ADAPTIVE 2-D DOA ESTIMATION USING SUBSPACE FITTING

Jie Zhuang[1], Lu Yang[2], Guo-Yong Ning[2], Ishak Ali Hussein[1], and Wei Wang[1]

[1]School of Information and Communication Engineering,
University of Electronic Science and Technology of China (UESTC), Chengdu, 611731, China.
[2]Chengdu Aerospace Communication Device Company, Chengdu, 610052, China.

ABSTRACT

Direction-of-arrival (DOA) estimation is a ubiquitous task in array processing. In this paper, we propose an adaptive 2-dimensional direction finding framework to track multiple moving targets by using the subspace fitting method. First, we expand the steering vectors of the current snapshot in a Taylor series around the DOAs of the previous snapshot. Then we transform the subspace fitting problem into a set of linear equations. As a result, the DOAs of each snapshot can be updated by solving a set of linear equations and we no longer need to search the 2-D spatial spectrum. In comparison with the traditional 2-D MUSIC, the proposed method not only reduces the computational complexity considerably but also has better estimation performance.

Index Terms— 2-D DOA Estimation, subspace fitting, DOA tracking.

1. INTRODUCTION

Direction-of-arrival (DOA) estimation is a ubiquitous task concerned in array processing, which has been widely used in wireless communication, radar, sonar, acoustics, astronomy, medical imaging, and other areas. In this paper, both the azimuth and elevation angles are of interest and we assume that they are time-varying.

Consider an array of N sensors operating in the presence of M uncorrelated narrowband signals via unknown directions. The $N \times 1$ signal vector received at the time instant t can be expressed as

$$\mathbf{x}(t) = \mathbf{A}(t)\mathbf{m}(t) + \mathbf{n}(t) \tag{1}$$

where $\mathbf{m}(t)$ collects the M complex narrowband signal envelopes and $\mathbf{n}(t)$ represents the additive white Gaussian noise with covariance $\sigma_n^2\mathbf{I}$ (σ_n^2 is the noise power). The notation $\mathbf{I}$ denotes an identity matrix. The matrix $\mathbf{A}(t)$ has the steering vectors of the M signals, i.e.,

$$\mathbf{A}(t) = [\mathbf{a}(\theta_1(t), \phi_1(t)), \ldots, \mathbf{a}(\theta_M(t), \phi_M(t))] \tag{2}$$

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant No. 61571090, and Sichuan Science and Technology Program under Grant 2017JY0223.

where $\theta \in [0, 2\pi)$ and $\phi \in [0, \pi]$ denote the azimuth and elevation angle respectively. The n-th element of the $N \times 1$ steering vector $\mathbf{a}_m \triangleq \mathbf{a}(\theta_m(t), \phi_m(t))$ is given by

$$[\mathbf{a}_m]_n = e^{-j\pi(x_n \sin\theta_m \sin\phi_m + y_n \cos\theta_m \sin\phi_m + z_n \cos\phi_m)} \tag{3}$$

where $\{x_n, y_n, z_n\}$ are the coordinates of the n-th array sensor in units of half-wavelengths. The notation $[\cdot]_n$ denotes the n-th element of a vector.

In practical applications, the time-varying covariance matrix of the received vector $\mathbf{x}(t)$ can be obtained as follows [1]

$$\widehat{\mathbf{R}}(t) = \sum_{k=1}^{t} \beta^{t-k}\mathbf{x}(k)\mathbf{x}^H(k) = \beta\widehat{\mathbf{R}}(t-1) + \mathbf{x}(t)\mathbf{x}^H(t) \tag{4}$$

where $\beta \in (0, 1]$ is commonly known as the forgetting factor. Eq. (4) tells us that all the received sample vectors available in the time interval $1 \le k \le t$ are involved in the estimation and the data in the distant past should be downweighted. Then performing eigenvalue decomposition on $\widehat{\mathbf{R}}(t)$ produces

$$\widehat{\mathbf{R}}(t) = \mathbf{E}_s(t)\mathbf{\Lambda}_s(t)\mathbf{E}_s^H(t) + \mathbf{E}_n(t)\mathbf{\Lambda}_n(t)\mathbf{E}_n^H(t) \tag{5}$$

where $\mathbf{E}_s(t) \in \mathcal{C}^{N\times M}$ is the eigenvectors associated with the largest M eigenvalues and $\mathbf{E}_n(t) \in \mathcal{C}^{N\times(N-M)}$ represents the eigenvectors corresponding to the remaining small eigenvalues. Commonly $\mathbf{E}_s(t)$ and $\mathbf{E}_n(t)$ are referred to as the signal-subspace eigenvectors and noise-subspace eigenvectors. The diagonal matrices $\mathbf{\Lambda}_s(t)$ and $\mathbf{\Lambda}_n(t)$ have diagonal elements associated with the signal and noise eigenvalues respectively.

In order to find the true DOAs, a null-spectrum cost function is formed as follows

$$(\widehat{\theta}, \widehat{\phi}) = \arg\min_{\theta,\phi}\{\mathbf{a}(\theta, \phi)^H\mathbf{E}_n(t)\mathbf{E}_n^H(t)\mathbf{a}(\theta, \phi)\}. \tag{6}$$

Due to the orthogonality between the signal and noise subspace, the steering vectors of the source signals correspond to the minima of the above cost function. Therefore, we can find the true DOAs by searching the continuous array manifold vector over the area of θ and ϕ. This is the basic idea of the well-known MUltiple SIgnal Classification (MUSIC) method [2]. However, such spectral search process may be unaffordable for some real-time implementations since a matrix product of $\mathbf{a}(\theta,\phi)^H \mathbf{E}_n(t)\mathbf{E}_n^H(t)\mathbf{a}(\theta,\phi)$ has to be computed for each search point. This drawback becomes particularly apparent in joint estimation of azimuth and elevation since we have to search over two dimensions [3, 4]. Moreover, eigen-decomposition operation is also computationally expensive and thus in many real-time applications we cannot conduct it when each new snapshot arrives.

In this paper, we propose a computation attractive 2-D DOA estimator. First, we employ existing signal subspace tracking methods to update the signal subspace when each snapshot arrives, thereby avoiding the computationally expensive eigen-decomposition process. Then a subspace fitting problem is formulated to find the DOAs for each snapshot. The trick to solve such subspace fitting problem is that we expand the array steering vectors of the current snapshot in a Taylor series around the DOAs of the previous snapshot. In doing so, the subspace fitting problem is transformed to a set of linear equations. Thus, the DOAs of each snapshot can be updated by solving a set of linear equations and we no longer need to search the 2-D spatial spectrum.

2. PROPOSED ADAPTIVE 2-D DOA ESTIMATOR

Here we assume that both the azimuth and elevation angles change constantly due to the moving targets. This section provides an adaptive 2-D direction finding framework of tracking multiple moving targets. Next, we will introduce how to expand the array steering vectors of the current snapshot in a Taylor series around the DOAs of the previous snapshot.

2.1. Steering Vector Expansion

Let $\boldsymbol{\eta} = [\theta_1(t), \phi_1(t), \ldots, \theta_M(t), \phi_M(t)]^T$ denote the full angular parameter vector. Suppose that our knowledge of the angles is inaccurate, viz. we wrongly assume that the DOAs are $\boldsymbol{\eta}_0 = [\theta_{1,0}(t), \phi_{1,0}(t), \ldots, \theta_{M,0}(t), \phi_{M,0}(t)]^T$ in lieu of the true values $\boldsymbol{\eta}$. Using the Taylor expansion and retaining the terms up to the first order, the true array steering vectors can be approximated by

$$\mathbf{A}(t) \approx \mathbf{A}_0(t) + \sum_{k=1}^{2M} \Delta_k \left. \frac{\partial \mathbf{A}}{\partial [\boldsymbol{\eta}]_k} \right|_{\boldsymbol{\eta}_0} \tag{7}$$

where $\mathbf{A}_0(t)$ can be computed by (2) using the inaccurate information $\boldsymbol{\eta}_0$ and $\Delta_k = [\boldsymbol{\eta}]_k - [\boldsymbol{\eta}_0]_k$. The partial derivatives of the array steering matrix with respect to the m-th DOAs are given by

$$\begin{aligned} \frac{\partial \mathbf{A}}{\partial \theta_m} &= \left[\mathbf{0}, \ldots, \frac{\partial \mathbf{a}_m}{\partial \theta_m} \ldots, \mathbf{0}\right], \\ \frac{\partial \mathbf{A}}{\partial \phi_m} &= \left[\mathbf{0}, \ldots, \frac{\partial \mathbf{a}_m}{\partial \phi_m} \ldots, \mathbf{0}\right] \end{aligned} \tag{8}$$

where only the m-th column is non-zero and

$$\left[\frac{\partial \mathbf{a}_m}{\partial \theta_m}\right]_n = j\pi[\mathbf{a}_m]_n(y_n \sin\theta_m \sin\phi_m - x_n \cos\theta_m \sin\phi_m) \tag{9}$$

and

$$\begin{aligned} \left[\frac{\partial \mathbf{a}_m}{\partial \phi_m}\right]_n \quad &= \quad -j\pi[\mathbf{a}_m]_n(x_n \sin\theta_m \cos\phi_m \\ &\quad + y_n \cos\theta_m \cos\phi_m - z_n \sin\phi_m). \end{aligned} \tag{10}$$

2.2. Subspace Fitting

In the presence of moving sources, we have to compute the signal and noise subspaces repeatedly since they vary constantly with time. This implies that the usefulness of the block-processing-based subspace techniques may not be realistic due to the high computational complexity associated with either the eigen-decompositions of the covariance data matrix or searching the minima over 2-D MUSIC spatial spectrum.

In order to overcome the above difficulties, we employ the subspace tracking technique [5] to update the eigenbasis recursively on the arrival of a new data snapshot. In our proposed approach, the Fast Approximated Power Iteration (FAPI) method presented in [6] is adopted for the recursive signal subspace estimation. The FAPI algorithm is a fast implementation of the classical power iteration method. When each new snapshot arrives, the FAPI method can produce an arbitrary orthonormal basis of the signal subspace, represented by $\mathbf{W}(t) \in \mathcal{C}^{N\times M}$. In other words, despite $\mathbf{W}(t) \neq \mathbf{E}_s(t)$, the associated subspace projection matrices are equal, i.e., $\mathbf{W}(t)\mathbf{W}^H(t) = \mathbf{E}_s(t)\mathbf{E}_s^H(t)$. For more details, refer to [6].

Once we have the knowledge of the signal subspace, we can obtain the DOAs of each snapshot by solving the following multidimensional fitting problem [7]:

$$\widehat{\boldsymbol{\eta}} = \arg \min_{\boldsymbol{\eta},\mathbf{T}} \parallel \mathbf{W}(t) - \mathbf{A}(\boldsymbol{\eta})\mathbf{T} \parallel^2 \tag{11}$$

which aims to find an $\mathbf{A}(\boldsymbol{\eta})$ such that the two subspaces spanned by $\mathbf{A}(\boldsymbol{\eta})$ and $\mathbf{W}(t)$ are as close as possible. This method suffers from a costly multidimensional optimization. However, we can observe that there are no mixed constraints in (11) for the two variables $\mathbf{A}(\boldsymbol{\eta})$ and $\mathbf{T}$, which implies that (11) is separable in $\mathbf{A}(\boldsymbol{\eta})$ and $\mathbf{T}$. In other words, we can fix either $\mathbf{A}(\boldsymbol{\eta})$ or $\mathbf{T}$ and optimize the other, which is easier to solve than (11) in its entirety [8, 9]. For instance, if we fix $\mathbf{A}(\boldsymbol{\eta})$, the optimal solution for $\mathbf{T}$ can be readily obtained by

$$\mathbf{T}^\star = \mathbf{A}^\dagger(\boldsymbol{\eta})\mathbf{W}(t) \tag{12}$$

where $\mathbf{A}^\dagger(\boldsymbol{\eta}) = (\mathbf{A}^H(\boldsymbol{\eta})\mathbf{A}(\boldsymbol{\eta}))^{-1}\mathbf{A}^H(\boldsymbol{\eta})$. Then we assume that $\mathbf{T}$ is known and rewrite the cost function of (11) as follows

$$f = \mathrm{Tr}\left\{(\mathbf{W}(t) - \mathbf{A}(\boldsymbol{\eta})\mathbf{T})^H (\mathbf{W}(t) - \mathbf{A}(\boldsymbol{\eta})\mathbf{T})\right\} \tag{13}$$

where the notation Tr$\{\cdot\}$ denotes the trace of the matrix between brackets. Differentiating f with respect to the i-th element of $\boldsymbol{\eta}$, we have

$$\frac{\partial f}{\partial[\boldsymbol{\eta}]_i} = \mathrm{Tr}\left\{-\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\mathbf{T}\right)^H(\mathbf{W}(t)-\mathbf{A}(\boldsymbol{\eta})\mathbf{T})\right\}+(\cdots)^H \tag{14}$$

where the notation $(\cdots)^H$ means the same expression appears again with conjugate transpose. Then inserting (7) into (14) produces (15) where the notation Re$\{\cdot\}$ denotes the real part. By setting (15) to zero, we have (16).

Now the equations $\left[\frac{\partial f}{\partial[\boldsymbol{\eta}]_1}, \ldots, \frac{\partial f}{\partial[\boldsymbol{\eta}]_{2M}}\right]^T = \mathbf{0}^T$ can be written more compactly as

$$\mathbf{H}\boldsymbol{\Delta} = \mathbf{b} \tag{17}$$

where

$$[\mathbf{H}]_{i,k} = \mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_k}\right|_{\boldsymbol{\eta}_0}\mathbf{T}\right\}\right\} \tag{18}$$

and

$$[\mathbf{b}]_i = \mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H(\mathbf{W}(t)-\mathbf{A}_0\mathbf{T})\right\}\right\} \tag{19}$$

with $i, k = 1, 2, \ldots, 2M$. Consequently, the error vector can be readily computed by

$$\boldsymbol{\Delta} = \mathbf{H}^{-1}\mathbf{b}. \tag{20}$$

2.3. Summary of the Proposed Method

We summarise the proposed DOA estimation method in Algorithm 1. Theoretically speaking, iterating between Step 1 and 2 may not lead to the global solution to (11). However, this work can yield an estimate of the steering vectors closer to the signal subspace than the steering vectors of the previous snapshot. In the simulation part, it is found that the average of iteration times is less than five. Although the matrix inverse operation is involved in each iteration, the computational complexity is low since the dimensions of the matrices $(\mathbf{A}^H(\boldsymbol{\eta})\mathbf{A}(\boldsymbol{\eta}))$ and $\mathbf{H}$ (whose inverses we need to compute) are of M and $2M$ only. Before closing this section, we would like to point out that our framework is not limited to the FAPI method. Other subspace tracking approaches can be accommodated in our framework as well.

3. SIMULATION RESULTS

In this section we evaluate the effectiveness of the proposed adaptive 2-D DOA estimation approach. Consider an L-shaped array of $N = 10$ sensors with x-y Cartesian coordinates (measured in half-wavelength) given by

$$\begin{aligned}\mathbf{x} &= [-2,-2,-2,-2,-2,-2,-1,0,1,2]\\ \mathbf{y} &= [5,4,3,2,1,0,0,0,0,0].\end{aligned}$$

We assume that the forgetting factor is $\beta = 0.7$ and the signal number is $M = 2$.

In the first example, we consider the Gauss-Markov mobility model [10] in which the true azimuth and elevation are generated as follows:

$$\begin{aligned}\theta(t) &= \alpha\theta(t-1) + (1-\alpha)\overline{\theta}(t) + \sqrt{1-\alpha^2}\theta_x\\ \phi(t) &= \alpha\phi(t-1) + (1-\alpha)\overline{\phi}(t) + \sqrt{1-\alpha^2}\phi_x\end{aligned} \tag{21}$$

where $\overline{\theta}(t)$ and $\overline{\phi}(t)$ represent the mean azimuth and elevation, θ_x and ϕ_x are two random variables drawn from a Gaussian distribution $\mathcal{N}(0,1)$, and $\alpha \in [0,1]$ is the tuning parameter which determines the degree of randomness. When α is zero, completely random movement (i.e. Brownian motion) is obtained. If α is set to be 1, movement becomes predictable, losing all randomness. Here we choose $\alpha = 0.7$. Moreover, we set $\overline{\theta}(t) = \overline{\theta}(t-1)+0.2^\circ$ and $\overline{\phi}(t) = \overline{\phi}(t-1)+0.2^\circ$, which means that at each new arrival snapshot, the mean angles are increased by 0.2°. The mean angles at the first snapshot are $(\overline{\theta}_1(1), \overline{\phi}_1(1)) = (30^\circ, 20^\circ)$ and $(\overline{\theta}_2(1), \overline{\phi}_2(1)) = (50^\circ, 40^\circ)$ for the two targets. In Fig. 1 and 2, the true and estimated azimuths and elevations are illustrated for one trial when the snapshot number varies from 10 to 100 and the input signal-to-noise ratio (SNR) is fixed at 10dB. As shown in Fig. 1 and 2, the proposed subspace-fitting-based adaptive direction finding approach works well for both moving targets.

In the second simulation experiment, we investigate the root-mean-square errors (RMSEs) between the estimated and real DOAs. The snapshot number also changes from 10 to 100. The initial angles of the second source moves close to the first source with $(\overline{\theta}_2(1), \overline{\phi}_2(1)) = (40^\circ, 30^\circ)$. In Fig 3, we plot both the azimuth and the elevation estimation RMSEs versus the SNR for the proposed method and the traditional 2-D MUSIC method (see (6)), where the SNR varies from 5dB

Algorithm 1 Proposed Subspace-Fitting-Based Method

For each time step do:
Input: signal eigenvectors $\mathbf{W}(t)$ and previous DOA estimations $\boldsymbol{\eta}_0 = \boldsymbol{\eta}(t-1)$.
Iterate the following steps until convergence:

1. Compute the matrix $\mathbf{T}$ by (12) with $\boldsymbol{\eta}_0$,
2. Compute the error vector $\boldsymbol{\Delta}$ using (20) where $\mathbf{H}$ and $\mathbf{b}$ are defined in (18) and (19),
3. Update $\boldsymbol{\eta}_0 = \boldsymbol{\eta}_0 + \boldsymbol{\Delta}$,
4. Repeat Step 1, 2 and 3 a few times.

$$\frac{\partial f}{\partial[\boldsymbol{\eta}]_i} = \mathrm{Tr}\left\{-\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H\left(\mathbf{W}(t) - \mathbf{A}_0\mathbf{T} - \sum_{k=1}^{2M}\Delta_k \left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_k}\right|_{\boldsymbol{\eta}_0}\mathbf{T}\right)\right\} + (\cdots)^H \tag{15}$$

$$= -2\mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H(\mathbf{W}(t) - \mathbf{A}_0\mathbf{T})\right\}\right\} + 2\sum_{k=1}^{2M}\mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_k}\right|_{\boldsymbol{\eta}_0}\mathbf{T}\right\}\right\}\Delta_k$$

$$\sum_{k=1}^{2M}\mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_k}\right|_{\boldsymbol{\eta}_0}\mathbf{T}\right\}\right\}\Delta_k = \mathrm{Tr}\left\{\mathrm{Re}\left\{\mathbf{T}^H\left(\left.\frac{\partial \mathbf{A}}{\partial[\boldsymbol{\eta}]_i}\right|_{\boldsymbol{\eta}_0}\right)^H(\mathbf{W}(t) - \mathbf{A}_0\mathbf{T})\right\}\right\} \tag{16}$$

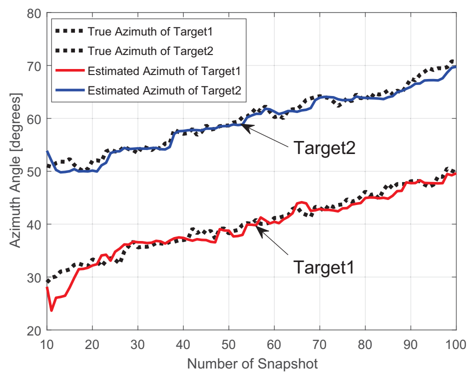

Fig. 1. True and estimated azimuths of two moving targets; first example.

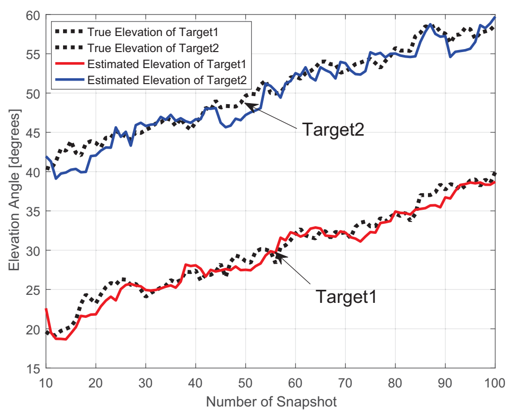

Fig. 2. True and estimated elevations of two moving targets; first example.

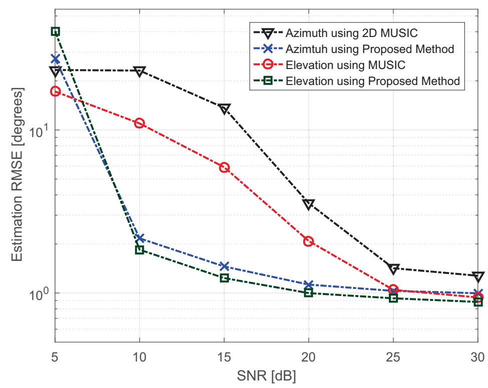

Fig. 3. Azimuth and elevation estimation RMSEs versus input SNR; second example.

to 30dB and the search grid of MUSIC is $0.1°$. Each simulation point is averaged from 100 Monte-Carlo independent runs. We can see that when the input SNR is above 5dB our algorithm outperforms the 2-D MUSIC method.

4. CONCLUSION

We propose an adaptive 2-D direction finding framework to track multiple moving targets. The subspace tracking technique is applied to update the eigenbasis recursively on the arrival of a new data snapshot. Then a subspace fitting problem is formed to find the DOAs of each snapshot. We find that the costly multidimensional optimization problem can be solved by computing the solution of a set of linear equations. From the simulation results, it has been demonstrated that the proposed subspace-fitting-based algorithm is capable of tracking target DOAs with very good accuracy.

5. REFERENCES

[1] W. Li, "Cooperative arrayed wireless sensor networks," PhD Thesis, Imperial College London, Apr. 2011.

[2] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propagat.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[3] J. Zhuang, H. Xiong, W. Wang, and X. Cai "FFT-based Adaptive 2-D DOA Estimation for Arbitrary Array Structures," in *International Conference on Digital Signal Processing*, pp. 2229–2232, Aug. 2017.

[4] J. Zhuang, C. Duan, W. Wang and Z. Chen, "Joint Estimation of Azimuth and Elevation via Manifold Separation for Arbitrary Array Structures," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5585–5596, Jul. 2018.

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

[6] R. Badeau, B. David, and G. Richard, "Fast approximated power iteration subspace tracking," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2931–2941, Aug. 2005.

[7] M. Viberg, and B. Ottersten, "Sensor array processing based on subspace fitting," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1110–1121, May 1991.

[8] M. W. Trosset, "Extensions of classical multidimensional scaling: Computational theory", Department of Mathematics, College of William & Mary, www.convexoptimization.com/TOOLS/TrossetPITA.pdf, 2001.

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

[10] J. P. Rohrer, E. K. Cetinkaya, H. Narra, D. Broyles, K. Peters, and J.P.G. Sterbenz, "AeroRP performance in highly-dynamic airborne networks using 3D Gauss-Markov mobility model," in *Military communications confernece (MILCOM)*, pp. 834–841, Nov. 2011.