A Newton-type Forward Backward Greedy Method for Multi-Snapshot Compressed Sensing

Ahmad Bazzi*†, Dirk T.M. Slock*, and Lisa Meilhac†
*EURECOM Mobile Communications Department, 450 route des Chappes, 06410 Biot Sophia Antipolis, France
Email: {bazzi,slock}@eurecom.fr
†CEVA-RivieraWaves, 400, avenue Roumanille Les Bureaux, Bt 6, 06410 Biot Sophia Antipolis, France
Email:{ahmad.bazzi, lisa.meilhac}@ceva-dsp.com

Abstract—**Parameter estimation has applications in many applications of signal processing, such as Angle-of-Arrival (AoA) estimation. Compressed sensing is a widely growing paradigm that can be applied to parameter estimation via sparse recovery. In this paper, we propose a Newton-type Forward Backward Greedy method that performs sparse recovery, given the observed data over multiple snapshots. This method is applied to the AoA estimation problem, where we have observed better performance, in terms of Mean-Squared Error and faster convergence when compared to existing methods. More information can be found in the conclusions section.**

***Index terms*— Sparse Recovery, Compressed Sensing, Angle-of-Arrival Estimation, Forward Backward Greedy, Multiple Measurement Vectors (MMV)**

I. Introduction

The problem of angles of arrival (AoA) of multiple sources is a fundamental one, which appears in several areas such as radar [1], [2] and localization [3], [4]. However, multipath [5] deteriorates the estimation performance if not taken into account. Therefore, we must take into account multipath parameters and implement fast algorithms to estimate all related parameters. Indeed, AoA estimation could be used to infer user position, such as in [6].

Compressed sensing seems to be a promising approach for data acquisition and compression, in a sense that reconstruction is possible even beyond the Nyquist rate. The papers in [7]–[9] give certain conditions where compressed sensing could be used to reduce the time and space needed to reconstruct the desired signal. To be more precise, assume the linear model

$$\boldsymbol{y} \simeq \boldsymbol{A}\boldsymbol{x}$$

where $\boldsymbol{y}$ is the observed vector, $\boldsymbol{A}$ is an over-complete dictionary and $\boldsymbol{x}$ is the desired signal. This problem has an infinite number of solutions; however, ℓ_p $(0 \leq p \leq 1)$ constraints on $\boldsymbol{x}$ force sparsity on $\boldsymbol{x}$ and hence favor sparse solutions, so the problem is no longer considered to be over-determined.

Greedy methods [10] have gained popularity over the recent years due to their fast nature to converge to a local optimum of the ℓ_0-norm problem. In particular, greedy methods select at each step the most suited column, or atom, of $\boldsymbol{A}$ according to some criterion. For instance, the Matching Pursuit (MP) [11] algorithms atoms based on maximum norm projections and the Orthogonal Matching Pursuit [12] selects atoms that best correlate with the residual part of the signal. Improvements of MP and OMP were implemented, such as the Directional Pursuit [13] and the Adaptive Forward Backward (AdFoBa) Greedy method [14]. In this paper, we are more interested in the latter, due to its ability to correct "bad selections" of atoms, thanks to its backward step. In addition to greedy methods, pusuit algorithms were established so as to relax the ℓ_0 problem to an ℓ_1. This offers convexity to the problem we have and therefore some methods could be derived and solved using fixed point methods. Popular algorithms that are used for the l_1 optimisation problem are the Iterative Shrinkage Thresholding Algorithm (ISTA) [15] and the Basis Pursuit Denoising (BPDN) [16]. Another category of methods are based on the *Bayesian* approach, where entries of $\mathbf{x}$ are assumed to have *a priori* statistical information. For instance, the Fast Matching Bayesian Pursuit (FBMP) [17] models the entries of $\boldsymbol{x}$ as Gaussian.

In this paper, we are interested in the Forward Backward Greedy scheme, namely the adaptive one in [14], i.e. AdFoBa. We propose a faster and better version of the adaptive scheme in [14]. To be more precise, a Newton type optimization is embedded in the forward step. Thanks to this step, we have observed faster convergence in the algorithm. Furthermore, the backward step also differs from the one in [14], to adapt with the Newton type version herein. Simulations show the performance and speed of this method, compared to [14] and others as well.

This paper is organized as follows: Section II presents the system model. In Section III, the proposed method is introduced and a pseudo code of the algorithm is also given. Section IV presents some simulation results. Finally, we conclude the paper in Section V.

Notations: Upper-case and lower-case boldface letters denote matrices and vectors, respectively. $(.)^T$ and $(.)^H$ represent the transpose and the transpose-conjugate operators. The matrix $\boldsymbol{I}$ is the identity matrix with appropriate dimensions. For a vector $\boldsymbol{x}$, the notation $[\boldsymbol{x}]_k$ denotes the k^{th} entry of $\boldsymbol{x}$. For a matrix $\boldsymbol{X}$ and set of indices $\boldsymbol{\Omega}$, the notation $\boldsymbol{X}_{(\boldsymbol{\Omega},:)}$ is a submatrix of $\boldsymbol{X}$ obtained by extracting the rows indexed by $\boldsymbol{\Omega}$. Similarly, the notation $\boldsymbol{X}_{(:,\boldsymbol{\Omega})}$ is obtained by extracting the columns indexed by $\boldsymbol{\Omega}$. Finally, for a set $\boldsymbol{\Omega}$, the notation $\boldsymbol{\Omega}/j$ is obtained by removing j from set $\boldsymbol{\Omega}$.

II. System Model

Assume a planar arbitrary array of N antennas. Furthermore, consider $q < N$ narrowband sources attacking the array from different angles, i.e. $\theta_1 \dots \theta_q$. Collecting L time snapshots, we could say the following

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{\chi} \tag{1}$$

where $\boldsymbol{Y} \in \mathbb{C}^{N\times L}$ is the collected data. The overcomplete basis $\boldsymbol{A} \in \mathbb{C}^{N\times K}$ is assumed known in the absence of array perturbations. For planar arrays, the j^{th} atom, or column, of $\boldsymbol{A}$ is $\boldsymbol{a}(\theta_j)$, where the k^{th} entry is

$$[\boldsymbol{a}(\theta_j)]_k = \frac{1}{\sqrt{N}} e^{-j\frac{w_c}{c}(\bar{x}_k \sin(\theta_j) + \bar{y}_k \cos(\theta_j))} \tag{2}$$

where $(\bar{x}_k, \bar{y}_k)$ is the position of the k^{th} antenna in a Cartesian system. The term $w_c = 2\pi f_c$ is the angular frequency, and c is the speed of light in vacuum. The matrix $\boldsymbol{\chi}$ is modeled as a white circular complex Gaussian process of zero mean and covariance $\sigma^2\boldsymbol{I}$ and independent from $\mathbf{X}$. Recall that this model is ideal, in a sense that the matrix $\boldsymbol{A}$ is known.

III. Newton-Type Forward Backward Method

Consider the following optimization problem

$$\begin{aligned} \underset{\boldsymbol{X}}{\text{minimize}} \quad & \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{X}\|_{2,0} \leq q \end{aligned} \tag{3}$$

where $\|.\|_{2,0}$ is the $\ell_{2,0}$ norm defined as

$$\|\boldsymbol{X}\|_{2,0} = \text{card}\ \{k : \|\boldsymbol{X}_{k,:}\|_2 \neq 0\} \tag{4}$$

where $\boldsymbol{X}_{k,:}$ is the k^{th} row of $\boldsymbol{X}$ and $\|\|_2$ is the ℓ_2 norm. So, basically, the $\ell_{2,0}$ norm counts the number of rows that have at least one non-zero entry. In this section, we develop *Forward-Backward Greedy approach* that optimizes (3).

A. Forward Step

At an n^{th} iteration, we propose to choose an atom that minimizes

$$j^{(n)} = \arg \min_{j \notin \boldsymbol{\Omega}^{(n-1)}} \min_{\boldsymbol{\beta}} \frac{\|\boldsymbol{Y} - \boldsymbol{A}\big(\boldsymbol{X}^{(n)} + \boldsymbol{e}_j \boldsymbol{\beta}^H\big)\|_2^2}{\|\big[\nabla_{\boldsymbol{X}}(\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X})\big]_{j,:}\|_2^2} \tag{5}$$

which is easily verified to be

$$j^{(n)} = \arg \min_{j \notin \boldsymbol{\Omega}^{(n-1)}} \frac{\|\boldsymbol{Y} - \boldsymbol{A}\big(\boldsymbol{X}^{(n)} + \boldsymbol{e}_j \boldsymbol{A}^H_{:,j}[\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}]\big)\|_2^2}{\|\big[\boldsymbol{A}^H(\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X})\big]_{j,:}\|_2^2} \tag{6}$$

Here, we are *"wiggling"* the weights corresponding to the j^{th} atom, or column, in $\boldsymbol{A}$ and choosing the atom index that is *least affected* with this perturbation. Moreover, we have included the Gradient term in the denominator of the above cost function, similar to the *Newton's* method. Although it may seem natural, this additional term helps in speeding up the convergence of the algorithm, yet achieving better performance as well. After finding this index and appending it in the support set, namely

$$\boldsymbol{\Omega}^{(n)} \leftarrow \boldsymbol{\Omega}^{(n-1)} \cup \{j^{(n)}\} \tag{7}$$

We estimate an updated version of $\boldsymbol{X}$ as follows

$$\boldsymbol{X}^{(n+1)} = (\boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})}\boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)})})^{-1}\boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})}\boldsymbol{Y} \tag{8}$$

Also, let $\epsilon^{(n)}$ denote the relative error at iteration n as

$$\epsilon^{(n)} = |\|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}^{(n+1)}\|_2^2 - \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}^{(n)}\|_2^2| \tag{9}$$

B. Backward Step

To allow flexibility of the proposed greedy method, we propose a backward scheme. The backward scheme will indeed depend on the value of the error $\epsilon^{(n)}$ at iteration (n). If the error is "relatively" small, we can go on to another forward step $n + 1$, otherwise a correction is needed. A natural question arises here: *"What should $\epsilon^{(n)}$ be compared to?"* Well, we can ask an alternative question, which is the following *"What if the atom added at iteration (n) corresponding to index $j^{(n)}$ increases the cost function $\|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2$ and not decrease it ?"* To check

Algorithm 1: Pseudo-code of the Newton-type Forward-Backward Greedy Algorithm

```
Input: $\boldsymbol{Y}$, $\boldsymbol{A}$, $\delta$
Output: $\boldsymbol{X}$
Initialize: $\boldsymbol{X}^{(0)} = \boldsymbol{0}$, $n = 1$, $\boldsymbol{\Omega}^{(0)} = \phi$, $\epsilon^{(0)} = 10^{15}$
do
    $j^{(n)} = \arg \min_{j \notin \boldsymbol{\Omega}^{(n-1)}} \frac{\|\boldsymbol{Y} - \boldsymbol{A}(\boldsymbol{X}^{(n)} + \boldsymbol{e}_j \boldsymbol{A}^H_{:,j}[\boldsymbol{Y} - \boldsymbol{AX}])\|_2^2}{\|[\boldsymbol{A}^H(\boldsymbol{Y} - \boldsymbol{AX})]_{j,:}\|_2^2}$
    $\boldsymbol{\Omega}^{(n)} \leftarrow \boldsymbol{\Omega}^{(n-1)} \cup \{j^{(n)}\}$
    $\boldsymbol{X}^{(n+1)} = (\boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)})})^{-1} \boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{Y}$
    $\epsilon^{(n)} = |\|\boldsymbol{Y} - \boldsymbol{AX}^{(n+1)}\|_2^2 - \|\boldsymbol{Y} - \boldsymbol{AX}^{(n)}\|_2^2|$
    $n \leftarrow n + 1$
    Compute for $i \in \boldsymbol{\Omega}^{(n)}$
    $\vartheta_i^{(n)} = \left| \|\boldsymbol{Y} - \boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)}/\{i\})} \boldsymbol{X}^{(n+1)}_{(\boldsymbol{\Omega}^{(n)}/\{i\},:)}\|_2^2 - \|\boldsymbol{Y} - \boldsymbol{AX}^{(n)}\|_2^2 \right|$
    $\vartheta^{(n)} = \min \{\vartheta_i^{(n)}\}_{j \in \boldsymbol{\Omega}^{(n)}}$
    $i^{(n)} = \operatorname{argmin}_i \vartheta_i^{(n)}$
    if $\vartheta^{(n)} \leq \epsilon^{(n)}$ then
        $\boldsymbol{\Omega}^{(n)} \leftarrow \boldsymbol{\Omega}^{(n)}/\{i^{(n)}\}$
        $\boldsymbol{X}^{(n+1)} = (\boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)})})^{-1} \boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{Y}$
        $n \leftarrow n - 1$
    else
        do nothing
while $\epsilon^{(n)} > \delta$;
```

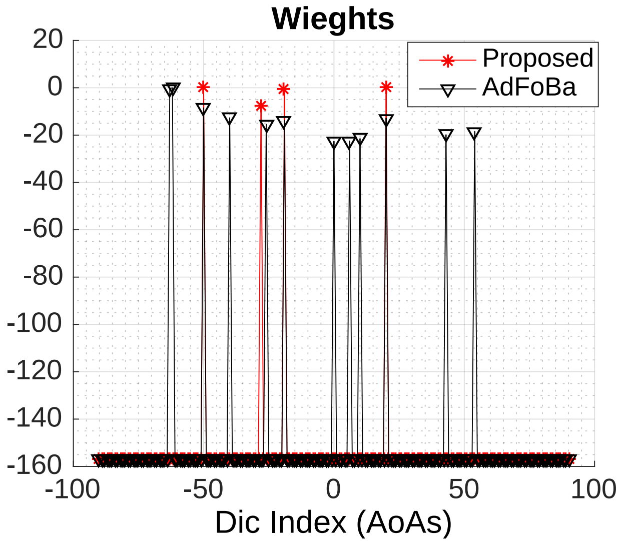

Fig. 1: Comparison of spectra for $q = 4$ sources.

for this case, we compare the error $\epsilon^{(n)}$ to an error $\vartheta^{(n)}$, which is computed if the support $\boldsymbol{\Omega}^{(n)}$ contains 1 less element[1]. More precisely, define

$$\vartheta_i^{(n)} = \left| \|\boldsymbol{Y} - \boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)}/\{i\})} \boldsymbol{X}^{(n+1)}_{(\boldsymbol{\Omega}^{(n)}/\{i\},:)}\|_2^2 - \|\boldsymbol{Y} - \boldsymbol{AX}^{(n)}\|_2^2 \right| \tag{10}$$

for all $i \in \boldsymbol{\Omega}^{(n)}$. Now choose the smallest error amongst all $\vartheta_i^{(n)}$, i.e.

$$\vartheta^{(n)} = \min \{\vartheta_i^{(n)}\}_{j \in \boldsymbol{\Omega}^{(n)}} \tag{11}$$

and denote

$$i^{(n)} = \operatorname*{argmin}_i \vartheta_i^{(n)} \tag{12}$$

Here, if $\vartheta^{(n)} > \epsilon^{(n)}$, we say that the error at iteration n is acceptable and there doesn't seem to be any over-fitting. On the other hand, if $\vartheta^{(n)} \leq \epsilon^{(n)}$, we should remove this "defected atom", which corresponds to index $i^{(n)}$

$$\boldsymbol{\Omega}^{(n)} \leftarrow \boldsymbol{\Omega}^{(n)}/\{i^{(n)}\} \tag{13}$$

Re-modify the weighting matrix

$$\boldsymbol{X}^{(n+1)} = (\boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{A}_{(:,\boldsymbol{\Omega}^{(n)})})^{-1} \boldsymbol{A}^H_{(:,\boldsymbol{\Omega}^{(n)})} \boldsymbol{Y} \tag{14}$$

and finally go one step backward

$$n \leftarrow n - 1 \tag{15}$$

The forward backward procedure is repeated until error $\epsilon^{(n)} \leq \delta$, where δ is a given tolerance value.

IV. Simulations

In this section, we present some computer simulations. In Fig 1, we have used $N = 15$ antennas and a dictionary of size $K = 181$ discritized at steps of 1°. Furthermore, $L = 1$ snapshot was used at SNR = 20 dB. We have $q = 4$ sources at $\theta_1 = -50°$, $\theta_2 = -27°$, $\theta_3 = -20°$, and $\theta_4 = 20°$. We can clearly see the difference between the proposed Newton-type method and the one AdFoBa [14]. Our method avoids overfitting of sources, whereas the AdFoBa overestimates then number of existing sources.

In Experiment 1 (Fig. 2), we are interested in the MSE performance of existing sparse recovery methods compared to the proposed here. We compare the Newton-type Forward Backward proposed method with AdFoBa [14], BPDN [16] and FBMP [17]. Here we have set $L = 10^2$, $N = 10$, $q = 2$ with $\theta_1 = 0°$ and $\theta_2 = 10°$. Also $K = 181$ as before. The MSE is computed using 10^4 Monte Carlo trials. Here, in case of overfitting, we choose the q largest peaks in the weights. We can see that both Forward-Backward schemes (the

[1] This could be seen as over-fitting.

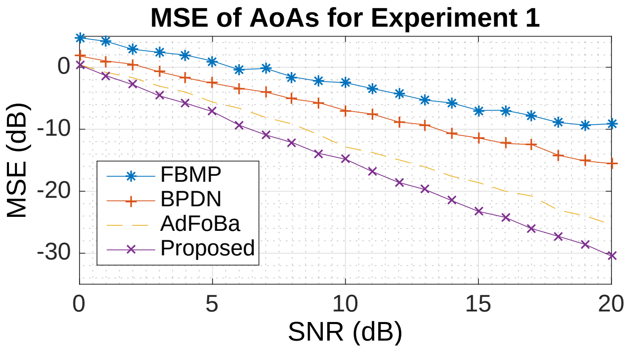

Fig. 2: MSE of AoAs for Exp. 1

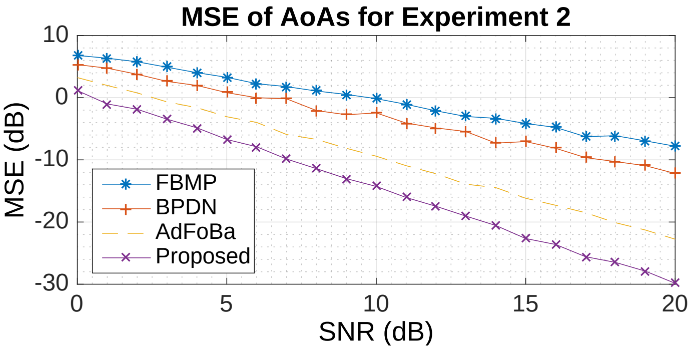

Fig. 3: MSE of AoAs for Exp. 2

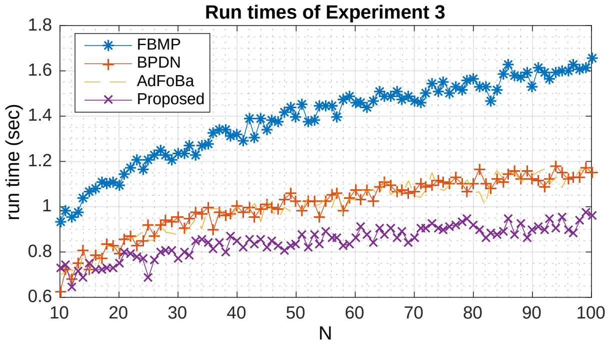

Fig. 4: Run times of Exp. 3

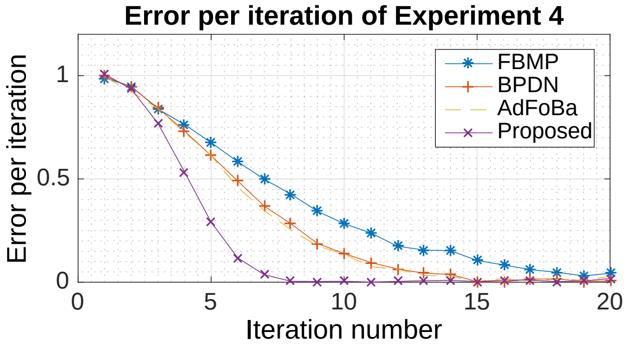

Fig. 5: Errors per iteration of Exp. 4

proposed one and AdFoBa) perform better than BPDN and FBMP, due to their adaptive ability of ”correcting themselves” in case of any overfitting or false selected atoms. Moreover, the proposed one performs better than AdFoBa, due to the different backward step criterion. We can see an 5 dB between the proposed method and the AdFoBa at sufficiently high SNR. In Experiment 2 (Fig. 3), we have used the same parameters as in Experiment 1, except that we have changed L to $L = 10$. We can also observe the phenomena as above. Note that here we have a higher MSE for all methods due to less observed samples. Nevertheless, we can see that the MSE gaps between the different methods are still the same as that in Experiment 1. We can see an 8 dB between the proposed method and the AdFoBa at sufficiently high SNR.

Another important aspect is the algorithm complexity or the number of operations required before the algorithm terminates. In Experiment 3 (i.e. Fig 4), we study the speed of the algorithms mentioned above as a function of number of antennas N. To assess generality, we have also averaged the speeds over 10^4 Monte Carlo trials. We can see that the proposed algorithm terminates before all the other ones mentioned above, thanks to the gradient factor in the cost function of equation (5). If $N = 100$ antennas were used, we can see a gain of speed of about 0.6 seconds compared to the FBMP algorithm and 0.2 seconds compared to AdFoBa and BPDN.

Finally, in Experiment 4, i.e. Fig 5, we have fixed the parameters as in Experiment 1 and studied the behavior of the error for different algorithms, in the sense of

$$\left| \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}^{(n)}\|^2 - \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}^{(n-1)}\|^2 \right| \qquad (16)$$

This means that when no improvement occurs, the above error should become negligible. Also, we can see that the proposed algorithm converges in about 8 to 9 iterations. The AdFoBa and the BPDN require around 15 iterations to achieve the same error as the proposed one. Additionally, we can see that the FBMP needs more than 20 iterations to achieve this accuracy.

V. Conclusion

From the outcomes of our investigations, it seems to be possible that the propsed Newton-type forward backward greedy method performs faster, in terms of convergence and number of operations, and better, in terms of Mean-Squared-Error of AoAs. At high SNR, we have been able to achieve a 5 to 8 dB MSE improvement compared to the Adaptive Forward Backward greedy method (AdFoBa) and a 0.2 sec faster run time for large number of antennas.

Acknowledgment

This work was supported in part by RivieraWaves, a CEVA company, and a Cifre scholarship. EURECOM's research is partially supported by its industrial members and by the H2020 EU project HIGHTS.

References

[1] V. Gupta, *et al.* "SBL-based joint target imaging and Doppler frequency estimation in monostatic MIMO radar systems," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[2] A. Daniel and D. C. Popescu. "MIMO radar waveform design for multiple extended target estimation based on greedy SINR maximization." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[3] N. Garcia, *et al.* "Direct Localization for Massive MIMO." arXiv preprint arXiv:1607.00946 , 2016.

[4] E. Leitinger, *et al.* "Cognitive indoor positioning and tracking using multipath channel information." arXiv preprint arXiv:1610.05882, 2016.

[5] S. Hinteregger, E. Leitinger, *et al.* "Bandwidth dependence of the ranging error variance in dense multipath." *IEEE European Signal Processing Conference (EUSIPCO)*, 2016.

[6] G.C. Alexandropoulos, "Position Aided Beam Alignment for Millimeter Wave Backhaul Systems with Large Phased Arrays," arXiv preprint arXiv:1701.03291, 2017.

[7] R.G. Baraniuk, "Compressive sensing [lecture notes]." *IEEE Signal Processing Magazine* 24.4, 118-121, 2007.

[8] E. Candes, J. Romberg, and T. Tao. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information." *EEE Transactions on information theory*, 52.2: 489-509, 2006.

[9] E. Candes, and M.B. Wakin. "An introduction to compressive sampling." *IEEE signal processing magazine*, 25.2: 21-30, 2008.

[10] J.A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information theory* 50.10: 2231-2242, 2004.

[11] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, 41:33973415, 1993.

[12] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," In *Proc. of the 27th Asilomar Conference on Signals, Systems and Computers*, 1:40-44, 1993.

[13] T. Blumensath and M. Davies, "Gradient pursuits," *IEEE Transactions on Signal Processing*, 56(6):23702382, 2008.

[14] T. Zhang, "Adaptive forward-backward greedy algorithm for learning sparse representations." IEEE transactions on information theory 57.7 (2011): 4689-4708.

[15] I. Daubechies, M. Defrise, and C. de Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math.*, 57 (2004), pp. 1413-1457.

[16] S. Chen, D. Donoho, and M. Saunders, Atomic decomposition by basis pursuit, *SIAM J. Sci. Comput.*, 20,pp. 33-61, 1998.

[17] P. Schniter, L. C. Potter, and J. Ziniel, "Fast bayesian matching pursuit," in *Proc. Workshop Inform. Theory Appl. (ITA)*, 2008.