Reduced-Rank DOA Estimation Algorithms Based on Alternating Low-Rank Decomposition

Linzheng Qiu, Yunlong Cai, Rodrigo C. de Lamare, and Minjian Zhao

Abstract—In this work, we propose an alternating low-rank decomposition (ALRD) approach and novel subspace algorithms for direction-of-arrival (DOA) estimation. In the ALRD scheme, the decomposition matrix for rank reduction consists of a set of basis vectors. A low-rank auxiliary parameter vector is then employed to compute the output power spectrum. Alternating optimization strategies based on recursive least squares (RLS), denoted as ALRD-RLS and modified ALRD-RLS (MALRD-RLS), are devised to compute the basis vectors and the auxiliary parameter vector. Simulations for large sensor arrays with both uncorrelated and correlated sources are presented, showing that the proposed algorithms are superior to existing techniques.

Index Terms—DOA estimation, low-rank decomposition, parameter estimation.

I. INTRODUCTION

Array signal processing has been widely used in areas such as radar, sonar and wireless communications. Many applications related to array signal processing require the estimation of the direction-of-arrival (DOA) of the sources impinging on a sensor array [1]. Among the well-known DOA estimation schemes are the Capon method and subspace-based algorithms [2] such as Multiple-Signal Classification (MUSIC) [3] and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [4]. The Capon method calculates the output power spectrum for each scanning angle according to the constrained minimum variance (CMV) criterion. Then the estimated DOAs can be obtained by finding the peaks of the output power spectrum [5]. MUSIC, ESPRIT and their improved versions [6]–[11] estimate the DOAs by exploiting the signal and the noise subspaces of the signal correlation matrix. Due to the eigenvalue decomposition (EVD) and/or the singular-value decomposition (SVD), MUSIC and ESPRIT require a high computational cost, especially for large sensor arrays. The recently proposed subspace-based auxiliary vector (AV) [12], the conjugate gradient (CG) [13] and the joint iterative optimization (JIO) algorithms [14] employ basis vectors to build the signal subspace instead of the EVD or the SVD. However, the iterative construction of the basis vectors yields a complexity comparable to the EVD. Moreover, the AV and CG algorithms cannot provide a satisfactory performance for large sensor arrays with many sources.

L. Qiu, Y. Cai and M. Zhao are with the Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China, 310027. R. C. de Lamare is with CETUC, PUC-Rio Rua Marques de So Vicente, 22493-900 Rio de Janeiro, Brazil and University of York, Heslington, YO10 5DD, York, England, United Kingdom. Email: lqiu@zju.edu.cn, ylcai@zju.edu.cn, rcdl500@ohm.york.ac.uk, mjzhao@zju.edu.cn.

This work was supported in part by CNPq and FAPERJ, Brazil, the National Natural Science Foundation of China under Grants 61471319, Zhejiang Provincial Natural Science Foundation of China under Grant LY14F010013, the Fundamental Research Funds for the Central Universities, and the National High Technology Research and Development Program (863 Program) of China under Grant 2014AA01A707.

The performance of direction finding algorithms depends on the data record and the array size. Resorting to larger data records leads to higher estimation performance. However, large data records are not always available in wireless environments that change rapidly. Nowadays, large sensor arrays have gained importance for applications such as radar and future communication systems. Nevertheless, direction finding for large sensor arrays is associated with high computational costs. In this regard, the development of low-complexity DOA estimation algorithms for large sensor arrays and scenarios with short data records is an important research problem.

In this paper, we present an alternating low-rank decomposition (ALRD) approach for DOA estimation in large sensor arrays with a large number of sources. In the ALRD scheme, a subspace decomposition matrix which consists of a set of basis vectors and an auxiliary low-rank parameter vector are employed to compute the output power spectrum. In order to avoid matrix inversions, we develop recursive least squares (RLS) type algorithms [15] to compute the basis vectors and the auxiliary parameter vector, which reduces the computational complexity. The proposed DOA estimation algorithms are referred to as ALRD-RLS and modified ALRD-RLS (MALRD-RLS), which employs a single basis vector. Simulations show that the proposed ALRD-RLS and MALRD-RLS algorithms achieve superior performance to existing techniques for large arrays with short data records.

The paper is organized as follows. In Section II, we outline the system model and the problem of DOA estimation. The proposed ALRD scheme and algorithms are presented in Section III. In Section IV, we illustrate and discuss the simulation results. Finally, Section V concludes this work.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a uniform linear array (ULA) with $M$ omnidirectional sensor elements and suppose that $K$ narrowband source signals impinge on the ULA from directions $\theta_1, \theta_2, \ldots, \theta_K$, respectively, where $M$ is a large number with $K \ll M$. The $i$th snapshot of the received signal can be expressed by an $M \times 1$ vector as

$$
\mathbf{r}(i) = \sum_{k=1}^{K} \mathbf{a}(\theta_k) \mathbf{b}_k(i) + \mathbf{n}(i),
$$

where $\mathbf{b}_k(i)$ is the $k$th source signal with power $\sigma^2_b$. The vector $\mathbf{n}(i)$ is noise vector, which is assumed to be temporally and spatially white Gaussian with zero mean and variance $\sigma^2_n$. The array steering vector $\mathbf{a}(\theta_k)$ is defined as

$$
\mathbf{a}(\theta_k) = [1, e^{-2\pi j \frac{\lambda c}{\lambda c} \cos \theta_k}, \ldots, e^{-2\pi j (M-1) \frac{\lambda c}{\lambda c} \cos \theta_k}]^T,
$$

A low-rank parameter vector arises from the solution of a low-rank system of equations. This rank also corresponds to the dimension or length of the parameter vector.
where \((\cdot)^T\) denotes the transpose operation and \(\lambda_i\) is the signal wavelength. The parameter \(d_s = \frac{\lambda_i}{2}\) represents the array inter-element spacing. Direction finding algorithms aim to estimate the DOAs \(\theta = [\theta_1 \ldots \theta_K]^T\) by processing \(r(\cdot)\). The correlation matrix of \(r(\cdot)\) is given by

\[
\mathbf{R} = \mathbb{E}\{r(\cdot)r^H(\cdot)\} = \sum_{k=1}^{K} \mathbf{a}(\theta_k)\mathbf{R}_{b,k}\mathbf{a}^H(\theta_k) + \sigma_n^2\mathbf{I}_M, \tag{3}
\]

where \(\mathbb{E}\{\cdot\}\) denotes expectation, \((\cdot)^H\) is the Hermitian operator and \(\mathbf{I}_M\) is an identity matrix with dimension \(M\). \(\mathbf{R}_{b,k} = \mathbb{E}\{\mathbf{n}(\cdot)\mathbf{n}^H(\cdot)\}\) is the correlation matrix of the \(k\)th signal. \(\mathbf{R}_n = \mathbb{E}\{\mathbf{n}(\cdot)\mathbf{n}^H(\cdot)\} = \sigma_n^2\mathbf{I}_M\) is the correlation matrix of the noise vector. Note that the exact knowledge of \(\mathbf{R}\) is difficult to obtain, thus estimation by sample averages is employed in practice, which employs \(\hat{\mathbf{R}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{r}(i)\mathbf{r}^H(i)\), with \(N\) being the number of available snapshots.

III. PROPOSED ARLD SCHEME ARLD-RLS AND MALRD-RLS ALGORITHMS

In this section, we detail the proposed ARLD scheme and the ARLD-RLS and MALRD-RLS DOA estimation algorithms. The ARLD scheme divides the received vector into several segments and processes each segment with an individual basis vector. The basis vectors constitute the columns of the decomposition matrix, which performs dimensionality reduction. Then, a lower dimensional data vector is processed by the auxiliary parameter vector to construct the output power spectrum. The ARLD-RLS and the MALRD-RLS algorithms are based on an alternating optimization procedure of the basis vectors and the reduced-rank auxiliary parameter vector.

\[\begin{array}{c}
\text{Decomposition matrix } \mathbf{T}(i) \\
\mathbf{r}(i) \rightarrow \mathbf{T}(i) \rightarrow \overline{\mathbf{r}}(i) \rightarrow \mathbf{p}(0_1)
\end{array}\]

Fig. 1. Block diagram of the ARLD scheme

A. Proposed ARLD Scheme

The block diagram of the ARLD scheme is depicted in Fig. 1. The received vector \(\mathbf{r}(i) = [r_1(\cdot) \ldots r_{M-1}(\cdot)]^T\) is processed by an \(M \times D\) decomposition matrix \(\mathbf{T}(i)\), which is constructed by a set of \(I \times 1\) basis vectors \(\mathbf{s}_d(i)\), where \(d \in \{1, \ldots, D\}\). The \(D \times 1\) reduced-rank data vector can be expressed by

\[
\overline{\mathbf{r}}_D(i) = \mathbf{T}^H(i)\mathbf{r}(i) = \sum_{d=1}^{D} \mathbf{q}_d\mathbf{d}_d^H \mathcal{R}(i)\mathbf{s}_d^H(i), \tag{4}
\]

where \(\mathbf{q}_d\) is a \(D \times 1\) vector with a one in the \(d\)th position and zeros elsewhere, \(\mathbf{d}_d\) is the \(M \times 1\) selection vector to divide \(\mathbf{r}(i)\) into \(D\) segments, which are defined as:

\[
\mathbf{d}_d = [0 \ldots 0, 1, 0 \ldots 0]^T, \quad \mu_d \text{ zeros, } \quad M-\mu_d-1 \text{ zeros,} \tag{5}
\]

where \(\mu_d\) is the selection pattern chosen as \(\mu_d = (d-1)[\frac{M}{D}]\). Then the \(d\)th column of \(\mathbf{T}(i)\) can be described as

\[
\mathbf{t}_d(i) = [0 \ldots 0, \mathbf{s}_d^T(i), 0 \ldots 0]^T. \tag{6}
\]

The \(M \times I\) matrix \(\mathcal{R}(i)\) with the samples of \(\mathbf{r}(i)\) has a Hankel structure [16], which is described by

\[
\mathcal{R}(i) = \left(\begin{array}{cccc}
\mathbf{r}_0(i) & \mathbf{r}_1(i) & \cdots & \mathbf{r}_{M-1}(i) \\
\mathbf{r}_{M-1}(i) & \mathbf{r}_{M-1+1}(i) & \cdots & \mathbf{r}_{M-2}(i) \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{r}_{M-1}(i) & 0 & \cdots & 0
\end{array}\right). \tag{7}
\]

After dimensionality reduction, \(\overline{\mathbf{r}}_D(i)\) is processed by a \(D \times 1\) auxiliary parameter vector \(\vec{\omega}(i)\) to compute the output power spectrum. As seen from Fig. 1, \(\mathbf{s}_d(i)\) and \(\vec{\omega}(i)\) are alternately optimized according to a prescribed criterion, which is introduced in what follows.

B. Proposed ARLD-RLS DOA Estimation Algorithm

The ARLD-RLS algorithm based on the CMV criterion solves the optimization problem:

\[
\begin{aligned}
\min_{\vec{\omega}(i)} & \quad \sum_{l=1}^{I} \alpha^{I-l} |\vec{\omega}^H(i)\sum_{d=1}^{D} \mathbf{q}_d\mathbf{d}_d^H \mathcal{R}(l)\mathbf{s}_d(i)|^2 \\
\text{subject to } & \quad \vec{\omega}^H(i) \sum_{d=1}^{D} \mathbf{q}_d\mathbf{d}_d^H \mathcal{A}_n\mathbf{s}_d(i) = 1
\end{aligned} \tag{8}
\]

where \(\alpha\) is a forgetting factor close to but smaller than 1 [15]. \(\mathcal{A}_n\) is the \(M \times I\) Hankel matrix of the scanning steering vector \(\mathbf{a}(\theta_n) = [a_0(\theta_n) \ldots a_{M-1}(\theta_n)]^T\), which is given by

\[
\mathcal{A}_n(i) = \left(\begin{array}{cccc}
a_{M-1}(\theta_n) & a_{M-1+1}(\theta_n) & \cdots & a_{M-2}(\theta_n) \\
a_{M-1+1}(\theta_n) & a_{M-1+2}(\theta_n) & \cdots & a_{M-3}(\theta_n) \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0
\end{array}\right). \tag{9}
\]

The optimization problem in (8) can be solved by the method of Lagrange multipliers whose Lagrangian is described by

\[
\mathcal{L}(i) = \sum_{l=1}^{I} \alpha^{I-l} \left|\vec{\omega}^H(i) \sum_{d=1}^{D} \mathbf{q}_d\mathbf{d}_d^H \mathcal{R}(l)\mathbf{s}_d(i)\right|^2 + 2\Re\{\lambda |\vec{\omega}^H(i) \sum_{d=1}^{D} \mathbf{q}_d\mathbf{d}_d^H \mathcal{A}_n\mathbf{s}_d(i) - 1\}, \tag{10}
\]

where \(\Re\{\cdot\}\) selects the real part of the argument. By taking the gradient of (10) with respect to \(\mathbf{s}_d(i)\), we obtain

\[
\frac{\partial \mathcal{L}(i)}{\partial \mathbf{s}_d(i)} = \sum_{l=1}^{I} \alpha^{I-l} \mathcal{R}^H(l)\mathbf{d}_d\mathbf{q}_d^H \vec{\omega}(i)\vec{\omega}^H(i)\mathbf{q}_d\mathbf{d}_d^H \mathcal{R}(l)\mathbf{s}_d(i) \\
+ \sum_{l=1}^{I} \alpha^{I-l} \mathcal{R}^H(l)\mathbf{d}_d\mathbf{q}_d^H \vec{\omega}(i)\vec{\omega}^H(i)\sum_{j \neq d}^{D} \mathbf{q}_j\mathbf{d}_j^H \mathcal{R}(l)\mathbf{s}_j(i) \\
+ \lambda^* \mathcal{A}_n^H\mathbf{d}_d\mathbf{q}_d^H \vec{\omega}(i). \tag{11}
\]

By equating (11) to zero and solving for \(\mathbf{s}_d(i)\), we have

\[
\mathbf{s}_d(i) = -\mathbf{R}_s^{-1}(i) \sum_{l=1}^{I} \lambda \mathbf{R}_s^{-1}(i) \mathcal{A}_n^T \mathbf{d}_d\mathbf{q}_d^H \vec{\omega}(i). \tag{12}
\]

where
\[ R_{s_d}(i) = \sum_{l=1}^{i} \alpha^{i-l} R^{T}(l) d_d q_d^H \bar{\omega}^T(i) q_d^H R^{*}(l), \]

\[ P_{s_d}(i) = \sum_{l=1}^{i} \alpha^{i-l} R^{T}(l) d_d q_d^H \bar{\omega}^T(i) q_d^H R^{*}(l) s_d(i). \]

Substituting (12) into (8), we obtain the Lagrange multiplier:

\[ \frac{\partial}{\partial \omega} \lambda = 1 - \prod_{i \neq d}^{D} \frac{P_{s_d}(i)}{P_{s_d}(i)} \prod_{i \neq d}^{D} (i) \prod_{i \neq d}^{D} (i), \]

where \( \prod_{i} = q_d^H \bar{\omega}(i) d_d^H A_5^s(i, i-1). \) Based on (12) and (15), we obtain the \( d \)th basis vector \( s_d(i) \).

Next, we consider the update of \( R_{s_d}^{-1}(i) \). By applying the matrix inversion lemma [15] to (13), we obtain

\[ g_{s_d}(i) = \frac{R_{s_d}^{-1}(i-1) R^{T}(i) d_d}{\alpha + P_{s_d}(i)} \]

\[ R_{s_d}^{-1}(i) = \alpha R_{s_d}^{-1}(i-1) - \alpha^{-2} g_{s_d}(i) d_d^H \bar{\omega}^T(i) q_d^H R^{*}(i) s_d(i). \]

By employing (12)-(18), we can update \( s_d(i) \) for \( d \in \{1, \ldots, D\} \). Given the values of \( s_d(i) \), we can compute \( \bar{\omega}(i) \). Defining \( \bar{a}(i) = \sum_{d=1}^{D} q_d^H A_5 s_d(i) \) and

\[ \bar{r}(i) = \sum_{d=1}^{D} q_d^H R \bar{a}(i), \]

we can modify as

\[ \min_{\bar{\omega}(i)} \sum_{i=1}^{D} \alpha^{-1} |H(i) \bar{r}(i)|^2 \text{ subject to } \bar{\omega}(i) \bar{a}(i) = 1. \]

Solving for \( \bar{\omega}(i) \), we have

\[ g_D(i) = \frac{R_{D}^{-1}(i-1) \bar{r}(i)}{\alpha + \bar{r}(i)^H \bar{r}(i)} \]

\[ R_{D}^{-1}(i) = \alpha^{-1} R_{D}^{-1}(i-1) - \alpha^{-2} g_D(i) R^{H}(i) R_{D}^{-1}(i-1), \]

\[ \bar{\omega}(i) = \frac{R_{D}^{-1}(i) \bar{a}(i)}{\alpha + \bar{r}(i)^H \bar{r}(i)} R_{D}^{-1}(i) \bar{a}(i). \]

Next, we discuss the update of \( \bar{\omega}(i) \). By redefining \( \bar{a}(i) = Q \bar{A}_0 s^*(i) \), the cost function for the update of \( \bar{\omega}(i) \) is the same as that in (19). Hence \( \bar{\omega}(i) \) can also be constructed by (20)-(22) in the MALRD-RLS algorithm.

### Table I

<table>
<thead>
<tr>
<th>The MALRD-RLS DOA estimation algorithm.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Set ( N ) and ( \alpha )</td>
</tr>
<tr>
<td>2. For each scanning angle ( \theta_n ) do</td>
</tr>
<tr>
<td>3. Initialize ( R_{\theta_n}^{-1}(0), q(0), \bar{\omega}(0) )</td>
</tr>
<tr>
<td>4. For each snapshot ( i = 1, \ldots, N ) do</td>
</tr>
<tr>
<td>5. Update ( s(i) ) based on (26)-(28)</td>
</tr>
<tr>
<td>6. Update ( \bar{\omega}(i) ) based on (20)-(22)</td>
</tr>
<tr>
<td>7. Calculate the output power ( P(\theta_n) = (\bar{\omega}(N) R_{\theta_n}^{-1}(N) \bar{a}(N))^{-1} )</td>
</tr>
<tr>
<td>8. Estimate the DOA ( \theta = \arg \max_{\theta_n} P(\theta_n) )</td>
</tr>
</tbody>
</table>

After the update of \( s(i) \) and \( \bar{\omega}(i) \), we calculate the output power spectrum based on (23). The peaks of the power spectrum are the estimated DOAs. A brief summary of the MALRD-RLS algorithm is illustrated in Table II.
D. Computational Complexity

Here we detail the computational complexity of the proposed ALRD-RLS and MALRD-RLS algorithms and several existing DOA estimation algorithms. ESPRIT uses an EVD of $\mathbf{R}$, which has complexity of $O(M^3)$. MUSIC employs both the EVD and grid search, resulting in a cost of $O(M^3 + (180/\Delta)M^2)$, with $\Delta$ being the search step. Matrix inversions and grid searches are essential for Capon, whose complexity is $O(M^3 + (180/\Delta)M^2)$. For the AV and CG algorithms, the construction of the basis vectors leads to a complexity which is higher than that of the ESPRIT algorithm [12] [13]. The JIOL-RS algorithm has a cost of $O((180/\Delta)N(M^2 + D^2))$, with $D$ being the length of the reduced-rank received vector. ALRD-RLS avoids the EVD, the matrix inversion and the construction of the transformation matrix, and the update of $D$ basis vectors and an auxiliary parameter vector requires $O((180/\Delta)N(DI^2 + D^2))$. MALRD-RLS only uses one basis vector and costs $O((180/\Delta)N(I^2 + D^2))$. The computational complexity of the analyzed algorithms is depicted in Table III. Even in a large sensor array, $I$ and $D$ are small numbers, with $I \ll M$ and $D \ll M$, the cost of ALRD-RLS and MALRD-RLS can be less than those of the existing algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESPRIT [4]</td>
<td>$O(M^3)$</td>
</tr>
<tr>
<td>MUSIC [3]</td>
<td>$O(M^3 + (180/\Delta)M^2)$</td>
</tr>
<tr>
<td>Capon [5]</td>
<td>$O(M^3 + (180/\Delta)M^2)$</td>
</tr>
<tr>
<td>AV [12]</td>
<td>$O((180/\Delta)KM^2)$</td>
</tr>
<tr>
<td>CG [13]</td>
<td>$O((180/\Delta)KM^2)$</td>
</tr>
<tr>
<td>JIOL-RS [14]</td>
<td>$O((180/\Delta)N(M^2 + D^2))$</td>
</tr>
<tr>
<td>ALRD-RLS</td>
<td>$O((180/\Delta)N(DI^2 + D^2))$</td>
</tr>
<tr>
<td>MALRD-RLS</td>
<td>$O((180/\Delta)N(I^2 + D^2))$</td>
</tr>
</tbody>
</table>

IV. SIMULATIONS

In this section, we evaluate the ALRD-RLS and MALRD-RLS algorithms through simulations. A ULA with $M = 60$ elements is adopted in the experiments. $K = 15$ narrowband source signals impinge on the ULA from directions $\theta_k, k \in \{1, \cdots, K\}$, with 2 of them being correlated and the others uncorrelated. The separation between $\theta_k$ and $\theta_{k+1}$ is assumed to be $3^\circ$. The correlated sources are generated as follows:

$$b_1 \sim \mathcal{N}(0, \sigma_b^2) \quad \text{and} \quad b_2(i) = \rho b_1(i) + \sqrt{1 - \rho^2} e(i), \quad (29)$$

where $e \sim \mathcal{N}(0, \sigma_e^2)$, $\rho$ is the correlation coefficient fixed as 0.7 in this work. We assume that a small number of snapshots are available and set $N = 20$ in the simulations. The source signals are modulated by a binary phase shift keying (BPSK) scheme, 13 out of 15 sources have powers $\sigma_b^2 = 1$, one with power $\sigma_b^2 = 2$, another with power $\sigma_b^2 = 4$. The search step is chosen as $0.3^\circ$ for the algorithms based on grid search. In each experiment, $L = 100$ independent Monte Carlo runs are conducted to obtain the curves.

In the first experiment we assess the effects of $D$ and $I$ in terms of root mean square error (RMSE) performance of the proposed algorithms, which is calculated as

$$RMSE = \sqrt{\frac{1}{L/M} \sum_{l=1}^{L} \sum_{k=1}^{K} (\hat{\theta}_{k,l} - \theta_k)^2}.$$ 

We investigate the RMSE of the ALRD-RLS and MALRD-RLS algorithms with an input signal-to-noise ratio (SNR) of 5dB. Simulation result in Fig. 2 shows that the performance gets better for both the ALRD-RLS and MALRD-RLS algorithms as $D$ and $I$ become larger. Nevertheless, when $D$ is greater than 7, the performance gain becomes negligible, especially for MALRD-RLS. Considering that larger $D$ and $I$ lead to a higher computational cost, we choose $I = 12$, $D = 5$ for both algorithms to achieve an attractive tradeoff between performance and complexity.

![Fig. 2. RMSE versus $I$ for different $D$s for proposed algorithms. (a) ALRD-RLS (b) MALRD-RLS](image)

We then evaluate the RMSE performance of the analyzed algorithms. We compare ALRD-RLS and MALRD-RLS with MUSIC, ESPRIT, Capon, CG, AV and the JIOL-RS algorithms. We set the parameters for JIO-RLS to $D = 5$ and $\alpha = 0.998$. For MALRD-RLS and ALRD-RLS, we choose $I = 12$, $D = 5$, $\alpha = 0.998$. From Fig. 3, MALRD-RLS provides a superior RMSE performance with the lowest threshold SNR and the lowest RMSE level in high SNRs, followed by ALRD-RLS, MUSIC, JIOL-RS, Capon and ESPRIT. The AV and CG algorithms present poor performance for large-scale sensor arrays. Note that MUSIC, ESPRIT, Capon, AV and CG require forward backward averaging (FBA) [17], [18] to ensure satisfactory performance for correlated signals.

![Fig. 3. RMSE versus input SNR.](image)

V. CONCLUSION

In this paper, we have proposed the ALRD scheme and the ALRD-RLS and MALRD-RLS subspace DOA estimation algorithms based on alternating optimization. The proposed algorithms are suitable for large sensor arrays and have a lower computational cost than existing techniques. Simulation results show that MALRD-RLS and ALRD-RLS outperform previously reported algorithms.
REFERENCES


