An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization

F. ELVANDER, T. KRONVALL, S. I. ADALBJORNSSON, AND A. JAKOBSSON

Published in: Elsevier Signal Processing
doi:10.1016/j.sigpro.2016.02.015

Lund 2016

Mathematical Statistics
Centre for Mathematical Sciences
Lund University

An Adaptive Penalty Multi-Pitch Estimator with Self-Regularization ☆

F. Elvander*[,a], T. Kronvall[a], S. I. Adalbjörnsson[a], A. Jakobsson[a]

[a] *Department of Mathematical Statistics, Lund University, P.O. Box 118, SE-221 00 Lund, Sweden*

Abstract

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

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

1. Introduction

Pitch estimation is a problem arising in a variety of fields, not least in audio processing. It is a fundamental building block in several music information retrieval applications, such as automatic music transcription, i.e., automatic sheet music generation from audio (see, e.g.,[1, 2]). Pitch estimation could also be used as a component in methods for cover song detection and music querying, possibly improving currently available services. For example, the popular query service Shazam [3] operates by matching hashed portions of spectrograms of user-provided samples against a large music database. As a change of instrumentation would alter the spectrogram of a song, such algorithms can only identify recordings of a song that are very similar to the actual recording present in the database. Thus, services such as Shazam might fail to identify, e.g., acoustic alternate versions of rock songs. A query algorithm based on pitch estimation could on the other hand correctly match the acoustic version to the original electrified one as it would recognize, e.g., the main melody. The applicability of pitch estimation to music is due to the fact that the notes produced by many instruments used in Western tonal music, e.g., woodwind instruments such as the clarinet, exhibit a structure that is well modeled using a harmonic sinusoidal structure [4]. However, for some plucked stringed instruments, such as the guitar and the piano, the tension of the string results in the harmonics deviating from perfect integer multiples of the fundamental frequency, a phenomenon called inharmonicity. For some instruments, such as the piano, there are models describing the structure of the inharmonicity based on physical properties of the instrument [5]. Such signals require agile pitch estimation algorithms allowing for this form of deviations (see, e.g., [6–8]). In this work, we will assume such deviations to be small, although noting that one may extend the here presented work along the lines in [6–8]. Estimating the fundamental frequencies of multi-pitch signals is generally a difficult problem. There are many methods available, see, e.g., [9], but most of them require *a priori* model order knowledge, i.e., they require knowledge of the number of

☆This work was supported in part by the Swedish Research Council and Carl Trygger's foundation. Parts of this material have been published at the EUSIPCO 2015 conference.

*Corresponding author. Phone: +46722004363.

Email addresses: filipelv@maths.lth.se (F. Elvander), ted@maths.lth.se (T. Kronvall), sia@maths.lth.se (S. I. Adalbjörnsson), aj@maths.lth.se (A. Jakobsson)

pitches present in the signal, as well as the number of active harmonics for each pitch.[1] Three such methods will be used in this work as reference estimators. The first method, here referred to as ORTH, exploits orthogonality between the signal and noise subspaces to form pitch frequency estimates. The second method is an optimal filtering method based on the Capon estimator, and is therefore here referred to as Capon. The third method is an approximate non-linear least squares method, here referred to as ANLS [10–12] (see also [9] for an overview of these methods). Methods not requiring *a priori* model order knowledge have also been proposed. For example, [13] uses a sparse dictionary representation of the signal and regularization penalties to implicitly choose the model order. A similar, but less general, method was introduced in [14], which used a dictionary specifically tailored to piano notes for estimating pitch frequencies generated by pianos. Other source specific methods include [15] and [16]. In [17], the author proposes a sparsity-exploiting method, where the dictionary atoms are learnt from databases of short-time Fourier transforms of musical notes. A similar idea is used in [18] for pitch-tracking in music. In [16] and [19], pitch estimation is based on the assumption of spectral smoothness, i.e., the amplitudes of the harmonics within a pitch are assumed to be of comparable magnitudes. Another field of research is performing multi-pitch estimation, often in the context of automatic music transcription, by decomposing the spectrogram of the signal into two matrices, one that describes the frequency content of the signal and one that describes the time activation of the frequency components. This method makes use of the non-negative matrix factorization, first introduced in this context in [20] and since then widely used, such as in, e.g., [21]. There are also more statistical approaches to multi-pitch estimation, posing the estimation as a Bayesian inference problem (see, e.g., [22]).

The approach to multi-pitch estimation presented in this work is to solve the problem in a group sparse modeling framework, which allows us to avoid making explicit assumptions on the number of pitches, or on the number of harmonics in each pitch. Instead, the number of components in the signal is chosen implicitly, by the setting of some tuning parameters. These tuning parameters determine how appropriate a given pitch candidate is to be present in the signal and may be set using cross-validation, or by using some simple heuristics. The sparse modeling approach has earlier been used for audio (see, e.g., [23]), and specifically for sinusoidal components in [24]. We extend on these works by exploiting the harmonic structure of the signals in a block sparse framework, where each block represents a candidate pitch. A similar method was introduced in [13], where block sparsity was enforced using block-norms, penalizing the number of active pitches. As the block-norm penalty, under some circumstances, cannot distinguish a true pitch from its sub-octave, i.e., the pitch with half the true fundamental frequency, the method is also complemented by a total variation penalty, which is shown to solve such issues. Total variation penalties are often applied in image analysis to obtain block-wise smooth image reconstructions (see, e.g., [25]). For audio data, one can similarly assume that signals often are block-wise smooth, as the harmonics of a pitch are expected to be of comparable magnitude [19]. Enforcing this feature will specifically deal with octave errors, i.e., the choosing of the sub-octave instead of the true pitch, as, in the noise free case, only every other harmonic of the sub-octave will have non-zero power. In this paper, we show that a total variation penalty, in itself, is enough to enforce a block sparse solution, if utilized efficiently. More specifically, by making the penalty function adaptive, we may improve upon the convex approximation used in [13], allowing us to drop the block-norm penalty altogether, and so reduce the number of tuning parameters. In some estimation scenarios, e.g., when estimating chroma using the approach in [26], this would simplify the tuning procedure significantly. Furthermore, we show that the proposed method performs comparably to that of [13], albeit with the notable improvement of requiring fewer tuning parameters. The method operates by solving a series of convex optimization problems, and to solve these we present an efficient algorithm based on the alternating directions method of multipliers (ADMM) (see, e.g., [27] for an overview of ADMM in the context of convex optimization). As the proposed method requires two tuning parameters to operate, we also present a scheme for automatic selection of appropriate model orders, thereby avoiding the need of user-supplied parameters. The remainder of this work is organized as follows; in the following section, we introduce the signal model, followed in Section 3 by the proposed estimation algorithm. Section 4 summarizes the efficient ADMM implementation whereas Section 5 examines how to adaptively choose the regularization parameters. Numerical results illustrating the achieved performance are presented in Section 6. Finally, Section 7 concludes upon the work.

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

2. Signal model

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

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

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

[Figure 1 about here.]

In general, selecting model orders in (1) may be a daunting task, with both the number of sources, K, and the number of harmonics in each of these sources, L_k, being unknown, as well as often being structured such that different sources may have spectrally overlapping overtones. In order to remedy this, this work proposes a relaxation of the model onto a predefined grid of $P \gg K$ candidate fundamentals, each having $L_{\max} \geq \max_k L_k$ harmonics. Here, $L_{\max}$ should be selected to ensure that the corresponding highest frequency harmonic is limited by the Nyquist frequency, and could thus vary depending on the considered candidate frequency (see also [13]). For notational simplicity, we will hereafter, without loss of generality, use the same $L_{\max}$ for all candidate frequencies. Assume that the candidate fundamentals are chosen so numerous and so closely spaced that the approximation

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

holds reasonably well. As only K pitches are present in the actual signal, we want to derive an estimator of the amplitudes $a_{p,\ell}$ such that only few, ideally $\sum_{k=1}^{K} L_k$, of the amplitudes in (2) are non-zero. This approach may be seen as a sparse linear regression problem reminiscent of the one in [24] and has been thoroughly examined in the context of pitch estimation in, e.g., [13, 29, 30]. For notational convenience, define the set of all amplitude parameters to be estimated as

$$\boldsymbol{\Psi} = \left\{ \boldsymbol{\Psi}_{\omega_1}, \ldots, \boldsymbol{\Psi}_{\omega_P} \right\} \tag{3}$$

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

where, as described above, most of the $a_{p,\ell}$ in $\boldsymbol{\Psi}$ will be zero. Note that $\boldsymbol{\Psi}$ will be sparse, i.e., having few non-zero elements. Also, the pattern of this sparsity will be group wise, meaning that if a pitch with fundamental frequency ω_p is not present, then neither will any of its harmonics, i.e., $\boldsymbol{\Psi}_{\omega_p} = \mathbf{0}$. Due to the harmonic structure of the signal, candidate pitches having fundamental frequencies at fractions of the present pitches fundamentals will have a partial fit of their harmonics. This may cause misclassification, i.e., erroneously identifying a present pitch as one or more non-present candidate pitches. This is the cause of the so-called sub-octave problem, which is mistaking the true pitch with fundamental frequency ω_p for the candidate pitch with fundamental frequency $\omega_p/2$. This may occur if the candidate set $\boldsymbol{\Psi}$ is structured such that the sub-octave pitch may perfectly model the true pitch, which is when $L_{\max} \geq 2L_p$. This is illustrated in Figure 1, displaying an extreme case with a pitch with fundamental frequency 100 Hz and four harmonics and as well as its sub-octave, i.e., a pitch with fundamental frequency 50 Hz and eight harmonics where only the even-numbered harmonics are non-zero. Relating to music signals, this is the same as mistaking a pitch for the pitch an octave below it. Thus, when estimating the elements of $\boldsymbol{\Psi}$, one also has to take into account the structure of the block sparsity, in order to avoid erroneously selecting sub-octaves.

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

3. Proposed estimation algorithm

Consider N samples of a noise-corrupted measurement of the signal in (1), $y(t)$, such that it may be well modeled as $y(t) = x(t) + e(t)$, where $e(t)$ is a broadband noise signal. A straightforward approach to estimate $\mathbf{\Psi}$ would then be to minimize the residual cost function

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

However, setting

$$\hat{\mathbf{\Psi}} = \underset{\mathbf{\Psi}}{\operatorname{argmin}}\ g_1(\mathbf{\Psi}) \tag{6}$$

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

$$\hat{\mathbf{\Psi}} = \underset{\mathbf{\Psi}}{\operatorname{argmin}}\ g_1(\mathbf{\Psi}) + \lambda||\mathbf{\Psi}||_0 \tag{7}$$

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

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

The resulting problem

$$\min_{\mathbf{\Psi}}\ g_1(\mathbf{\Psi}) + \lambda g_2(\mathbf{\Psi}) \tag{9}$$

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

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

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

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

where we define

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

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

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

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

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

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

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

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

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

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

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

$$\mathbf{y} = \left[\begin{array}{ccc} y(1) & \ldots & y(N) \end{array} \right]^T \tag{18}$$

$$= \sum_{p=1}^{P} \mathbf{W}_p \, \mathbf{a}_p + \mathbf{e} \triangleq \mathbf{W}\mathbf{a} + \mathbf{e} \tag{19}$$

where

$$\mathbf{W} = \left[\begin{array}{ccc} \mathbf{W}_1 & \ldots & \mathbf{W}_P \end{array} \right] \tag{20}$$

$$\mathbf{W}_p = \left[\begin{array}{ccc} \mathbf{z}_p^1 & \ldots & \mathbf{z}_p^{L_{\max}} \end{array} \right] \tag{21}$$

$$\mathbf{z}_p = \left[\begin{array}{ccc} e^{i\omega_p 1} & \ldots & e^{i\omega_p N} \end{array} \right]^T \tag{22}$$

$$\mathbf{a} = \left[\begin{array}{ccc} \mathbf{a}_1^T & \ldots & \mathbf{a}_P^T \end{array} \right]^T \tag{23}$$

$$\mathbf{a}_p = \left[\begin{array}{ccc} a_{p,1} & \ldots & a_{p,L_{\max}} \end{array} \right]^T \tag{24}$$

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

$$\hat{\mathbf{a}}^{(k)} = \underset{\mathbf{a}}{\operatorname{argmin}} \ \frac{1}{2}\left\|\mathbf{y}-\mathbf{H}_1^{(k)}\mathbf{a}\right\|_2^2 + \lambda_2\left\|\mathbf{H}_2^{(k)}\mathbf{a}\right\|_1 + \lambda_4\left\|\mathbf{H}_4^{(k)}\mathbf{a}\right\|_1 \tag{25}$$

where

$$\mathbf{H}_1^{(k)} = \mathbf{W} \tag{26}$$

$$\mathbf{H}_2^{(k)} = \operatorname{diag}\left(1/\left(\left|\hat{\mathbf{a}}^{(k-1)}\right| + \epsilon\right)\right) \tag{27}$$

$$\mathbf{H}_4^{(k)} = \mathbf{F}\operatorname{diag}\left(\arg\left(\hat{\mathbf{a}}^{(k-1)}\right)\right)^{-1} \tag{28}$$

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

$$f_{k,\ell} = \begin{cases} 1 & \text{if } k=\ell=1 \\ -1 & \text{if } k=\ell, \ell\neq 1 \\ 1 & \text{if } k=\ell+1 \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

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

4. ADMM implementation

In order to solve (25), we proceed to introduce an efficient ADMM implementation. To this end, let $\mathbf{z}\in\mathbb{C}^{PL_{\max}}$ be the primal optimization variable and introduce the auxiliary variables $\mathbf{u}_1\in\mathbb{C}^N$, $\mathbf{u}_2\in\mathbb{C}^{PL_{\max}}$, and $\mathbf{u}_4\in\mathbb{C}^{P(L_{\max}+1)}$ and let

$$\mathbf{G}^{(k)} = \begin{bmatrix} \mathbf{H}_1^{(k)T} & \mathbf{H}_2^{(k)T} & \mathbf{H}_4^{(k)T} \end{bmatrix}^T \tag{30}$$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1^T & \mathbf{u}_2^T & \mathbf{u}_4^T \end{bmatrix}^T \tag{31}$$

Thus, we want to solve

$$\min_{\mathbf{z}} f(\mathbf{G}^{(k)}\mathbf{z}) \tag{32}$$

where

$$f(\mathbf{G}^{(k)}\mathbf{z}) = \frac{1}{2}\left\|\mathbf{y}-\mathbf{H}_1^{(k)}\mathbf{z}\right\|_2^2 + \lambda_2\left\|\mathbf{H}_2^{(k)}\mathbf{z}\right\|_1 + \lambda_4\left\|\mathbf{H}_4^{(k)}\mathbf{z}\right\|_1 \tag{33}$$

Algorithm 1 The proposed PEBSI-Lite algorithm

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

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

$$\begin{aligned} &\min_{\mathbf{z},\mathbf{u}} f(\mathbf{u}) + \frac{\mu}{2}\left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u}\right\|_2^2 \\ &\text{subject to } \mathbf{G}^{(k)}\mathbf{z} - \mathbf{u} = \mathbf{0} \end{aligned} \tag{34}$$

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

$$\mathbf{d} = \left[\begin{array}{ccc} \mathbf{d}_1^T & \mathbf{d}_2^T & \mathbf{d}_4^T \end{array}\right]^T \tag{35}$$

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

$$L_\mu(\mathbf{z},\mathbf{u},\mathbf{d}) = f(\mathbf{u}) + \frac{\mu}{2}\left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u} - \mathbf{d}\right\|_2^2 - \frac{\mu}{2}\left\|\mathbf{d}\right\|_2^2 \tag{36}$$

Also, define

$$\boldsymbol{\zeta}(j) = \left[\begin{array}{ccc} \boldsymbol{\zeta}_1^T(j) & \boldsymbol{\zeta}_2^T(j) & \boldsymbol{\zeta}_4^T(j) \end{array}\right]^T \tag{37}$$

where

$$\boldsymbol{\zeta}_\ell(j) = \mathbf{H}_\ell^{(k)}\mathbf{z}(j+1) - \mathbf{d}_\ell(j)\ ,\ \ell = 1,2,4 \tag{38}$$

The Lagrangian (36) is separable in the variables $\mathbf{z}$, $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_4$ and one may thus form an updating scheme similar to that in [25], as

$$\mathbf{z}(j+1) = \underset{\mathbf{z}}{\text{argmin}}\ \left\|\mathbf{G}^{(k)}\mathbf{z} - \mathbf{u}(j) - \mathbf{d}(j)\right\|_2^2 \tag{39}$$

$$\mathbf{u}_1(j+1) = \underset{\mathbf{u}_1}{\text{argmin}}\ \frac{1}{2}\left\|\mathbf{y} - \mathbf{u}_1\right\|_2^2 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_1(j) - \mathbf{u}_1\right\|_2^2 \tag{40}$$

$$\mathbf{u}_2(j+1) = \underset{\mathbf{u}_2}{\text{argmin}}\ \lambda_2\left\|\mathbf{u}_2\right\|_1 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_2(j) - \mathbf{u}_2\right\|_2^2 \tag{41}$$

$$\mathbf{u}_4(j+1) = \underset{\mathbf{u}_4}{\text{argmin}}\ \lambda_4\left\|\mathbf{u}_4\right\|_1 + \frac{\mu}{2}\left\|\boldsymbol{\zeta}_4(j) - \mathbf{u}_4\right\|_2^2 \tag{42}$$

$$\mathbf{d}(j+1) = \mathbf{u}(j+1) - \boldsymbol{\zeta}(j) \tag{43}$$

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

$$\mathbf{z}(j+1) = \left(\mathbf{G}^{(k)H}\mathbf{G}^{(k)}\right)^{-1}\mathbf{G}^{(k)H}\left(\mathbf{u}(j) + \mathbf{d}(j)\right) \tag{44}$$

and

$$\mathbf{u}_1(j+1) = \frac{\mathbf{y} + \mu\boldsymbol{\zeta}_1(j)}{1+\mu} \tag{45}$$

respectively. Using the element-wise shrinkage function,

$$\mathbf{T}\left(\mathbf{x}, \xi\right) = \frac{\max(|\mathbf{x}| - \xi, 0)}{\max(|\mathbf{x}| - \xi, 0) + \xi} \odot \mathbf{x} \tag{46}$$

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

$$\mathbf{u}_2(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_2(j), \frac{\lambda_2}{\mu}\right) \tag{47}$$

and

$$\mathbf{u}_4(j+1) = \mathbf{T}\left(\boldsymbol{\zeta}_4(j), \frac{\lambda_4}{\mu}\right) \tag{48}$$

respectively. The resulting PEBSI-Lite algorithm is summarized in Algorithm 1, where the solution is given as $\hat{\mathbf{a}} = \hat{\mathbf{a}}^{(k_{\text{end}})}$ with k_{end} denoting the last iteration index of the outer loop. The complexity of the resulting algorithm will be dominated by the computation of step 5 in Algorithm 1. This system of equations can be solved efficiently by storing the Cholesky factorization of the matrix to be inverted, with a one-time cost of $\mathcal{O}(p^3)$ operations, where p denotes the number of variables (here, assumed to be larger than the number of data points). Furthermore, at each iteration, one needs to perform a back solve costing $\mathcal{O}(p^2)$ operations.

5. Self-regularization

The quality of the pitch estimates produced by the PEBSI-Lite algorithm depend on the values of the regularization parameters λ_2 and λ_4. In general, large values of λ_2 encourage sparse solutions while large values of λ_4 encourage solutions that are smooth within blocks. As the model order is unknown, it is generally hard to determine how sparse the solution should be in order to be considered the desired one. Therefore, one often determines the values of the regularization parameters using cross-validation schemes, making the performance of the methods user dependent. Instead, one would like to have a systematic and preferable automatic method for choosing λ_2 and λ_4, and thereby the model order. A common approach to solving model order problems is to use information criteria such as AIC or BIC [35], which measure the fit of the model to the data, while penalizing high model orders, resulting in a trade-off criterion that should take its optimal value for the correct model order. For the LASSO problem, there have been suggestions of appropriate model order criteria [36], [37]. In [13], the authors suggest a BIC-style criterion for multi-pitch estimation for given regularization parameters. However, this criterion can only be used to determine which of the found pitches are true and which are spurious, and not to determine the appropriate regularization parameters. Thus, even if one has an efficient criterion for choosing between different models, one first has to form a set of candidate models, in effect running Algorithm 1 for different values of λ_2 and λ_4. For the simpler case of the LASSO, the analog is to solve (9) for all $\lambda \in \mathbb{R}_+$, for which there are algorithms such as LARS [38]. There have also been methods suggested to solve the LASSO for only a finite number of values λ, i.e., only values of the regularization parameter where the number of active components of the solution change (see, e.g., [37]). For our problem, the analog is to find solutions for the set of parameter values

$$\{(\lambda_2, \lambda_4) | (\lambda_2, \lambda_4) \in \mathbf{R}_+ \times \mathbf{R}_+\} \tag{49}$$

[Figure 2 about here.]

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

$$\min_{\boldsymbol{\beta}} \frac{1}{2} \|\mathbf{y} - \mathbf{W}\boldsymbol{\beta}\|_2^2 + \lambda \|\mathbf{D}\boldsymbol{\beta}\|_1 \tag{50}$$

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

Despite the above efficient ADMM implementation, it is computationally cumbersome to conduct a search on (49) in order to find an appropriate model order, with the computation complexity increasing both in the case of longer signals, and when using more elements in the dictionary. Instead of constructing a fully general path algorithm for PEBSI-Lite, we therefore proceed to propose a scheme for constructing a reduced size signal adapted dictionary that combined with a parametrization of the regularization parameters (λ_2, λ_4) will allow us to form good pitch estimates without having to predefine values of the regularization parameters, by means of a simple line search instead of searching through (49). The proposed dictionary construction begins by estimating the frequency content of the signal without imposing any harmonic structure. This estimation may be performed by any standard method, such as ESPRIT (see, e.g., [42]). As the number of sinusoidal components is unknown, estimates corresponding to different model orders can be evaluated using, for instance, the BIC criterion (see, e.g., [35])

$$\text{BIC}_k = 2N \log \hat{\sigma}_k^2 + (5k + 1) \log N \tag{51}$$

where $\hat{\sigma}_k^2$ is the maximum likelihood estimate of the residual variance corresponding to the model constituted by k estimated sinusoids, in order to choose a suitable model order. The accuracy of the frequency estimates produced by ESPRIT will suffer if a too low model order is determined, whereas it is less sensitive to cases when the model order is moderately overestimated. Thus, we propose to increase the robustness of the frequency estimates by using $k+\delta$, $\delta \geq 1$, estimated sinusoids for the case when order k is determined optimal by the BIC. As the only interesting pitch candidates are those having at least one harmonic corresponding to a present sinusoidal component, we can then design a considerably reduced dictionary, containing only pitches with such matching harmonics. If one has some prior knowledge of the nature of the signal, one could impose stronger assumptions on the candidate pitches in order to reduce the dictionary further, e.g., by allowing only pitches whose first harmonic is found in the set of estimated sinusoids. Using the obtained dictionary, one could then proceed to conduct a search for λ_2 and λ_4. Although considerably cheaper as compared to when performed using a full dictionary, a complete evaluation of the $\lambda_2\lambda_4$-plane is still somewhat expensive. To avoid a full grid search, the following heuristic concerning the connection between λ_2 and λ_4 can be used. Assume that we have a single-pitch signal where all L_k harmonics have equal magnitude r. Further, assume that when setting $\lambda_4 = 0$, λ' is the largest value of λ_2 resulting in a nonzero solution, where each harmonic amplitude is estimated to r_0. If we would instead set $\lambda_2 = 0$, and consider which value of λ_4 that should result in the same solution, this value should be

$$\lambda_4 = \frac{L_k}{2} \lambda' \tag{52}$$

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

$$\lambda_2 = \alpha\lambda' \ , \ \lambda_4 = (1 - \alpha)\frac{L_k}{2}\lambda' \tag{53}$$

yields the penalty $\lambda' \, L_k r_0$ for all $\alpha \in [0, 1]$. If we assume (53) to be true, we should, for spectrally smooth signals, expect to see ridges in the solution surface where the number of pitches present in the solution changes, and the shapes of the ridges in the $\lambda_2\lambda_4$-plane should be described by lines similar to (53). This is

Algorithm 2 Self-Regularized PEBSI-Lite

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

illustrated in Figure 2, presenting a plot of the number of pitches present in the solution for different values (λ_2, λ_4) for a signal consisting of three pitches with fundamental frequencies 400, 550 and 700 Hz, and with 4, 8, and 12 harmonics, respectively. The magnitude of each harmonic amplitude has been drawn uniformly on $(0.9, 1.1)$ and each phase has been drawn uniformly on $(0, 2\pi)$. The signal was sampled at frequency 20 kHz in a time frame of length 40 ms, generating 800 samples of the signal. The Signal-to-Noise Ratio (SNR), as defined in (56), was 20 dB. On the plateau with two pitches, the pitch with four harmonics have been forced to zero, whereas on the plateau with one pitch present, only the pitch with twelve harmonics is present. Note the shape of the different plateaus: seen in the $\lambda_2\lambda_4$-plane, the slopes of the ridges seem to be well described by (53) where $L_k = 4, 8$, and 12, for the three ridges corresponding to changes from three to two, from two to one, and from one to zero pitches, respectively. The signal corresponding to Figure 2 has a relatively low level of noise. Increasing the noise level, the least regularized solutions, i.e., with λ_2 and λ_4 close to zero, results in more than three non-zero pitches. Guided by this observation, one could reduce the search for (λ_2, λ_4) from a 2-D to a 1-D search by using a re-parametrization. Keeping the plateaus in Figure 2 and our assumption of spectral smoothness in mind, we should expect a desirable solution to correspond to a (λ_2, λ_4)-pair with $\lambda_2 \leq \lambda_4$. In order to get solutions regularized with respect to spectral smoothness, while keeping the risk of getting only zero solutions low, the following parametrization can be used. Let λ denote the only free parameter and set

$$\lambda_2 = \lambda \tag{54}$$

$$\lambda_4 = \frac{L}{2}\lambda \tag{55}$$

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

6. Numerical results

We proceed to examine the performance of the proposed algorithm using signals simulated from the pitch model (1) as well as synthetic audio signals generated from MIDI, and measured audio signals.

6.1. Two-pitch signal

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

[Figure 3 about here.]

[Table 1 about here.]

$$\text{SNR} = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2} \tag{56}$$

where σ_x^2 and σ_e^2 is the power of the signal and the noise, respectively. For a pitch signal generated by (1), under the simplifying assumption of distinct sinusoidal components, the power of the signal is given by

$$\sigma_x^2 = \sum_{k=1}^{K} \sum_{\ell=1}^{L_k} \frac{|a_{k,\ell}|^2}{2} \tag{57}$$

At each SNR, 200 Monte Carlo simulations were performed, each simulation generating a signal with fundamental frequencies of 600 and 730 Hz. As PEBS and PEBS-TV rely on a predefined frequency grid, the fundamental frequencies were randomly chosen at each simulation uniformly on $600 \pm d/2$ and $730 \pm d/2$, where d is the grid point spacing, to reflect performance in present of off-grid effects. The phases of the harmonics in each pitch were chosen uniformly on $[0, 2\pi)$, whereas all had unit magnitude. The signal was sampled at $f_s = 48$ kHz on a time frame of 10 ms, yielding $N = 480$ samples per frame. As a result, the pitches were spaced by approximately f_s/N Hz, which is the resolution limit of the periodogram. This is also seen in Figure 3, illustrating the resolution of the periodogram as well as the frequencies of the harmonics, at SNR $= -5$ dB. From the figure, it may be concluded that the signal contains more than one harmonic source, as the observed peaks are not harmonically related. Furthermore, it is clear that the fundamental frequencies are not separated by the periodogram, indicating that any pitch estimation algorithm based on the periodogram would suffer notable difficulties. For PEBSI-Lite, the estimates are formed using Algorithm 2 with $\tau = 0.1$ and $\epsilon = 0.05$. The safety margin for the sinusoidal model order is $\delta = 1$. For PEBS and PEBS-TV, the estimation procedure is initiated using a coarse dictionary, with candidate pitches uniformly distributed on the interval $[280, 1500]$ Hz, thus also including $\omega_p/2$ and $2\omega_p$ for both pitches. The coarse resolution was $d = 10$ Hz, i.e., still a super-resolution of $f_s/10N$. After estimation on this grid, a zooming step was taken where a new grid with spacing $d/10$ was laid $\pm 2d$ around each pitch having non-zero power. The regularization parameter values used for PEBS-TV and PEBS are presented in Table 1. The values where selected using *manual cross-validation* for similar signals. Comparisons were also made with the ANLS, ORTH, and the harmonic Capon estimators, which had been given the *oracle* model orders (see [9] for more details on these methods). The simulation and estimation procedure was performed for two cases; one where the number of harmonics L_k were set to 5 and 6, and one where L_k were set to 10 and 11. In the former case, $L_{\max} = 10$ and in the latter $L_{\max} = 20$, i.e., well above the true number of harmonics. Figures 4 and 5 show the percentage of pitch estimates where both lie within ±2 Hz from the true values for the six compared methods, for the case of 5 and 6 as well as 10 and 11 harmonics, respectively. In this setting, PEBS performs poorly, as the generous choices of $L_{\max}$ allows it to pick the sub-octave, as predicted. As can be seen in Figure 4, PEBSI-Lite performs better than all reference methods for SNRs above and including 10 dB despite not having the model order information given to

[Figure 4 about here.]

[Figure 5 about here.]

[Figure 6 about here.]

[Figure 7 about here.]

ORTH, ANLS, and Capon, nor having the supervised regularization parameters choices of PEBS and PEBS-TV. Though, in higher noise settings, the performance of PEBSI-Lite degrades and its pitch frequency estimates are worse than those produced by the reference methods for SNRs below 10 dB. For the case of 10 and 11 harmonics, PEBSI-Lite performs on par with the reference methods for SNRs above and including 15 dB, while performing worse in higher noise settings. As shown in Figures 6 and 7, the drop in performance for lower SNRs results from the difficulty of accurately estimating the total number of sinusoids, as used by the ESPRIT step, for such signals. In Figure 6, the percentage of the estimates in which the the BIC criterion (51) correctly determines the number of sinusoidal components in the signal is presented, whereas Figure 7 shows the percentage of the estimates in which the BIC criterion (51) determines a too low model order. As is clear from the figures, the model order estimates strongly degrade for lower SNRs, thus causing the PEBSI-Lite dictionary to be inaccurate. Clearly, all the other methods here shown using oracle model order information would suffer drastically from such inaccuracies, although it should be stressed that one may expect these methods to suffer further, as they also need to perform an exhaustive combinatorial search to determine the number of pitches given the found number of sinusoids.

6.2. Three-pitch signal

To further examine the performance of Algorithm 2, it was evaluated using a simulated triple-pitch signal, measured in white Gaussian noise at different SNR levels, ranging from 0 dB to 25 dB, in steps of 5 dB. Instead of using unit magnitudes of the harmonics, as was the case for the above presented two-pitch setting, the spectral envelopes of the three pitch components were constructed from periodograms of three different speech recordings. The formants of the three pitches are displayed in Figure 8. The pitches had fundamental frequencies 200, 350, and 530 Hz, and 7, 8, and 11 harmonics, respectively. At each level of SNR, 1000 Monte Carlo simulations were performed, where the fundamental frequencies were chosen uniformly on 200 ± 2.5, 350 ± 2.5, and 530 ± 2.5 Hz, respectively, and the phase of each harmonic was chosen uniformly on $[0, 2\pi)$. The signal was sampled in a 40 ms window at a sampling frequency of 20 kHz, generating 800 samples of the signal. The algorithm settings were $\tau = 0.1$, $\epsilon = 0.05$, and $\delta = 1$. Here, Algorithm 2 was compared to the ANLS, ORTH, harmonic Capon, as well as PEBS-TV estimators. The three first comparison methods were given the oracle model orders. To illustrate the fact that the choice of regularization parameter values is not universal, the values found using cross-validation for the two-pitch case (see Table 1) were used for PEBS-TV initially. However, this resulted in such poor performance that the parameter values had to be slightly altered in order to make PEBS-TV an interesting reference method. As a compromise, the parameter values corresponding to SNR 20 dB in Table 1 were used for all SNRs in this simulation setting.

[Figure 8 about here.]

[Figure 9 about here.]

For the dictionaries of PEBSI-Lite and PEBS-TV, $L_{\max} = 16$ was used, well above the true model orders. Figure 9 shows the percentage of the pitch estimates where all three pitch estimates lie within ±2 Hz of the true values for the five different methods. As can be seen, the performance of PEBSI-Lite is again poor for low SNRs while improving considerably for lower noise levels. The low scoring for PEBSI-Lite for low SNRs is mainly due to the selection of wrong model orders. This is illustrated in Figure 10, which shows the percentage of the estimates in which PEBSI-Lite and PEBS-TV selects the correct number of pitches. As can be seen, for an SNR of 0 dB, PEBSI-Lite selects the true model order in less than 10% of the simulations. Mostly, a too high model order is selected, which is to be expected as the model order choice is based on the power of the model residual and that the pitch estimates depend on the accuracy of the initial ESPRIT estimates. Arguably, one could improve on these results by either using prior knowledge of the noise level or by estimating it, and based on this make the model order selection scheme more robust. Figure 11 shows the root mean squared error (RMSE) for the estimated fundamental frequencies. Instead of presenting three separate RMSE plots, Figure 11 shows an aggregate version where the MSE for the three pitches have been summed. In order to compute relevant RMSE values for PEBSI-Lite and PEBS-TV, estimates where the model order has not been correctly determined have been discarded. Thus, for an SNR level of 0 dB, the RMSE values for PEBSI-Lite is based on quite few samples. However, as PEBSI-Lite finds the correct model order for high SNR levels with high probability, the corresponding RMSE values are more trustworthy in these regions. For the reference methods ORTH, ANLS, Capon, and PEBS-TV, some of the estimates deviate from the true pitch frequencies with as much as 100 Hz, resulting in very large RMSE values should all estimates be used in their computation. Thus, in order to obtain RMSE values comparable to that of the PEBSI-Lite estimates, only estimates found within 2 Hz of the true pitch frequencies are used when computing RMSE for the reference methods. With this, as can be seen in Figure 11, PEBSI-Lite performs worse than the reference methods for SNRs below and including 10 dB, while outperforming all reference methods except Capon for SNRs above and including 20 dB. Though, one should bear in mind that the RMSE values for Capon for these SNRs are based on only 15% respectively 8% of the available pitch estimates, as can be seen in Figure 9, and that the Capon method has been allowed oracle model order knowledge. Also presented in Figure 11 is the root Cramér-Rao lower bound (CRLB) for the estimates of the pitch frequencies. As the frequencies of the harmonics in this case are distinct and the additive noise is white Gaussian, the lower limit for the variance of an unbiased pitch frequency estimate $\hat{f}_k$ is given by [9]

$$\mathrm{Var}(\hat{f}_k) \geq \frac{6\sigma^2(f_s/2\pi)^2}{N(N^2-1)\sum_{\ell=1}^{L_k}|a_{k,\ell}|^2\ell^2} \tag{58}$$

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

$$\mathrm{CRLB} = \sum_{k=1}^{3} \frac{6\sigma^2(f_s/2\pi)^2}{N(N^2-1)\sum_{\ell=1}^{L_k}|a_{k,\ell}|^2\ell^2} \tag{59}$$

As can bee seen in Figure 11, PEBSI-Lite, as well as the other methods, fail to reach the CRLB. In an attempt to improve the PEBSI-Lite estimates for SNR levels above and including 15 dB, a non-linear least squares (NLS) search was performed, using the presented algorithm estimate as and initial estimate of all the unknown parameters, including the model orders. This means that we obtain refined estimates of the pitch frequencies f_k contained in the vector $\mathbf{f}$ as (see, e.g, [42])

$$\mathbf{f} = \underset{\mathbf{f}}{\operatorname{argmax}}\ \mathbf{y^H B (B^H B)^{-1} B^H y} \tag{60}$$

where $\mathbf{B}$ is a block matrix consisting of K blocks,

$$\mathbf{B} = [\mathbf{B}_1, \ldots, \mathbf{B}_K] \tag{61}$$

where each block $\mathbf{B}_j$ corresponds to a separate pitch and is constructed as

$$\mathbf{B}_j = \begin{bmatrix} e^{i2\pi f_j/f_s t_1} & \ldots & e^{i2\pi L_j f_j/f_s t_1} \\ \vdots & & \vdots \\ e^{i2\pi f_j/f_s t_N} & \ldots & e^{i2\pi L_j f_j/f_s t_N} \end{bmatrix} \tag{62}$$

Given that the PEBSI-Lite estimates are fairly close to the true pitch frequencies, we expect the NLS scheme to converge if we solve (60) using routines like MATLAB's *fminsearch* initialized with the PEBSI-Lite estimates. However, the success of such a scheme is not only dependent on good initial frequency estimates, we also need the true number of harmonics L_j for each pitch. Figure 12 presents a plot of the average absolute error in the number of detected harmonics for each pitch for the test signal when using PEBSI-Lite. As can be seen, the number of detected harmonics is only correct for the third pitch even for the largest SNRs. The errors in number of harmonics for the first and second pitches are due to the relatively small amplitudes of both pitches highest order harmonics, as shown in Figure 8, making these harmonics prone to occasionally being cancelled out by the PEBSI-Lite regularization penalties. Using erroneous harmonic orders as input to the NLS search, we expect the resulting pitch frequency estimates to be somewhat biased. Indeed, this is what happens. Figure 13 presents a plot of the RMSE of the pitch frequency estimates when the PEBSI-Lite estimates for SNRs above and including 15 dB have been post-processed using NLS. As can be seen, the estimator still fails to reach the CRLB, although the estimation errors have become smaller. Note also that the slopes of the RMSE curve for PEBSI-Lite and CRLB are now somewhat different, which is due to that the erroneous harmonic orders induces varying degrees of bias in the estimates. Considering computational complexity, ANLS and ORTH are by far the fastest methods, with average running times of 0.03 and 1.6 seconds per estimation cycle on a regular PC, respectively. For Capon and PEBS-TV, the corresponding running times are 6.1 and 6.4 seconds for the considered example, respectively, while running PEBSI-Lite using Algorithm 2 requires on average 40.1 seconds per estimation cycle. As a comparison, it may be noted that if one replaces Algorithm 1 in Algorithm 2 to instead use SeDuMi or SDPT3, the computation time for this step of Algorithm 2 increases almost tenfold[3].

[Figure 10 about here.]

[Figure 11 about here.]

[Figure 12 about here.]

[Figure 13 about here.]

[Figure 14 about here.]

[Figure 15 about here.]

[Figure 16 about here.]

[Figure 17 about here.]

[Figure 18 about here.]

Although Algorithm 2 is considerably more expensive to run than the reference methods, it should be noted that the method does not require any user input in terms of regularization parameter values. PEBS-TV could arguably be tuned to perform on par with PEBSI-Lite if one is allowed to change the values of its regularization parameters. However, PEBS-TV needs the setting of three parameter values and after trying only seven such triplets, the computational time is the same as running Algorithm 2 in its entirety.

6.3. MIDI and measured audio signals

Figure 14 shows a plot of the spectrogram of a signal consisting of three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz. The signal was sampled initially at 44 kHz and then down sampled to 20 kHz. The 311 Hz saxophone starts out alone and is after 0.45 seconds joined by the 277 Hz saxophone and after 0.95 seconds by the 440 Hz saxophone. The image is quite blurred for the later parts of the signal, but for the first half second, one can clearly see the harmonic structure of the saxophone pitch. It is worth noting that a large number of harmonics is present. Figure 15 shows pitch estimates produced by Algorithm 2, using $\tau = 0.1$ and $L_{\max} = 15$, when applied to the same signal, using windows of lengths 40 ms. As can be seen, the estimates are quite accurate, with the exception of the beginning of the first tone and for a single frame where the 440 Hz pitch is mistaken for a 220 Hz pitch. It is worth noting that such errors may be avoided using the information resulting from earlier frames, for instance using an approach similar to [22]. The figure also shows the estimated pitch tracks obtained using the ESACF estimator [43]; this estimator requires *a priori* knowledge of the number of sources in the signal, but is, given this information, able to estimate the number of harmonics of each source. Here, EACF has thus been provided *oracle* knowledge of the number of sources, with each source given the same maximum harmonic order as used by PEBSI-Lite (as before, the latter also has to estimate the number of sources). As can be seen from the figure, the ESACF estimator fails to track the pitches in several of the frames. In particular, it fails to estimate the pitch with fundamental frequency 440 Hz altogether.

Furthermore, Figure 16 examines the performance of the PEBSI-Lite estimator when applied to a measured audio signal. The considered signal consists of three trumpets playing the notes A4, B4, and $C^{\#}4$, which, using concert tuning, corresponds to the fundamental frequencies 440, 493.883, and 554.365 Hz, respectively. However, it should be noted, that as the musicians play with vibrato, the fundamental frequencies

[3] For all algorithms, the given execution times are those of direct implementations of the corresponding methods. Clearly, these methods can be more efficiently implemented by fully exploiting their inherent structures.

are not constant across the frames, which may also be seen in the resulting estimates. To facilitate for a comparison, the ground truth estimates of the fundamental frequencies have been obtained using the joint order and (single) pitch estimation algorithm ANLS, presented in [11], when applied to each individual trumpet separately. As a comparison, the figure also shows the three fundamental frequencies obtained using the ESACF estimator (which has here, again, been allowed *oracle* knowledge of the number of sources, but using the same maximum number of harmonics as used by PEBSI-Lite). As can be seen, PEBSI-Lite accurately tracks each of the three pitches, even catching the pitch variations caused by the vibrato. As before, it may be noted that the estimates produced by ESACF have lower accuracy as compared to PEBSI-Lite, with the ESACF estimator here erroneously picking one of the sub-octaves in some of the frames. The trumpet signal was sampled at 8 kHz. The pitch estimates where formed in non-overlapping frames of length 30ms.

The performance of PEBSI-Lite and ESACF on real audio was also evaluated on the Bach10 dataset [44]. This dataset consists of ten string quartets composed by Johann Sebastian Bach. The parts are performed by a violin, a clarinet, a saxophone, and a bassoon, with each piece being approximately 30 seconds long. Each piece was sampled at 44.1 kHz, then downsampled to 22.05 kHz, and divided into non-overlapping frames of length 30 ms. Estimates of the ground truth fundamental frequencies in each frame were obtained by applying YIN [45] to each individual channel. Obvious errors in the YIN estimates were then corrected manually. As before, to yield its best possible performance, ESACF was given *oracle* knowledge of the number of present pitches and both methods were given a maximum harmonic order of 15. For PEBSI-Lite, $\tau = 0.1$ was used. Table 2 presents the resulting measures of the *accuracy*, *precision*, and *recall* for the dataset, defined as

$$\text{Accuracy} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i) + \text{FP}(t,i) + \text{FN}(t,i)} \tag{63}$$

$$\text{Precision} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i) + \text{FP}(t,i)} \tag{64}$$

$$\text{Recall} = \frac{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i)}{\sum_{i=1}^{I}\sum_{t=1}^{T_i} \text{TP}(t,i) + \text{FN}(t,i)} \tag{65}$$

where $\text{TP}(t,i)$, $\text{FP}(t,i)$, and $\text{FN}(t,i)$ denote the number of true positive, false positive, and false negative pitch estimates, respectively, for frame t in music piece i. Furthermore, T_i is the number of frames for music piece i, whereas I is the number of music pieces. Here, an estimated pitch is associated with a ground truth pitch only if its fundamental frequency lies within a quarter tone, or 3%, of the ground truth pitch (see also, e.g., [46]). To avoid the most non-stationary frames, where we cannot expect the estimates produced by PEBSI-Lite and ESACF, nor the ground truth, to be reliable, frames containing note onsets, defined as frames where one of the ground truth pitches change with more than a semi-tone, have been excluded when computing the measures. As can be seen from the table, PEBSI-Lite performs better than ESACF for all of the three considered measures *accuracy*, *precision*, and *recall.* As PEBSI-Lite does, for now, not incorporate information between adjacent frames, these results are most promising for what might be achievable when extended to include such information. As an illustration of the performance, Figures 17 and 18 present pitch tracks produced by PEBSI-Lite and ESACF when applied to the first 15 seconds of one of the pieces in the dataset, namely *Ach, lieben Christen.* As can be seen from the figures, PEBSI-Lite tracks the fundamental frequencies of the violin, the saxophone, and the bassoon fairly well, while having trouble with the clarinet. This problem is caused by the shape of the spectral envelope of the clarinet, as it is dominated by a large peak at the fundamental frequency, with very weak overtones, and thus deviates from the here used model assumption of spectral smoothness. It may also be noted that PEBSI-Lite has better performance at the stationary parts of the signal, while producing more erroneous estimates at note on- and offsets due to quickly changing spectral content. The ESACF estimator on the other hand has serious problems tracking the violin and clarinet, often picking sub-octaves estimates instead of the correct pitch, although being able to track the saxophone and bassoon fairly well.

[Table 2 about here.]

7. Conclusions

The proposed algorithm PEBSI-Lite has been shown to be an accurate method for multi-pitch estimation. The method was shown to perform as good as, or better than, state-of-the-art methods. As compared to related methods, the presented algorithm requires fewer regularization parameters, simplifying the calibration of the method. Furthermore, the work introduces an adaptive dictionary scheme for determining suitable regularization parameters. Combined with this scheme, PEBSI-Lite was shown to outperform other multi-pitch estimation methods for high levels of SNR, while breaking down in too noisy settings. However, even if this scheme would fail to select the correct model order, the obtained efficient dictionary facilitates a more rigorous grid search in terms of computational complexity. Such a grid search could also exploit information about the solution surface obtained from the line search. Using an additional refinement step, the proposed algorithm is found to yield estimates reasonably close to being efficient, if considering that the method has not been allowed any knowledge of the model order of the signal.

References

[1] M. Müller, D. P. W. Ellis, A. Klapuri, G. Richard, Signal Processing for Music Analysis, IEEE J. Sel. Topics Signal Process. 5 (6) (2011) 1088–1110.

[2] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, A. Klapuri, Automatic Music Transcription: Challenges and Future Directions, Journal of Intelligent Information Systems 41 (3) (2013) 407–434.

[3] A. Wang, An Industrial Strength Audio Search Algorithm, in: 4th International Conference on Music Information Retrieval, Baltimore, Maryland USA, 2003.

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

[5] H. Fletcher, Normal vibration frequencies of stiff piano string, Journal of the Acoustical Society of America 36 (1).

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

[7] S. M. Nørholm, J. R. Jensen, M. G. Christensen, On the Influence of Inharmonicities in Model-Based Speech Enhancement, in: European Signal Processing Conference, Marrakesh, 2013.

[8] T. Nilsson, S. I. Adalbjörnsson, N. R. Butt, A. Jakobsson, Multi-Pitch Estimation of Inharmonic Signals, in: European Signal Processing Conference, Marrakech, 2013.

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

[10] M. G. Christensen, S. H. Jensen, S. V. Andersen, A. Jakobsson, Subspace-based Fundamental Frequency Estimation, in: European Signal Processing Conference, Vienna, 2004.

[11] M. G. Christensen, A. Jakobsson, S. H. Jensen, Joint High-Resolution Fundamental Frequency and Order Estimation, IEEE Trans. Audio, Speech, Lang. Process. 15 (5) (2007) 1635–1644.

[12] M. G. Christensen, P. Stoica, A. Jakobsson, S. H. Jensen, Multi-pitch estimation, Signal Processing 88 (4) (2008) 972–983.

[13] S. I. Adalbjörnsson, A. Jakobsson, M. G. Christensen, Multi-Pitch Estimation Exploiting Block Sparsity, Elsevier Signal Processing 109 (2015) 236–247.

[14] M. Genussov, I. Cohen, Multiple fundamental frequency estimation based on sparse representations in a structured dictionary, Digit. Signal Process. 23 (1) (2013) 390–400.

[15] C. Kim, W. Chang, S.-H. Oh, S.-Y. Lee, Joint Estimation of Multiple Notes and Inharmoncity Coefficient Based on f0-Triplet for Automatic Piano Transcription, IEEE Signal Processing Letters 21 (12) (2014) 1536–1540.

[16] V. Emiya, R. Badeau, B. David, Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle, IEEE Trans. Audio, Speech, Lang. Process. 18 (6) (2010) 1643–1654.

[17] K. O'Hanlon, Structured Sparsity for Automatic Music Transcription, in: 37th IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Kyoto, 2012.

[18] M. Bay, A. Ehmann, J. Beauchamp, P. Smaragdis, J. Downie, Second Fiddle is Important Too: Pitch Tracking Individual Voices in Polyphonic music, in: 13th Annual COnference of the International Speech Communication Association, Portland, 2012.

[19] A. Klapuri, Multiple fundamental frequency estimation based on harmonicity and spectral smoothness, IEEE Trans. Acoust., Speech, Signal Process. 11 (6) (2003) 804–816.

[20] P. Smaragdis, J. Brown, Non-Negative Matrix Factorization for Polyphonic Music Transcription, in: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2003, pp. 177–180.

[21] N. Bertin, R. Badeau, E. Vincent, Enforcing Harmonicity and Smoothness in Bayesian Non-Negative Matrix Factorization Applied to Polyphonic Music Transcription, IEEE Trans. Acoust., Speech, Language Process. 18 (3) (2010) 538–549.

[22] S. Karimian-Azari, A. Jakobsson, J. R. Jensen, M. G. Christensen, Multi-Pitch Estimation and Tracking using Bayesian Inference in Block Sparsity, in: 23rd European Signal Processing Conference, Nice, France, 2015.

[23] R. Gribonval, E. Bacry, Harmonic decomposition of audio signals with matching pursuit, IEEE Trans. Signal Process. 51 (1) (2003) 101–111.

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

[25] M. A. T. Figueiredo, J. M. Bioucas-Dias, Algorithms for imaging inverse problems under sparsity regularization, in: Proc. 3rd Int. Workshop on Cognitive Information Processing, 2012, pp. 1–6.

[26] T. Kronvall, M. Juhlin, S. I. Adalbjörnsson, A. Jakobsson, Sparse Chroma Estimation for Harmonic Audio, in: 40th IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Brisbane, 2015.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, Found. Trends Mach. Learn. 3 (1) (2011) 1–122.

[28] S. L. Marple, Computing the discrete-time "analytic" signal via FFT, IEEE Trans. Signal Process. 47 (9) (1999) 2600–2603.

[29] T. Kronvall, S. I. Adalbjörnsson, A. Jakobsson, Joint DOA and Multi-Pitch Estimation Using Block Sparsity, in: 39th IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Florence, 2014.

[30] T. Kronvall, S. I. Adalbjörnsson, A. Jakobsson, Joint DOA and Multi-pitch estimation via Block Sparse Dictionary Learning, in: 22nd European Signal Processing Conference (EUSIPCO), Lisbon, 2014.

[31] R. Tibshirani, Regression shrinkage and selection via the Lasso, Journal of the Royal Statistical Society B 58 (1) (1996) 267–288.

[32] E. J. Candès, J. Romberg, T. Tao, Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information, IEEE Trans. Inf. Theory 52 (2) (2006) 489–509.

[33] E. J. Candès, M. B. Wakin, S. Boyd, Enhancing Sparsity by Reweighted l_1 Minimization, Journal of Fourier Analysis and Applications 14 (5) (2008) 877–905.

[34] R. H. Tutuncu, K. C. Toh, M. J. Todd, Solving semidefinite-quadratic-linear programs using SDPT3, Mathematical Programming Ser. B 95 (2003) 189–217.

[35] P. Stoica, Y. Selén, Model-order Selection — A Review of Information Criterion Rules, IEEE Signal Process. Mag. 21 (4) (2004) 36–47.

[36] C. D. Austin, R. L. Moses, J. N. Ash, E. Ertin, On the Relation Between Sparse Reconstruction and Parameter Estimation With Model Order Selection, IEEE J. Sel. Topics Signal Process. 4 (2010) 560–570.

[37] A. Panahi, M. Viberg, Fast Candidate Point Selection in the LASSO Path, IEEE Signal Processing Letters 19 (2) (2012) 79–82.

[38] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, The Annals of Statistics 32 (2) (2004) 407–499.

[39] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, K. Knight, Sparsity and Smoothness via the Fused Lasso, Journal of the Royal Statistical Society B 67 (1) (2005) 91–108.

[40] H. Hoefling, A Path Algorithm for the Fused Lasso Signal Approximator, Journal of Computational and Graphical Statistics 19 (4) (2010) 984–1006.

[41] R. Tibshirani, J. Taylor, The Solution Path of the Generalized Lasso, The Annals of Statistics 39 (3) (2011) 1335–1371.

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

[43] T. Tolonen, M. Karjalainen, A computationally efficient multipitch analysis model, IEEE Trans. Audio, Speech, Lang. Process. 8 (6) (2000) 708–716.

[44] Z. Duan, B. Pardo, Bach10 dataset, http://music.cs.northwestern.edu/data/Bach10.html, accessed December 2015.

[45] A. de Cheveigné, H. Kawahara, YIN, a fundamental frequency estimator for speech and music, J. Acoust. Soc. Am. 111 (4) (2002) 1917–1930.

[46] M. Bay, A. Ehmann, J. Downie, Evaluation of Multiple-F0 Estimation and Tracking Systems, in: International Society for Music Information Retrieval Conference, Kobe, Japan, 2009.

List of Figures

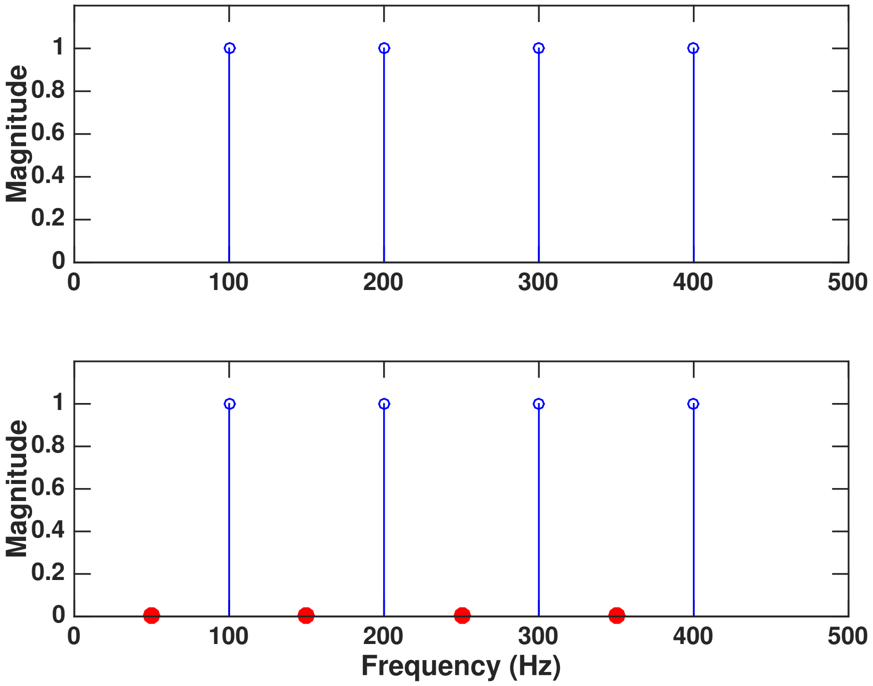

Figure 1: The upper picture depicts a pitch with fundamental frequency 100 Hz and four harmonics. The lower picture depicts a pitch with fundamental frequency 50 Hz and eight harmonics where all odd-numbered harmonics are zero (marked red dots).

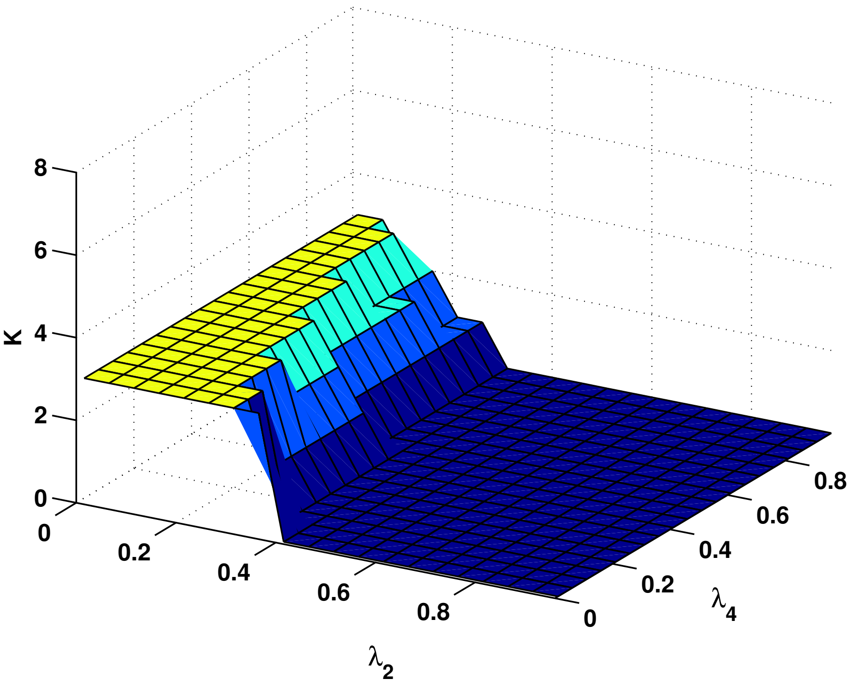

Figure 2: Number of pitches, K, present in the solution of PEBSI-Lite for different values (λ_2, λ_4) when applied to a three pitch signal with 4, 8, and 12 harmonics, respectively.

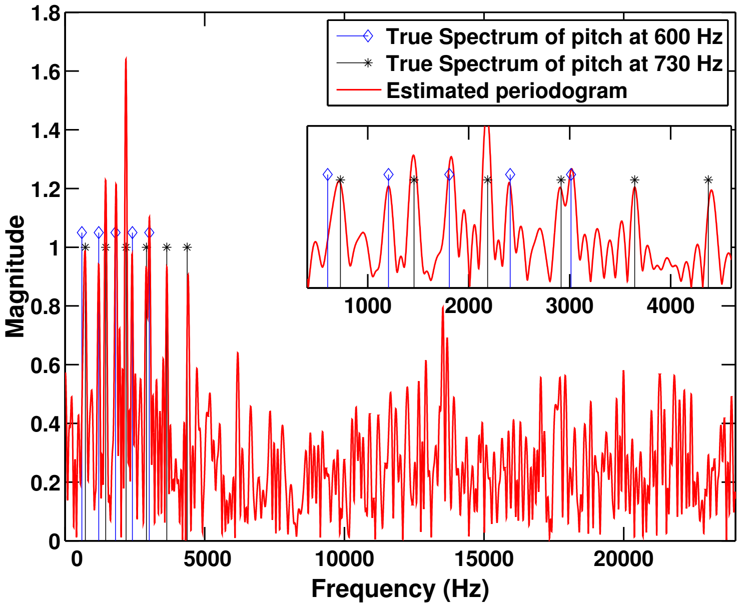

Figure 3: The periodogram estimate and the true signal studied in Figure 4.

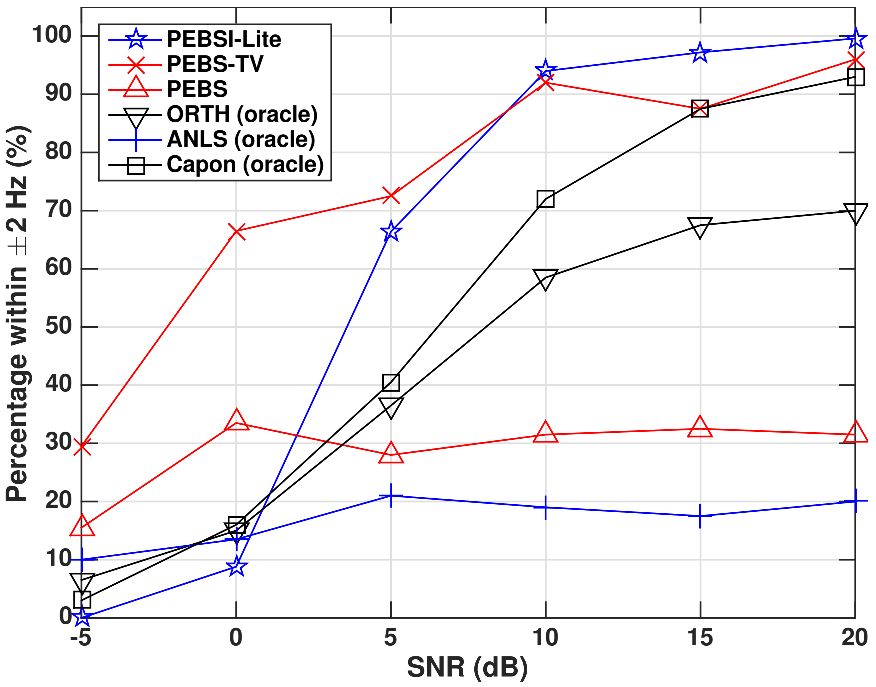

Figure 4: Percentage of estimated pitches where both fundamental frequencies lie at most 2 Hz, or $d/5 = 1/50N$, from the ground truth, plotted as a function of SNR. Here, the pitches have $[5, 6]$ harmonics, respectively, and $L_{\max} = 10$.

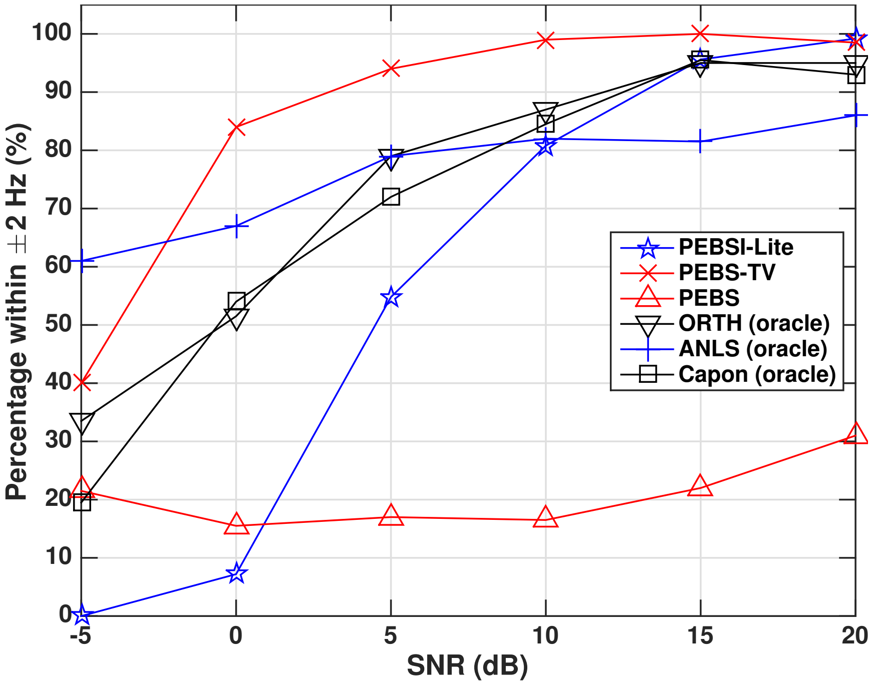

Figure 5: Percentage of estimated pitches where both fundamental frequencies lie at most 2 Hz, or $d/5 = 1/50N$, from the ground truth, plotted as a function of SNR. Here, the pitches have $[10, 11]$ harmonics, respectively, and $L_{\max} = 20$.

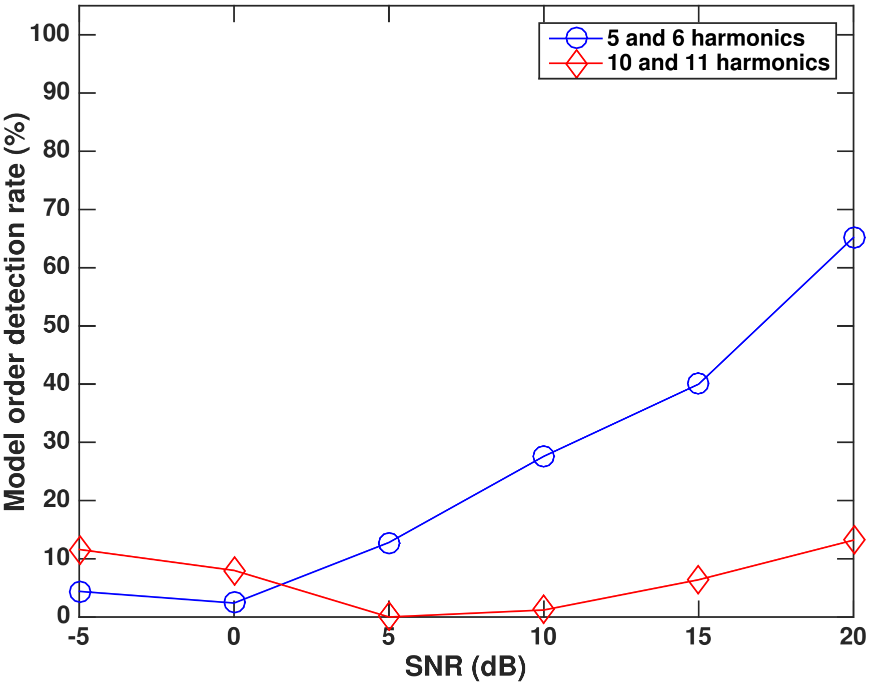

Figure 6: The percentage of the estimates in which the model order choice criterion (51) correctly determines the number of sinusoidal components in the two-pitch signal, for the case of 5 and 6 harmonics, and 10 and 11 harmonics, respectively.

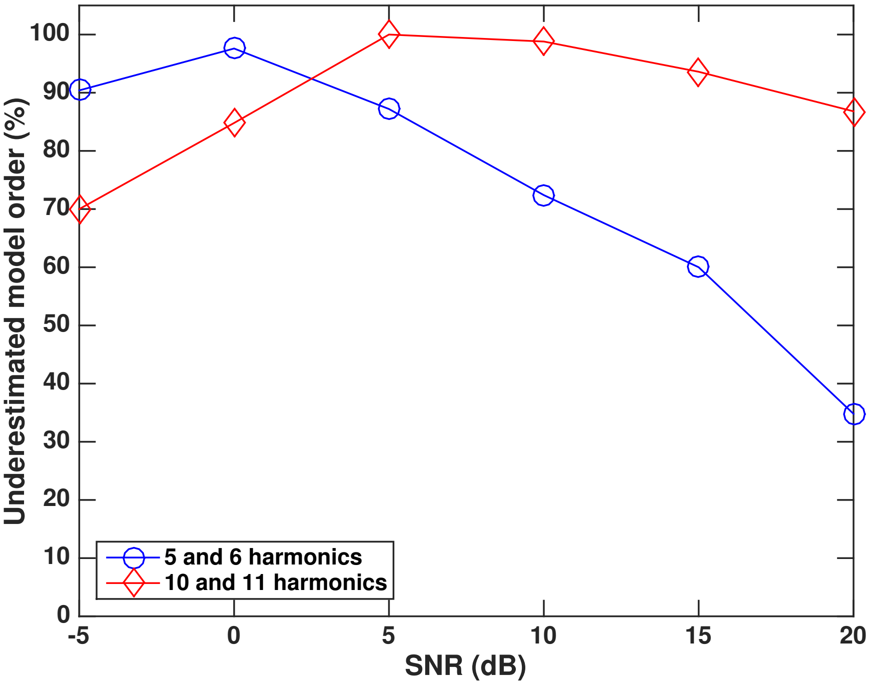

Figure 7: The percentage of the estimates in which the model order choice criterion (51) selects a model with too few sinusoidal components for the two-pitch signal, for the case of 5 and 6 harmonics, and 10 and 11 harmonics, respectively.

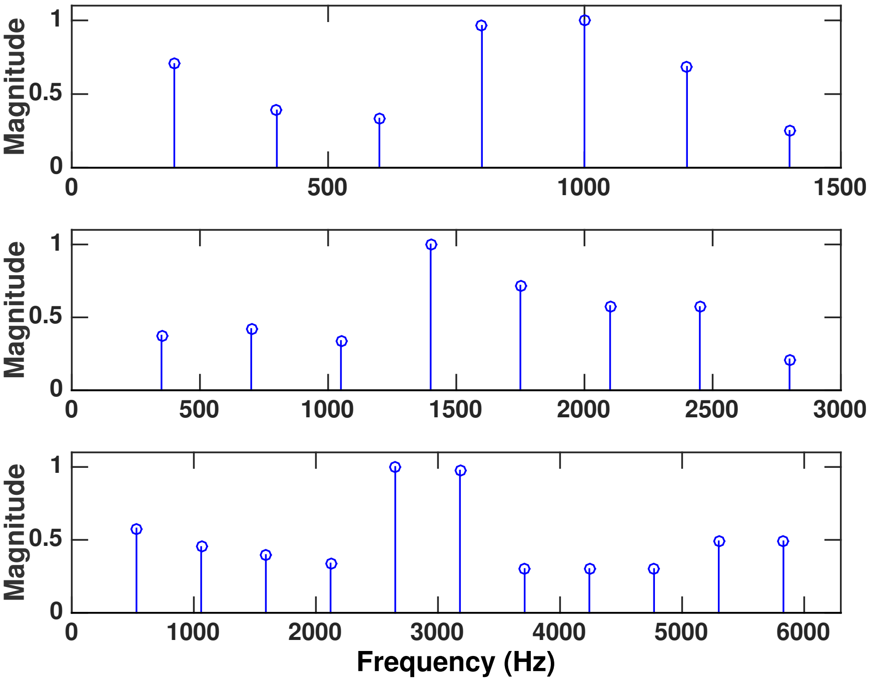

Figure 8: Magnitudes for the harmonics of the three pitches constituting the test signal for the Monte Carlo simulations.

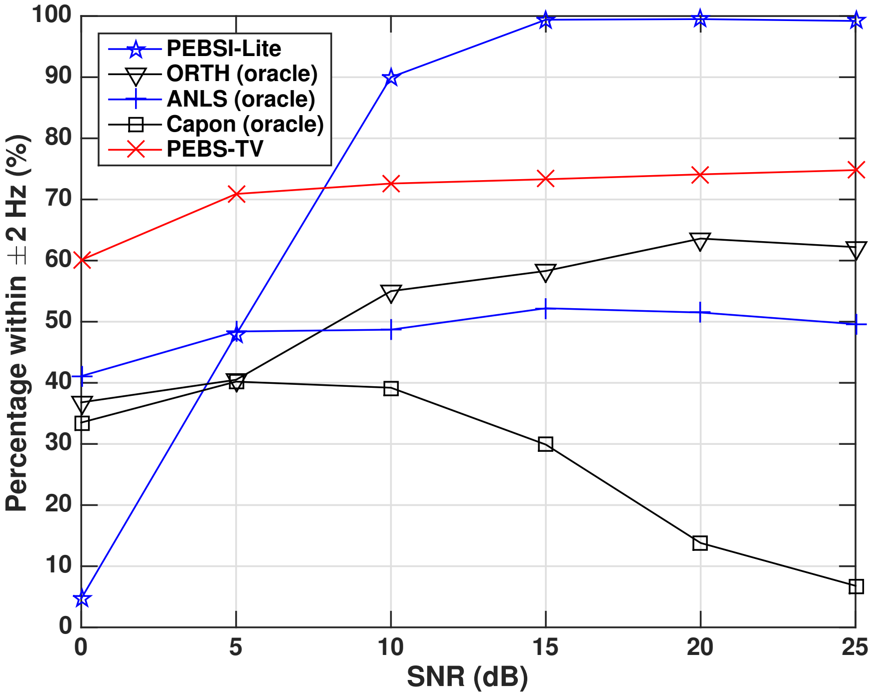

Figure 9: Percentage of estimated pitches where all three fundamental frequencies lie at most 2 Hz from the ground truth.

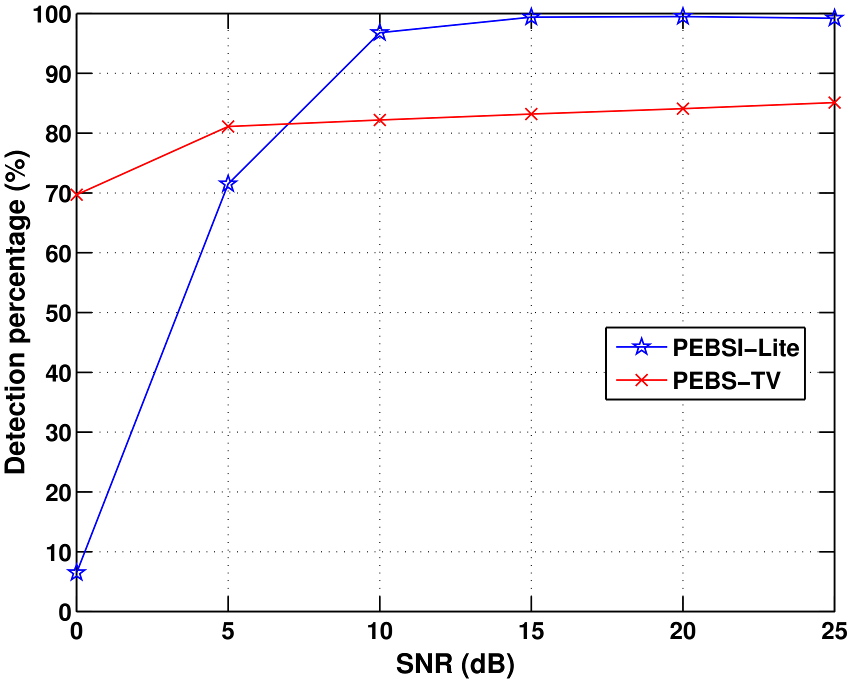

Figure 10: Estimated probability of PEBSI-Lite determining the correct number of pitches for the triple pitch test signal.

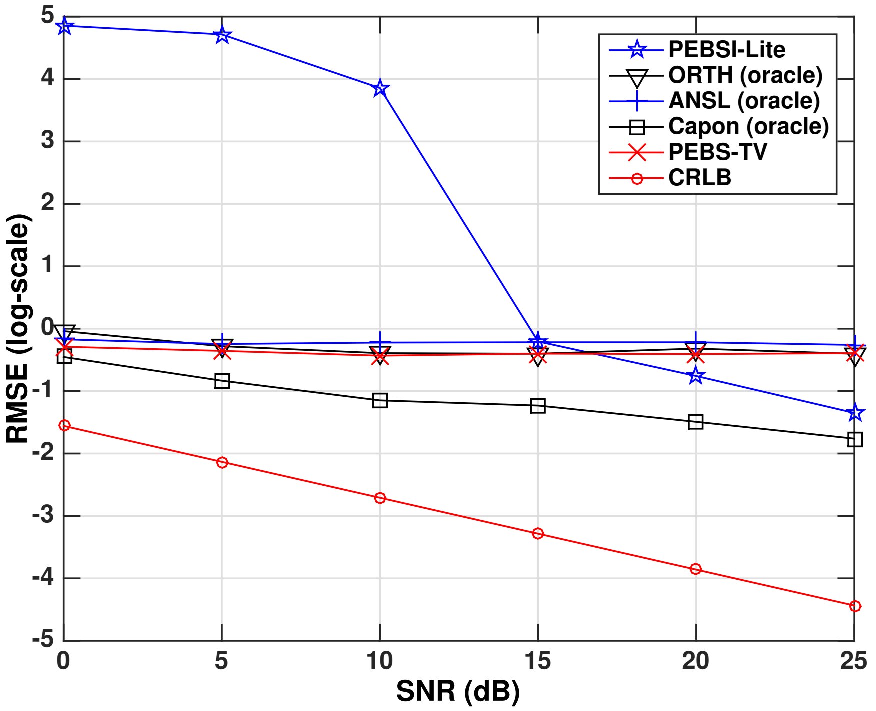

Figure 11: The RMSE for the fundamental frequency estimates for the triple pitch test signal, as compared to the (root) CRLB. For PEBSI-Lite and PEBS-TV, only estimates where the number of pitches is found are considered. For the reference methods ORTH, ANLS, Capon, and PEBS-TV only estimates where all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

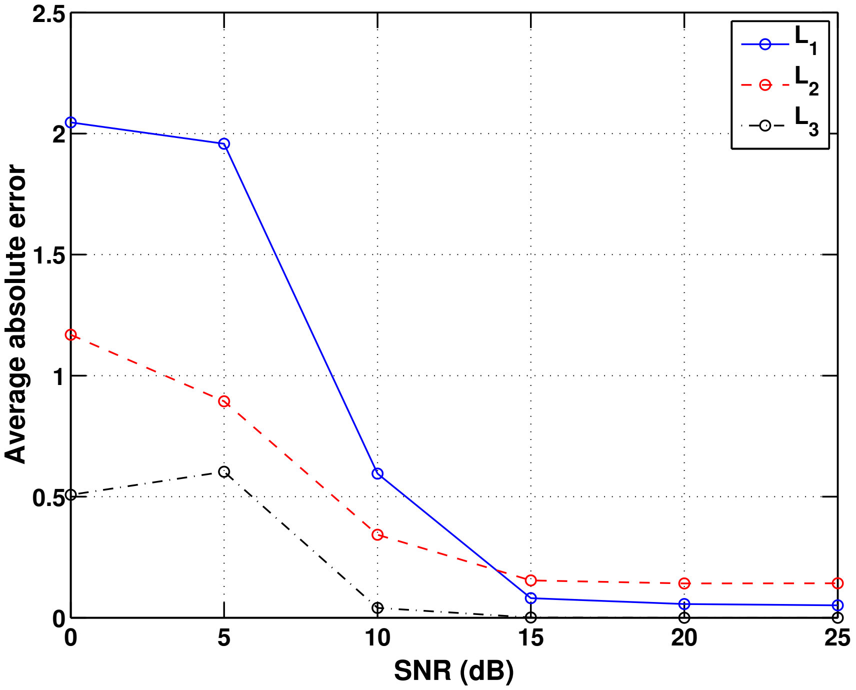

Figure 12: The average absolute error in the number of detected harmonics (L_1, L_2, L_3) for the three pitches of the test signal when using PEBSI-Lite. Only estimates where the correct number of pitches is found are considered.

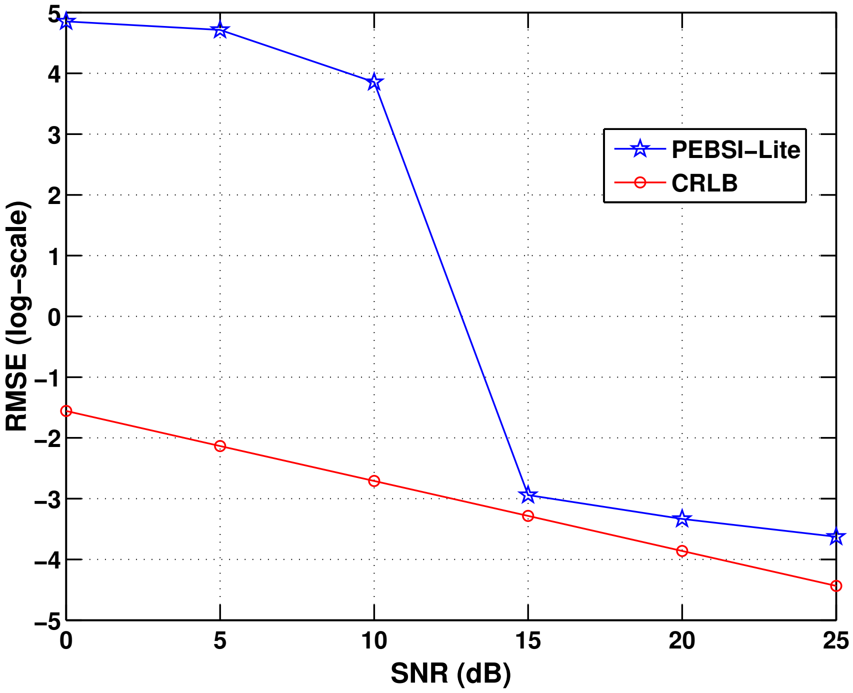

Figure 13: The RMSE for the fundamental frequency estimates where the estimates obtained using PEBSI-Lite have been improved using NLS for SNR levels 15, 20, and 25 dB, as compared to the (root) CRLB. Only estimates where the number of pitches is found are considered.

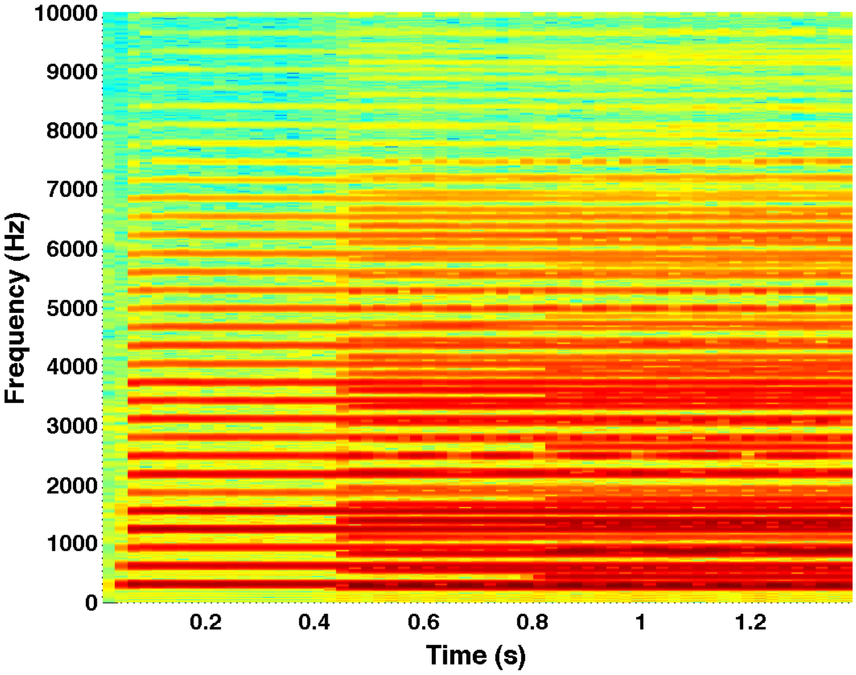

Figure 14: Spectrogram for a signal consisting of one, two and lastly three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz, respectively.

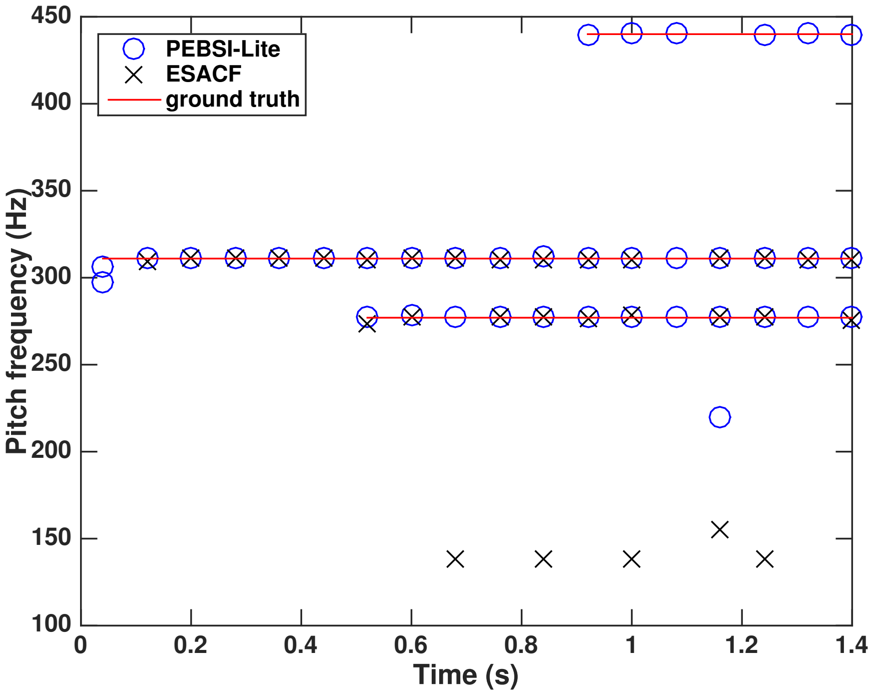

Figure 15: Pitch tracks for a signal consisting of one, two, and lastly three MIDI-saxophones playing notes with fundamental frequencies 311, 277, and 440 Hz, respectively.

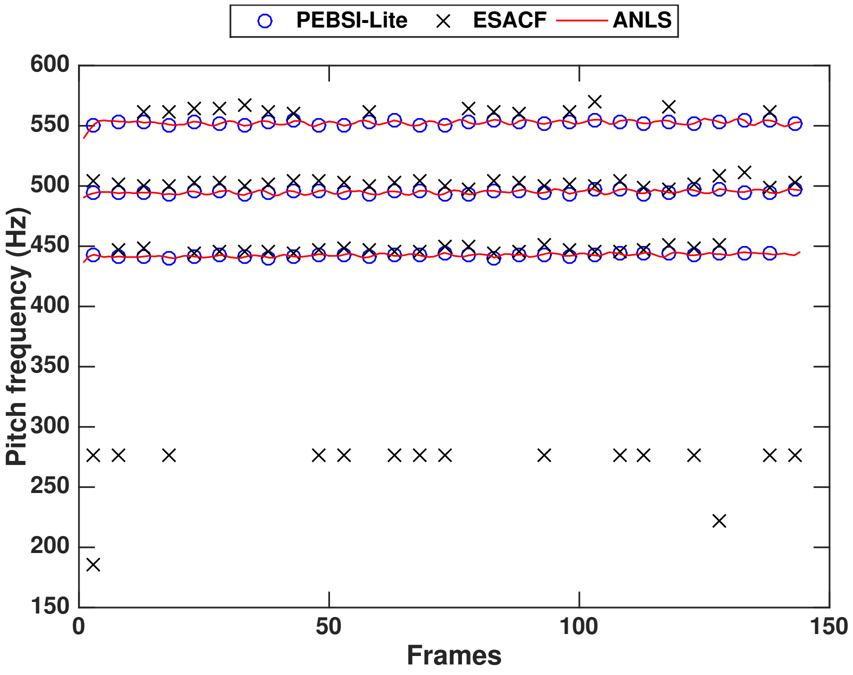

Figure 16: Pitch tracks produced by PEBSI-Lite as well as ESACF when applied to a triple-pitch signal consisting of three trumpets. The ground truth has been obtained using ANLS applied to the single source signals.

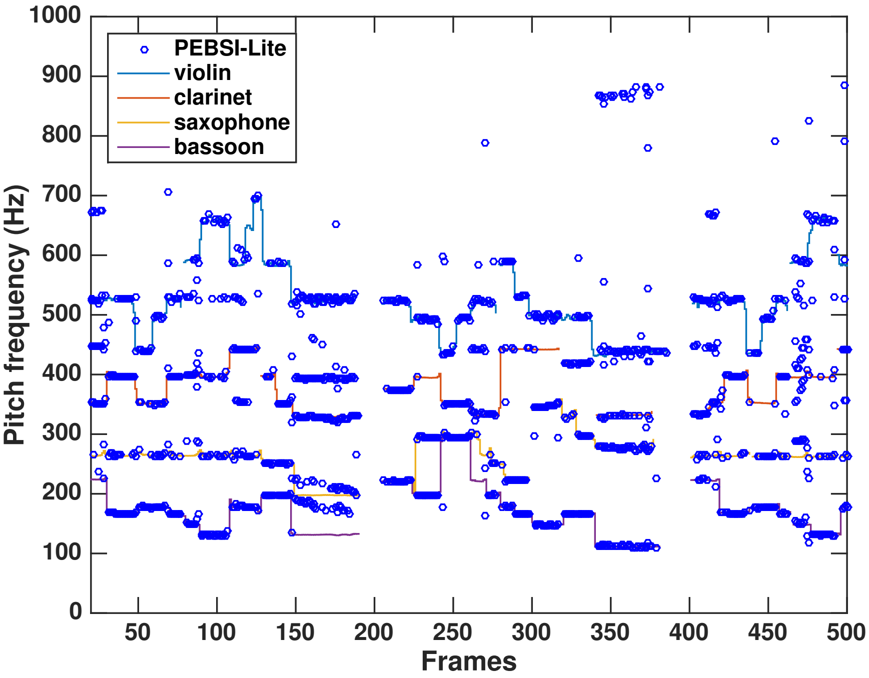

Figure 17: Pitch tracks produced by PEBSI-Lite when applied to first 15 seconds of J.S. Bach's *Ach, lieben Christen*, performed by a violin, a clarinet, a saxophone, and a bassoon. The ground truth has been obtained using YIN applied to the single source signals.

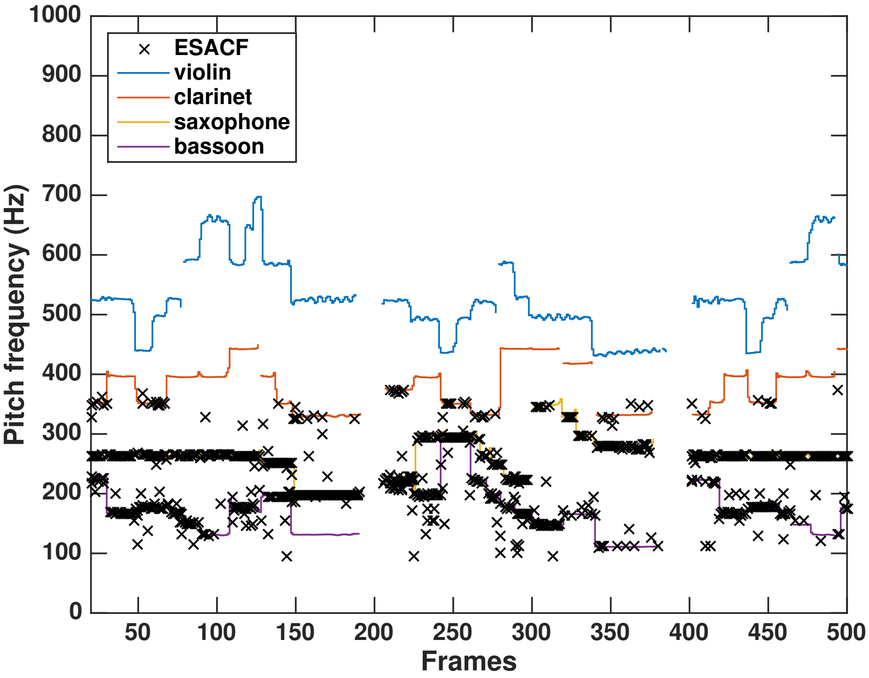

Figure 18: Pitch tracks produced by ESACF when applied to the first 15 seconds of J.S. Bach's *Ach, lieben Christen*, performed by a violin, a clarinet, a saxophone, and a bassoon. The ground truth has been obtained using YIN applied to the single source signals.

List of Tables

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

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

	PEBSI-Lite	ESACF
Accuracy	0.466	0.363
Precision	0.641	0.776
Recall	0.631	0.406

Table 2: Performance measures for PEBSI-Lite and ESACF when evaluated on the Bach10 dataset.