A Novel Method of DOA Tracking by Penalized Least Squares

Ashkan Panahi, Mats Viberg
Signal Processing Group, Chalmers University, Sweden
Email:{*ashkanp, viberg*}@*chalmers.se*

***Abstract*—This work develops a new DOA tracking technique by proposing a novel semi-parametric method of sequential sparse recovery for a dynamic sparsity model. The proposed method iteratively provides a sequence of spatial spectrum estimates. The final process of estimating direction paths from the spectrum sequence is not considered. However, the simulation results show concentration of the spectrum around the true directions, which simplifies DOA tracking, for example, using a pattern recognition approach. We have also proved analytical results indicating consistency in terms of spectral concentration, which we omit in the interest of space and postpone to a more extensive work. The semi-parametric nature of the proposed method avoids highly complex data association and makes the method robust against crossing. The computational complexity per time sample is proportional to grid size , which can be contrasted to a single-snapshot LASSO solution that has a polynomial complexity order.**

I. Introduction

The method of penalized least squares has offered new possibilities to analyze of partially linear models. It provides a robust implementation an has interesting theoretical properties as a statistical shrinkage method. In particular, it can be used to provide sparse solutions, which is suitable for the problem of Direction Of Arrival (DOA) estimation. A very well known example is the Least Absolute Shrinkage and Selection Operator (LASSO) [1]. This work aims to generalize penalized least squares for sequentially tracking a set of temporally changing DOAs.

DOA estimation can be viewed as both a parametric estimation and a non-parametric selection. While the parametric methods provide more accurate and sophisticated machinery, they suffer the intrinsic permutation ambiguity. On the other hand, the penalized least squares are related to parameterizing the essentially non-parametric model of DOA estimation by a huge (possibly infinite), but sparse vector of parameters. Presented in this semi-parametric framework, penalized least squares enjoy the superior properties of a parametric design, avoiding to associate parameters with corresponding DOA values.

The permutation ambiguity is not restrictive in a static estimation scenario. However, in the general dynamic case of interest, it results in a data association problem, which needs an extra process reducing the total performance, especially when the individual tracks cross, appear or vanish. This so called DOA tracking problem has been concerned by many previous works, providing a rich literature referring to various model-based tracking techniques. Among these attempts, one may find sequential Bayesian ideas, e.g. [2], [3], as well as subspace tracking techniques. The subspace methods, such as [4] are low-computational with a guaranteed performance for slow-varying models with an uncorrelated noise. However, they are sensitive to the noise model mismatch. Furthermore, it is not straightforward to adapt them for different time evolution models. Unlike subspace based techniques, the recursive Bayesian filter may be simply designed for different evolution models. Still, recovering a set of tracks with a dynamic order, caused by crossover, creation or annihilation is considered as a difficult problem for such approaches. In this context, the semi-parametric method of penalized least square is appealing as it does not concern permutation and can be easily designed by different time models. Trained from the past observations, a suitable semi-parametric tracking filter provides a spatio-temporal spectrum which leads to precise track estimates by methods such as [5] without considering data association.

The above idea is also addressed in sequential Compressed Sensing (CS) and sparsity recovery literatures . Many of these works such as [6] concern a static sparse vector measured dynamically or like [7] assume a static support. However the more general setup is addressed, e.g. in [8]. However, it may not be relevant to the current application as first it assumes a strong temporal relation in waveform and second it still need a heavy computational process in each iteration. The work in [9] is also conceptually similar to the current approach, where only a small percentage of the support is assumed to vary slowly at each time. Hence, it may still not meet the current qualifications, where a very small support evolves totally and rapidly. Finally, unlike [8], [9] and many other CS-based attempts we do not directly impose sparsity at each iteration, but promote a strongly concentrated spectrum, from which estimation becomes simpler.

A. *How to Track?*

The tracking problem is connected with a functional estimation, tractable only when performed sequentially. Such a recursive design in various tracking methods share a common recipe; Start from sequential estimation through a static model instead. Then, add uncertainties at each iteration accounting for the parameter dynamics. For example, the prediction step in sequential Bayesian estimators (like Kalman filter) dissipates the posterior probability density. This is clearer noting that simplified by removing prediction step, the sequential Bayesian filter boils down to dynamically implementing the

978-1-4673-3146-3/13/$31.00 ©2013IEEE

Maximum Likelihood (ML) estimator of static parameters. Similarly, the well know Projection Approximation Subspace Tracking (PAST) [4] results by successively perturbing an interesting optimization equivalent to signal and noise subspace splitting.

The current study offers the same type of design by sequentially converging to the solution of the sum-RMS penalized least square (G-LASSO) in Section III, which is shown to be consistent in many practical situations with a proper choice of parameters [10]. This gives a non-parametric estimator of the statiic sources similar to [11]. Later in Section IV we modify each iteration to admit dynamic parameters.

II. Mathematical Modeling

Consider the following model:

$$\mathbf{x}(t)=\sum_{k=1}^{n}\mathbf{a}(\theta_k(t))s_k(t)+\mathbf{n}(t) \tag{1}$$

where $\mathbf{x}(t)\in\mathbb{C}^m$ is the known observed data vector at discrete time $t=1,2,\dots$ and $\mathbf{a}(\theta):[0\ 2\pi]\rightarrow\mathbb{C}^m$ is the array manifold. Moreover, $\mathbf{s}(t)=[s_1(t)\ s_2(t)\dots s_n(t)]^T$ and $\mathbf{n}(t)\in\mathbb{C}^m$ are the unknown source (or waveform) and measurement noise vectors, respectively. The measurement noise is assumed to be centered, circularly symmetric, Gaussian complex-valued vector process, while it is difficult to assume a satisfactory statistical model for waveforms. Hence, we treat them deterministically. The question is to find the track vector $\boldsymbol{\theta}(t)=[\theta_1(t)\ \theta_2(t)\dots\ \theta_n(t)]^T$ from the above and some further assumptions on the nature of them. Although ineffective in derivations, we have in mind the following random walk DOA $\theta_k(t)$ evolution model:

$$\theta_{k+1}(t)=\theta_k(t)+\epsilon_k(t) \tag{2}$$

where $\boldsymbol{\epsilon}(t)=[\epsilon_1(t)\ \epsilon_2(t)\dots\epsilon_n(t)]^T$ is a white uncorrelated zero-mean Gaussian process. In this case, the Maximum A-Posteriori (MAP) estimator for first T snapshots is given by

$$\min\frac{1}{2}\sum_{t=1}^{T}\|\mathbf{x}(t)-\mathbf{A}(\boldsymbol{\theta}(t))\mathbf{s}(t)\|_2^2+\beta\sum_{t=2}^{T}\|\boldsymbol{\theta}(t)-\boldsymbol{\theta}(t-1)\|_2^2 \tag{3}$$

where the minimum is taken over $(\{\theta_k(t)\},\{\mathbf{s}(t)\})$, we define $\mathbf{A}(\boldsymbol{\theta})=[\mathbf{a}(\theta_1)\ \mathbf{a}(\theta_2)\dots\ \mathbf{a}(\theta_n)]$ and β is a positive fraction of the noise and the process variances, controlling the relative importance of the second term.

When DOAs are stationary, i.e. $\boldsymbol{\epsilon}(t)=\mathbf{0}$, the DOA track vector expression reduces to $\theta_k(t)=\theta_k$ where θ_k denotes the k^{th} stationary DOA. Then, $\beta=0$ in (3). However, it is still difficult to solve it directly. In this case it is proposed to introduce a large set $\boldsymbol{\theta}_G=[\theta_1^g,\theta_2^g,\dots,\theta_N^g,]$ of N candidate DOAs and confine search over its elements, so that DOA estimation becomes selecting a subset $I\subset\boldsymbol{\theta}_G$. It is also known that the selection procedure can be accomplished by taking the following so-called G-LASSO convex optimization [1]:

$$\min_{\{\mathbf{s}^g(t)\}}\frac{1}{2}\sum_{t=1}^{T}\|\mathbf{x}(t)-\mathbf{A}_G\mathbf{s}^g(t)\|_2^2+\lambda\sum_{k=1}^{N}\sqrt{\sum_{t=1}^{T}|s_k(t)|^2} \tag{4}$$

The matrix $\mathbf{A}_G$ is equal to $\mathbf{A}(\boldsymbol{\theta}_G)=[\mathbf{a}(\theta_1^g)\ \mathbf{a}(\theta_2^g)\dots\ \mathbf{a}(\theta_N^g)]$ and the design parameter λ controls the number of nonzero elements in the optimal solution of G-LASSO. To emphasize on its dependence on the window size T, we may index λ as λ_T. The vector $\mathbf{s}^g$ also denotes the generalized source vector corresponding to the elements of $\boldsymbol{\theta}_G$. The DOA estimates then correspond to the nonzero entries of the optimal point. It is possible to show that under some mild conditions [10], the support of G-LASSO solution well-approximate the true directions when $\lambda_T/\sqrt{T}$ tends to a proper finite value. Hence, for simplicity and without loss of generality we assume that $\lambda_T=\lambda\sqrt{T}$ where λ is a proper design value. Moreover, despite the unbounded growth of the G-LASSO dimension, it can be performed by a bounded complexity as its solution only depends on the covariance $\mathbf{R}=\mathcal{E}(\mathbf{x}(t)\mathbf{x}^T(t))$.

III. Stochastic G-LASSO Program

In this part, we propose a stochastic recursion, whose output converges to the solution of G-LASSO (4) when $T\rightarrow\infty$. We later generalize this idea to include parameter dynamics. We simply write the optimality condition of G-LASSO and suitably alter it to obtain a recursion. We may then prove its desired convergence. As shown in [12], the optimal solution of (4) satisfies

$$\mathbf{A}^H\mathbf{z}(t)=\lambda_T\mathbf{P}_T^{-1}\mathbf{s}(t) \tag{5a}$$

$$\forall\theta\quad\sqrt{\sum_{t=1}^{T}|\mathbf{a}^H(\theta)\mathbf{z}(t)|^2}\leq\lambda_T \tag{5b}$$

where $\mathbf{A}=\mathbf{A}(I)$ denotes the collection matrix of the steering vectors corresponding to the support of the optimal solution and $\mathbf{s}(t)$ indicate the the values of its corresponding nonzero elements. In other words, assuming $I=\{\theta_{i_1}^g\ \theta_{i_2}^g\dots\theta_{i_n}^g\}$, we have that $\mathbf{s}(t)=[s_{i_1}^g(t)\ s_{i_2}^g(t)\dots s_{i_n}^g(t)]$ where $s_k^g(t)$ denotes the k^{th} element of the optimal point. Naturally, $s_k^g(t)=0$ if $\theta_k^g\notin I$. Moreover, the vector $\mathbf{z}(t)=\mathbf{x}(t)-\mathbf{A}\mathbf{s}(t)$ is the residual vector and $\mathbf{P}$ denotes a diagonal matrix whose k^{th} element equals to the RMS value p_{i_k} where

$$p_k(T)=\sqrt{\sum_{t=1}^{T}|s_k^g(t)|^2}. \tag{6}$$

From (5a) and after straightforward manipulations we conclude that

$$\mathbf{s}(t)=\mathbf{P}_T\mathbf{A}^H(\mathbf{A}\mathbf{P}_T\mathbf{A}^H+\lambda_T\mathbf{I})^{-1}\mathbf{x}(t). \tag{7}$$

Let us introduce $\sigma_k(T)=p_k(T)/\sqrt{T}$ and $\boldsymbol{\Sigma}_T$ as the $N\times N$ diagonal matrix, whose k^{th} element equals $\sigma_k(T)$. Then (7) equivalently leads to

$$\mathbf{s}^g(t)=\boldsymbol{\Sigma}_T\mathbf{A}_G^H(\mathbf{A}_G\boldsymbol{\Sigma}_T\mathbf{A}_G^H+\lambda\mathbf{I})^{-1}\mathbf{x}(t). \tag{8}$$

Note that (8) does not solve for $\mathbf{s}(t)$ as the term $\boldsymbol{\Sigma}_T$ depends on $\mathbf{s}(t)$. Moreover, unlike (7) the expression in (8) includes all entries of $\mathbf{s}^g$ although it leads to a trivial equality for elements outside I.

The relation in (8), where the value of $\sigma_k(T)$ depends on future values of $s_k(t)$ cannot be directly used in a sequential procedure. Hence, we propose substituting $\sigma_k(T)$ by an approximation $\hat{\sigma}_k(t)$ from the previous estimates $\mathbf{s}^g(t')$ for $t' < t$. Then,

$$\mathbf{s}^g(t) = \hat{\mathbf{\Sigma}}_t \mathbf{A}_G^H (\mathbf{A}_G \hat{\mathbf{\Sigma}}_t \mathbf{A}_G^H + \lambda \mathbf{I})^{-1} \mathbf{x}(t) \tag{9}$$

where $\hat{\mathbf{\Sigma}}_t$ denotes a diagonal matrix with $\hat{\sigma}_k^g(t)$ as its k^{th} element. This may indeed include different types of dynamical models as well. We discuss the dynamical case in more details in the next section. Let us take a simple example in the static case, where $\hat{\sigma}_k(t) = p_k(t-1)/\sqrt{t-1}$. This can be recursively implemented by

$$\hat{\sigma}_k^2(t+1) = \frac{t\hat{\sigma}_k^2(t) + |s_k^g(t)|^2}{t+1}. \tag{10}$$

Neglecting the technical details, it can be seen that the iterations in (10) and (9) may not provide satisfactory properties. It only converges for high enough values of λ which leads to under estimated order. This is mainly because the updating step in (10) shrinks faster than the uncertainty at each iteration so that the vector $\sigma_k(t)$ diverges with $O(\log t)$. This may be fixed by considering the following more general form of (10),

$$\hat{\sigma}_k^2(t+1) = (1-\alpha(t))\hat{\sigma}_k^2(t) + \alpha(t)|s_k^g(t)|^2 \tag{11}$$

with a proper choice of $\alpha(t)$. However, as our main goal is to design a dynamical filter, we postpone the current discussion on a consistent choice of $\alpha(t)$ to a future work.

IV. A Spectrum Tracking Method

To design a tracker recursion, we exploit the framework obtained in the previous section; Fixing a sequence of step lengths $\alpha(t)$ and iterating by (11) and (9). Obviously, the value of $\alpha(t)$ defines how much the current value of $\hat{\sigma}_k(t)$ depends on the previous observations. Hence, a lower value of $\alpha(t)$ corresponds to a more dynamical model. Incidentally, a static model may be obtained by letting $\alpha(t) \to 1$ with a proper convergence speed. In contrast, when $\alpha(t)$ is fixed to some value α, the iterative approach continuously forgets the effect of relatively old samples, which best suits a stationary dynamical model. In this case, it is not difficult to see that

$$\hat{\sigma}_k^2(t) = \alpha \sum_{t'<t} (1-\alpha)^{t-t'} |\hat{s}_k(t')|^2 \tag{12}$$

and that this is equivalent to the more familiar recursion of

$$\hat{\sigma}_k^2(t+1) = (1-\alpha)\hat{\sigma}_k^2(t) + |s_k^g(t)|^2, \tag{13}$$

without any multiplier in the second summand, together with (9) when λ is replaced by $\lambda/\sqrt{\alpha}$. As both λ and α are parameters of design, we may use (13) and (9) to obtain another valid recursive estimator.

Although the preceding approach recovers some aspects of a dynamic setup, it surprisingly rules out the uncertainty at each iteration in favor of a static model, leading to undesired discontinuous estimates. This is better presented in the following unproven statement:

For a stationary process $\mathbf{x}(t)$, the estimated support through iterative application of (13) and (9) converges in the sense that the values of $\hat{\sigma}_k$ tend to zero for corresponding inactive indexes. In other words, the spectrum $\hat{\sigma}_k$ arbitrarily concentrates around a fixed support after a sufficient number of iterations.

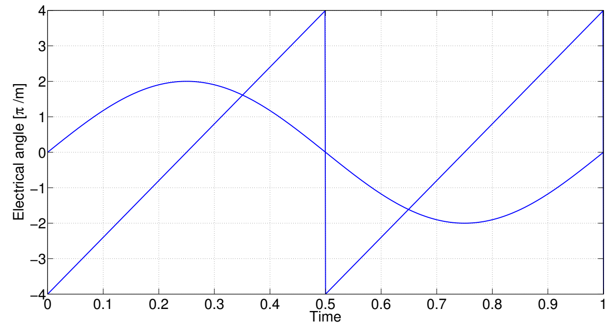

Fig. 1. The exact trajectory of two DOAs in terms of electrical angle.

Unfortunately, as the simulations in the next section suggest, the above result also applies to many non-stationary cases of interest. To cope with this, we propose to widen the $\hat{\sigma}_k$ spectrum in each iteration by smoothing the result of (13) before applying it to the next one. Due to our selected model in (2), we heuristically propose to update $\hat{\sigma}_k$ to a spatial neighborhood average which retains the linear complexity with N. Finally, a single iteration of the proposed method is given by

$$\mathbf{s}^g(t) = \hat{\mathbf{\Sigma}}_{t|t-1} \mathbf{A}_G^H (\mathbf{A}_G \hat{\mathbf{\Sigma}}_{t|t-1} \mathbf{A}_G^H + \lambda \mathbf{I})^{-1} \mathbf{x}(t) \tag{14a}$$

$$\hat{\sigma}_k^2(t|t) = (1-\alpha)\hat{\sigma}_k^2(t|t-1) + |s_k^g(t)|^2 \tag{14b}$$

$$\hat{\sigma}_k^2(t+1|t) = \frac{1}{2K+1} \sum_{k'=k-K}^{k+K} \hat{\sigma}_{k'}^2(t|t) \tag{14c}$$

where K is the spatial averaging radius and $\hat{\mathbf{\Sigma}}_{t|t-1}$ is a diagonal matrix with $\hat{\sigma}_k(t|t-1)$ as the diagonal element.

V. Numerical Results

In this part, we demonstrate the result of recursively applying the steps in (14) in the following relatively difficult scenario of two DOA tracks. The signals are measured by a half wavelength Uniform Linear Array (ULA) of $m = 4$ sensors. The first DOA, $\theta_1(t)$ uniformly orbits the sensor array such that $\phi_1(t) = 4\pi t - \pi$ for $t \in [0\ 1)$, where $\phi_i(t) = \pi \cos(\theta_i(t))$ is the corresponding electrical angle. Note that ϕ is modular, i.e. the values ϕ and $\phi \pm 2\pi$ are interchangeable. The second one $\theta_2(t)$ oscillates with $\phi_2(t) = \pi/2 \sin(2\pi t)$. Figure (1) depicts them over time. There is no discontinuity in the second path as the points $\phi = \pi$ and $\phi = -\pi$ correspond to the same DOA. The signal waveform and noise sequences were

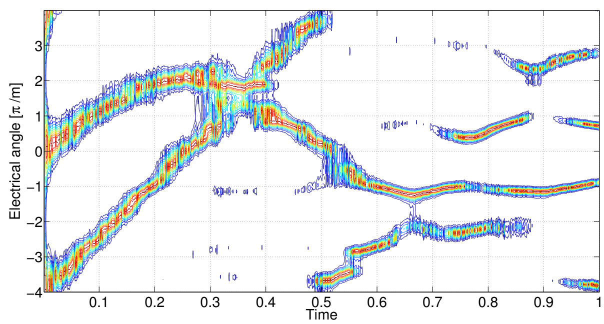

Fig. 2. The result of applying the recursion with forgeting factor and no spatial smoothing.

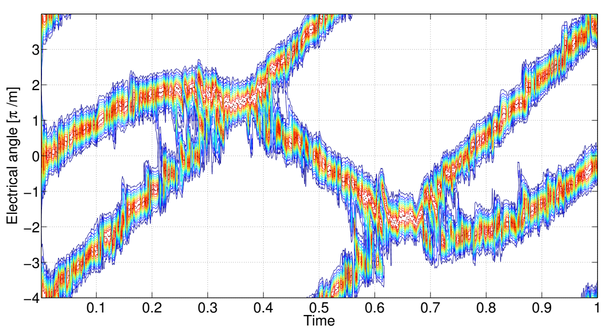

Fig. 3. The result of applying the recursive method by spatial smoothing. K=2

simulated by zero-mean, white, Gaussian, uncorrelated vectors with variances 1 and 0.1 respectively, which correspond to SNR=10 dB. We took a uniform grid of $N = 600$ DOA points and constructed $\mathbf{A}_G$ by the following array manifold:

$$\mathbf{a}(\theta) = [1 \; e^{j\phi} \; e^{j2\phi} \ldots \; e^{j(m-1)\phi}]^T \tag{15}$$

All plots depict 1000 samples in a unit time interval. We first considered the recursive method including (13) and (9). Figure 2 shows the contour plot of the scaled spectrum at different times. The white region corresponds to zero value of $\sigma(t)$. As seen, the spectrum $\sigma_k(t)$ first concentrates around the desired path. However, it eventually converges to some stationary points as predicted by the statement in Section IV. We then considered the steps in (14), where smoothing is included. Figure 3 correspond to $K = 2$ in (14). The values of λ and α were fixed to 0.5 and 0.2 respectively. Obviously, a larger value of K leads to smoother, but wider (less informative) spectrum estimates. This is comparable to the role of the process noise level in classical Kalman filters. Finally, Figure 4 compares the track estimates of the current method and the PAST algorithm (with MUSIC spectrum) by taking the spectrum peak points. As seen, the proposed method recovers cross points remarkably easier than PAST.

VI. Conclusion

In this work, we proposed a new low-complex method of DOA tracking, summarized in (14). We show its superior

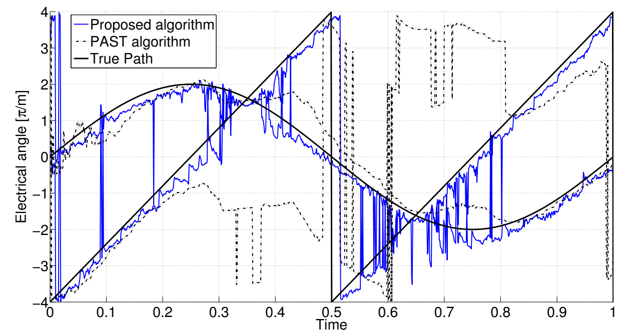

Fig. 4. The DOA estimates for the current method compared to that of the PAST algorithm.

properties by simulation in a difficult scenario. More elaborate results and comparisons as well as analytical discussions are postponed to future work due to lack of space. However, our preliminary results show that the proposed method resolve the problem of association and is robust against crossing.

To maintain low complexity, we made a very simple use of the evolution model by considering spatial smoothing. A more sophisticated way of connecting neighboring samples in space and time might lead to a similar reduced noise sensitivity with less resolution loss. But this might lead to more computational complexity, thus loosing the original goal of this work.

References

[1] D. Malioutov, M. Çetin, and A. S. Willsky, “A sparse signal reconstruction perspective for source localization with sensor arrays,” *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, 2005.

[2] K. Reif and R. Unbehauen, “The extended kalman filter as an exponential observer for nonlinear systems,” *IEEE Trans. Signal Process.*, vol. 47, no. 8, pp. 2324–2328, 1999.

[3] C. Rao, C. R. Sastry, and B. Zhou, “Tracking the direction of arrival of multiple moving targets,” *IEEE Trans. Signal Process.*, vol. 42, no. 5, pp. 1133–1144, 1994.

[4] B. Yang, “Projection approximation subspace tracking,” *IEEE Trans. Signal Process.*, vol. 43, no. 1, pp. 95–107, 1995.

[5] P. Heidenreich, L. A. Cirillo, and A. M. Zoubir, “Morphological image processing for fm source detection and localization,” *Signal Process.*, vol. 89, no. 6, pp. 1070–1080, 2009.

[6] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, “Online adaptive estimation of sparse signals: Where rls meets the¡ formula formulatype=,” *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3436–3447, 2010.

[7] D. Sejdinovic, C. Andrieu, and R. Piechocki, “Bayesian sequential compressed sensing in sparse dynamical systems,” in *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on.* IEEE, 2010, pp. 1730–1736.

[8] D. Angelosante, G. Giannakis, and E. Grossi, “Compressed sensing of time-varying signals,” in *Digital Signal Processing, 2009 16th International Conference on.* IEEE, 2009, pp. 1–8.

[9] N. Vaswani, “Kalman filtered compressed sensing,” in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on.* IEEE, 2008, pp. 893–896.

[10] Y. C. Eldar and H. Rauhut, “Average case analysis of multichannel sparse recovery using convex relaxation,” *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 505–519, 2010.

[11] T. Yardibi, J. Li, P. Stoica, M. Xue, and A. B. Baggeroer, “Source localization and sensing: A nonparametric iterative adaptive approach based on weighted least squares,” *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 46, no. 1, pp. 425–443, 2010.

[12] A. Panahi and M. Viberg, “Fast candidate points selection in the lasso path,” *IEEE Signal Process. Lett.*, vol. 19, no. 2, pp. 79–82, 2012.