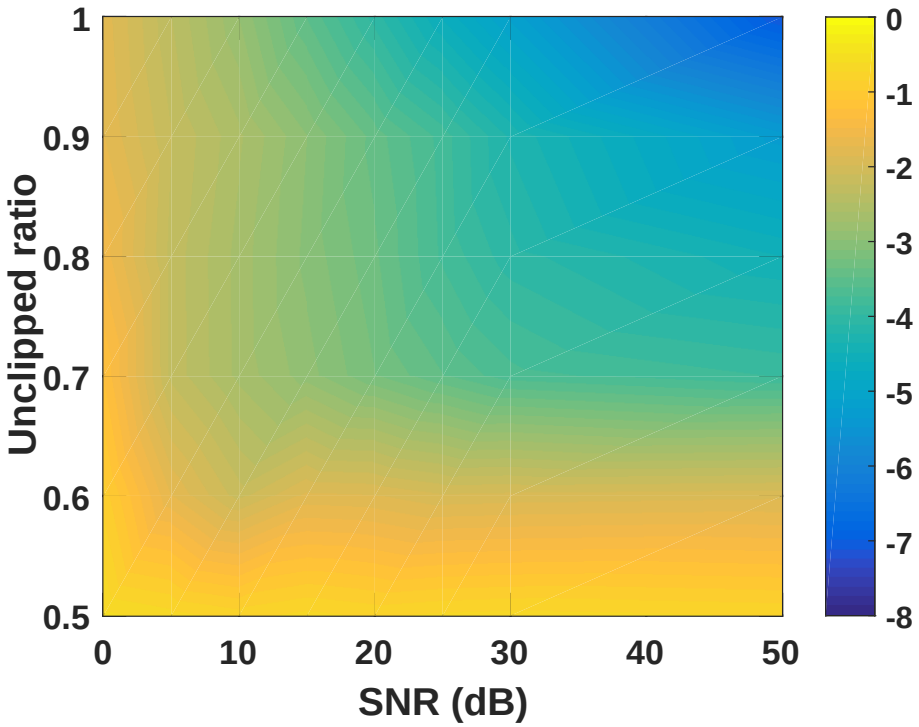

Figure 8: RMSE for the amplitude estimates produced by the 2-D atomic norm estimator as a function of the fraction of unclipped data and the SNR level.

2-D atomic norm proposed in [27], which treats the clipped samples as unknown. The frequency estimates were all retrieved using the MaPP estimator from [28][1]. The RMSE was evaluated for a range of different SNR levels and clipping ratios. The SNR ranged from 0 to 50, and the clipping ratios varied from 0.5 to 1. For each setting of SNR and clipping ratio, 500 Monte-Carlo simulations were done and the user parameters were set to $\mu = \lambda = 1$. Figure 5 and Figure 6 show the performance of the proposed multidimensional estimator and the 2-D atomic norm, respectively. It can be seen from the figures that the result from two estimators differs on two key points. First, the proposed estimator seems relatively

[1]The authors would like to thank Dr. Zai Yang for providing the code for the MaPP method, as well as making us aware of several interesting references.

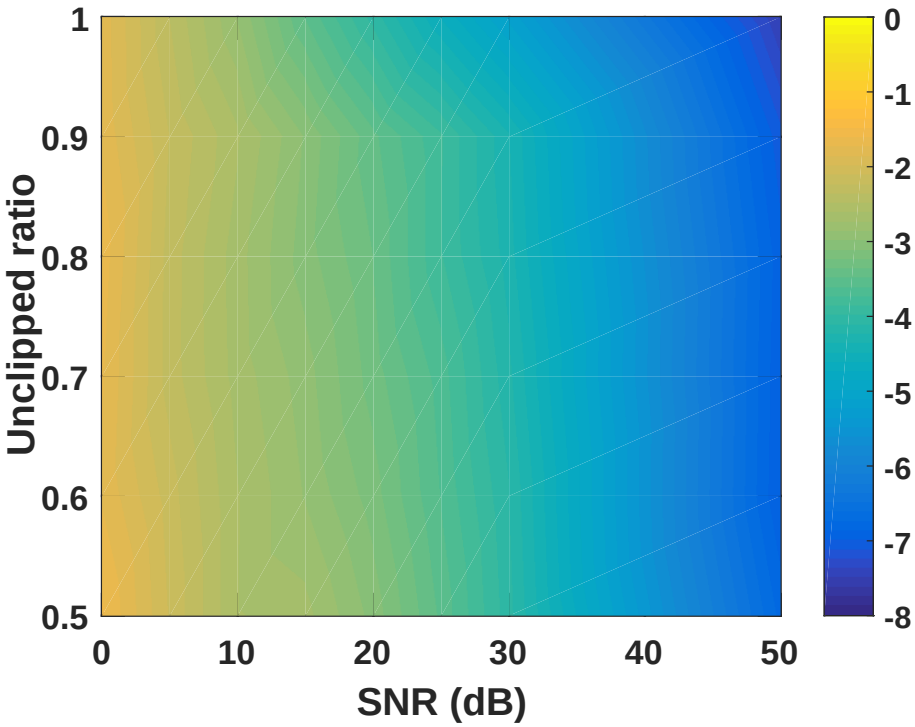

Figure 9: RMSE for the amplitude estimates produced by the constrained ($\lambda = 1$) LS estimator, given oracle knowledge of the true frequencies, as a function of the fraction of unclipped data and the SNR level.

unaffected by the clipping ratio; it is first when the SNR level drops to about 0 dB that any degradation starts to be noticeable. For the 2-D atomic norm estimator, the performance degrades both for low SNR levels and when the number of clipped samples increase. This figure corresponds well with the results for the 1-D case shown in Figures 1 and 3. Incorporating information that the clipped sample should be above (below) the clipping threshold, as well as including the noise effect using $\boldsymbol{\varepsilon}$, clearly shows its benefits.

As in the 1-D case, we proceed to examine the resulting RMSE for the amplitude estimates comparing the proposed method to the 2-D atomic norm estimator, as well as the three LS estimators described above, which are given full knowledge about the true frequencies. Figures 7 and 8 show the RMSE of the

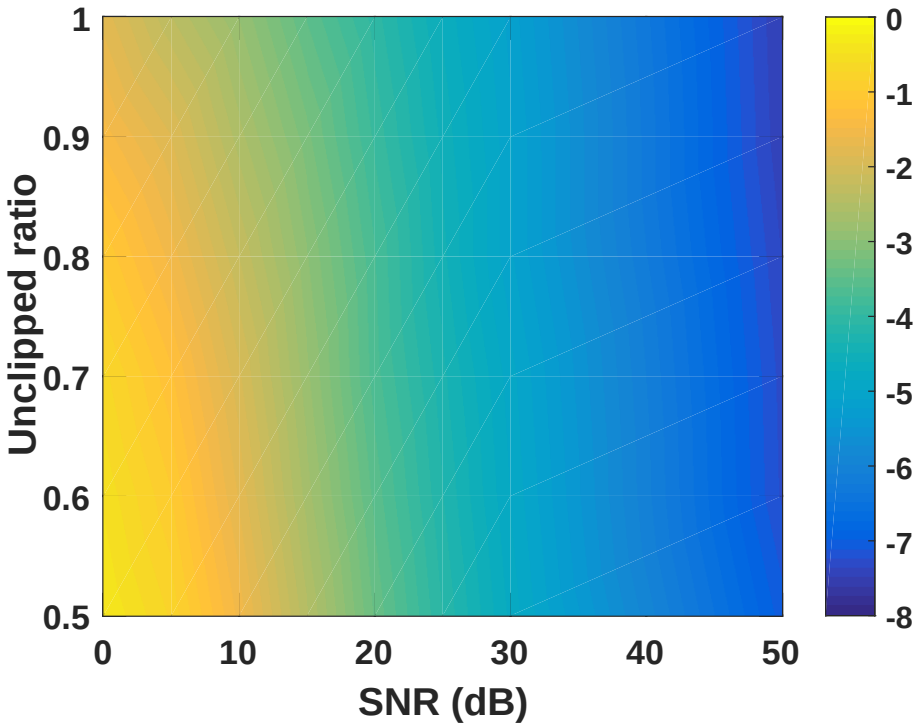

Figure 10: RMSE for the amplitude estimates produced by the unconstrained ($\lambda = 0$) LS estimator, given oracle knowledge of the true frequencies, as a function of the fraction of unclipped data and the SNR level.

amplitude estimates produced by solving (15) using the frequency estimate from the proposed method and the 2-D atomic norm estimator, respectively. Similar to the frequency estimation, the proposed method manage to better estimate the amplitudes. This is not surprising, as it also produced better frequency estimates. Figures 9, 10, and 11 show the three LS estimates with total knowledge of the true frequencies, corresponding to solving (15), with $\lambda = 1$, $\lambda = 0$, and $\lambda \to +\infty$, respectively. As can be seen from the figures, the proposed method in Figure 7 can be seen to produce similar results as the oracle LS estimator with $\lambda = 1$ in Figure 9. The unconstrained LS estimate, shown in Figure 10, seems to be more sensitive to the SNR, especially when the number of clipped samples is large. The final LS estimator, corresponding to the case when $\lambda \to +\infty$, seems even more sensitive

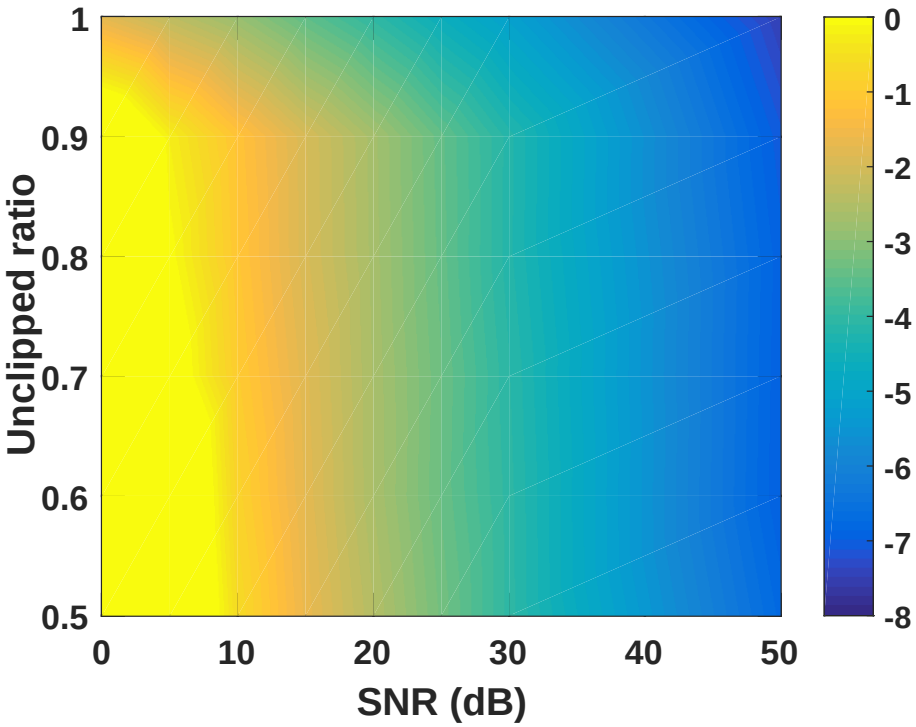

Figure 11: RMSE for the amplitude estimates produced by the hard constrained ($\lambda \to +\infty$) LS estimator, given oracle knowledge of the true frequencies, as a function of the fraction of unclipped data and the SNR level.

to low SNR levels. We can conclude that the proposed method provides a better frequency estimate than the traditional 2-D atomic norm estimator. Furthermore, given these estimates, it is shown that the amplitude estimates are almost as good as given full knowledge about the true frequencies.

4 Conclusions

In this work, we have introduced a sparse reconstruction technique allowing for saturated signal samples. By exploiting the 1-bit information of the saturated samples, as well as allowing for the possibility that the noise causes the signal sat-

uration of signals close to the saturation limits, the proposed estimator is shown to outperform alternative estimators not exploiting such information. The proposed estimator is formed using an atomic norm formulation allowing for a continuous parameter space, and does thus not suffer from the off-grid effects that often deteriorates dictionary based techniques.

References

[1] J. S. Abel and J. O. Smith, "Restoring a Clipped Signal," in *16th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Toronto, Canada, Apr. 14-17 1991, pp. 1745–1748.

[2] B. Porat and B. Friedlander, "ARMA spectral estimation of time series with missing observations," *IEEE Trans. Inform. Theory*, vol. 30, no. 4, pp. 601–602, July 1986.

[3] Y. Wang, J. Li, and P. Stoica, *Spectral Analysis of Signals - The Missing Data Case*, Morgan & Claypool, 2005.

[4] E. Gudmundson, P. Stoica, J. Li, A. Jakobsson, M. D. Rowe, J. A. S. Smith, and J. Ling, "Spectral Estimation of Irregularly Sampled Exponentially Decaying Signals with Applications to RF Spectroscopy," *J. Magn. Reson.*, vol. 203, no. 1, pp. 167–176, March 2010.

[5] P. Smaragdis, "Dynamic Range Extension Using Interleaved Gains," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 17, no. 5, pp. 966–973, Jul. 2009.

[6] F. Esqueda, S. Bilbao, and V. Välimäki, "Aliasing Reduction in Clipped Signals," *IEEE Trans. Signa*, vol. 64, no. 20, pp. 5255–5267, Oct. 2016.

[7] A. Adler, V. Emiya, M. G. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, "Audio Inpainting," *IEEE Trans. Signal Process.*, vol. 20, no. 3, pp. 922–932, Mar. 2012.

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

[9] B. Defraene, N. Mansour, S. De Hertogh, T. van Waterschoot, M. Diehl, and M. Moonen, "Declipping of Audio Signals Using Perceptual Compressed Sensing," *IEEE Trans. Acoust., Speech, Language Process.*, vol. 21, no. 12, pp. 2627–2637, Dec. 2013.

[10] S. Kitic, L. Jacques, N. Madhu, M. P. Hopwood, A. Spriet, and C. De Vleeschouwer, "Consistent Iterative Hard Thresholding for Signal Declipping," in *38th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Vancouver, Canada, May 26-31 2013, pp. 5939–5943.

[11] A. J. Weinstein and M. B. Wakin, "Recovering a Clipped Signal in Sparseland," *Sampling Theory in Signal and Image Processing*, vol. 12, no. 1, pp. 55–69, Jan. 2013.

[12] M. J. Harvilla and R. M. Stern, "Efficient Audio Declipping using Regularized Least Squares," in *40th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Brisbane, Australia, Apr. 19-24 2015, pp. 221–225.

[13] K. Siedenburg, M. Kowalski, and M. Dörfler, "Audio Declipping with Social Sparsity," in *39th IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Florence, Italy, May 4-9 2014, pp. 1577–1581.

[14] **c**C. Bilen, A. Ozerov, and P. Pérez, "Audio Declipping via Nonnegative Matrix Factorization," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2015, pp. 1–5.

[15] C. Studer, P. Kuppinger, G. Pope, and H. Bolcskei, "Recovery of Sparsely Corrupted Signals," *IEEE Trans. Inf. Theor.*, vol. 58, no. 5, pp. 3115–3130, May 2012.

[16] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-Bit Compressive Sensing via Binary Stable Embeddings of Sparse Vectors," *IEEE Trans. Inf. Theor.*, vol. 59, no. 4, pp. 2082–2102, Apr. 2013.

[17] Y. Plan and R. Vershynin, "Robust 1-bit Compressed Sensing and Sparse Logistic Regression: A Convex Programming Approach," *IEEE Trans. Inf. Theor.*, vol. 59, no. 1, pp. 482–494, Jan. 2013.

[18] A. Zymnis, S. Boyd, and E. Candes, "Compressed Sensing With Quantized Measurements," *IEEE Signal Process. L*, vol. 17, no. 2, pp. 149–152, Feb. 2010.

[19] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The Convex Geometry of Linear Inverse Problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, Dec 2012.

[20] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed Sensing Off the Grid," *IEEE Trans. Inform. Theory*, vol. 59, no. 11, pp. 7465–4790, Nov 2013.

[21] J. Swärd, S. I. Adalbjörnsson, and A. Jakobsson, "Generalized Sparse Covariance-based Estimation," *Elsevier Signal Processing*, 2017, Accepted for publication.

[22] C. Zhou, Z. Zhang, F. Liu, and B. Li, "Gridless compressive sensing method for line spectral estimation from 1-bit measurements," *Digit. Signal Process.*, vol. 60, pp. 152–162, Jan. 2017.

[23] T. L. Jensen and L. Vandenberghe, "Multi-pitch Estimation using Semidefinite Programming," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, New Orleans, LA, USA, March 2017.

[24] P. Stoica, J. Li, and J. Ling, "Missing Data Recovery via a Nonparametric Iterative Adaptive Approach," *IEEE Signal Process. Lett.*, vol. 16, no. 4, pp. 241–244, April 2009.

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

[26] Z. Yang and L. Xie, "On Gridless Sparse Methods for Line Spectral Estimation From Complete and Incomplete Data," *IEEE Trans. Signal Process.*, vol. 63, no. 12, pp. 3139–3153, June 2015.

[27] Y. Chi and Y. Chen, "Compressive Two-Dimensional Harmonic Retrieval via Atomic Norm Minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1030–1042, Feb 2015.

[28] Z. Yang, L. Xie, and P. Stoica, "Vandermonde Decomposition of Multilevel Toeplitz Matrices With Application to Multidimensional Super-Resolution," *IEEE Trans. Inf. Theor.*, vol. 62, no. 6, pp. 3685–3701, June 2016.

[29] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

I

Paper I

Designing Sampling Schemes for Multi-Dimensional Data

Johan Swärd[1], Filip Elvander[1], and Andreas Jakobsson[1]

[1] *Centre for Mathematical Sciences, Lund University, Lund, Sweden*

Abstract

In this work, we propose a method for determining a non-uniform sampling scheme for multi-dimensional signals by solving a convex optimization problem reminiscent of the sensor selection problem. The resulting sampling scheme minimizes the sum of the Cramér-Rao lower bound for the parameters of interest, given a desired number of sampling points. The proposed framework allows for selecting an arbitrary subset of the parameters detailing the model, as well as weighing the importance of the different parameters. Also presented is a scheme for incorporating any imprecise *a priori* knowledge of the locations of the parameters, as well as defining estimation performance bounds for the parameters of interest. Numerical examples illustrate the efficiency of the proposed scheme.

Key words: sampling schemes, convex optimization, CRLB

1 Introduction

Determining how to suitably sample a signal is an important problem in many signal processing applications, such as sensor positioning and selection in network monitoring [1,2], localization and tracking [3], magnetic resonance imaging (MRI) [4], graph signal processing [5, 6], and selecting the temporal sampling [7]. In general, these problems can be viewed as sampling a multi-dimensional field containing partly known signal components. For high-dimensional data, it quickly becomes infeasible to sample the field uniformly, especially, in areas such as nuclear magnetic resonance (NMR) spectroscopy when examining living

cells, which have limited lifetimes. For example, a recent study of 4-D NMR measurements that would have taken about 2.5 years to perform using regular sampling was shown to be possible to construct in merely 90 hours using a non-uniform sampling scheme [8]. This has caused an interest in formulating sampling schemes for NMR signals, allowing for notable improvements [7,9–12].

Among the developed schemes are some exploiting a compressive sensing framework, allowing for an accurate signal reconstruction using fewer samples than the Nyquist-Shannon sampling theorem necessitates for uniformly sampled signals (see, e.g., [11–14]). However, the developed schemes typically do not optimize the sampling scheme with respect to the expected signals, even though these are often fairly well known. In this work, we strive to exploit this knowledge in order to design a sampling scheme that would allow for a optimal estimation accuracy given the assumed prior knowledge.

There are many related problems to the here studied sampling scheme problem. In [15], the problem of how to optimally measure a signal in problems related to propagating wave-fields was studied. More specifically, the authors studied how to best recover the input wave field from noise measurements of the output field given that each measurement is associated with a cost, where the selected cost was set higher for measurement devices with better resolution. The results were presented as trade-off curves between the error of estimation and the total cost budget. In [16], a framework for joint hypothesis testing and estimation using a minimal sampling size was developed. The proposed framework guarantees, under a Bayesian setup, that the overall detection and estimation performance, given the minimization of the samples size, is the best possible. In [17], the optimal placement of phasor measurement units on power grids was studied. Other works have been studying problems related to sampling in random fields [18, 19] and wireless sensor networks [20]. A notable example of the latter category is [20], where the problem of target tracking in wireless sensor networks is studied. The sensors with the most information are found by utilizing a proposed probabilistic sensor management scheme based on the compressed sensing framework. This scheme is decided based on the probability of transmission at each node, found by maximizing the trace of the Fisher information matrix (FIM). Using this approach, sensors with less information can be discarded, implying that fewer sensors need to communicate, thus leading to energy savings.

Lately, for the related problem of optimal sensor placement, there has been several methods proposed in which the combinatorial problem of selecting a sub-

set of sensors is relaxed using convex optimization. In [21], the authors consider the case when signal measurements are linear in the unknown parameters and propose a sensor selection scheme based on solving a convex optimization problem inspired by the determinant criterion (D-optimality) of experimental design [22]. This work was then developed in [2, 23–26], wherein the authors consider non-linear measurement equations, as well as replacing D-optimality with the average variance criterion (A-optimality) as a performance measure. Specifically, as A-optimality can be interpreted as the sum of the diagonal elements of the Cramér-Rao lower bound (CRLB) for the signal parameters, the problem was formulated as to minimize the number of required sensors subject to an upper bound on the resulting diagonal sum of the CRLB. Assuming that the bound is tight, the method thus finds a sparse set of sensors, i.e., activates a few out of a set of candidate sensors, while keeping the variance of the estimated parameters below a fixed level.

In this paper, we expand on this idea, proposing a method for finding a suitable sampling scheme in order to estimate the parameters for signal models where, in general, the signal measurements are non-linear functions of the unknown parameters. By taking the available prior information of the signal into consideration, we propose a sampling scheme that is found by solving a convex optimization problem that guarantees a bound on the worst case CRLB. The sampling pattern is selected via a variable vector, corresponding to the available sample positions, which is penalized using the ℓ_1-norm, resulting in a sampling scheme that is limited in the number of samples. Furthermore, we reformulate the optimization problem into a semidefinite program (SDP) problem that allows for more flexibility and can be used for adding additional constraints on the optimization. In general, when estimating a set of parameters, it might be that the scale of the parameters, as well as the accuracy with which they can be estimated, are significantly different. Also, some of the unknown parameters might be of greater interest than the others; again, using NMR as an example, the signal decay is often of more interest than the signal frequencies, the latter often being relatively well known for a given substance, whereas the former measures the sought interactions. We here propose to use a weighting scheme in order to allow for a relative balancing of the variances of the different parameters, allowing for designing sampling schemes specifically tailored to yield good estimation accuracy for the parameters of interest.

In some applications, one may assume some prior knowledge of the signal of

interest, such as, for example, knowledge of the subspace where the signal parameters are to be found. Again using NMR as an illustrative example, the signals of interest consist of decaying modes, being well modeled as a sum of damped sinusoids. These modes are, as noted, often well known in frequency, at least within some reasonably well defined frequency band, whereas the uncertainty of, and the interest in, the signal decays is often more significant. Typically, the problem of interest is thus to specify the damping parameter as accurately as possible using as few samples as possible. To allow for this case, we herein propose using a gridding of the parameter space in order to guarantee performance within certain bounds, allowing for uncertainty in the parameters.

This paper is organized as follows. In section 2, we introduce the problem statement and derive the proposed optimization problem. In Section 3, we present extensive numerical simulations and results that validates our proposed method. Finally, in Section 4, we conclude upon our work.

2 Problem statement and proposed sampling scheme

Consider a measured signal $y(\boldsymbol{\vartheta}_n)$, defined on a D-dimensional space with N potential D-dimensional sampling points, $\boldsymbol{\vartheta}_n$, $n = 1, 2, \ldots, N$. It is assumed that the probability density function (pdf) of $y(\boldsymbol{\vartheta}_n)$, here denoted with $p\left(y(\boldsymbol{\vartheta}_n); \boldsymbol{\vartheta}\right)$, is parametrized by the parameter vector $\boldsymbol{\vartheta} \in \mathbb{R}^P$ and that two samples $y(\boldsymbol{\vartheta}_n)$ and $y(\boldsymbol{\vartheta}_m)$ are independent if $\boldsymbol{\vartheta}_n \neq \boldsymbol{\vartheta}_m$. FIM for sample $y(\boldsymbol{\vartheta}_n)$ may then be defined as

$$\mathbf{F}(\boldsymbol{\vartheta}_n; \boldsymbol{\vartheta}) = \mathbb{E}\left\{ \nabla_{\boldsymbol{\vartheta}} \log\left(p(y(\boldsymbol{\vartheta}_n); \boldsymbol{\vartheta})\right) \nabla_{\boldsymbol{\vartheta}}^H \log\left(p(y(\boldsymbol{\vartheta}_n); \boldsymbol{\vartheta})\right) \right\} \tag{1}$$

where $\mathbb{E}\left\{\cdot\right\}$, $\nabla_{\boldsymbol{\vartheta}}$, and $(\cdot)^H$ denote the statistical expectation, the gradient with respect to $\boldsymbol{\vartheta}$, and the conjugate transpose, respectively. The here proposed sampling scheme is designed such that it is optimal in the sense of either minimizing the CRLB of the parameters of interest, given that M of the N potential uniform samples are used, or conversely, to minimize the number of samples used given a desired upper bound on the CRLB of the parameters. It is worth noting that as the potential signal samples are assumed to be independent, for any set of samples indices $\boldsymbol{\Omega}$, it holds that

$$\sum_{n \in \boldsymbol{\Omega}} \mathbf{F}(\boldsymbol{\vartheta}_n; \boldsymbol{\vartheta}) \tag{2}$$

is the corresponding FIM using this sample scheme. Let the N-dimensional vector $\mathbf{w}$ denote the possible sampling points in the D-dimensional sampling space, such that if the n:th index, w_n, is set to one, this sampling point is used, whereas if it is set to zero, it is not. Reminiscent of the case of optimal sensor selection, the resulting sampling design problem may then be formulated as (see also [23])

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}} \quad & \text{tr}\left(\sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta})\right)^{-1} \\ \text{subject to} \quad & \|\mathbf{w}\|_1 \leq \lambda \\ & w_n \in \{0,1\}\ ,\ n = 1,2,\ldots,N \end{aligned} \tag{3}$$

where $\lambda > 0$ and $\text{tr}(\cdot)$ denotes the trace operator. The choice of objective function is related to the so-called A-optimality criterion from design of experiments [22] as the trace of the inverse FIM corresponds to the sum of the CRLBs of the signal parameters in $\boldsymbol{\vartheta}$. Here, the parameter λ constitutes an upper bound on the ℓ_1-norm of the sample selection vector. The sampling design scheme (3) is not convex due to the restriction that w_n, for $n = 1,\ldots,N$, is defined over a non-convex set. A convex approximation to this problem may be found by relaxing the binary constraint and instead allowing w_n to take any value in the range $[0,1]$ (see, e.g., [24]), resulting in

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}} \quad & \text{tr}\left(\sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta})\right)^{-1} \\ \text{subject to} \quad & \mathbf{1}^T\mathbf{w} \leq \lambda \\ & w_n \in [0,1]\ ,\ n = 1,2,\ldots,N \end{aligned} \tag{4}$$

where $\mathbf{1}$ is a vectors of ones with appropriate dimension. It should be noted that we can here replace $||\mathbf{w}||_1$ with simply $\mathbf{1}^T\mathbf{w}$, since each element in $\mathbf{w}$ is equal to or greater than zero. Given a solution $\hat{\mathbf{w}}$ to (4), we define the FIM for the corresponding sampling pattern as

$$\mathcal{I}(\hat{\mathbf{w}};\boldsymbol{\vartheta}) = \sum_{\ell\in\boldsymbol{\Omega}} \mathbf{F}(\boldsymbol{\vartheta}_\ell;\boldsymbol{\vartheta}), \quad \boldsymbol{\Omega} = \{\ell \mid \hat{w}_\ell > \xi\} \tag{5}$$

where $\xi \geq 0$ is a threshold determining whether a sample weight $\hat{w}_\ell$ should be rounded toward one or zero, i.e., whether the sampling point should be included or not. This formulation allows for the minimization of the sum of the CRLBs

given an upper bound on the number of samples used. Note that the problem could alternatively be formulated as minimizing the number of sampling points given an upper bound on the sum of the CRLBs.

However, the sampling design in (4) does not allow for the case when one is primarily interested in a subset of the available parameters. Neither does the formulation take into account that the different parameters might have significantly different variances. For example, for a sum of damped sinusoids, the trace constraint in (4) will clearly be dominated by the CRLB for the amplitudes, as these are orders of magnitude larger than those of the frequencies, and the optimization will therefore put an emphasis on minimizing the CRLB of the amplitude parameter. In order to allow for sampling schemes that put an emphasis on a selection of the parameters of interest, we recently proposed to introduce a weighting matrix, $\mathbf{A}(\boldsymbol{\vartheta})$, acting upon the FIM in [27]. Specifically, instead of minimizing the cost function using the FIM, we proposed to perform the minimization using weighted FIMs

$$\tilde{\mathbf{F}}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) = \mathbf{A}(\boldsymbol{\vartheta})\mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta})\mathbf{A}^T(\boldsymbol{\vartheta}) \ , \tag{6}$$

i.e., performing a linear transformation of the variables and minimizing the sum of the CRLBs corresponding to the transformed parameters $\tilde{\boldsymbol{\vartheta}} = \mathbf{A}(\boldsymbol{\vartheta})\boldsymbol{\vartheta}$. However, although this formulation allows for shifting emphasis to the parameters of interest, it does not allow for complete disregard of nuisance parameters as $\mathbf{A}(\boldsymbol{\vartheta})$ has to be definite in order for the matrix inverse to be defined. In order to allow for an arbitrary weighting, we note the following useful identity holds for an invertible matrix $\mathbf{B}$,

$$\operatorname{tr}\mathbf{B}^{-1} = \sum_{p=1}^{P} \mathbf{e}_p^T\mathbf{B}^{-1}\mathbf{e}_p \tag{7}$$

where $\mathbf{e}_p$ denotes the pth canonical basis vector, i.e., a vector with all its elements equal to zero except the pth being equal to one. Furthermore, it is noted that for a positive definite matrix $\mathbf{B}$, a scalar μ, and an arbitrary vector $\mathbf{a}$, it follows from the Schur complement (see, e.g., [28]) that

$$\mu - \mathbf{a}^T\mathbf{B}^{-1}\mathbf{a} \geq 0 \iff \begin{bmatrix} \mathbf{B} & \mathbf{a} \\ \mathbf{a}^T & \mu \end{bmatrix} \succeq \mathbf{0} \tag{8}$$

where $\mathbf{X} \succeq \mathbf{0}$ indicates that the matrix $\mathbf{X}$ is positive semi-definite. Thus, it follows that

$$\underset{\mathbf{B}\succ 0}{\text{minimize}}\ \mathbf{a}^T\mathbf{B}^{-1}\mathbf{a} \tag{9}$$

and

$$\begin{aligned} &\underset{\mu,\mathbf{B}\succ 0}{\text{minimize}} && \mu \\ &\text{subject to} && \begin{bmatrix} \mathbf{B} & \mathbf{a} \\ \mathbf{a}^T & \mu \end{bmatrix} \succeq \mathbf{0} \end{aligned} \tag{10}$$

are minimized by the same matrix $\mathbf{B}$. Here, $\mathbf{B} \succ \mathbf{0}$ indicates that the matrix $\mathbf{B}$ is positive definite. This observation allows us to reformulate (4) as the semidefinite program (SDP) (cf. [2, 17])

$$\begin{aligned} &\underset{\boldsymbol{\mu},\mathbf{w}}{\text{minimize}} && \sum_{p=1}^{P} \psi_p \mu_p \\ &\text{subject to} && \begin{bmatrix} \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) & \mathbf{e}_p \\ \mathbf{e}_p^T & \mu_p \end{bmatrix} \succeq \mathbf{0},\ \forall p \\ & && \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) \succ \mathbf{0} \\ & && \mathbf{1}^T\mathbf{w} \leq \gamma \quad , \quad w_n \in [0,1],\ \forall n \end{aligned} \tag{11}$$

where ψ_p are weight parameters allowing for putting emphasis on different components of the vector $\boldsymbol{\vartheta}$. For example, if $\psi_q = 1$ and $\psi_p = 0$, $\forall p \neq q$, then the CRLB for the parameter ϑ_q will be the only one minimized, as μ_q precisely corresponds to this lower bound, whereas the CRLBs for the other parameters ϑ_p, $p \neq q$ will be disregarded. Similarly, for $\psi_p = 1$, $\forall p$, the problems (4) and (11) are equivalent. Another benefit of this formulation is that it allows for a straightforward way of incorporating performance constraints in the minimization problem, such as if, for instance, there is some upper tolerance bound λ_p for the CRLB of parameter ϑ_p. This kind of performance specifications can then be incorporated

in the minimization problem via linear inequality constraints according to

$$\begin{aligned}
\underset{\boldsymbol{\mu},\mathbf{w}}{\text{minimize}} \quad & \sum_{p=1}^{P} \psi_p \mu_p \\
\text{subject to} \quad & \begin{bmatrix} \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) & \mathbf{e}_p \\ \mathbf{e}_p^T & \mu_p \end{bmatrix} \succeq \mathbf{0}, \ \forall p \\
& \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) \succ \mathbf{0} \\
& \mathbf{1}^T \mathbf{w} \leq \gamma \quad , \quad w_n \in [0,1], \ \forall n \\
& \mu_p \leq \lambda_p \ , \ \forall p
\end{aligned} \tag{12}$$

Furthermore, one may not only be interested in designing a sampling scheme for a single parameter vector $\boldsymbol{\vartheta}$, but rather for a set of parameter vectors. For example, consider the case when the parameters in $\boldsymbol{\vartheta}$ are only partly known, such that one may assume that $\boldsymbol{\vartheta}$ instead lies in a set of possible parameters, Θ. In such cases, it may be desired to treat some of the parameters as known, whereas others are only partly known, within some set of uncertainty. To allow for this, as well as taking the weighting into account, we further generalize (12) such that the sampling scheme is designed as

$$\begin{aligned}
\underset{\boldsymbol{\mu},\mathbf{w}}{\text{minimize}} \quad & \sum_{p=1}^{P} \psi_p \mu_p \\
\text{subject to} \quad & \begin{bmatrix} \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) & \mathbf{e}_p \\ \mathbf{e}_p^T & \mu_p \end{bmatrix} \succeq \mathbf{0}, \ \forall p, \forall \boldsymbol{\vartheta} \in \Theta \\
& \sum_{n=1}^{N} w_n \mathbf{F}(\boldsymbol{\vartheta}_n;\boldsymbol{\vartheta}) \succ \mathbf{0} \\
& \mathbf{1}^T \mathbf{w} \leq \gamma \quad , \quad w_n \in [0,1], \ \forall n \\
& \mu_p \leq \lambda_p \ , \ \forall p
\end{aligned} \tag{13}$$

Using this formulation, the optimal μ_p will, assuming that $\psi_p > 0$, now correspond to a worst case CRLB for the pth component of $\boldsymbol{\vartheta}$, when $\boldsymbol{\vartheta} \in \Theta$, i.e., for the obtained sampling sampling scheme

$$\mu_p = \max_{\boldsymbol{\vartheta} \in \Theta} \mathbf{e}_p^T \mathcal{I}(\hat{\mathbf{w}};\boldsymbol{\vartheta})^{-1} \mathbf{e}_p \tag{14}$$

Thus, the solution to (13) is a sampling scheme minimizing the worst case CRLB for the parameters of interest if the parameter vector $\boldsymbol{\vartheta}$ is known to be in the set Θ.

Further, one could also consider the case where there is some cost associated with changing sampling points in one of the dimensions. For instance, if one of the sampling dimensions corresponds to a certain setting of a machine, e.g., time delay or magnetic flow, it could be more costly to acquire many different sample points in this dimension. Illustrating this in the 2-D case, one could include such a cost in the optimization by forming the $N_1 \times N_2$ matrix $\mathbf{W}$ by reshaping the vector $\mathbf{w}$, and adding the constraints

$$\left|\left|\mathbf{W}^T\right|\right|_{2,1} = \sum_{n=1}^{N_1} \left|\left|\mathbf{W}_{(:,n)}\right|\right|_2 \leq \gamma_1 \tag{15}$$

$$||\mathbf{W}||_{2,1} = \sum_{n=1}^{N_2} \left|\left|\mathbf{W}_{(n,:)}\right|\right|_2 \leq \gamma_2 \tag{16}$$

to (13). Here, γ_1 and γ_2 are tuning parameters that may be set according to the associated cost. This constraint can easily be omitted simply by setting $\gamma_1 = \gamma_2 = \infty$.

It is also worth noting that when relaxing (3) in favor for (4), we can no longer guarantee that the weights are exactly 0 or 1. In this case, as is noted in (5), we simple choose an appropriate threshold such that values above the threshold are deemed as ones, and the values below are deemed as zeros. However, a better approximation of (3) is found by using re-weighting. This may be done by first solving (13), yielding the estimated $\mathbf{w}^{(1)}$, where the superscript $(\cdot)^{(j)}$ denotes jth iteration. Then, (13) is solved again, but this time with

$$\frac{1}{w_n^{(1)} + \varepsilon} \tag{17}$$

as a scaling factor for each w_n, where ε is a small number added to the denominator to avoid numerical problems. This procedure can then repeated until convergence. The re-weighting is a better approximation of the ℓ_0-norm, and thus is more likely to produce weights with values close to zero or one. As we have empirically found that using re-weighting for the here studied examples offers only a marginal improvement, while significantly increasing the computational cost due to the iterative procedure, we have in our examples chosen to use the simpler thresholding approach.

3 Numerical results

3.1 Illustration in 1-D

To illustrate the proposed sampling scheme, we consider the NMR signal model, as noted being formed as a sum of damped sinusoids (for ease of notation, we initially focus on the 1-D case), such that

$$y(t_n) = \sum_{k=1}^{K} \alpha_k \exp\{2i\pi f_k t_n - \beta_k t_n + i\varphi_k\} + \varepsilon(t_n) \tag{18}$$

for $n = 1, \ldots, N$, where α_k, f_k, β_k, and φ_k denote the magnitude, frequency, damping, and phase of the k:th component, respectively, and where ε is an additive noise term, here assumed to be well modeled as a white, circularly symmetric Gaussian noise with variance σ^2, with t_n being the time at sample n. For simplicity, we consider uniformly sampled candidate sampling times, t_n. As an illustration, Figure 1 shows an example of sampling schemes found by solving (13) for two different levels of decay for a single damped sinusoid such that $\beta = 1/10$ for the top figure, and $\beta = 1/20$ for the bottom figure, but otherwise identical signal parameters. In both cases, $\gamma = 13$ so that $M = 13$ sample points, out of $N = 50$ possible candidates, are selected. Also, $\psi_p = 1$, $p = 1, \ldots, 4$, i.e., all signal parameters are considered in the minimization. As can be seen, the placing of the samples are determined by the damping parameter. As may be expected, for both values of β, some samples are placed in the beginning of the signal, where the signal to noise ratio (SNR) is at its maximum. To allow for an accurate estimation of the damping constant, one can also note that a further set of samples are selected later in the signal, with the more strongly decaying signal selecting them earlier than the less damped version, agreeing with the intuition that the more rapidly decaying signal contains less information at later sampling times.

As a further example, we next consider an example showing the resulting sample scheme for a signal containing two linear chirp components on the form

$$y(t_n) = \sum_{k=1}^{2} \alpha_k \exp\left\{2i\pi\left(f_k^0 + f_k^1 t_n\right) t_n + i\varphi_k\right\} + \varepsilon(t_n) \tag{19}$$

where f_k^0 and f_k^1 denote the frequency starting point and the slope of the chirp component k, respectively. Figure 2 shows the three sampling schemes yielded by the proposed method for three different setting on γ, namely $\gamma = 15$, $\gamma = 20$,

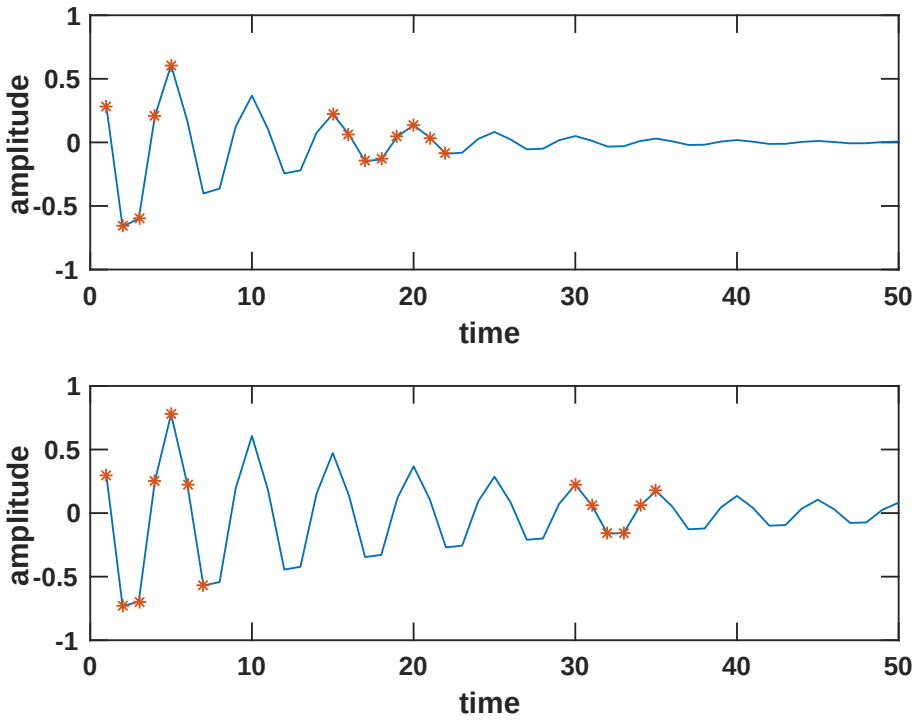

Figure 1: The resulting sample scheme for two different values of β plotted against the real part of the signal. The upper most figure details the sampling scheme for $\beta = \frac{1}{10}$ and the bottom figure the sampling scheme for $\beta = \frac{1}{20}$.

and $\gamma = 25$. The here used parameters had the values $\alpha_1 = \alpha_2 = 5, f_1^0 = 0.1$, $f_2^0 = 0.5, f_1^1 = 0.01, f_2^1 = -0.003$, and the phases were set to $\varphi_1 = \pi/2$, and $\varphi_2 = \pi/3$. Due to the linear drift in frequency, it is reasonable to assume that the resulting sample scheme should have at least two clusters; one in the beginning of the signal, and one at the end of the signal. Looking at the sampling schemes in Figure 2 supports this intuition; three clusters are present for all three settings of γ. When γ increases the two first clusters gets bigger, whereas the last cluster remains more or less unchanged.

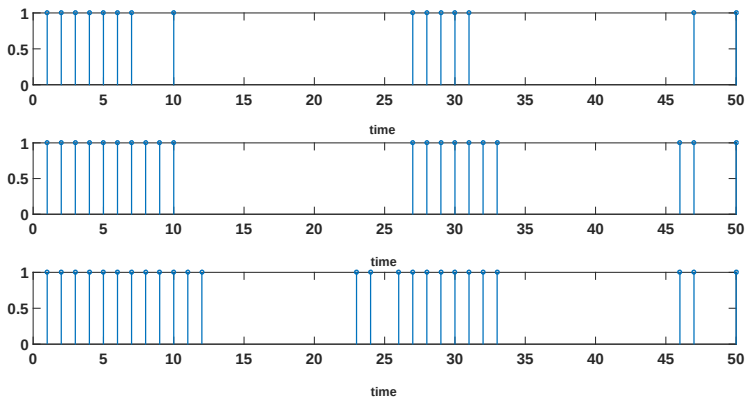

Figure 2: The resulting sample scheme for three different settings of γ, namely $\gamma = 15$, $\gamma = 20$, and $\gamma = 25$, where the signal contains two linear chirps.

3.2 Illustration in 2-D

As further illustration of the impact of the choice of weight parameters ψ_p, consider the 2-D case with one damped sinusoid, i.e.,

$$y(t_1, t_2) = \alpha e^{2i\pi(f_1 t_1 + f_2 t_2) - (\beta_1 t_1 + \beta_2 t_2) + i\varphi} + \varepsilon(t_1, t_2) \tag{20}$$

with $\alpha = 1$, $f_1 = 0.2$, $f_2 = 0.5$, $\beta_1 = 1/20$, $\beta_2 = 1/10$, $\varphi = 1/2$, and noise variance $\sigma^2 = 0.1$. Figure 3 presents the sampling scheme found by solving (11) with $\gamma = 50$, i.e., 50 sampling points are chosen, for the case when $\psi_p = 1$ for all parameters. As can be seen, the optimal sampling pattern here consists of three clusters of selected sampling points; one close to the origin and two close to the two time axes. Note that this is analogous to the 1-D case as the sampling cluster close to the first time axis is located further from the origin due to the decay in the first dimension being slower.

In contrast, Figure 4 displays the corresponding scheme found when solving (11), again with $\gamma = 50$, but only giving weight to the frequency and damping parameters, i.e., the ψ_p corresponding to the amplitude and phase parameters are set to zero. As can be seen, assigning the amplitude and phase parameters zero weight has the effect of shifting sampling points away from the origin to the clusters close to the t_1 and t_2 axes, in order to put more emphasis on the frequency

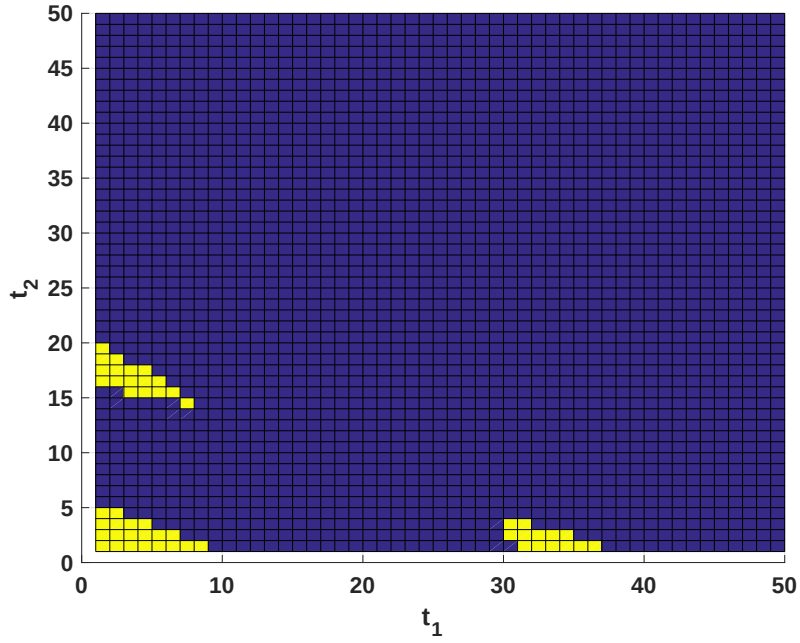

Figure 3: The resulting sampling scheme consisting of 50 selected samples for a signal consisting of a 2-D damped sinusoid as found when solving (11) with all $\psi_p = 1$.

and damping parameters. Indeed, the sum of the CRLBs for the parameters, as given by the sampling scheme in Figure 3, is $2.31 \cdot 10^{-2}$, whereas it is $3.61 \cdot 10^{-2}$ for the sampling scheme in Figure 4. However, if one considers the sum of the CRLBs for the frequency and damping parameters, these are $6.53 \cdot 10^{-4}$ and $4.42 \cdot 10^{-4}$ for Figures 3 and 4, respectively.

3.3 Simulations in 1-D

3.3.1 Optimization vs simulation

In Figure 5, we motivate that solving (13) is indeed a reasonable approach to determine optimal sampling patterns. The figure shows the obtained sum of the CRLBs for the parameters, i.e., $\mathrm{tr}\left(\mathcal{I}(\hat{\mathbf{w}}; \boldsymbol{\vartheta})^{-1}\right)$, where the sampling pattern is

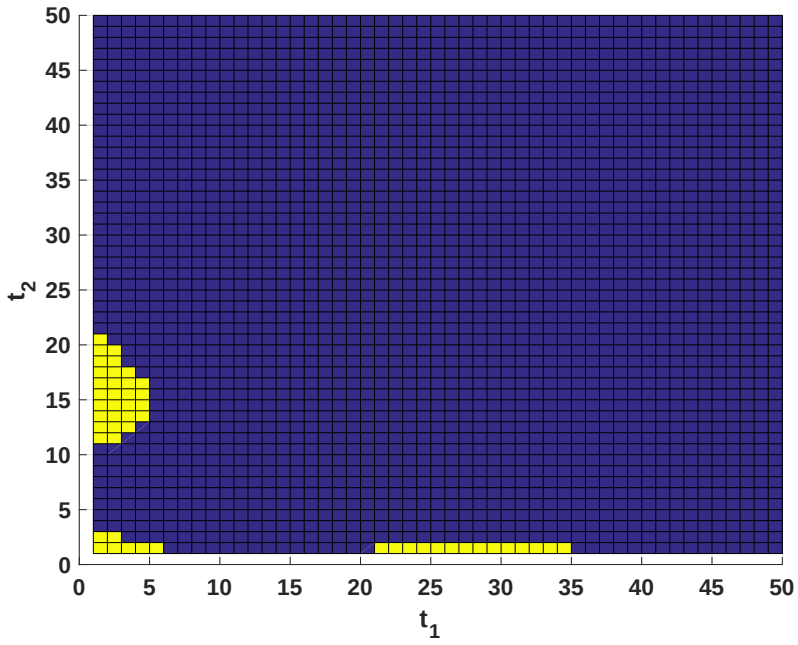

Figure 4: The resulting sampling scheme consisting of 50 selected samples for a signal consisting of a 2-D damped sinusoid as found when solving (11) with all $\psi_p = 1$ except for the amplitude and phase parameters, for which $\psi_p = 0$.

obtained by solving (13) for the case of $K = 1$ using the model (18), for a singleton set Θ. This is done for varying values of γ such that the number of samples used vary between $M = 5$ and $M = 25$. As a comparison, for each sample size M, we carry out 10^6 Monte Carlo simulations, in which we randomly decide on which M sampling points to use. We then compute which of these 10^6 sampling patterns that results in the lowest sum of CRLBs. As can be seen from the figure, the randomized approach achieves better results for small sample sizes, this as the simulations then become an exhaustive search, i.e., the simulations will with high likelihood find the exact solution to (3). However, as the sample size increases, so does the number of possible sampling patterns, which is $\binom{N}{M}$. As can be seen from the figure, the sampling scheme determined by (13) is then able to achieve an optimal performance as the sample size increases.

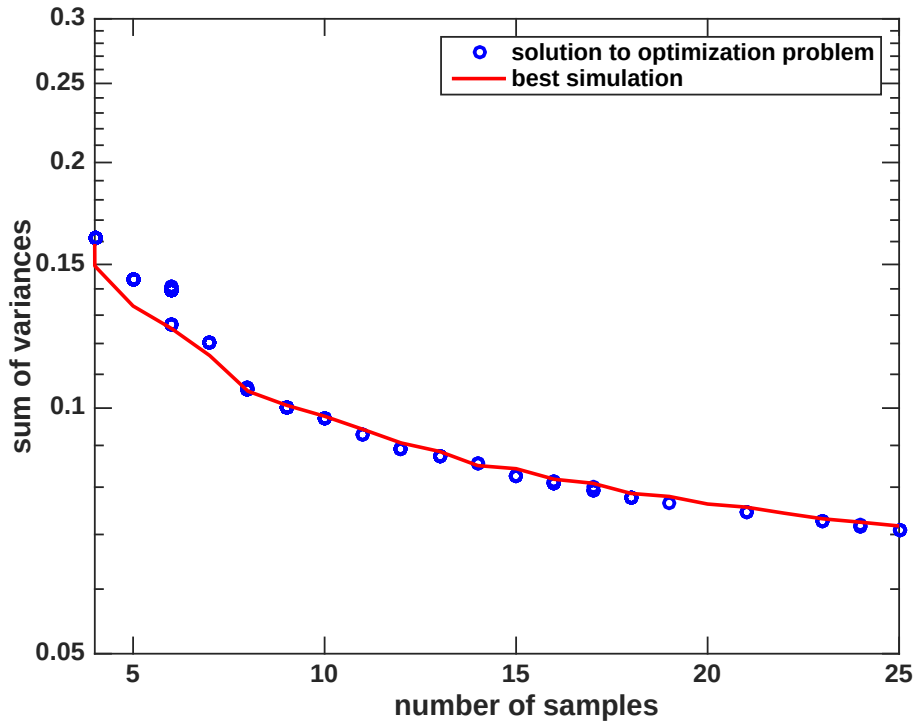

Figure 5: Sum of CRLBs for the parameters, i.e., tr $\left(\mathcal{I}(\hat{\mathbf{w}};\vartheta)^{-1}\right)$, for the sampling patterns given by the optimization problem and the best simulation, respectively, for different number of sampling points.

3.3.2 Weighting

In Figures 6 and 7, we proceed to examine the effect of using the weighted FIM in (13). This is done for a signal consisting of two damped sinusoids with parameters $(\alpha_1, f_1, \beta_1, \varphi_1) = (1, 0.2, 1/12, 0.5)$ and $(\alpha_2, f_2, \beta_2, \varphi_2) = (1, 0.65, 1/20, \pi/5)$. The noise variance was $\sigma^2 = 0.01$ and $N = 50$. Assuming that we are interested only in the frequencies f_1, f_2, and the damping factors β_1, β_2, but not in the amplitudes or the phases, the weight parameters ψ_p are set to one for the frequency and damping parameters, whereas they are set to zero for the amplitudes and phases. Thus, the sought sampling pattern will be designed to increase the accuracy for the frequency and damping parameters at the expense of the amplitude and phase

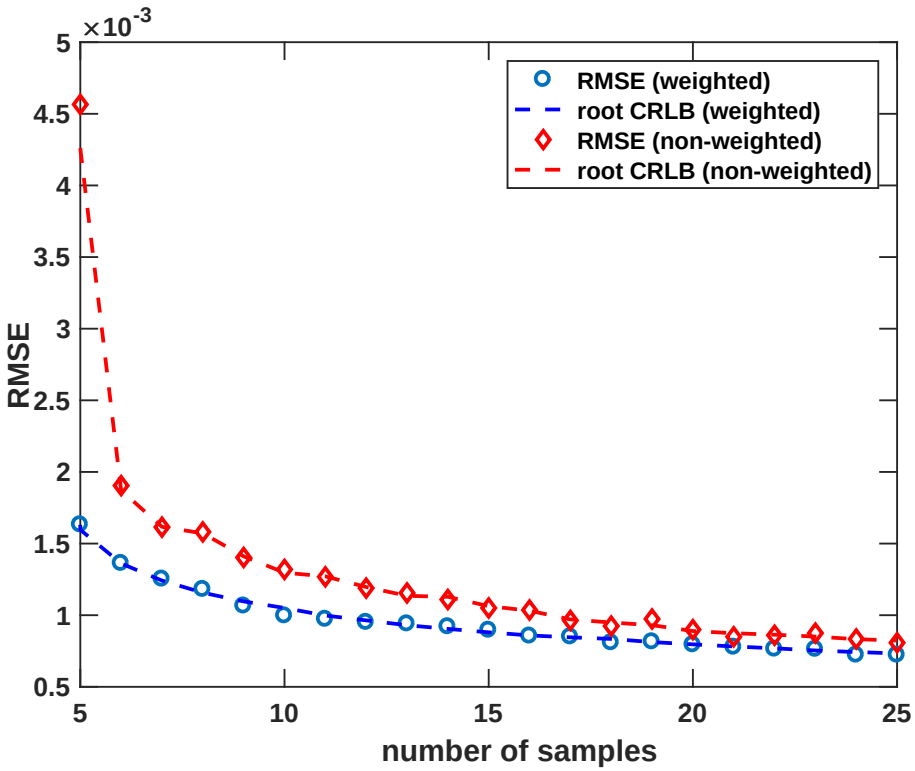

Figure 6: Obtained RMSE for the frequencies, when using the sampling patterns for the weighted and non-weighted cases, respectively.

parameters.

The resulting root CRLB, as a function of the number of samples used, for the frequencies f_1 and f_2 and the dampings β_1 and β_2 are shown in Figures 6 and 7, respectively. The root CRLB for the frequencies f_1 and f_2 is here defined as the root of the sum the individual CRLBs, and correspondingly for the dampings, β_1 and β_2. For comparison, the figures also present the root CRLBs corresponding to the optimal sampling patterns obtained for the case when no weighting is applied, i.e., $\psi_p = 1, \forall p$. As can be seen, the weighting scheme results in sampling patterns that decreases the CRLB for the parameters of interest, in this case the frequencies and dampings. Also plotted is the obtained root mean squared error (RMSE) for the frequency and damping parameters, respectively, obtained when estimating

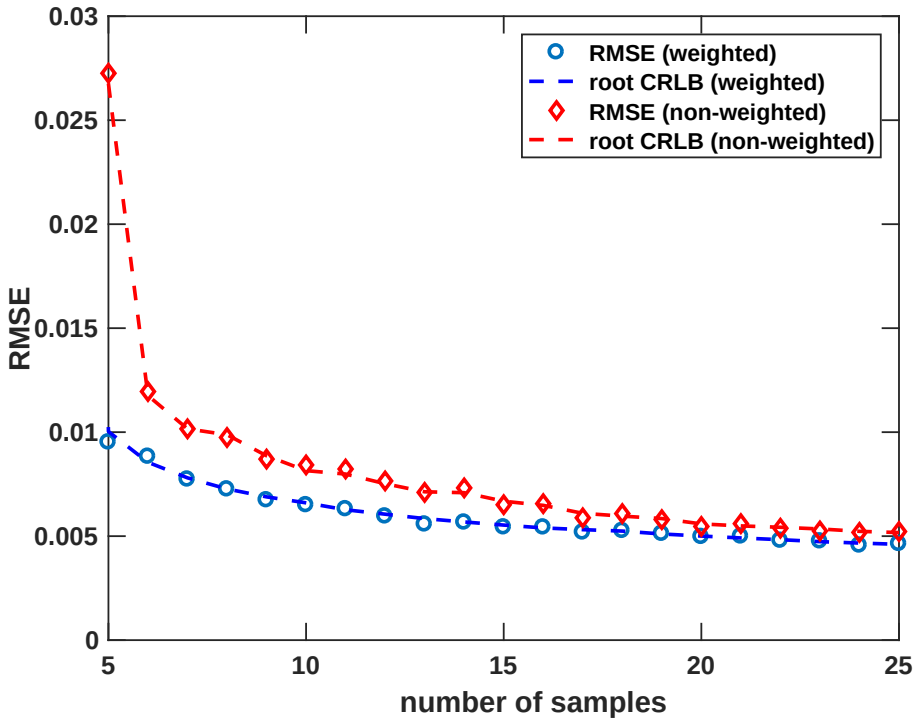

Figure 7: Obtained RMSE for the damping, when using the sampling patterns for the weighted and non-weighted cases, respectively.

these parameters using non-linear least squares (NLS) applied to simulated signals. The NLS estimate is found by solving

$$\hat{\boldsymbol{\vartheta}} = \underset{\boldsymbol{\vartheta}}{\operatorname{argmin}} \quad \frac{1}{2}||\mathbf{y} - g(\boldsymbol{\vartheta})||_2^2 \tag{21}$$

where $\mathbf{y}$ is the data and $g(\vartheta)$ is the (non-linear) data model with parameter ϑ. In this paper, a minimum of (21) is found by evaluating the cost function over a grid of parameter values ϑ. The ϑ that achieves the lowest value of (21) then becomes the resulting estimates. The RMSE is here defined as the root of the sum of the individual MSEs for the frequencies and dampings, respectively. As can be seen, the RMSE coincides with the root CRLB, implying that the bound is tight.

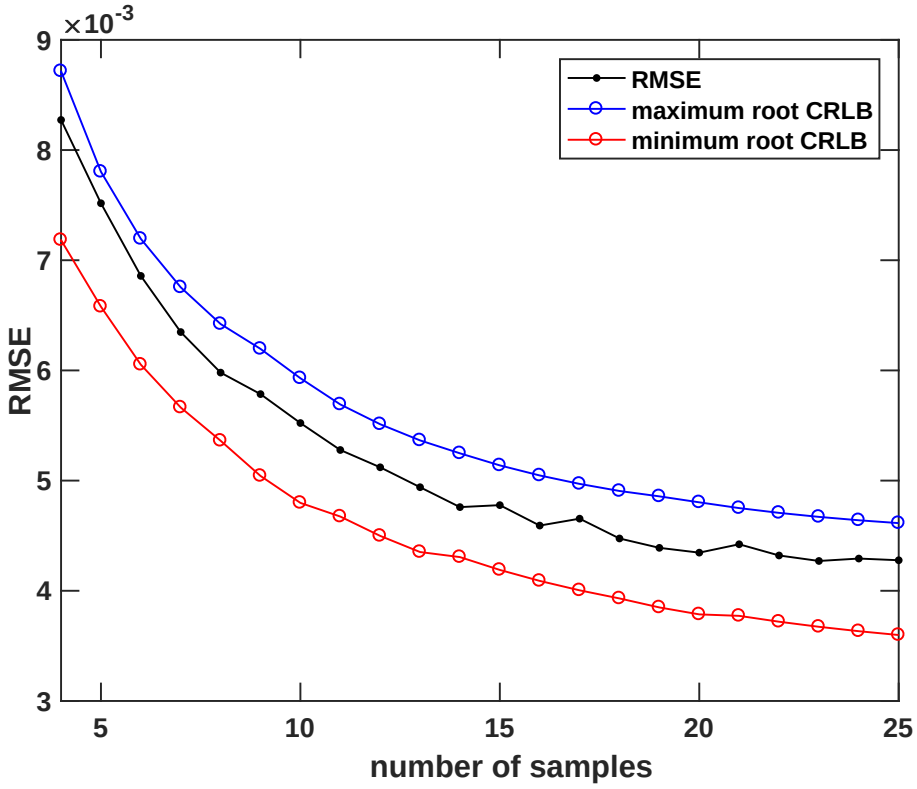

Figure 8: Obtained RMSE for the frequency f, when estimating ϑ for the sampling pattern obtained for a grid of damping parameters β.

3.3.3 Gridding

Figures 8 and 9 show the effect of finding an optimal sampling pattern for a set of parameters $\vartheta \in \Theta$ when solving (13). The results are obtained for a single decaying sinusoid. Here, we let $\Theta = \{\vartheta_\ell\}_{\ell=1}^{L}$ express uncertainty in only the damping parameter β by fixing α, f, and φ and letting Θ be a gridding over the damping parameter β, such that the parameter vectors constituting Θ are $\vartheta_\ell = (\alpha, f, \beta_\ell, \varphi)^T$, where

$$\beta_\ell = \beta_{\text{lower}} + \frac{\ell - 1}{L}\Delta_\beta \tag{22}$$

with Δ_β denoting the grid spacing, in effect letting β reside in the uncertainty

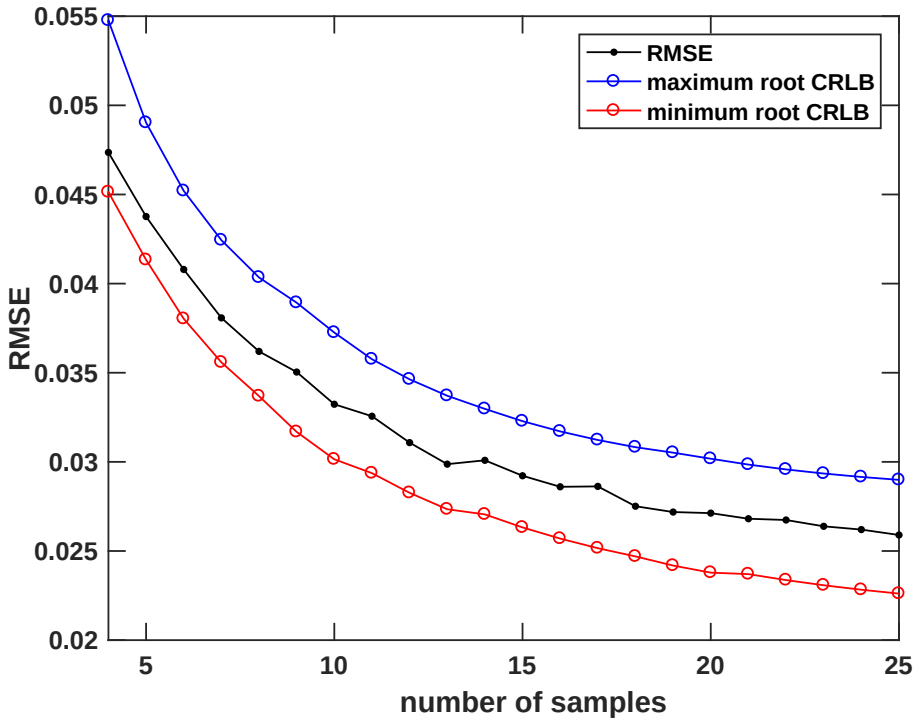

Figure 9: Obtained RMSE for the damping β, when estimating $\boldsymbol{\vartheta}$ for the sampling pattern obtained for a grid of damping parameters β.

interval

$$\mathcal{J}_\beta = \left[\beta_{\text{lower}}, \beta_{\text{lower}} + \frac{L-1}{L}\Delta_\beta\right] \tag{23}$$

The parameters used are $\alpha = 1$, $\varphi = 0.5$, $\sigma^2 = 0.1$, $\beta_{\text{lower}} = 0.1$, $\Delta_\beta = 0.022$, and $L = 10$. Using this, we solve (13) to get optimal sampling patterns as the number of samples grows. To evaluate the performance of the obtained sampling schemes, we then randomly sample the parameter vectors $\boldsymbol{\vartheta}$ where β is sampled uniformly on $\mathcal{J}_\beta$, i.e., on the interval covered by the grid Θ, but not on the grid points β_ℓ, $\ell = 0, 1, \ldots, L-1$. We then estimate $\boldsymbol{\vartheta}$ using NLS and compute the RMSE for the parameters $\boldsymbol{\vartheta}$. The figures show the obtained MSE using 5000 Monte Carlo simulations for the frequency f and the damping β, respectively.

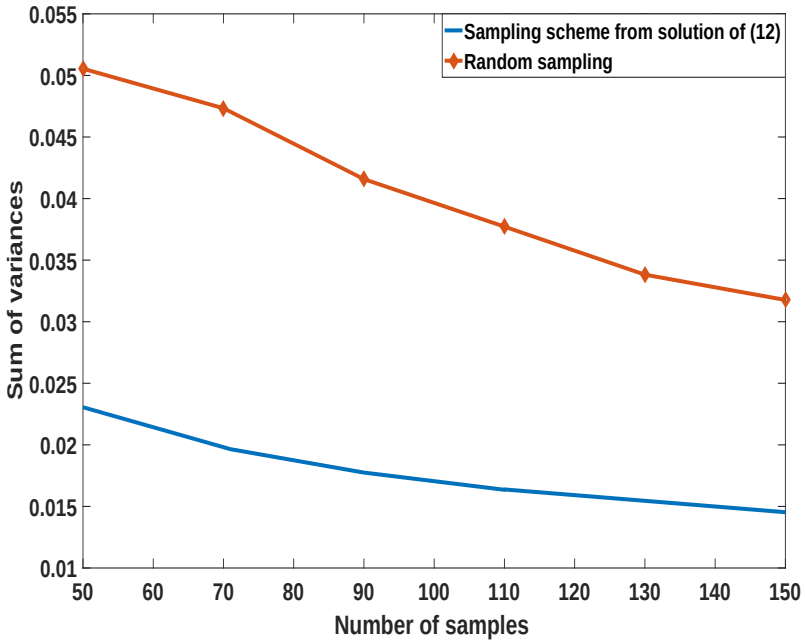

Figure 10: The sum of variances of the parameters of interest as a function of the number of selected samples.

Also presented are the best and worst case root CRLBs found on the grid Θ for each parameter. The obtained RMSE lies between the lowest and highest on-grid root CRLB for both parameters and for all considered sample sizes, suggesting that (13) indeed yields sampling schemes with a guaranteed worst case performance, as well as a lower limit on the possible RMSE.

3.4 Simulations in 2-D

3.4.1 Optimization vs simulation

As was seen in the 1-D setting, the optimization scheme was able to outperform the method of randomly selecting sampling points and then choosing the scheme minimizing the sum of the CRLB. In 2-D, this becomes even more apparent as the number of potential sampling points increase rapidly with increasing dimension.

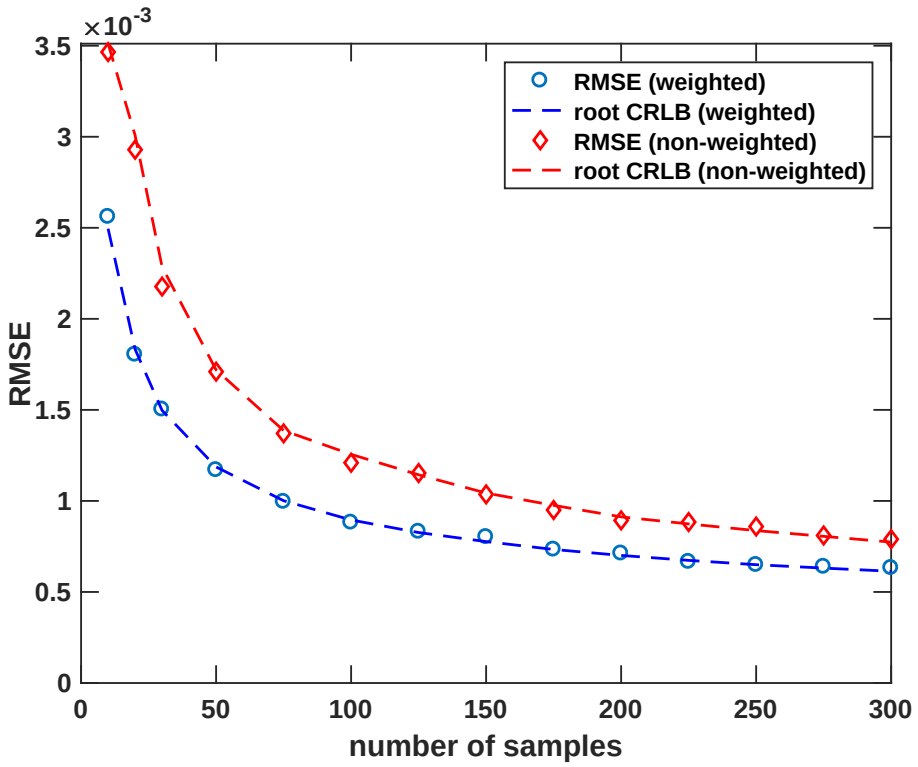

Figure 11: Obtained RMSE for the frequencies in the first dimension, when using the sampling patterns for the weighted and non-weighted cases, respectively.

An illustration of this is shown in Figure 10, showing the sum of the CRLBs obtained when solving for varying number of desired sampling points. The signal considered is the 2-D damped sinusoid in (20) with parameters $\alpha = 1$, $f_1 = 0.2$, $f_2 = 0.5$, $\beta_1 = 1/20$, $\beta_2 = 1/10$, $\varphi = 1/2$, and $\sigma^2 = 0.1$. We here let $\psi_p = 1$, $\forall p$, and consider a sampling space of 50×50 potential sampling times. Also presented is the sum of the CRLBs for the best (defined as the one with smallest sum of CRLBs) among 10^7 sampling scheme obtained by randomly choosing sampling points. As can be seen from the figure, the proposed method outperforms the random sampling for all numbers of selected samples. It is worth noting that the computational time to evaluate the 10^7 sampling schemes was three times longer than solving the proposed problem using a off-the-shelf convex

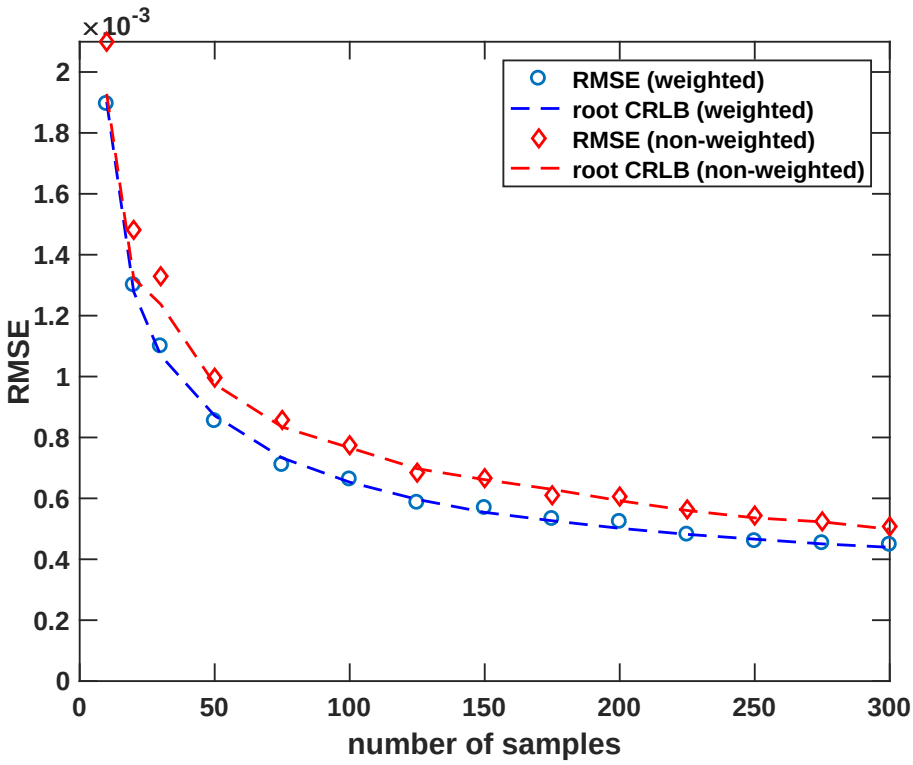

Figure 12: Obtained RMSE for the frequencies in the second dimension, when using the sampling patterns for the weighted and non-weighted cases, respectively.

solver [29].

3.4.2 Weighting

We here consider the case of a signal consisting of two 2-D damped sinusoid, i.e.,

$$y(t_1,t_2)=\sum_{k=1}^{K}\alpha_k e^{i\varphi_k}\Pi_{d=1}^{2}e^{2i\pi f_{k,d}t_d-\beta_{k,d}t_d}+\varepsilon(t_1,t_2) \tag{24}$$

for $K=2$. Let the parameters be $(f_{1,1},f_{2,1})=(0.1,0.2)$ and $(\beta_{1,1},\beta_{2,1})=(0.1,0.1)$ for the first dimension, $(f_{1,1},f_{2,1})=(0.1,0.2)$ and $(\beta_{1,1},\beta_{2,1})=(0.1,0.1)$ for the second dimension, and let $\alpha_1=1$, $\alpha_2=1.3$, $\varphi_1=\frac{\pi}{3}$, $\varphi_2=\frac{\pi}{3}$, and

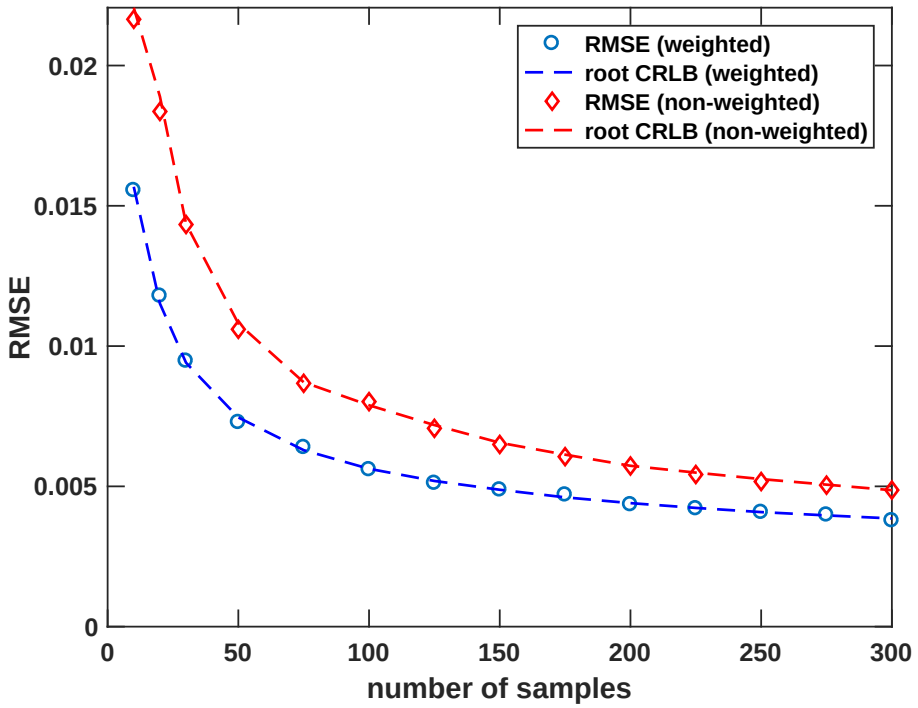

Figure 13: Obtained RMSE for the dampings in the first dimension, when using the sampling patterns for the weighted and non-weighted cases, respectively.

$\sigma^2 = 0.01$. We then determine optimal sampling schemes by solving (11) for varying number of sampling points. This is done for both the equally weighted case, i.e., with $\psi_p = 1$ for all p, as well as for the case when only the frequency and damping parameters are given weight, i.e., with $\psi_p = 0$ for the amplitude and phase parameters. The results are shown in Figures 11-14. In Figure 11, the root of the sum of the CRLBs for the frequencies in the first dimension, i.e., $f_{1,1}$ and $f_{2,1}$, are shown. Similarly, Figure 12 corresponds to the frequencies in the second dimension, while Figures 13 and 14 corresponds to the damping parameters in the first and second dimension, respectively. Also presented is the corresponding RMSE obtained when estimating the parameters using NLS. As can be seen, the obtained RMSEs coincides with the CRLBs for both the weighted and non-weighted case, implying that the bound is tight. Note also that the schemes

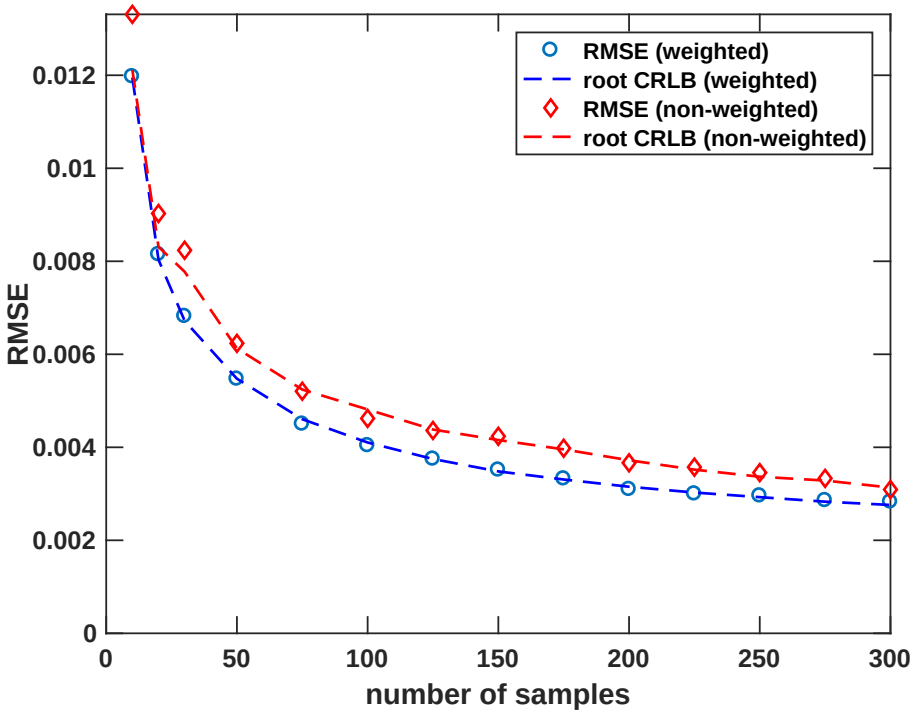

Figure 14: Obtained RMSE for the dampings in the second dimension, when using the sampling patterns for the weighted and non-weighted cases, respectively.

corresponding to assigning no weight to the amplitude and phase parameters all result in a lower sum of CRLB for the frequency and damping parameters than the non-weighted schemes. This comes at the price of a larger sum of CRLB for the amplitudes α_1 and α_2, which is illustrated in Figure 15. As can be seen in the figure, the non-weighted sampling scheme here leads to more accurate estimates of the amplitudes.

4 Conclusion

In this work, we have proposed a convex optimization problem for finding suitable sampling schemes for multidimensional data models. The optimization problem is formed such that the number of used samples, chosen from a col-

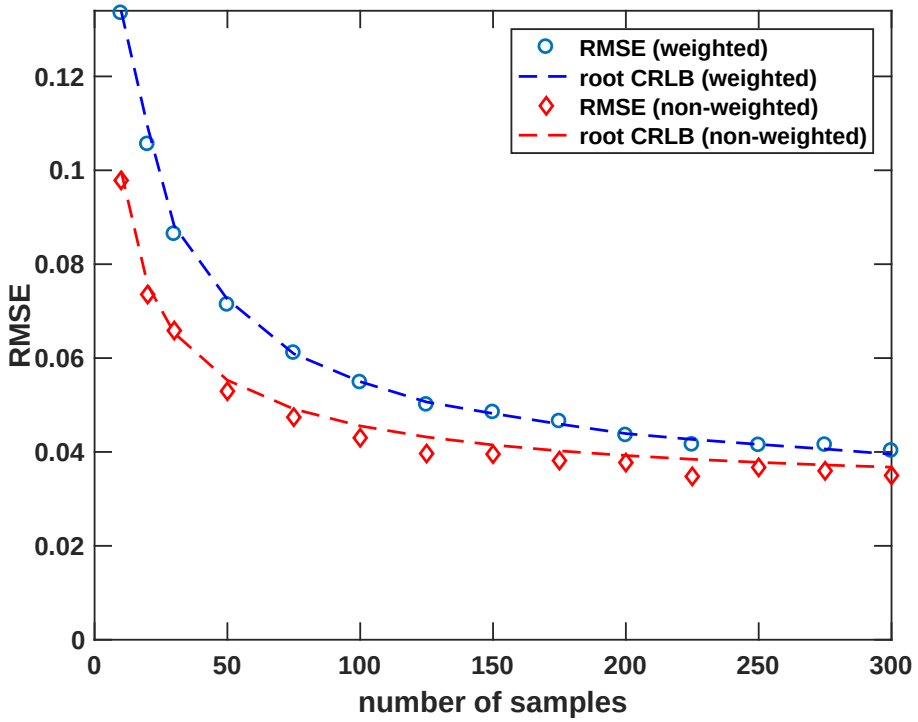

Figure 15: Obtained RMSE for the amplitudes, when using the sampling patterns for the weighted and non-weighted cases, respectively.

lection of available sampling points, is minimized while the sum of the variance of parameters of interest are guaranteed to be below a certain level. Due to the structure of the optimization problem, it is easy to add additional constraints, e.g., adding performance bounds on selected parameters, or putting more emphasize on a subset of the parameters, and to model for the uncertainty in *a-priory* assumptions of the parameter values. In the numerical section, we show that solving the proposed optimization problem is a more efficient approach than randomly selecting the sampling points, especially in the multi-dimensional setting. Further, we show that using the sampling schemes found by solving the proposed optimization problem, will provide a lower Cramér-Rao lower bound than that found from using ordinary uniform sampling. By using an efficient parameter estimator on the signal sampled according to the found sampling scheme, we show

that these Cramér-Rao lower bounds are, in fact, tight.

References

[1] S. Liu, M. Fardad, E. Masazade, and P. K. Varshney, "Optimal Periodic Sensor Scheduling in Networks of Dynamical Systems," *IEEE Trans. Sig*, vol. 62, no. 12, pp. 3055–3068, December 2014.

[2] H. Jamali-Rad, A. Simonetto, X. Ma, and G. Leus, "Distributed Sparsity-Aware Sensor Selection," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 5951–5964, November 2015.

[3] S. P. Chepuri, G. Leus, and A. J. van der Veen, "Sparsity-Exploiting Anchor Placement for Localization in Sensor Networks," in *21st European Signal Processing Conference*, 9-13 September 2013, pp. 1–5.

[4] S. Ravishankar and Y. Bresler, "Adaptive Sampling Design for Compressed Sensing MRI," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Boston, Massachusetts, 30 Aug.-3 Sept 2011, pp. 3751–3755.

[5] F. Gama, A. G. Marques, G. Mateos, and A. Ribeiro, "Rethinking Sketching as Sampling: Linear Transformation of Graph Signals," in *50th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, November 2016.

[6] A. Anis, A. Gadde, and A. Ortega, "Efficient Sampling Set Selection for Bandlimited Graph Signals Using Graph Spectral Proxies," *IEEE Trans. Signal Process.*, vol. 64, no. 14, pp. 3775–3789, July 2016.

[7] P. Schmieder, A. S. Stern, G. Wagner, and J. C. Hoch, "Application of nonlinear sampling scheme to COSY-type spectra," *Journal of Biomolecular NMR*, vol. 3, pp. 569–576, 1993.

[8] K. Kazimierczuk, A Zawadzka-Kazimierczuk, and W. Koźmiński, "Non-uniform frequency domain for uniform exploitation of non-uniform sampling," *J. Magn. Reson.*, vol. 205, pp. 286–292, 2010.

[9] S. G. Hyberts, K. Takeuchi, and G. Wagner, "Poisson-Gap Sampling and Forward Maximum Entropy Reconstruction for Enhancing the Resolution and Sensitivity of Protein NMR Data," *J Am Chem Soc.*, vol. 132, pp. 2145–2147, 2010.

[10] P. J. Sidebottom, "A new approach to the optimisation of non-uniform sampling schedules for use in the rapid acquisition of 2D NMR spectra of small molecules," *Magn Reson Chem*, vol. 54, no. 8, pp. 689–694, August 2016.

[11] K. Kazimierczuk and V. Y. Orekhov, "Accelerated NMR Spectroscopy by Using Compressed Sensing," *Angewandte Chemie International Edition*, vol. 50, no. 24, June 2011.

[12] K. Kazimierczuk and V. Y. Orekhov, "A comparison of convex and non-convex compressed sensing applied to multidimensional NMR," *J. Magn. Reson.*, vol. 223, pp. 1–10, 2012.

[13] S. G. Hyberts, H. Arthanari, S. A. Robson, and G. Wagner, "Perspectives in magnetic resonance: NMR in the post-FFT era," *J. of Magn. Reson.*, vol. 241, pp. 60–73, 2014.

[14] P. C. Aoto, R. B. Fenwick, G. J. A. Kroon, and P. E. Wright, "Accurate Scoring of Non-uniform Sampling Schemes for Quantitative NMR," *Journal of Magnetic Resonance*, vol. 246, pp. 31–35, Sept 2014.

[15] A. Özcelikkale, H. M. Ozaktas, and E. Arikan, "Signal Recovery with Cost-Constrained Measurements," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3607–3617, July 2010.

[16] Y. Yilmaz, S. Li, and X. Wang, "Sequential Joint Detection and Estimation: Optimum Tests and Applications," *IEEE Trans. Signal Process.*, vol. 64, no. 20, pp. 5311, Oct 2016.

[17] V. Kekatos, G. B. Giannakis, and B. Wollenberg, "Optimal Placement of Phasor Measurement Units via Convex Relaxation," *IEEE Trans. on Power Systems*, vol. 27, no. 3, pp. 1521–1530, Aug 2012.

[18] T. C-Gulcu and H. M. Ozaktas, "Choice of Sampling Interval and Extent for Finite-Energy Fields," *IEEE Trans. Signal Process.*, vol. 65, no. 7, pp. 1741–1751, April 2017.

[19] H. Zhang, J. M. F. Moura, and B. K. Krogh, "Dynamic Field Estimation Using Wireless Sensor Networks: Tradeoffs Between Estimation Error and Communication Cost," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2383–2395, June 2009.

[20] S. Liu, E. Masazade, and P. K. Varshney, "Temporally Staggered Sensing for Field Estimation with Quantized Data in Wireless Sensor Networks," in *IEEE Statistical Signal Processing Workshop (SSP)*, Ann Arbor, MI, USA, August 2012.

[21] S. Joshi and S. Boyd, "Sensor Selection via Convex Optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, February 2009.

[22] F. Pukelsheim, *Optimal design of experiments*, Wiley series in probability and mathematical statistics. Wiley, New York, 1993.

[23] S. P. Chepuri, *Sparse Sensing for Statistical Inference - Theory, Algorithms, and Applications*, Ph.D. thesis, Delft University of Technology, 2015.

[24] S. P. Chepuri and G. Leus, "Sparsity-Promoting Sensor Selection for Non-Linear Measurement Models," *IEEE Trans. Signal Process.*, vol. 63, no. 3, pp. 684–698, February 2015.

[25] S. Liu, S. P. Chepuri, M. Fardad, E. Masazade, and G. Leus P. K. Varshney, "Sensor Selection for Estimation with Correlated Measurement Noise," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3509–3522, July 2016.

[26] S. P. Chepuri and G. Leus, "Continuous Sensor Placement," *IEEE Signal Process. L*, vol. 22, no. 5, pp. 544–548, May 2015.

[27] J. Swärd, F. Elvander, and A. Jakobsson, "Designing Optimal Sampling Schemes," in *25th European Signal Processing Conference*, Aug 28 - Sep 2 2017.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

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