Open access • Proceedings Article • DOI:10.1109/ICASSP.2013.6638862

Spectral compressive sensing with polar interpolation — Source link

Karsten Fyhn, Hamid Dadkhahi, Marco F. Duarte

Institutions: Aalborg University, University of Massachusetts Amherst

Published on: 26 May 2013 - International Conference on Acoustics, Speech, and Signal Processing

Topics: Spectral density estimation, Compressed sensing and Coherence (signal processing)

Spectral Compressive Sensing with Polar Interpolation

Fyhn, Karsten; Dadkhahi, Hamid; F. Duarte, Marco

Published in:
Proceedings of the 2013 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)

Publication date:
2013

Document Version
Early version, also known as pre-print

SPECTRAL COMPRESSIVE SENSING WITH POLAR INTERPOLATION

Karsten Fyhn[*]*, Hamid Dadkhahi*[†]*, Marco F. Duarte*[†]

[*]Dept. of Electronic Systems, Aalborg University, Denmark.
[†]Dept. of Electrical and Computer Engineering, University of Massachusetts Amherst, USA.

ABSTRACT

Existing approaches to compressive sensing of frequency-sparse signals focuses on signal recovery rather than spectral estimation. Furthermore, the recovery performance is limited by the coherence of the required sparsity dictionaries and by the discretization of the frequency parameter space. In this paper, we introduce a greedy recovery algorithm that leverages a band-exclusion function and a polar interpolation function to address these two issues in spectral compressive sensing. Our algorithm is geared towards line spectral estimation from compressive measurements and outperforms most existing approaches in fidelity and tolerance to noise.

Index Terms— Compressive sensing, frequency-sparse signals, spectral estimation, polar interpolation

1. INTRODUCTION

One of the most popular thrusts in compressive sensing (CS) research has focused on the recovery of signals that are spectrally sparse (i.e., that have a sparse frequency-domain representation) from a reduced number of measurements [1–5]. Such *frequency-sparse signals* bring up a novel issue in the formulation of the CS recovery problem: frequency-domain representations have a continuous parameter space, while CS is inherently rooted on discretized signal representations.

Aiming for an increasingly dense sampling of the frequency parameter space introduces performance issues in sparsity-leveraging algorithms. In particular, increasing the resolution of the parameter sampling worsens the coherence of the dictionary that provides sparsity for relevant signals. This both prevents certain algorithms from finding the sparse representation successfully and introduces ambiguity on the choice of representations available for a signal in the dictionary. Initial contributions address such issues by modifying the sparsity prior, the recovery algorithm, or both, to be tailored to the intricacies of the signal representation [5–8].

Interestingly, CS recovery of frequency-sparse signals can be formalized in two different ways: recovery of the signal samples, and recovery of the signal's component frequencies. Previous contributions have almost exclusively focused on the former; their performance for the latter goal is limited by the representation leveraged during CS. Particularly, the required discretization of the parameter space explicitly limits the performance of compressive frequency estimation.

In this paper, we improve over existing approaches by introducing interpolation steps within CS recovery algorithms that break the discretization barrier implicit in CS and are able to improve the quality of frequency parameter estimation. While such interpolation is considered briefly and integrated to a simple recovery algorithm in [5], we introduce a novel polar interpolation approach that leverages the fact that frequency-sparse signals are translation-invariant in the frequency domain. We couple polar interpolation with a more sophisticated CS greedy recovery approach to improve the performance of spectral CS over existing algorithms. We provide experimental evidence that shows improved frequency estimation performance against approaches previously proposed for spectral CS signal recovery: in some cases, our estimates are more precise than those from the baseline approaches, while in other cases we match the precision of the baseline with greatly reduced computational complexity.

2. BACKGROUND AND RELATED WORK

Compressive sensing (CS) is a technique to simultaneously acquire and reduce the dimensionality of sparse signals in a randomized fashion. More precisely, in the CS framework, a signal $\mathbf{f} \in \mathbb{C}^N$ is sampled by M linear measurements of the form $\mathbf{y} = \mathbf{A}\mathbf{f}$, where $\mathbf{A}$ is an $M \times N$ sensing matrix and $M \ll N$. In practice, the measurements are acquired in the presence of noise $\mathbf{z}$, in which case we have $\mathbf{y} = \mathbf{A}\mathbf{f} + \mathbf{z}$.

In many applications, the signal $\mathbf{f}$ is not sparse but has a sparse representation in some dictionary $\mathbf{D}$. In other words, we have $\mathbf{f} = \mathbf{D}\mathbf{x}$, where $\mathbf{x}$ is K-sparse (i.e. $||\mathbf{x}||_0 \leq K$). Under certain conditions on the matrix $\mathbf{A}$ [9, 10], we can recover $\mathbf{x}$ from the measurements $\mathbf{y}$ through the following ℓ_1-minimization problem (which we refer to as ℓ_1-*synthesis*):

$$\hat{\mathbf{x}} = \min_{\tilde{\mathbf{x}} \in \mathbb{C}^N} ||\tilde{\mathbf{x}}||_1 \text{ s.t. } ||\mathbf{A}\mathbf{D}\tilde{\mathbf{x}} - \mathbf{y}||_2 \leq \epsilon, \qquad (1)$$

where ϵ is an upper bound on the noise level $||\mathbf{z}||_2$. Note that

E-mails: kfn@es.aau.dk, {hdadkhahi,mduarte}@ecs.umass.edu. The first two authors contributed equally to this work. This work was partially supported by The Danish Council for Strategic Research under grant number 09-067056 and by NSF under grant number ECCS-1201835.

optimal recovery of $\mathbf{x}$ from the optimization in (1) is feasible only when the elements of the dictionary $\mathbf{D}$ form an orthonormal basis, and thus are incoherent [1, 11]. However, in many applications, the signal of interest is sparse in an overcomplete dictionary or a frame, rather than in a basis.

This paper focuses on frequency-sparse signals, which can be modeled as a superposition of K complex sinusoids with arbitrary frequencies $\tilde{\boldsymbol{\omega}} = \{\omega_1, \omega_2, \ldots, \omega_K\}$. The signal $\mathbf{f} = \begin{bmatrix} f_1 & f_2 & \ldots f_N \end{bmatrix}^T$ is given by

$$f_n = \sum_{k=1}^{K} x_k e^{j2\pi\tilde{\omega}_k n},\ \tilde{\omega}_k \in [0,1],\ n \in \{1,2,\ldots,N\}. \quad (2)$$

Such signals are sparse in the discrete-time Fourier transform (DTFT), when defined using an infinite dictionary. In practice, a finite-length representation of the signal is required, and the transform of choice is the discrete Fourier transform (DFT). Unfortunately, the DFT coefficients for such a frequency-sparse signal are sparse only when the frequencies of the constituent sinusoids are integral. One way to remedy this problem would be to employ a dictionary corresponding to a finer discretization of the Fourier representation. We call such a dictionary a DFT frame of redundancy $c \in \mathbb{N}$, containing $P = c \cdot N$ elements, defined as:

$$\mathbf{D} = \begin{bmatrix} \mathbf{d}(\omega_1) & \mathbf{d}(\omega_2) & \cdots & \mathbf{d}(\omega_P) \end{bmatrix},\ \omega_p = \frac{p}{P},$$
$$\mathbf{d}(\omega_p) = \begin{bmatrix} d_1(\omega_p) & d_2(\omega_p) & \ldots & d_N(\omega_p) \end{bmatrix}^T, \quad (3)$$

where $d_n(\omega) = \frac{1}{\sqrt{N}} e^{j2\pi\omega n}$. However, the DFT frame violates the incoherence requirement for the dictionary [5].

It has recently been shown in [6] that as far as the recovery of signal $\mathbf{f}$ (instead of the sparse coefficient vector $\mathbf{x}$) is concerned, the coherence condition of the dictionary is not necessary, provided that the matrix $\mathbf{D}^H\mathbf{D}$ is sufficiently sparse, where $(\cdot)^H$ designates the Hermitian operation. In this case, the signal $\mathbf{f}$ can be recovered via ℓ_1-*analysis*. However, the matrix $\mathbf{D}^H\mathbf{D}$ is not sufficiently sparse for DFT frames.

Alternatively, one can take advantage of structured sparsity in spectral CS recovery by using a coherence inhibition model [5]. The resulting structured iterative hard thresholding (SIHT) algorithm can recover the frequency-sparse signal with a DFT frame by avoiding dictionary elements with high coherence. A variation of this method uses a band-exclusion function to achieve the same avoidance [8]. We can define the η-coherence band of the index set S as

$$B_\eta(S) = \bigcup_{k \in S} \{i \mid \mu(i,k) > \eta\},\ i \in \{1,2,\ldots,P\}, \quad (4)$$

where $\mu(i,k) = |\langle \mathbf{d}(\omega_i), \mathbf{d}(\omega_k) \rangle|$ is the coherence between two atoms in the dictionary. The authors use the band-exclusion function to avoid selecting coherent dictionary elements in various greedy algorithms, including Band-excluded Orthogonal Matching Pursuit (BOMP).

More recently, it has been shown that one can recover a frequency-sparse signal from a random subset of its samples using atomic norm minimization [7]. The atomic norm of $\mathbf{f}$ is defined as the size of the smallest scaled convex hull of a continuous dictionary of complex exponentials. Thus, the recovery procedure searches over a continuous dictionary rather than a discretized one. The atomic norm minimization can be implemented as a semidefinite program (SDP), which can be computationally expensive. In addition, this formulation does not account for measurement noise, and it is not clear if guarantees can be given for arbitrary measurement settings. Nonetheless, [7] motivates our formulation of algorithms that push past the discretization of the frequency parameter space.

3. POLAR INTERPOLATION FOR FREQUENCY ESTIMATION

One way to remedy the discretization of the frequency parameter space implicit in CS is to use interpolation. In [12], a *polar interpolation* approach for translation-invariant signals has been derived. Such signals can be written as a linear combination of shifted versions of a waveform. In a nutshell, the interpolation procedure exploits the fact that translated versions of a waveform form a manifold which lies on the surface of a hypersphere. Thus, any sufficiently small segment of the manifold can be well-approximated by an arc of a circle, and an arbitrarily-shifted waveform can be closely approximated by a point in such arc.

The complex exponentials that compose a DFT frame also form a manifold over a hypersphere, and thus can be approximated by an arc of a circle. This is motivated by the fact that complex exponentials have translation-invariant Fourier transforms, which correspond to an isometric rotation of the time-domain vectors. In this case, the DFT frame samples the frequency parameter space with a steps size $\Delta = 1/c$, and we approximate a segment of the manifold $\mathbf{d}(\tilde{\omega}_i) : \tilde{\omega}_i \in [\omega_p - \frac{\Delta}{2}, \omega_p + \frac{\Delta}{2}]$ by a circular arc containing the three exponentials $\{\mathbf{d}(\omega_p - \frac{\Delta}{2}), \mathbf{d}(\omega_p), \mathbf{d}(\omega_p + \frac{\Delta}{2})\}$. Making use of trigonometric identities, the polar interpolator approximates exponentials $\mathbf{d}(\tilde{\omega}_i)$, $\tilde{\omega}_i \in [\omega_p - \frac{\Delta}{2}, \omega_p + \frac{\Delta}{2}]$, using linear combinations of the three exponentials [12]:

$$\mathbf{d}(\tilde{\omega}_i) \approx \mathbf{c}(\omega_p) + r\cos\left(\frac{2\tilde{\omega}}{\Delta}\theta\right)\mathbf{u}(\omega_p) + r\sin\left(\frac{2\tilde{\omega}}{\Delta}\theta\right)\mathbf{v}(\omega_q),$$

$$\begin{bmatrix} \mathbf{c}(\omega_p)^T \\ \mathbf{u}(\omega_p)^T \\ \mathbf{v}(\omega_p)^T \end{bmatrix} = \begin{bmatrix} 1 & r\cos(\theta) & -r\sin(\theta) \\ 1 & r & 0 \\ 1 & r\cos(\theta) & r\sin(\theta) \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{d}(\omega_q - \frac{\Delta}{2})^T \\ \mathbf{d}(\omega_p)^T \\ \mathbf{d}(\omega_p + \frac{\Delta}{2})^T \end{bmatrix},$$

where r is the ℓ_2 norm of each element of the dictionary and θ is the angle between $\mathbf{d}(\omega_p)$ and $\mathbf{d}(\omega_p - \frac{\Delta}{2})$. In order to extend the above approximation to sums of J exponentials with frequencies $\Omega = \{\omega_1, \omega_2, \ldots, \omega_J\}$, we define:

$$\tilde{\mathbf{f}} = \mathbf{C}(\Omega)\boldsymbol{\alpha} - \mathbf{U}(\Omega)\boldsymbol{\beta} - \mathbf{V}(\Omega)\boldsymbol{\gamma}, \quad (5)$$

$$\begin{aligned}\mathbf{C}(\Omega) &= \begin{bmatrix}\mathbf{c}(\omega_1) & \mathbf{c}(\omega_2) & \cdots & \mathbf{c}(\omega_J)\end{bmatrix}, \\ \mathbf{U}(\Omega) &= \begin{bmatrix}\mathbf{u}(\omega_1) & \mathbf{u}(\omega_2) & \cdots & \mathbf{u}(\omega_J)\end{bmatrix}, \\ \mathbf{V}(\Omega) &= \begin{bmatrix}\mathbf{v}(\omega_1) & \mathbf{v}(\omega_2) & \cdots & \mathbf{v}(\omega_J)\end{bmatrix},\end{aligned} \tag{6}$$

where $\boldsymbol{\alpha}$ represents the amplitude of the signal and $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ controls the frequency translations. The three coefficient vectors can be estimated using the following constrained convex optimization problem [12]:

$$\begin{aligned}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) &= \mathrm{T}(\mathbf{y}, \mathbf{A}, \Omega) \\ &= \underset{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}}{\arg\min} \frac{1}{2\sigma^2}||\mathbf{y} - \mathbf{A}\tilde{\mathbf{f}}||_2^2 + ||\boldsymbol{\alpha}||_1 \\ &\text{s.t.} \left\{\begin{array}{c} \alpha_j \geq 0, \\ \sqrt{\beta_j^2 + \gamma_j^2} \leq \alpha_j^2 r^2, \\ \alpha_j r\cos(\theta) \leq \beta_j \leq \alpha_j r, \end{array}\right\} \text{for } j = 1, \ldots, J,\end{aligned} \tag{7}$$

where $\mathbf{A}$ is the measurement matrix, and $\mathbf{y}$ is the received compressed signal. The constraints for the optimization problem ensure that the solution consists of points on the arcs used for approximation. The first constraint ensures we have only nonnegative signal amplitudes. The second enforces the trigonometric relationship among each triplet α_j, β_j, and γ_j. The last constraint ensures that the angle between the solution and $\mathbf{d}(\omega_j)$ is restricted to the interval $[0, \theta]$. It is necessary to scale $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ after the optimization problem [12]:

$$(\beta_j, \gamma_j) \leftarrow \left(\frac{\beta_j \alpha_j r}{\sqrt{\beta_j^2 + \gamma_j^2}}, \frac{\gamma_j \alpha_j r}{\sqrt{\beta_j^2 + \gamma_j^2}}\right). \tag{8}$$

This is because the inequality of the second constraint should in fact be an equality. However, the equality would violate the convexity assumption of the optimization. After this normalization, we obtain the signal estimate from (6) and the frequency estimates using the one-to-one relation

$$\alpha_j \mathbf{c}(\omega_j) + \beta_j \mathbf{u}(\omega_j) + \gamma_j \mathbf{v}(\omega_j) = \alpha_j \mathbf{d}\left(\omega_j + \tfrac{\Delta}{2\theta}\tan^{-1}(\tfrac{\gamma_j}{\beta_j})\right). \tag{9}$$

The optimization (7), when applied with all parameter values used in the dictionary D, is named continuous basis pursuit (CBP) in [12]:

$$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \mathrm{T}(\mathbf{y}, \mathbf{A}, \Omega_{CBP}), \tag{10}$$

where $\Omega_{CBP} = \{\omega_1, \omega_2, \ldots, \omega_P\}$ is the set of all frequencies that appear in the DFT frame for our application of interest. As posed, CBP has a high computational complexity: it operates on matrices of size $3N$, whereas other CS algorithms operate on matrices of size N. However, its interpolation step has one important advantage: translation-invariance and interpolation enables CBP to reconstruct arbitrary frequency-sparse signal while requiring only a small subset of the corresponding dictionary. This makes it possible to incorporate the convex optimization solver into a greedy algorithm that quickly finds a rough estimate, which is then improved upon by a convex optimization solver.

4. BAND-EXCLUDED INTERPOLATING SUBSPACE PURSUIT

We incorporate the convex optimization (7) and band-exclusion (4) in a Subspace Pursuit algorithm [13]. We call this algorithm Band-Excluded Interpolating Subspace Pursuit (BISP), which is shown in Algorithm 1.

In the algorithm initialization, the best K correlating atoms are found and stored in S^n by generating a proxy for the sparse signal. The K atoms are found iteratively, which deviates from the original Subspace Pursuit algorithm where the K atoms are found in one step. In each iteration, we trim the proxy based on the found atom and the band exclusion function $B_\eta(S)$, as defined in (4). In the main loop, we find the K best atom indices and add them to S^n. From S^n, we form a set Ω consisting of all frequencies corresponding to the indices in S^n along with all adjacent indices. This is necessary because the frequencies present in $\mathbf{y}$ may not be sufficiently incoherent and may therefore skew the peaks of the proxy estimate. Therefore, as a precaution, we include the closest neighbors on each side. The set Ω is input to the convex optimization in (7) along with the measurement matrix and the received signal.

In practice, we found that for noisy measurements it is often preferable to move the minimization objective $||\mathbf{y} - \mathbf{A}\tilde{\mathbf{f}}||_2^2$ in (7) into a constraint. Moving this fidelity measure from the objective function to a constraint causes the optimization to return the sparsest set of coefficients that yields measurements within the noise range of the observation. If the output is non-existent or trivial, we move the fidelity metric from the objective function to the constraint (or vice versa).

Algorithm 1 BISP

INPUTS: Compressed signal $\mathbf{y}$, sparsity K, measurement matrix $\mathbf{A}$ and spacing between dictionary elements Δ.
OUTPUTS: Reconstructed signal $\tilde{\mathbf{f}}$ and frequency estimates $\tilde{\boldsymbol{\omega}}$.
INITIALIZE: $\boldsymbol{\Phi} = \mathbf{AD}$, $i = 1, S^0 = \emptyset$
while $i \leq K$ **do**
  $S^0 = S^0 \cup \arg\max_i |\langle \mathbf{y}, \mathbf{A}_i \rangle|,\ i \notin B_0(S^0),\ i = i + 1$
end while
$\mathbf{y}_r^0 = \mathbf{y} - \boldsymbol{\Phi}_{S^0}\boldsymbol{\Phi}_{S^0}^\dagger \mathbf{y},\ n = 1$
LOOP:
repeat
  $i = 1, S^n = S^{n-1}$
  while $i \leq K$ **do**
    $S^n = S^n \cup \arg\max_i |\langle \mathbf{y}, \mathbf{A}_i \rangle|, i \notin B_0(S^n),\ i = i+1$
  end while
  $\mathbf{a} = (\boldsymbol{\Phi}_{S^n})^\dagger \mathbf{y}$
  $S^n = \mathrm{supp}(\mathrm{thresh}(\mathbf{a}, K))$
  $\Omega = \cup\{\Delta(s-1), \Delta s, \Delta(s+1) | s \in S^n\}$
  From $\mathrm{T}(\mathbf{y}, \mathbf{A}, \Omega)$ obtain $\tilde{\mathbf{f}}$ and $\tilde{\boldsymbol{\omega}}$ using (9) and (6)
  $\mathbf{y}_r^n = \mathbf{y} - \mathbf{A}\tilde{\mathbf{f}},\ n = n + 1$
until $||\mathbf{y}_r^n||_2 < \epsilon \cdot ||\mathbf{y}_r^{n-1}||_2 \vee n \leq K$

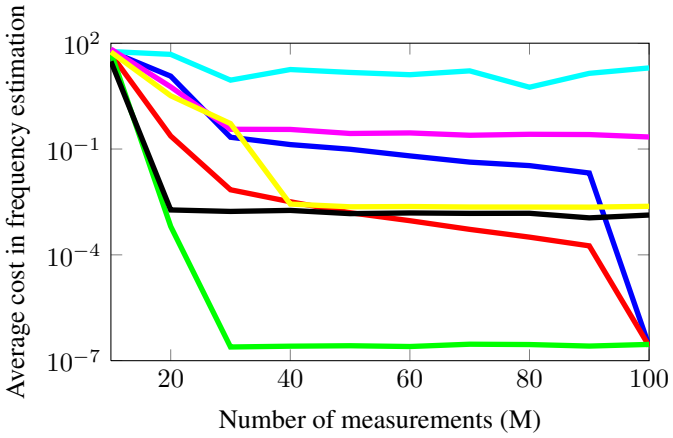

Fig. 1. Frequency estimation performance in noise-less case. The legend is shown in Fig. 2.

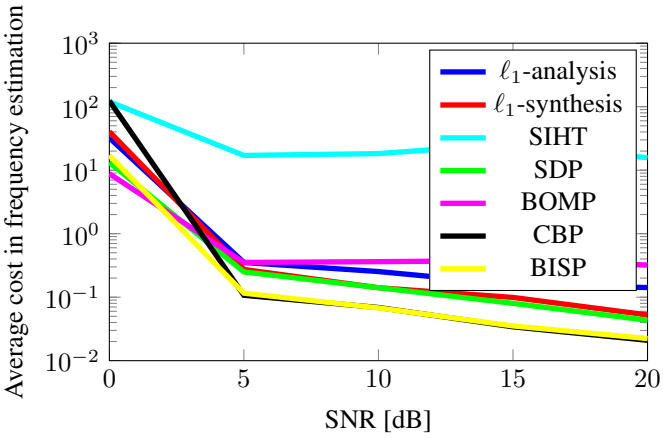

Fig. 2. Frequency estimation performance in noisy case.

	Noiseless	Noisy
ℓ_1-analysis	9.5245	8.8222
ℓ_1-synthesis	2.9082	2.7340
SIHT	0.2628	0.1499
SDP	8.2355	9.9796
BOMP	0.0141	0.0101
CBP	46.9645	40.3477
BISP	5.4265	1.4060

Table 1. Average computation times in seconds.

5. NUMERICAL EXPERIMENTS

To evaluate Algorithm 1, we have performed two numerical experiments.[1] We generated frequency-sparse signals of length $N = 100$ containing $K = 4$ complex sinusoids with frequencies selected uniformly at random. We used a DFT frame with $c = 5$ ($\Delta = 0.2$Hz), and considered well-separated tones so that no two tones are closer than 1Hz of each other. We performed Monte Carlo experiments and averaged over 30 experiments. As measurement matrix[2] we used a Gaussian matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$. We set $M = \kappa N$, where $\kappa \in (0, 1]$ is the CS subsampling rate. We compare our proposed Algorithm 1 with six state-of-the-art methods: ℓ_1-synthesis, ℓ_1-analysis, SIHT, SDP, BOMP, and CBP. As performance measure, we use the Hungarian algorithm [15, 16] to find the best matching between the estimated and true frequencies. For the algorithms that return a dense DFT coefficient vector or a reconstructed signal (ℓ_1-synthesis, ℓ_1-analysis, SIHT, and SDP), we apply the MUSIC algorithm [17] on the reconstructed signal to estimate its frequencies. In the BISP and BOMP algorithms, we exclude atoms with coherence $\eta > 0.25$ using (4).

For the first experiment, we explore a range of subsampling ratios κ with noiseless measurements to verify the level of compression that allows for successful estimation. We set $\epsilon = 10^{-10}$ for the relevant algorithms. The result of the numerical experiment is shown in Figure 1. In the noiseless case, SDP obtains the best result. The polar interpolation algorithms (CBP and BISP) both converge to a given estimation precision, which corresponds to the level of approximation error. When the number of measurements M is sufficiently small, CBP outperforms ℓ_1-synthesis. The performance of BOMP and SIHT is worst among the algorithms tested. Surprisingly, while the DFT coefficients $\mathbf{x}$ found by ℓ_1-synthesis are not sparse and do not match the original frequencies, the signal $\mathbf{f}$ is still reconstructed accurately, and so the MUSIC algorithm recovers the frequencies adequately.

For the second experiment, we include measurement noise in the signal model. We fix $\kappa = 0.5$ and vary the signal-to-noise ratio (SNR) from 0 to 20 dB. In the noisy case, the polar interpolation algorithms perform best. This is because their interpolation step relies less on the sparsity of the signal and more on the known signal model and the fitting to a circle on the manifold. Additionally, the presence of noise renders the measurements non-sparse in the dictionaries used by the non-interpolating algorithms, hindering their performance.

The computation time of the algorithms is also of importance, and we have listed the average computation times in Table 1. We observed that most algorithms exhibit computation time roughly independent of M, with the exception of ℓ_1-synthesis and CBP[3]. The table shows that the excellent performance of SDP in Figure 1 is tempered by its high computational complexity, as well as its lack of flexibility on the measurement scheme. Moreover, the relaxation in BISP that accounts for the presence of noise reduces its computation time, increasing its performance advantage over SDP and CBP.

[1] The documentation and code for these experiments are made freely available at `http://www.sparsesampling.com/scspi`, following the principle of Reproducible Research [14].

[2] For the SDP algorithm we used a random subsampling matrix, as the algorithm is only defined for such a measurement matrix. The authors would like to thank Gongguo Tang for providing the implementation of SDP.

[3] See results at `http://www.sparsesampling.com/scspi`.

6. REFERENCES

[1] E. J. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, June 2007.

[2] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.

[3] M. Mishali and Y. C. Eldar, "From theory to practice: Sub-Nyquist sampling of sparse wideband analog signals," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 375–391, Apr. 2010.

[4] M. Wakin, S. Becker, E. Nakamura, M. Grant, E. Sovero, D. Ching, J. Yoo, J. Romberg, A. Emami-Neyestanak, and E. Candès, "A nonuniform sampler for wideband spectrally-sparse environments," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 3, pp. 516–529, Sep. 2012.

[5] M. F. Duarte and R. G. Baraniuk, "Spectral compressive sensing," *Applied and Computational Harmonic Analysis*, 2012, To appear.

[6] E. J. Candès, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59 – 73, 2011.

[7] G. Tang, B. Bhaskar, P. Shah, and B. Recht, "Compressed sensing off the grid," *Preprint (available at: http://arxiv.org/abs/1207.6053)*, 2012.

[8] A. Fannjiang and W. Liao, "Coherence pattern-guided compressive sensing with unresolved grids," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 179–202, Feb. 2012.

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

[10] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, 2005.

[11] H. Rauhut, K. Schnass, and P. Vandergheynst, "Compressed sensing and redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 54, no. 9, pp. 2210–2219, Sep. 2008.

[12] C. Ekanadham, D. Tranchina, and E. P. Simoncelli, "Recovery of sparse translation-invariant signals with continuous basis pursuit," *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4735–4744, Oct. 2011.

[13] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.

[14] P. Vandewalle, J. Kovacevic, and M. Vetterli, "Reproducible research in signal processing – What, why, and how," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 37–47, May 2009.

[15] H. W. Kuhn and Bryn Yaw, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, pp. 83–97, 1955.

[16] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[17] P. Stoica and R. L. Moses, *Introduction to spectral analysis*, Prentice Hall, Upper Saddle River, NJ, 1997.