Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Cyclic block coordinate minimization algorithms for DOA estimation in co-prime arrays

Heeseong Yang[a,*], Joohwan Chun[b], Haris Vikalo[c]

[a] Agency for Defense Development, 488 Bugyuseong Daero, Yuseong, Daejeon, 34060, Republic of Korea
[b] School of Electrical Engineering, KAIST, 291 Daehak-Ro, Yuseong-Gu, Daejeon, Republic of Korea
[c] Department of Electrical and Computer Engineering, The University of Texas at Austin, 1 University Station C0803, Austin, TX, 78712-0240, USA

ARTICLE INFO

Article history:
Received 9 May 2017
Revised 20 November 2017
Accepted 2 December 2017
Available online 6 December 2017

Keywords:
DOA estimation
Cyclic block coordinate minimization
Gridless compressive sensing
Atomic norm
Co-prime arrays

ABSTRACT

We derive several closed-form expressions that generalize co-prime array system model and study a nonnegative gridless compressive sensing formulation of the problem of estimating direction-of-arrival (DOA) based on the derived model. To solve the problem, two computationally efficient cyclic block coordinate minimization algorithms are proposed; the algorithms perform atomic norm minimization of an objective function through a sequence of computationally efficient atom merging and atom activation steps conducted in subdomains of a continuous atom search space. The convergence properties of the developed algorithms are analyzed. Numerical simulations demonstrate that the proposed techniques outperform the joint sparsity reconstruction method (JLASSO) and the ESPRIT method with spatial smoothing (SS-ESPRIT) in terms of various criteria. It is also demonstrated that our methods are significantly faster and yield competitive performance in terms of root mean square error (RMSE), detection probability, and false alarms when compared to the recent convex optimization based methods, i.e. the gridless SPICE with ESPRIT (GLS-ESPRIT), the atomic norm minimization with dimension reduction and ESPRIT (ANM-ESPRIT), and the nuclear norm minimization with ESPRIT (NNM-ESPRIT).

1. Introduction

The problem of estimating direction-of-arrival (DOA) of the signals impinging on an array of sensors is encountered in a number of applications in radar, sonar, wireless communications, and other areas [1–5]. Source identifiability and DOA accuracy improve as the number of sensors grows [6]; however, increasing the number of sensors leads to higher cost and hardware complexity of the DOA estimation scheme. It is thus desirable to enable identifiability and accuracy of a DOA estimation scheme with as small number of sensors as possible. A fundamental limitation of the conventional uniform linear array (ULA) with $O(M_1+M_2)$ sensors is that it can identify no more than $O(M_1+M_2)$ sources. To overcome the limitations of ULAs, significant research efforts have been undertaken to develop non-uniform array structures [7–16]. Among those, co-prime arrays[1] have attracted significant attention due to their simplicity and efficacy (for details, see [11–16] and the references therein).

Rather than directly using raw signal measurements, a co-prime array system typically performs DOA estimation by relying on a single-snapshot of a virtual signal obtained after vectorizing the sample covariance matrix of multiple-snapshot raw measurements. In this framework, even when the number of sensors is smaller than that of the sources, the number of measurements of the virtual array may be sufficiently high to enable accurate identification of the sources. In particular, DOA estimation in co-prime arrays readily leads to compressive sensing problems [14–16] of the form

$$\min_{u_1,\ldots,u_N\in\boldsymbol{\Omega}} \min_{x_1,\ldots,x_N\geq 0} \frac{1}{2}\|\mathbf{r}-\mathbf{A}(\mathbf{u})\mathbf{x}\|_2^2+\lambda_t\sum_{k=1}^{N} x_k, \tag{1}$$

where $\mathbf{A}(\mathbf{u})$ denotes an $M\times N$ (complex-valued) matrix with the kth column (or the kth atom) $\mathbf{a}_k(u_k)=[e^{j\pi p_1 u_k},\ldots,e^{j\pi p_M u_k}]^T$ for $k=1,\ldots,N$, $p_1,\ldots,p_M$ are integer valued, $\mathbf{u}=[u_1,\ldots,u_N]^T$ (the components of $\mathbf{u}$ belong to a common bounded range $\boldsymbol{\Omega}$), $\mathbf{r}$ is a given $M\times 1$ noisy measurement vector, λ_t is constrained to be positive, and $\mathbf{x}=[x_1,\ldots,x_N]^T$ denotes a nonnegative sparse vector consisting of $K<N$ positive elements and $N-K$ zero elements. (We postpone the discussion on the connection between the variables in (1) and the parameters of DOA estimation in co-prime arrays to Section 2.) The problem (1) is encountered in a number of ap-

[*] Corresponding author.
E-mail address: hsyang@add.re.kr (H. Yang).

[1] As discussed in Section 2, the structure of co-prime arrays enables resolution of $O(M_1M_2)$ sources using only $O(M_1+M_2)$ sensors.

https://doi.org/10.1016/j.sigpro.2017.12.005

plications and has been intensively studied in the settings where $\mathbf{u}$ (and therefore $\mathbf{A}(\mathbf{u})$) is either known or restricted to a discrete set of possible values [17–20]. However, when restricting generally continuous parameters to a finite predefined grid, performance of compressive sensing is typically affected by the grid mismatch [21]. If one needs to estimate both the sparse vector $\mathbf{x}$ and the vector of parameters $\mathbf{u}$ in order to lead to accurate DOA estimation with no grid mismatch, optimization in (1) can be referred to as gridless[2] compressive sensing or the super-resolution problem [22–24]. In [23,24], it was shown that the general total variation minimization for super resolution problems can be reformulated as a computationally costly convex optimization followed by sequential root-finding to calculate a form of Fourier matrix. This idea was extended and applied to DOA estimation in co-prime arrays in [15].

Irrespective of whether raw signal measurements need to be manipulated before being used to formulate an optimization problem, many researches on gridless compressive sensing have been carried [25–31]. Sparse Bayesian inference/learning based DOA estimation algorithms were developed in [25,26]. They took into account similar but different approaches to overcome the grid mismatch; in [25] the array manifold matrix in the signal model was modified by using the first-order Taylor expansion and in [26] by using the linear interpolation. In [27–29], the convex optimization based methods for structured matrix recovery, e.g. the atomic norm minimization (ANM), the nuclear norm minimization (NNM), and the gridless sparse iterative covariance-based estimation (GLS), were proposed. ANM, NNM, and GLS were developed, particularly, in the literature on Hermitian Toeplitz covariance matrix recovery followed by a spectral estimation method such as MUSIC and ESPRIT [32]. Interestingly, estimating a Fourier matrix in [23,24] can be also categorized into structured matrix recovery. For a more comprehensive review of gridless compressive sensing, we recommend the articles by Yang and Xie [30] and Yang et al. [31].

Recently, [33] searched for practically feasible means of performing atomic decomposition [27,34] and in this context considered the line spectral estimation variant of (1); that work addressed grid mismatch and proposed a technique for atomic norm denoising based on a modified cyclic block coordinate minimization (CBCM) algorithm [35–37] which, if convergent, finds a solution satisfying the Karush–Kuhn–Tucker (KKT) optimality condition. A drawback of the method in [33] is the need to perform an a priori unknown number of correction steps wherein a judiciously selected element of the estimate of $\mathbf{x}$ is set to zero and the cost function is appropriately adjusted; the CBCM procedure is then run anew with $\mathbf{u}$ and $\mathbf{x}$ re-initialized as their current estimates.

In this paper, in order to solve the problem (1) (and therefore the DOA estimation problem in co-prime array systems), we propose two practical algorithms based on CBCM that do not require the correction steps of [33]. We refer to the first of these two algorithms as successive-atom-merging cyclic block coordinate minimization for atomic norm minimization (SAM-CBCMA), and to the second one as successive-atom-activating cyclic block coordinate minimization for atomic norm minimization (SAA-CBCMA). As their names suggest, the algorithms take different approaches to atom selection; while the SAM-CBCMA starts with superfluous atoms and repeatedly performs a step in which redundant atoms are merged and hence the number of parameters is reduced, the SAA-CBCMA starts with no atoms and repeatedly activates them in a greedy fashion. An important common feature of the two proposed algorithms is that they divide the bounded range $\mathbf{\Omega}$ and refine atom selection by performing optimizations in subdomains of $\mathbf{\Omega}$. This enables reduction of the computational complexity as compared to strategies performing atom refinement on the entire $\mathbf{\Omega}$. We observe that when the subdomains are appropriately constructed, the KKT condition of [33] is not violated.

The contributions of this paper can be summarized as follows:

- Two methods (SAM-CBCMA and SAA-CBCMA) for the nonnegative gridless compressive sensing formulation of DOA estimation in co-prime arrays are proposed.
- Guarantees that sequences of estimates generated by the proposed methods converge to a stationary point of the atomic norm minimization cost function are established.
- Closed-form expressions that generalize the co-prime array system model [11,12] are derived.
- Extensive benchmarking results are provided comparing the proposed methods with the various previous methods[3]: the joint sparsity reconstruction method (JLASSO) [16], the GLS with ESPRIT (GLS-ESPRIT) [29], the ANM with dimension reduction[4] and ESPRIT (ANM-ESPRIT) [27], the NNM with ESPRIT (NNM-ESPRIT) [28], and the ESPRIT method with spatial smoothing (SS-ESPRIT) [12].

The remainder of the paper is organized as follows. In Section 2, we introduce a co-prime array system model and its corresponding measurement model for DOA estimation. In Section 3, we provide detailed descriptions of the SAM-CBCMA and SAA-CBCMA algorithms and prove their convergence. In Section 4 we present results of a computational study of the co-prime array described in Section 2, demonstrating the superiorities of the proposed algorithms: they are much faster than the benchmarking algorithms except SS-ESPRIT; outperform JLASSO and SS-ESPRIT in terms of several criteria – specifically, root mean square error (RMSE), detection probability (P_d), false alarm count 1 ($P_{f,\,1}$), false alarm count 2 ($P_{f,\,2}$); and are comparable with the recent gridless sparse methods using computationally heavy convex optimization, i.e. GLS-ESPRIT, ANM-ESPRIT, and NNM-ESPRIT, in respect to RMSE, P_d, and false alarms. Moreover, we study convergence of the proposed algorithms. Finally, we conclude the paper in Section 5.

Throughout the paper, the notations $(\cdot)^T$ and $(\cdot)^H$ denote the transposition and conjugate transposition operators, respectively. $(\cdot)^*$ is the conjugate operator. $|\cdot|$ is the cardinality of a set, or the absolute value of a scalar. $\mathrm{Re}\{\cdot\}$ is the real part of a complex number. Unless otherwise noted, bold characters denote column vectors. $[\cdot]_m$ is the mth element of the vector in square blankets. $E\{\cdot\}$ is the expectation operator. $\hat{c}$ (or $\hat{\mathbf{c}}$) is the estimate of a scalar c (or a vector $\mathbf{c}$). $diag\{\cdot\}$ denotes a diagonal matrix whose diagonal entries are the elements of the vector in braces. The superscript of $(\cdot)^{(i)}$ indicates the ith iteration. $\otimes$ denotes the Kronecker product. $\circ$ is the Khatri-Rao product. $\mathbf{e}_k$ is the unit base vector having all zero elements except the kth element which is equal to one. $\lceil x \rceil$ and $\lfloor x \rfloor$ denote the least integer that is greater than or equal to x and the greatest integer that is less than or equal to x, respectively.

Preliminary results of this work were reported in [39].

[2] The word "gridless" implies that the parameters belong to an interval on a continuous line.

[3] The method proposed in [15] seems to be very sensitive to the choice of hyperparameters and for that reason is not included in the benchmarking study.

[4] In this paper, we adopt the dimension reduction scheme proposed in [38]. This enables us to reduce a large number of raw measurements to a much smaller number of effective measurements without deteriorating the final solution.

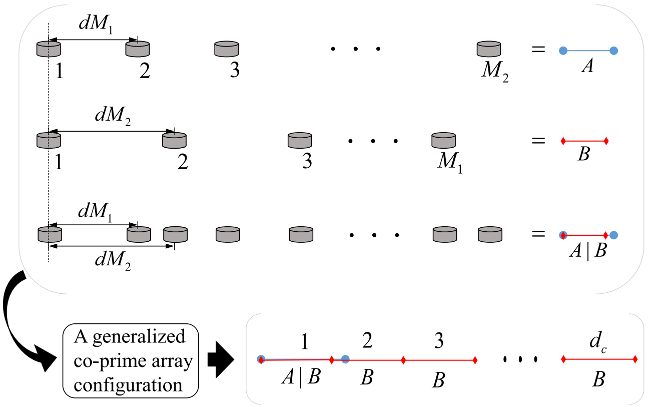


Fig. 1. A generalized co-prime array configuration.

2. Background on co-prime arrays and the formulation of the DOA problem

Consider a 1-D co-prime array where the co-prime numbers are M_1 and M_2 ($M_1 < M_2$) as illustrated in Fig. 1. Following the setting of [11,12], the co-prime array consists of two linear arrays that have M_2 sensors with their location set $\{ndM_1 | n = 0, 1, \ldots, M_2 - 1\}$ and d_cM_1 sensors with their location set $\{ndM_2 | n = 0, 1, \ldots, d_cM_1 - 1\}$, where d_c is a positive integer. Let us denote the set of unique differences between all possible pairs of elements of these two sets by P_D^S. Unlike in [11,12], the described setting does not restrict d_c to be 1 or 2. Moreover, the described setting is clearly different from the CACIS and CADiS presented in [13]. In this section, our main focus is on finding closed-form expressions for the elements of P_D^S and using those expressions to construct an effective measurement model by selecting nonredundant elements of the vectorized covariance matrix in a co-prime array.

For an illustration, suppose that sensor spacing d is one. After denoting the largest element of P_D^S by $M_{\max}$ we can write

$$M_{\max} = \begin{cases} M_1(M_2 - 1) & \text{if } d_c = 1, \\ M_2(d_cM_1 - 1) & \text{if } d_c > 1. \end{cases} \tag{2}$$

P_D^S can be thought of as a collection of integer grid points between $-M_{\max}$ and $M_{\max}$ where virtual sensors are located. The integers between $-M_{\max}$ and $M_{\max}$ not included in P_D^S are associated with the locations that are referred to as “holes”, i.e., those not associated with virtual sensors; therefore, P_D^S specifies the virtual array configuration and, thus, the effective measurement model for a co-prime array. To clarify this, let M_C denote the largest positive integer such that the consecutive integers between $-M_C$ and M_C do not include the positions of holes. Since M_1 and M_2 are co-prime, M_C can be found as

$$M_C = \begin{cases} M_1 + M_2 - 1 & \text{if } d_c = 1, \\ (d_c - 1)M_1M_2 + M_1 - 1 & \text{if } d_c > 1. \end{cases} \tag{3}$$

Clearly, the smallest positive number indicating the position of a hole is $M_C + 1$. Furthermore, all positive values indicating the positions of holes should satisfy condition

$$M_C + 1 + m_1M_1 + m_2M_2 < M_{\max}, \tag{4}$$

where m_1 and m_2 are nonnegative integers. For given m_1 and m_2, the left-hand side of the inequality in (4) corresponds to a position of a hole on the positive grid points. For convenience, we denote the positions of holes by

$$\begin{matrix} m_1^+M_1 + m_2^+M_2 & \text{if } d_c = 1, \\ (d_c - 1)M_1M_2 - M_2 + m_1^+M_1 + m_2^+M_2 & \text{if } d_c > 1. \end{matrix}$$

where m_1^+ and m_2^+ are positive integers. To find explicit expressions for the positions and the total number of holes, we only need to consider the two possible scenarios: (a) $d_c = 1$ and (b) $d_c > 1$. In particular:

(a) When $d_c = 1$, (4) is equal to $m_1^+M_1 + m_2^+M_2 < M_1(M_2 - 1)$ and, by mathematical induction, the range of m_2^+ for $m_1^+ = 1, 2, \ldots, \lceil M_2(\frac{M_1-1}{M_1}) - 1\rceil$ is calculated as

$$1 \le m_2^+ \le \left\lfloor \frac{M_1\left(M_2 - \left(1 + m_1^+\right)\right)}{M_2} \right\rfloor.$$

The total number of holes on the positive grid is

$$\sum_{k=1}^{\left\lceil M_2\left(\frac{M_1-1}{M_1}\right)-1\right\rceil} \left\lfloor \frac{M_1(M_2 - (1 + k))}{M_2} \right\rfloor.$$

(b) When $d_c > 1$, (4) is equal to $n_1^+M_1 + n_2^+M_2 < M_1M_2$ and, by mathematical induction, the range of m_2^+ for $m_1^+ = 1, 2, \ldots, \lceil M_2 - \frac{M_2}{M_1}\rceil$ is calculated as

$$1 \le m_2^+ \le \left\lfloor M_1 - \frac{m_1^+M_1}{M_2} \right\rfloor.$$

The total number of holes on the positive grid is

$$\sum_{k=1}^{\left\lceil M_2 - \frac{M_2}{M_1}\right\rceil} \left\lfloor M_1 - \frac{kM_1}{M_2} \right\rfloor.$$

Positions of holes on the negative grid are characterized by indices obtained by changing the sign of the indices of the holes on the positive grid. Following (a) and (b), all elements of P_D^S are known and scalable by a positive value d.

Let us consider a co-prime array system receiving K plane waves, i.e., there are K sources, incident from directions $\varphi_1, \ldots, \varphi_K$ (in radians), and denote the complex amplitude of the k^{th} plane wave and the noise of the l^{th} sensor by α_k and n_l, respectively. Then the received baseband measurement vector $\mathbf{y}$ is given by

$$\mathbf{y} = \mathbf{B}(\boldsymbol{\varphi})\boldsymbol{\alpha} + \mathbf{n}, \tag{5}$$

where $\mathbf{B}(\boldsymbol{\varphi})$ is a $(d_cM_1 + M_2 - 1) \times K$ array manifold matrix whose k^{th} column corresponds to φ_k, $\boldsymbol{\varphi} = [\varphi_1, \ldots, \varphi_K]^T$, $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_K]^T$, and $\mathbf{n} = [n_1, \ldots, n_{d_cM_1+M_2-1}]^T$. Assume that α_k and n_l are independent, zero-mean random variables. Then, the covariance matrix of $\mathbf{y}$ is given by

$$\mathbf{R_y} = E\left\{\mathbf{y}\mathbf{y}^H\right\} = \mathbf{B}(\boldsymbol{\varphi})\mathbf{R}_{\boldsymbol{\alpha}}\mathbf{B}^H(\boldsymbol{\varphi}) + \mathbf{R_n}, \tag{6}$$

where

$$\begin{aligned} \mathbf{R}_{\boldsymbol{\alpha}} &= E\left\{\boldsymbol{\alpha}\boldsymbol{\alpha}^H\right\} = diag\{\mathbf{s}\}, \\ \mathbf{R_n} &= E\left\{\mathbf{n}\mathbf{n}^H\right\} = diag\{\mathbf{s_n}\}, \\ \mathbf{s} &= [E\{\alpha_1\alpha_1^*\}, \ldots, E\{\alpha_K\alpha_K^*\}]^T, \\ \mathbf{s_n} &= \left[E\{n_1n_1^*\}, \ldots, E\left\{n_{d_cM_1+M_2-1}n_{d_cM_1+M_2-1}^*\right\}\right]^T. \end{aligned}$$

The vectorization of (6) yields

$$\mathbf{r}_{vec} = vec\{\mathbf{R_y}\} = (\mathbf{B}^*(\boldsymbol{\varphi}) \circ \mathbf{B}(\boldsymbol{\varphi}))\mathbf{s} + (\mathbf{I} \circ \mathbf{I})\mathbf{s_n}. \tag{7}$$

Let us denote the elements of P_D^S after having them sorted in ascending order by dp_m, $m = 1, \ldots, M$, where $M = \left|P_D^S\right|$ and p_m is an integer. Assuming that $d = \lambda/2$ where λ is the wavelength, every element of the kth column of $\mathbf{B}^*(\boldsymbol{\varphi}) \circ \mathbf{B}(\boldsymbol{\varphi})$ becomes one of $e^{j\pi p_1\varphi_k}, \ldots, e^{j\pi p_M\varphi_k}$.[5] We further refine $\mathbf{r}_{vec}$ by combining and

[5] p_1 and p_M are $-M_{\max}$ and $M_{\max}$, respectively.

averaging its elements that correspond to the identical rows of $\mathbf{B}^*(\boldsymbol{\varphi}) \circ \mathbf{B}(\boldsymbol{\varphi})$. The resulting $M \times 1$ processed measurement vector $\mathbf{r}$ in the co-prime array framework is of the form

$$\mathbf{r} = \mathbf{A}_e(\boldsymbol{\varphi})\mathbf{s} + s_{K+1}\mathbf{e}_{(M+1)/2} + \boldsymbol{\varepsilon}, \tag{8}$$

where $\mathbf{A}_e(\boldsymbol{\varphi})$ is the $M \times K$ matrix with the kth column $\mathbf{a}(\varphi_k) = [e^{j\pi p_1 \varphi_k}, \ldots, e^{j\pi p_M \varphi_k}]^T$, $\boldsymbol{\varepsilon}$ is a residual vector due to using a sample covariance matrix instead of the true one, $\mathbf{R_y}$, and $s_{K+1} = \frac{1}{d_c M_1 + M_2 - 1} \sum_{l=1}^{d_c M_1 + M_2 - 1} E\left\{n_l^* n_l\right\}$. Note that the number of rows of $\mathbf{A}_e(\boldsymbol{\varphi})$ is much larger than that of $\mathbf{B}(\boldsymbol{\varphi})$ in (5) and thus solving

$$\min_{u_1,\ldots,u_N} \min_{x_1,\ldots,x_N \geq 0} \frac{1}{2} \left\| \mathbf{r} - s_{K+1}\mathbf{e}_{(M+1)/2} - \mathbf{A}(\mathbf{u})\mathbf{x} \right\|_2^2 + \lambda_t \sum_{k=1}^{N} x_k \tag{9}$$

enables identification of more than $O(M_1 + M_2)$ plane waves – more precisely, it enables identification of $O(M_1 M_2)$ plane waves. This problem is of the same form as the optimization (1) except that $s_{K+1}\mathbf{e}_{(M+1)/2}$ should be estimated and then eliminated.[6] Note that the vectors $\mathbf{u}$ and $\mathbf{x}$ (which we are interested in estimating) are typically set to have longer lengths than $\boldsymbol{\varphi}$ and $\mathbf{s}$, respectively.

3. The algorithms and their convergence analysis

In this section, we describe the SAM-CBCMA and SAA-CBCMA algorithms and analyze their convergence properties. Recall that our methods perform optimizations in subdomains instead of the entire domain $\boldsymbol{\Omega}$, which can lead to computational efficiency without estimation performance degradation. A detailed discussion of subdomain is presented in Section 4.2.

3.1. SAM-CBCMA

Let $\hat{u}_k^{(i)}$ and $\boldsymbol{\Omega}_k^{(i)}$ denote the estimate of u_k and its corresponding subdomain in the ith iteration of the algorithm. We initialize SAM-CBCMA with equally sized $\boldsymbol{\Omega}_k^{(0)}$, $k = 1, \ldots, N$, with centers $\hat{u}_k^{(0)}$. Once $\hat{u}_k^{(0)}$, $k = 1, \ldots, N$, are selected, the initial x_k, denoted by $\hat{x}_k^{(0)}$, $k = 1, \ldots, N$, are found as the solution to (1) (ignoring the regularization term in the objective function). Next, at iteration i, if $\hat{x}_k^{(i-1)}$ is non-zero we update the pairs $(\hat{u}_k^{(i-1)}, \hat{x}_k^{(i-1)})$ sequentially according to the equations

$$\mathbf{r}_k = \mathbf{r} - \sum_{l=1}^{k-1} \hat{x}_l^{(i)} \mathbf{a}_l\left(\hat{u}_l^{(i)}\right) - \sum_{l=k+1}^{N} \hat{x}_l^{(i-1)} \mathbf{a}_l\left(\hat{u}_l^{(i-1)}\right), \tag{10}$$

$$\hat{u}_k^{(i)} = \operatorname*{arg\,max}_{u_k \in \boldsymbol{\Omega}_k^{(i-1)}} \operatorname{Re}\left\{\mathbf{a}_k^H(u_k)\mathbf{r}_k\right\}, \tag{11}$$

$$\boldsymbol{\Omega}_k^{(i)} = \left[\max\left\{-W, \hat{u}_k^{(i)} - W_s\right\}, \min\left\{W, \hat{u}_k^{(i)} + W_s\right\}\right], \tag{12}$$

$$\hat{x}_k^{(i)} = \max\left\{0, \operatorname{Re}\left\{\mathbf{r}_k^H \mathbf{a}_k\left(\hat{u}_k^{(i)}\right)\right\} - \lambda_t\right\}/M, \tag{13}$$

where W and W_s are positive values determining the boundaries of $\boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_k^{(i)}$, respectively (without a loss of generality, $\boldsymbol{\Omega} = [-W, W]$). It is worth pointing out that the atom refinement (11) is performed in a subdomain rather than the entire domain $\boldsymbol{\Omega}$.

If $(u_k^{(i)}, x_k^{(i)})$ and $(u_l^{(i)}, x_l^{(i)})$ formed in the ith iteration of the algorithm are virtually undistinguishable, one of them is considered redundant. To formalize the notion of redundancy, let us refer to the collection of pairs satisfying $|\hat{u}_l^{(\bar{j})} - \hat{u}_k^{(i)}| \leq \Delta u_{\max}$, $l \neq k$, as a cluster, where $\Delta u_{\max}$ denotes a pre-selected threshold on the separation between cluster elements and

$$\bar{j} = \begin{cases} i & \text{if } l < k, \\ i-1 & \text{if } l > k. \end{cases}$$

SAM-CBCMA merges all pairs in the cluster as follows. Let us define the index set P_C^S as

$$P_C^S = \left\{l | \hat{x}_l^{(\bar{j})} \neq 0 \text{ and } \left|\hat{u}_l^{(\bar{j})} - \hat{u}_k^{(i)}\right| \leq \Delta u_{\max},\ \forall l \neq k\right\}. \tag{14}$$

We re-evaluate $\hat{x}_k^{(i)}$ according to

$$\hat{x}_k^{(i)} \leftarrow \hat{x}_k^{(i)} + \sum_{m \in P_C^S} \hat{x}_m^{(\bar{j})}, \tag{15}$$

and force all $\hat{x}_l^{(\bar{j})}$ for $l \in P_C^S$ to zero. SAM-CBCMA is formalized as Algorithm 1, where $i_{\min}$ and δ_t denote the minimum number of iterations and the residual threshold for algorithm termination, respectively. The SAM-CBCMA algorithm terminates in the iteration i where the merging process does not take place and the termination condition (steps 18 and 20 in Algorithm 1) is satisfied. The minimum number of iteration, $i_{\min}$, is imposed to avoid premature termination. Note that without merging, $\left|P_C^S\right| + 1$ of pairs $(\hat{u}_k^{(i)}, \hat{x}_k^{(i)})$ would essentially share a common $\hat{u}_k^{(i)}$ and thus $\left|P_C^S\right|$ computations would be unnecessarily performed in the following iteration; merging allows us to avoid those computations and hence accelerates SAM-CBCMA (as evident from steps 5–7 of Algorithm 1), as well as to promote sparsity of the solution $\hat{\mathbf{x}}$.

3.2. SAA-CBCMA

Unlike SAM-CBCMA, the SAA-CBCMA algorithm starts with no atoms and increases cardinality of the support set, greedily selecting and activating atoms over $\boldsymbol{\Omega}$. Such an atom activation procedure is reminiscent of the active-set algorithm of Lawson and Hanson (LH) [40]. In particular, at iteration i, an atom is activated according to

$$\hat{u}_k^{(i)} = \operatorname*{arg\,max}_{u_k \in \left(\boldsymbol{\Omega} \setminus \bigcup_l \boldsymbol{\Omega}_l^{(i-1)}\right)} \operatorname{Re}\left\{\mathbf{a}_k^H(u_k)\mathbf{r}_k\right\}, \tag{16}$$

where the feasible range of u_k excludes the subdomains corresponding to previously activated atoms. Then, unless the newly activated atom has a negligible effect on the cost function, all the activated atoms including the most recent one are refined according to the update equations (10)–(13). If the newly activated atom is indistinguishable from a previously activated one, the SAA-CBCMA algorithm is terminated. The differences between the LH algorithm (applying to our problem) and SAA-CBCMA are that the LH algorithm optimizes a cost function with no regularization while SAA-CBCMA optimizes a regularized cost function and that the active-set refinement of the LH algorithm involves rejection of a newly activated atom and re-selection of another atom on a predefined discrete grid while SAA-CBCMA relies on atom refinement within a continuous sub-range, i.e. a subdomain, using (11). The SAA-CBCMA is formalized as Algorithm 2. Note that the steps 2–9 of Algorithm 2 implement atom activation while steps 10–18 are parts of the atom refinement procedure. The termination of Algorithm 2 is facilitated via steps 7–9.

3.3. Convergence analysis

In this subsection, we build upon the result of [37] to establish convergence of the proposed cyclic block coordinate minimization algorithms. For convenience, let us refer to a row vector of

[6] It is straightforward to eliminate $s_{K+1}\mathbf{e}_{(M+1)/2}$. See Section 4.2 for details.

Algorithm 1: SAM-CBCMA.

```
Initialization: $\hat{u}_k^{(0)}$, $\mathbf{\Omega}_k^{(0)}$, and $\hat{x}_k^{(0)}$ for all $k$. Termination = 1 and $i =$ 0.
while Termination do
    $i \leftarrow i + 1$
    Merge = 0.
    for $k = 1, \ldots, N$ do
        if $\hat{x}_k^{(i-1)} = 0$ then
            Continue.
        end if
        Calculate $\mathbf{r}_k$ using (10).
        $f_{k,1} = \frac{1}{2}\left\|\mathbf{r}_k - \mathbf{a}_k\left(\hat{u}_k^{(i-1)}\right)\hat{x}_k^{(i-1)}\right\|_2^2$
            $+ \lambda_t\left(\sum_{l \neq k}\hat{x}_l^{(\bar{j})} + \hat{x}_k^{(i-1)}\right).$
        Update $\hat{u}_k^{(i)}$, $\hat{x}_k^{(i)}$, and $\mathbf{\Omega}_k^{(i)}$ using (11)–(13).
        if $\left|P_C^S\right| > 0$ then
            Merge = 1.
            Do merge process using (15).
            for $l \in P_C^S$ do
                $\hat{x}_l^{(\bar{j})} \leftarrow 0.$
            end for
        end if
        if $i > i_{\min}$ and Merge = 0 then
            $f_{k,2} = \frac{1}{2}\left\|\mathbf{r}_k - \mathbf{a}_k\left(\hat{u}_k^{(i)}\right)\hat{x}_k^{(i)}\right\|_2^2$
                $+ \lambda_t\left(\sum_{l \neq k}\hat{x}_l^{(\bar{j})} + \hat{x}_k^{(i)}\right).$
            if $\left(f_{k,1} - f_{k,2}\right)/f_{k,1} < \delta_t$ then
                Termination = 0.
                Break.
            end if
        end if
    end for
end while
return $\hat{x}_k$ and $\hat{u}_k$ for $k = 1, \ldots, N$
```

length $\mathbf{t}_k$, $k = 1, \ldots, L_1$, consisting of the parameters to be estimated as the kth coordinate block. After denoting the cost function of the coordinate block group $\mathbf{t} = [\mathbf{t}_1, \ldots, \mathbf{t}_{L_1}]$ by $f\left(\mathbf{t}_1, \ldots, \mathbf{t}_{L_1}\right)$ (or $f(\mathbf{t})$) and the rth update of the coordinate block group by $\mathbf{t}^r = [\mathbf{t}_1^r, \ldots, \mathbf{t}_{L_1}^r]$, we can define the $(r+1)^{st}$ update of the coordinate block group as

$$\mathbf{t}^{r+1} = \left[\mathbf{t}_1^{r+1}, \ldots, \mathbf{t}_{s-1}^{r+1}, \hat{\mathbf{t}}_s, \mathbf{t}_{s+1}^{r+1}, \ldots, \mathbf{t}_{L_1}^{r+1}\right], \tag{17}$$

where

$$\hat{\mathbf{t}}_s = \arg\min_{\mathbf{t}_s} f\left(\mathbf{t}_1^r, \ldots, \mathbf{t}_{s-1}^r, \mathbf{t}_s, \mathbf{t}_{s+1}^r, \ldots, \mathbf{t}_{L_1}^r\right) \tag{18}$$

and $\mathbf{t}_j^{r+1} = \mathbf{t}_j^r$ for all $j \neq s$. We proceed by stating several definitions in which $\mathbf{z}$ and $\mathbf{d}$ denote row vectors in $\mathbb{R}^{L_2}$; dom $g = \{\mathbf{z}|g(\mathbf{z}) < \infty\}$; and a row vector $\bar{\mathbf{d}}_k = [0, \ldots, 0, \mathbf{d}_k, 0, \ldots, 0] \in \mathbb{R}^{L_2}$ for $k = 1, \ldots, L_1$ such that $\sum_{k=1}^{L_1}\bar{\mathbf{d}}_k = \left[\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_{L_1}\right]$.

Definition 1. The directional derivative of a function g at $\mathbf{z}$ in the direction $\mathbf{d}$ is

$$\begin{aligned} g'(\mathbf{z}; \mathbf{d}) &= \lim_{\eta \to 0_+} \frac{g(\mathbf{z} + \eta\mathbf{d}) - g(\mathbf{z})}{\eta} \\ &= \left.\frac{\partial}{\partial\eta} g(\mathbf{z} + \eta\mathbf{d})\right|_{\eta=0}. \end{aligned}$$

Definition 2. $\mathbf{z}$ is a stationary point of a function g if $\mathbf{z} \in \mathrm{dom}\, g$ and $g'(\mathbf{z};\, \mathbf{d}) \geq 0$ for all $\mathbf{d}$.

Algorithm 2: SAA-CBCMA.

```
Initialization: $\hat{x}_k^{(0)} = 0$ for all $k$ and $i = 0$.
while (1) do
    $i \leftarrow i + 1$
    Calculate $\mathbf{r}_i$ using (10).
    $f_{i,1} = \frac{1}{2}\|\mathbf{r}_i\|_2^2 + \lambda_t \sum_{l=1}^{i-1}\hat{x}_l^{(i-1)}.$
    Select $\hat{u}_i^{(i-1)}$ using (16) and update $\hat{x}_i^{(i-1)}$ using (13).
    $f_{i,2} = \frac{1}{2}\left\|\mathbf{r}_i - \mathbf{a}_i\left(\hat{u}_i^{(i-1)}\right)\hat{x}_i^{(i-1)}\right\|_2^2$
        $+\lambda_t\left(\sum_{l=1}^{i-1}\hat{x}_l^{(i-1)} + \hat{x}_i^{(i-1)}\right).$
    if $\left(f_{i,1} - f_{i,2}\right)/f_{i,1} < \delta_{t,1}$ and $i > i_{\min}$ then
        Break.
    end if
    for $k = 1, \ldots, i$ do
        Calculate $\mathbf{r}_k$ using (10).
        $f_{k,1} = \frac{1}{2}\left\|\mathbf{r}_k - \mathbf{a}_k\left(\hat{u}_k^{(i-1)}\right)\hat{x}_k^{(i-1)}\right\|_2^2$
            $+\lambda_t\left(\sum_{l \neq k}\hat{x}_l^{(\bar{j})} + \hat{x}_k^{(i-1)}\right).$
        Update $\hat{u}_k^{(i)}$, $\hat{x}_k^{(i)}$, and $\mathbf{\Omega}_k^{(i)}$ using (11)–(13).
        $f_{k,2} = \frac{1}{2}\left\|\mathbf{r}_k - \mathbf{a}_k\left(\hat{u}_k^{(i)}\right)\hat{x}_k^{(i)}\right\|_2^2$
            $+\lambda_t\left(\sum_{l \neq k}\hat{x}_l^{(\bar{j})} + \hat{x}_k^{(i)}\right).$
        if $\left(f_{k,1} - f_{k,2}\right)/f_{k,1} < \delta_{t,2}$ then
            Break.
        end if
    end for
end while
return $\hat{x}_k$ and $\hat{u}_k$ for $k = 1, \ldots, N$
```

Definition 3. $\mathbf{z}$ is a coordinate-wise local minimum (maximum) point of a function g if $\mathbf{z} \in \mathrm{dom}\, g$ and $g\left(\mathbf{z} + \bar{\mathbf{d}}_k\right) \geq g(\mathbf{z})$ $\left(g\left(\mathbf{z} + \bar{\mathbf{d}}_k\right) \leq g(\mathbf{z})\right)$ for all possible $\bar{\mathbf{d}}_k$ in the constraint that at least one of the nonzero elements of $\bar{\mathbf{d}}_k$ is within a local range.

Definition 4. A function g is regular at $\mathbf{z} \in \mathrm{dom}\, g$ if

$$g'(\mathbf{z}; \mathbf{d}) = \sum_{k=1}^{L_1} g'\left(\mathbf{z}; \bar{\mathbf{d}}_k\right).$$

Hereafter, confining our attention to the optimization (1) being solved by our algorithms, we set $L_1 = N$, $L_2 = 2N$, $\mathbf{t}_k = [u_k, x_k]$, denote the cost function of (1) by $f(\mathbf{t}_1, \ldots, \mathbf{t}_N)$, $\mathbf{d}_k \in \mathbb{R}^2$, $\mathbf{e}_k \in \mathbb{R}^N$, and $\bar{\mathbf{d}}_k = \mathbf{e}_k \otimes \mathbf{d}_k$.

Lemma 1. *Suppose that the sequence* $\{\mathbf{t}^r\}$ *for* $r = 1, 2, \ldots$ *is produced by using the update Eqs. (10)–(13). Then,* $\{\mathbf{t}^r\}$ *for* $r = 1, 2, \ldots$ *is bounded.*

Proof. Let us define a set $T^0 = \left\{\mathbf{z} : f(\mathbf{z}) \leq f\left(\mathbf{t}^0\right)\right\}$ where $\mathbf{t}^0$ is an initial coordinate block. The component u_k of any element of T^0 is in the closed interval $[-W, W]$. The other component x_k of that element is also in the closed interval $[0, \infty)$ since infinite x_k yields the infinite cost function. Therefore, T^0 is closed and bounded. Furthermore, since $f(\mathbf{t}^r)$ monotonically decreases or stay the same as r increases, it holds that $f(\mathbf{t}^r) \leq f\left(\mathbf{t}^0\right)$ so $\{\mathbf{t}^r\} \subset T^0$. The two results that "$T^0$ is closed and bounded" and "$\{\mathbf{t}^r\} \subset T^0$" indicate that the sequence $\{\mathbf{t}^r\}$ for $r = 1, 2, \ldots$ is bounded. □

Lemma 2. *f is regular.*

Proof. Note that the directional derivative of f at $\mathbf{d} = [\mathbf{d}_1, \ldots, \mathbf{d}_N]$, where $\mathbf{d}_k = [d_{u,k}, d_{x,k}]$, is given by

$$f'(\mathbf{t}_1, \ldots, \mathbf{t}_N; \mathbf{d})$$

$$\mathrm{Re}\left\{ -\left(\sum\nolimits_{k=1}^{N} d_{u,k} x_k \frac{\partial}{\partial \eta} \mathbf{a}(u_k) + \sum\nolimits_{k=1}^{N} d_{x,k} \mathbf{a}(u_k) \right)^H \right.$$
$$\left. \times \left(\mathbf{r} - \sum\nolimits_{k=1}^{N} x_k \mathbf{a}(u_k) \right) \right\} + \lambda_t \sum_{k=1}^{N} d_{x,k}. \tag{19}$$

The directional derivative of f at $\bar{\mathbf{d}}_k$ is given by

$$f'\left(\mathbf{t}_1, \ldots, \mathbf{t}_N; \bar{\mathbf{d}}_k\right)$$
$$= \mathrm{Re}\left\{ -\left(d_{u,k} x_k \frac{\partial}{\partial u_k} \mathbf{a}(u_k) + d_{x,k} \mathbf{a}(u_k) \right)^H \right.$$
$$\left. \times \left(\mathbf{r} - \sum\nolimits_{k=1}^{N} x_k \mathbf{a}(u_k) \right) \right\} + \lambda_t d_{x,k}. \tag{20}$$

Using (19) and (20), we obtain

$$f'(\mathbf{t}_1, \ldots, \mathbf{t}_N; \mathbf{d}) = \sum_{k=1}^{N} f'\left(\mathbf{t}_1, \ldots, \mathbf{t}_N; \bar{\mathbf{d}}_k\right). \tag{21}$$

Therefore, f is regular. □

Finally, we introduce the following theorem.

Theorem 1. *Suppose that the problem (11) for $k = 2, 3, \ldots, N-1$ has at most one solution. Then, the sequence $\{\mathbf{t}^r\}$ for $r = N-1, 2N-1, \ldots$ produced by the update Eqs. (10)–(13) converges to a stationary point of f.*

Proof. By Bolzano–Weierstrass theorem[7] [41] and Lemma 1, $\{\mathbf{t}^r\}$ for $r = 1, 2, \ldots$ forms a convergent sequence and any subsequence of $\{\mathbf{t}^r\}$ is also convergent. In other words, for any row vector $\bar{\mathbf{s}}^j$, $j = 1, 2, \ldots, N$, it holds that $\left\{\mathbf{t}^{r-N+1+j}\right\}$, $r = N-1, 2N-1, \ldots$ converges to $\bar{\mathbf{s}}^j = \left[\bar{\mathbf{s}}_1^j, \bar{\mathbf{s}}_2^j, \ldots, \bar{\mathbf{s}}_N^j\right]$. Note that $f(\mathbf{z})$ for $\mathbf{z} \in T^0$ is continuous and lower-bounded by zero; moreover, $f(\mathbf{t}^r)$ monotonically decreases as r increases. Therefore, $\{f(\mathbf{t}^r)\}$ for $r = 1, 2, \ldots$ must converge, i.e.,

$$f\left(\mathbf{t}^0\right) \geq \lim_{r \to \infty} f(\mathbf{t}^r) = f\left(\bar{\mathbf{s}}^1\right) = f\left(\bar{\mathbf{s}}^2\right) = \cdots = f\left(\bar{\mathbf{s}}^N\right). \tag{22}$$

To complete the proof of Theorem 1, we need to show that if the problem (11) for $k = 2, 3, \ldots, N-1$ has at most one solution, $f(\mathbf{t}_k)$ for $k = 2, 3, \ldots, N-1$ has at most one minimum over the corresponding subdomain. Recall the cost function in (1),

$$f(\mathbf{t}_1, \ldots, \mathbf{t}_N) = \frac{1}{2} \left\| \mathbf{r} - \sum\nolimits_{k=1}^{N} \mathbf{a}_k(u_k) x_k \right\|_2^2 + \lambda_t \sum_{k=1}^{N} x_k. \tag{23}$$

For the coordinate block $\mathbf{t}_k$,

$$f(\mathbf{t}_k) = \frac{M}{2} \left(x_k - \frac{\mathrm{Re}\left\{\mathbf{r}_k^H \mathbf{a}(u_k)\right\} - \lambda_t}{M} \right)^2 - c(u_k), \tag{24}$$

where

$$c(u_k) = \frac{\left(\mathrm{Re}\left\{\mathbf{r}_k^H \mathbf{a}(u_k)\right\} - \lambda_t\right)^2}{2M} + \frac{1}{2} \mathbf{r}_k^H \mathbf{r}_k + \lambda_t \sum_{l \neq k} x_l. \tag{25}$$

Note that once u_k is specified, so is x_k since $f(\mathbf{t}_k)$ is convex with respect to x_k. Consequently, if $\mathrm{Re}\left\{\mathbf{r}_k^H \mathbf{a}(u_k)\right\}$ has a finite number of local maxima in u_k, we can calculate the unique local minimizer of $f(\mathbf{t}_k)$.

Finally, we use the results of [37]: under the condition that $f(\mathbf{t}_k)$ for $k = 2, 3, \ldots, N-1$ has at most one minimum over the corresponding subdomain, (a) $\bar{\mathbf{s}}^1 = \bar{\mathbf{s}}^2 = \cdots = \bar{\mathbf{s}}^{N-1}$; (b) $\bar{\mathbf{s}}^j$ is a coordinate-wise local minimum point of f; and (c) $\bar{\mathbf{s}}^j$ is also a stationary point since f is regular by Lemma 2. □

Since the maximum number of atoms N is finite, both the number of merge steps of SAM-CBCMA and that of the atom activation steps of SAA-CBCMA are also finite. Therefore, if we use the estimated pair of x_k and u_k immediately following the merge step or atom activation step to initialize $\mathbf{t}^0$, it is clear that each of our algorithms would converge to a stationary point of the cost function. The selection of subdomains is further discussed in Section 4.2.

4. Simulation results

In this section, we first specify the five criteria used for performance evaluation, specifically the RMSE, P_d, $P_{f,\,1}$, $P_{f,\,2}$, and CPU time, and discuss how to choose the parameters of SAM-CBCMA and SAA-CBCMA considering a co-prime array with a positive integer d_c. Next, we empirically compare the performance of our algorithms with that of the existing methods,[8,9] i.e. JLASSO [16], GLS-ESPRIT [29], ANM-ESPRIT [27], NNM-ESPRIT [28], and SS-ESPRIT [12], in terms of the selected criteria and verify their convergence.

4.1. Performance evaluation criteria

To characterize the accuracy of the algorithms, let us set the lower and upper bound of the detection region of the kth plane wave, B_k^L and B_k^U, as

$$B_k^L = \max\{-W, \varphi_k - \Delta B\},$$
$$B_k^U = \min\{W, \varphi_k + \Delta B\}, \tag{26}$$

where ΔB is the maximum half width of a detection region. Denoting the detection threshold by γ_t, we make a decision on the kth plane wave depending on whether there exist at least one pair $(\hat{u}_l, \hat{x}_l)$, $l = 1, \ldots, N$, such that $B_k^L < \hat{u}_l \leq B_k^U$ and $\hat{x}_l > \gamma_t$. Consequently, the detection probability P_d is calculated as the number of detected plane waves/K. If there exist more than one pair $(\hat{u}_l, \hat{x}_l)$ in a detection region, the superfluous pairs count as type-1 false alarms. So, the false alarm count 1, $P_{f,\,1}$, is calculated as the number of type-1 false alarms/K. Moreover, $P_{f,\,2}$ is calculated as the number of type-2 false alarms which are defined as overall false alarms in the entire domain $\mathbf{\Omega}$. The average RMSE (in radians) of $\hat{u}_k$ is calculated based on one of the detected plane waves that has the maximum signal power in each detection region. The CPU time is the time (in seconds) required for a Monte Carlo run of an algorithm. It is worth pointing out that P_d and

[7] Every bounded infinite set (or sequence) of real numbers has at least one limit point.

[8] JLASSO, GLS-ESPRIT, ANM-ESPRIT, and NNM-ESPRIT are implemented by using the CVX package [42]. For simulation, we respectively set the number of grid points and the hyperparameter η of the JLASSO optimization (8) in [16] to $4(2M_2(2M_1 - 1) + 1)$ and $\max\left\{1.7, 3.5M\hat{\sigma}^2\right\}$ where $\hat{\sigma}^2$ is an initial estimate of noise power. The hyperparameter $0.5\lambda/\sqrt{N}$ of the ANM-ESPRIT optimization (190) in [31] is set to $5\sqrt{M\hat{\sigma}^2}$. The hyperparameter η of the NNM-ESPRIT optimization (173) in [31] is set to $1.25M\hat{\sigma}^2$.

[9] Semidefinite programming based methods including GLS-ESPRIT and ANM-ESPRIT can be implemented computationally more efficiently via another solver faster than those in the CVX package [42], e.g. the alternating direction method of multipliers (ADMM) [38,43]. ADMM guarantees global optimality and can increase the convergence speed by sacrificing the accuracy.

$P_{f,\,1}$ include information about grid mismatch sensitivity and target resolvability.

4.2. Parameter setting

We set the maximum number of atoms to $N = 2M + 1$. This guarantees that at least one of the columns of $\mathbf{A}(\mathbf{u})$ is linearly dependent on other columns. As a result, it is possible that the final estimate $\hat{\mathbf{x}}$ of CBCM includes at least one zero element and thus according to [33] satisfies the KKT condition of atomic norm minimization. In order to determine the entire domain $\boldsymbol{\Omega}$, W is set to $1 - 1/(2M_{\max} + 1)$ and thus $\boldsymbol{\Omega} = [-1 + 1/(2M_{\max} + 1), 1 - 1/(2M_{\max} + 1)]$. This allows every column of $\mathbf{A}(\mathbf{u})$ to represent a discrete Fourier vector sampled at Nyquist sampling rate but exclude several points (the holes). In order to determine the range of subdomains, we set empirically recommended value $W_s \in [0.5/(2M_{\max} + 1), 2/(2M_{\max} + 1)]$. Note that under the assumption of not having any holes, $2W/(2M_{\max} + 1)$ and $4W/(2M_{\max} + 1)$ are approximate 3-dB and null-to-null mainlobe widths of the Fourier-domain response, respectively. There is a trade-off where a large W_s increases possibility of finding the optimal u_k in each iteration but also increases computational complexity while the opposite happens with a small W_s. By adjusting the value of W_s, initial subdomains are either overlapped or their union does not initially cover the entire domain. The most influential parameter, λ_t, is in general hard to select to both promote the sparsity of $\mathbf{x}$ and suppress the data-fitting error. However, since λ_t plays a role of a soft threshold in $\mathbf{x}$ updates, as implied by (13), we can make an intuitively plausible choice. Recall $\mathrm{Re}\{\mathbf{r}_k^H \mathbf{a}_k^{(i)}\}$ in (13). This object can be recognized as the magnitude of the spectrum of the even part of $[\mathbf{r}_k]_m$ for $m = 1, 2, \ldots, N$ at the frequency $\hat{u}_k^{(i)}$, which can be approximated by $M\hat{x}_k^{(i-1)}$. In the regimes where the signal power is greater than the power of noise, it is expected that $\hat{x}_k^{(i)} > s_{K+1}$ for every i. Therefore, we set $\lambda_t = c_1 M s_{K+1}$ where c_1 is a positive control parameter. To promote sparsity of $\mathbf{x}$, λ_t can be set to $\max\{c_2, c_1 M s_{K+1}\}$ where c_2 is a positive value that controls the minimum sparsity level. Finally, c_1 and c_2 can be chosen by cross validation.

For the SAM-CBCMA algorithm, we initialize $\mathbf{u}$ with N evenly spaced points between $-1 + 1/(2M_{\max} + 1)$ and $1 - 1/(2M_{\max} + 1)$. Then we calculate the initial $\mathbf{x}$ and s_{K+1} in (9) by relying on matrix pseudo inversion and setting $\lambda_t = 0$, which leads to the minimum l_2-norm solution. The initial $\mathbf{x}$ is further refined by forcing its negative elements to be zero. The initial estimate of s_{K+1}, denoted by $\hat{s}_{K+1}$, is used to eliminate the effect of s_{K+1} in both of our algorithms by updating $\mathbf{r} \leftarrow \mathbf{r} - \hat{s}_{K+1}\mathbf{e}_{(M+1)/2}$ at the initialization.

Remark 1:. Solving (9) by using algorithms on subdomains from Section 3 yields a stationary point of the cost function. It is of interest to analyze optimality of such stationary points. When the subdomain concept is not used, from Theorem 1 in [33] it follows that the stopping point $\bar{\mathbf{s}} = [\bar{u}_1, \bar{x}_1, \bar{u}_2, \bar{x}_2, \ldots, \bar{u}_N, \bar{x}_N]$ of a CBCM algorithm satisfies the KKT condition for atomic norm minimization if at least one of $\bar{x}_k$ for $k = 1, \ldots, N$ is zero. However, in line spectral estimation problems including (9), the range where an optimal u_k exists is limited to the mainlobe width of the spectrum so we clearly do not have to optimize u_k over the entire $\boldsymbol{\Omega}$. Furthermore, once a subdomain is set to cover some portion of the width of a lobe (either sidelobe or mainlobe), our algorithms tend to find the peak of the lobe as the iterations proceed. This is because the preceding estimate of u_k is used as the center of the corresponding subdomain at the current iteration. In fact, our numerical results demonstrate that, in practical settings, the (approximate) 3 dB width of the mainlobe of a spectrum is a sufficiently large subdomain and enables comparable or better performance of the proposed algorithms as compared to state-of-the-art methods.

Table 1
values of the SAM-CBCMA parameters set in the simulations.

λ_t	$\max\{1.5, 1.25M\hat{s}_{K+1}\}$	N	$2M + 1$
W	$1 - 1/(2M_2(2M_1 - 1) + 1)$	γ_t	$0.05 \max\limits_m \hat{x}_m$
W_s	$1/(2M_2(2M_1 - 1) + 1)$	M_1, M_2	3, 5
$\Delta u_{\max}$	$1/(2M_2(2M_1 - 1) + 1)$	$i_{\min}$	3
ΔB	$2/(2M_2(2M_1 - 1) + 1)$	δ_t	5×10^{-10}

4.3. Discussion of the results

For simulations, we set $d_c = 2$ and assume that the plane waves are randomly incident from a limited angular range between -0.875 and 0.875 radian. We set the powers of waves (signals) to fluctuate in a way that their maximum value can be at most four times larger than their minimum value, and noise powers to be equal. Therefore, the SNR is defined as $\frac{M}{K}\sum_{k=1}^{K} E\{\alpha_k^* \alpha_k.\}/E\{n_1^* n_1\}$. The number of snapshots for constructing a sample covariance matrix is 500. The symbol $\mathrm{e} \pm c$ in Tables 2–4 denotes $10^{\pm c}$. Note that GLS-ESPRIT, ANM-ESPRIT, NNM-ESPRIT, and SS-ESPRIT require estimating rank of a covariance matrix estimator (model order). For this, we employ the model order selection method introduced in [44], that is, a combination of the second order statistic of eigenvalues (SORTE) [45] and the ratio of adjacent eigenvalues (RAE) for stable model order selection and list the means and standard deviations of the model order estimators in a form of fraction in the last parts of Tables 2–4 where the denominators and numerators indicate the means and standard deviations, respectively. The simulation parameter values of the SAM-CBCMA algorithm are given in Table 1.[10] For the SAA-CBCMA algorithm, the same parameter values are used except that $\delta_{t,1} = \delta_{t,2} = 10^{-7}$ and $i_{\min} = 1$. The simulation results are obtained using arithmetic average via 2000 Monte Carlo runs, except for Fig. 5, in the following three scenarios: $K = 1$, $K = 2$, and $K = 15$ targets. In the $K = 2$ case (two-close-target case), the true DOA values are separated by 0.03 radian while in the 15-target case the minimum separation between neighboring DOA values is set to be 0.05 radian. Note that $2W_s$, i.e., the range of subdomains, is set to 0.04 radian which is approximately the 3 dB mainlobe width of the spectrum in our co-prime array setting.

The results for $K = 1$ are shown in Fig. 2 and Table 2. Both SAM-CBCMA and SAA-CBCMA are comparable with the benchmarking algorithms in terms of RMSE. Moreover, the value of $P_{f,\,1}$ of our algorithms is small in all SNRs, implying that they overcome the grid mismatch problem. GLS-ESPRIT offers the best performance in most criteria but CPU time and, especially, almost achieves the Cramér Rao lower bound (CRLB) [14] at high SNRs. It is worthy noting that our algorithms are about more than ten times faster than the other methods except the SS-ESPRIT method that is the fastest because only model order selection and singular value decomposition are performed without other iterative procedures. SAM-CBCMA is about more than twice slower than SAA-CBCMA. The reason why SAM-CBCMA is slower than SAA-CBCMA is that the number of targets is $K = 1$ and thus activating atoms is more efficient than performing the merge step. As the number of targets increases, SAM-CBCMA becomes faster than SAA-CBCMA (as

[10] Since the number of sensors is given by $d_c M_1 + M_2 - 1$, the number of (physical) sensors in our simulation is 10. The value for γ_t is selected empirically but in our simulation we found the algorithms to be rather robust to the choice of γ_t between 0.05 and 0.1. Note that $\Delta u_{\max}$ is about 0.02, i.e., half of the 3 dB width of the mainlobe of the spectrum in our co-prime array setting. In general, this value should provide a guidance for making a decision whether two atoms are merged or not and is thus set to the center point between 0 and 0.04 (approximate mainlobe width or resolution of the spectrum).

Table 2
Performance evaluation in the $K = 1$ target case.

P_d	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	9.9800e−1	1.0000	1.0000	1.0000	1.0000
SAA-CBCMA	9.9800e−1	1.0000	1.0000	1.0000	1.0000
JLASSO	9.9900e−1	1.0000	1.0000	1.0000	1.0000
GLS-ESPRIT	9.9950e−1	1.0000	1.0000	1.0000	1.0000
ANM-ESPRIT	9.9950e−1	1.0000	1.0000	1.0000	1.0000
NNM-ESPRIT	1.0000	1.0000	1.0000	1.0000	1.0000
SS-ESPRIT	9.9800e−1	1.0000	1.0000	1.0000	1.0000
$P_{f,1}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	1.5000e−2	6.0000e−3	2.0000e−3	5.0000e−4	0.0000
SAA-CBCMA	1.6000e−2	1.9500e−2	2.4500e−2	1.9000e−2	2.5000e−3
JLASSO	3.7500e−2	2.0000e−3	5.0000e−3	9.0000e−3	2.0000e−3
GLS-ESPRIT	4.3500e−2	9.0000e−3	1.5000e−3	5.0000e−4	0.0000
ANM-ESPRIT	4.9500e−2	7.5000e−3	2.0000e−3	1.0000e−3	3.0000e−3
NNM-ESPRIT	4.7500e−2	1.0500e−2	1.5000e−3	1.0000e−3	0.0000
SS-ESPRIT	8.5000e−3	1.0000e−3	0.0000	0.0000	0.0000
$P_{f,2}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	8.3350e−1	3.1250e−1	1.3000e−2	5.0000e−4	1.0000e−3
SAA-CBCMA	9.5300e−1	3.3300e−1	3.3500e−2	1.9000e−2	2.5000e−3
JLASSO	8.5800e−1	5.4000e−2	7.5000e−3	9.0000e−3	2.0000e−3
GLS-ESPRIT	3.1765	5.1850e−1	1.0100e−1	1.1000e−2	0.0000
ANM-ESPRIT	3.4720	4.6200e−1	1.3400e−1	3.3000e−2	1.3150e−1
NNM-ESPRIT	3.3330	4.2800e−1	1.2300e−1	5.3500e−2	0.0000
SS-ESPRIT	1.9430	1.1000e−1	1.4500e−2	0.0000	0.0000
CPU time	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	1.7274e−2	1.7848e−2	1.9648e−2	2.4699e−2	2.1502e−2
SAA-CBCMA	6.5249e−3	7.1170e−3	9.2029e−3	1.3112e−2	6.1186e−3
JLASSO	8.0444e−1	8.0392e−1	8.0420e−1	8.6983e−1	7.2687e−1
GLS-ESPRIT	5.9484e−1	5.9569e−1	6.0463e−1	6.2615e−1	6.3086e−1
ANM-ESPRIT	8.7820e−1	8.9630e−1	8.9426e−1	9.3025e−1	1.0066
NNM-ESPRIT	7.7971	7.3701	8.9764	9.5310	1.2049e+1
SS-ESPRIT	2.8359e−3	1.1328e−3	7.8125e−4	5.7812e−4	6.1719e−4
Model order	−5 dB	0 dB	5 dB	10 dB	15 dB
GLS-ESPRIT	$\frac{7.7411}{4.1760}$	$\frac{3.3813}{1.5185}$	$\frac{1.5030}{1.1010}$	$\frac{4.9193e-1}{1.0110}$	$\frac{0.0000}{1.0000}$
ANM-ESPRIT	$\frac{8.0379}{4.4715}$	$\frac{3.1887}{1.4620}$	$\frac{1.7256}{1.1340}$	$\frac{8.5163e-1}{1.0330}$	$\frac{3.7186e-1}{1.1315}$
NNM-ESPRIT	$\frac{7.8827}{4.3330}$	$\frac{3.0698}{1.4280}$	$\frac{1.6548}{1.1230}$	$\frac{1.0706}{1.0535}$	$\frac{0.0000}{1.0000}$
SS-ESPRIT	$\frac{4.8003}{2.9410}$	$\frac{1.2664}{1.1100}$	$\frac{4.5869e-1}{1.0145}$	$\frac{0.0000}{1.0000}$	$\frac{0.0000}{1.0000}$

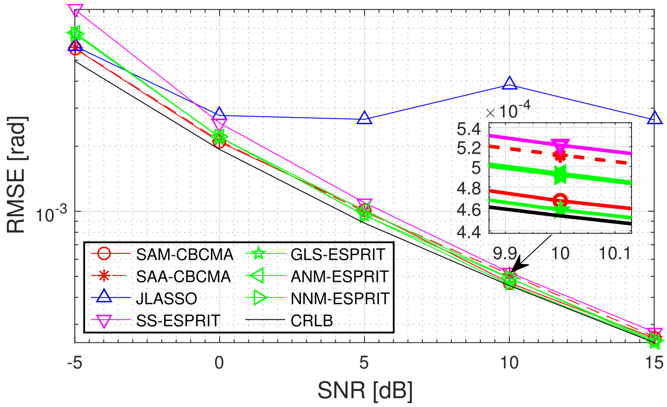


Fig. 2. The comparison of DOA estimation RMSEs in the $K = 1$ target case.

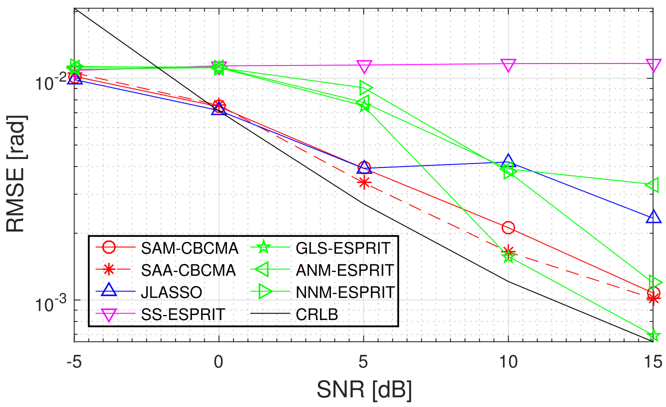


Fig. 3. The comparison of DOA estimation RMSEs in the $K = 2$ target case.

evident by the CPU time in the 15-target case). On all the criteria except CPU time, SAM-CBCMA and SAA-CBCMA are shown to exhibit similar performance.

Fig. 3 and Table 3 compare the target resolvability of the algorithms. We observe that if the SNR is 0 dB or higher, our algorithms and JLASSO perform well in terms of P_d and $P_{f,1}$ and, therefore, have good resolvability in $K = 2$ close-target cases. GLS-ESPRIT, ANM-ESPRIT, and NNM-ESPRIT also yield good resolvability at 15 dB but degraded resolvability at 10 dB or less SNRs. However, these convex optimization based methods have the potential for improvement of resolvability by improving model order selection since the degradation is mainly due to incorrectly selected model order (as evident by the standard deviations in Table 3). GLS-ESPRIT is the only one that almost achieves the CRLB in Fig. 3. The SS-ESPRIT algorithm achieves the lowest performance in terms of every criterion except CPU time. Specifically, P_d of SS-ESPRIT is

Table 3
Performance evaluation in the $K = 2$ target case.

P_d	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	5.9600e−1	8.4875e−1	9.9250e−1	9.9950e−1	9.9975e−1
SAA-CBCMA	6.2100e−1	9.4625e−1	9.9975e−1	1.0000	1.0000
JLASSO	5.5675e−1	8.8150e−1	9.9900e−1	9.8875e−1	9.9950e−1
GLS-ESPRIT	5.1300e−1	5.1450e−1	6.8675e−1	9.9250e−1	1.0000
ANM-ESPRIT	5.2275e−1	5.1450e−1	6.9300e−1	9.8075e−1	1.0000
NNM-ESPRIT	5.2150e−1	5.1375e−1	6.1225e−1	8.9225e−1	1.0000
SS-ESPRIT	5.0525e−1	5.0050e−1	5.0000e−1	5.0000e−1	5.0025e−1
$P_{f,1}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	7.5000e−3	1.0250e−2	5.2500e−3	7.5000e−3	0.0000
SAA-CBCMA	1.6750e−2	2.3500e−2	2.5000e−3	5.0000e−4	0.0000
JLASSO	5.2500e−3	1.3000e−2	5.7500e−3	1.4500e−2	5.0000e−4
GLS-ESPRIT	2.5000e−3	7.5000e−4	3.2500e−3	1.7500e−3	0.0000
ANM-ESPRIT	1.5000e−3	1.5000e−3	3.0000e−3	1.7500e−3	0.0000
NNM-ESPRIT	1.0000e−3	1.0000e−3	3.2500e−3	2.5000e−4	0.0000
SS-ESPRIT	0.0000	0.0000	0.0000	0.0000	0.0000
$P_{f,2}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	6.1050e−1	2.2700e−1	4.0000e−2	1.6000e−2	0.0000
SAA-CBCMA	6.8350e−1	2.3450e−1	2.5500e−2	1.0000e−3	0.0000
JLASSO	1.0850e−1	5.1000e−2	1.8500e−2	2.9000e−2	1.0000e−3
GLS-ESPRIT	9.2050e−1	4.7300e−1	4.6600e−1	2.4000e−1	1.0500e−2
ANM-ESPRIT	1.2150	5.0600e−1	7.7600e−1	2.7100e−1	0.0000
NNM-ESPRIT	1.1155	3.7500e−1	5.4450e−1	7.1500e−2	2.9500e−2
SS-ESPRIT	4.3200e−1	4.3500e−2	0.0000	0.0000	0.0000
CPU time	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	1.9373e−2	4.0730e−2	6.8117e−2	9.0320e−2	1.0333e−1
SAA-CBCMA	8.3934e−3	3.3149e−2	5.3319e−2	5.8106e−2	5.7392e−2
JLASSO	8.5630e−1	8.6775e−1	9.4743e−1	8.5615e−1	8.8710e−1
GLS-ESPRIT	5.9665e−1	6.1158e−1	6.3927e−1	6.6198e−1	6.6323e−1
ANM-ESPRIT	8.9207e−1	9.4296e−1	9.4379e−1	1.0500	9.6189e−1
NNM-ESPRIT	7.7721	7.9849	9.6924	1.0881e+1	1.1155e+1
SS-ESPRIT	2.3594e−3	1.4141e−3	1.7188e−3	1.0234e−3	6.0938e−4
Model order	-5 dB	0 dB	5 dB	10 dB	15 dB
GLS-ESPRIT	$\frac{4.5015}{1.9465}$	$\frac{3.2954}{1.5020}$	$\frac{3.2728}{1.8395}$	$\frac{2.2814}{2.2250}$	$\frac{4.6957\mathrm{e}{-1}}{2.0105}$
ANM-ESPRIT	$\frac{5.1636}{2.2605}$	$\frac{3.4054}{1.5350}$	$\frac{4.1636}{2.1620}$	$\frac{2.2950}{2.2325}$	$\frac{0.0000}{2.0000}$
NNM-ESPRIT	$\frac{4.9524}{2.1585}$	$\frac{2.9431}{1.4025}$	$\frac{3.5169}{1.7690}$	$\frac{1.2884}{1.8560}$	$\frac{1.9912\mathrm{e}{-1}}{2.0295}$
SS-ESPRIT	$\frac{2.4819}{1.4425}$	$\frac{8.1170\mathrm{e}{-1}}{1.0445}$	$\frac{0.0000}{1.0000}$	$\frac{0.0000}{1.0000}$	$\frac{2.2361\mathrm{e}{-2}}{1.0005}$

only about 0.5 at all SNRs, which means that SS-ESPRIT does not resolve the two close targets we set. Note that the strong performance in terms of RMSE at low SNRs is partly due to ignoring the estimates yielding large RMSE values during the detection process (those are being counted as "missing targets"). We should also point out that both of our algorithms are practically feasible although SAM-CBCMA is slower than SAA-CBCMA (same as in the $K = 1$ target case), and yield better resolvability and RMSE than ANM-ESPRIT. SAA-CBCMA has the best detection ability.

Fig. 4 and Table 4 illustrate the performance of the algorithms when the number of targets is larger than the number of sensors. We first observe that all the methods excluding the SS-ESPRIT method that yields the worst performance with the underestimated model order have capability of detecting 15 targets with 10 sensors. The RMSE of GLS-ESPRIT appreciably decreases as the SNR increases while those of the other methods including our algorithms are saturated. In fact, it is known that GLS-ESPRIT stems from a large-snapshot realization of the maximum likelihood estimator [46] and, thus, the RMSE of GLS-ESPRIT can be expected to be good at high SNRs. It is worth pointing out that our algorithms retain the computational efficiencies and the RMSEs comparable with those of ANM-ESPRIT and NNM-ESPRIT. SAM-CBCMA is now faster than SAA-CBCMA, as opposed to the small K settings. This is because the atom activation over the entire $\mathbf{\Omega}$ (excluding the union of the subdomains corresponding to previously activated atoms), done by SAA-CBCMA, imposes a significant computational burden when the number of targets is large. Meanwhile, SAM-CBCMA maintains computational efficiency when the number of targets is large.

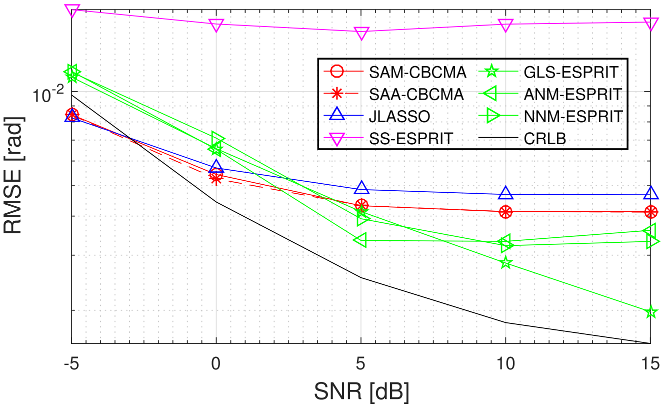


Fig. 4. The comparison of DOA estimation RMSEs in the $K = 15$ target case.

Fig. 5 illustrates the normalized power spectra of all the methods we consider. All the methods identify the four components of the ground truth. There are good coincidence between the ground truth and the spectra of GLS-ESPRIT and NNM-ESPRIT. The

Table 4
Performance evaluation in the $K = 15$ target case.

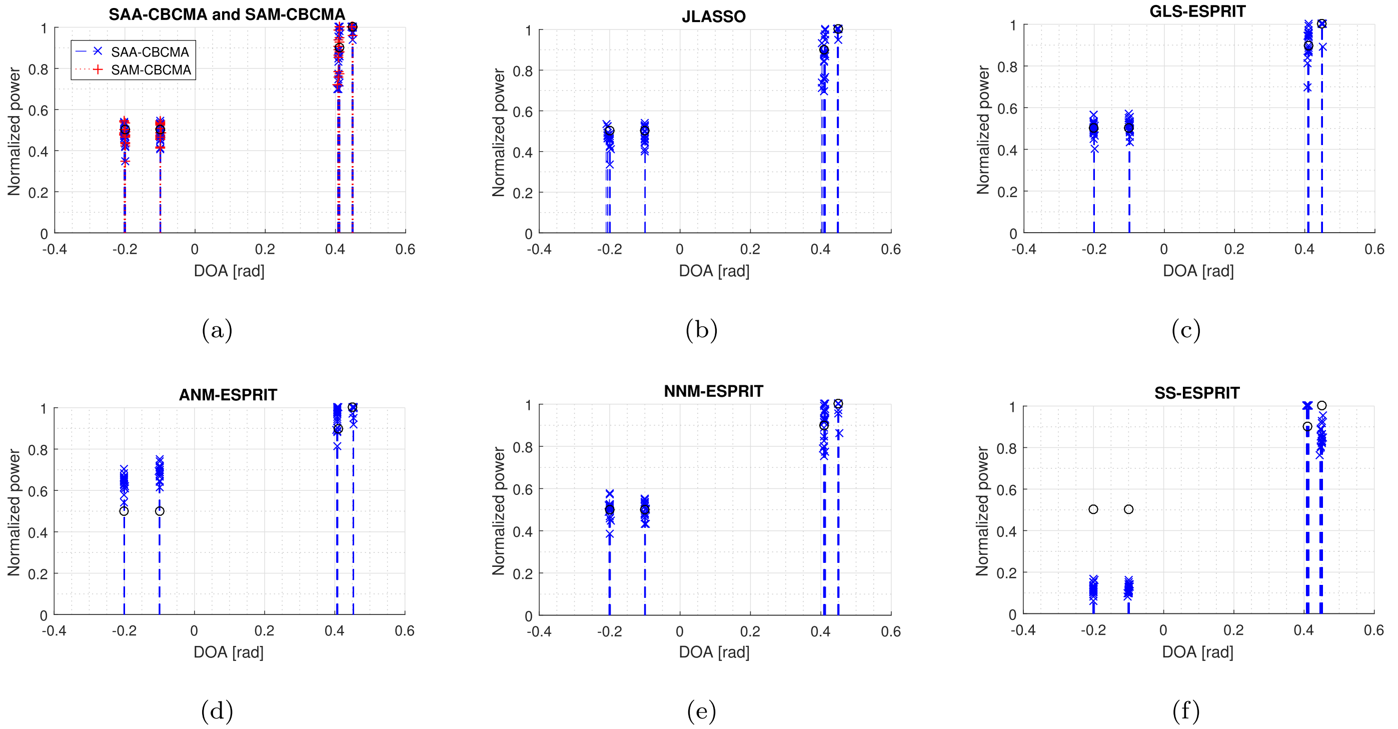
P_d	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	9.0150e−1	9.9610e−1	9.9537e−1	9.9570e−1	9.9467e−1
SAA-CBCMA	9.0077e−1	9.9687e−1	9.9597e−1	9.9603e−1	9.9530e−1
JLASSO	8.3557e−1	9.9083e−1	9.9007e−1	9.9213e−1	9.9033e−1
GLS-ESPRIT	7.3970e−1	9.4493e−1	9.9867e−1	9.9953e−1	9.9993e−1
ANM-ESPRIT	7.6967e−1	9.6970e−1	9.9713e−1	9.9993e−1	9.9987e−1
NNM-ESPRIT	7.6310e−1	9.7583e−1	9.9643e−1	9.9960e−1	9.9977e−1
SS-ESPRIT	6.1953e−1	7.7607e−1	8.3643e−1	8.1527e−1	8.0463e−1
$P_{f,\,1}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	6.1000e−3	1.5067e−2	2.3000e−3	1.5000e−3	1.3333e−3
SAA-CBCMA	2.3333e−3	3.5333e−3	6.0000e−4	3.0000e−4	3.6667e−4
JLASSO	2.9667e−3	8.4333e−3	1.5667e−3	1.3667e−3	9.6667e−4
GLS-ESPRIT	8.0167e−2	5.9167e−2	3.3900e−2	3.9333e−3	5.3333e−4
ANM-ESPRIT	8.7833e−2	6.6667e−2	4.2000e−3	3.2333e−3	2.9667e−3
NNM-ESPRIT	8.7800e−2	8.4333e−2	7.9000e−3	8.3000e−3	7.2333e−3
SS-ESPRIT	6.4333e−3	8.6333e−3	8.8333e−3	8.0000e−3	7.9333e−3
$P_{f,\,2}$	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	5.6900e−1	7.8500e−1	1.0700e−1	4.9000e−2	4.5000e−2
SAA-CBCMA	5.1100e−1	6.2750e−1	7.7500e−2	2.9500e−2	2.8000e−2
JLASSO	2.6150e−1	3.9950e−1	6.4000e−2	3.6000e−2	2.9500e−2
GLS-ESPRIT	4.1945	2.5960	1.4835	2.1450e−1	1.0000e−1
ANM-ESPRIT	4.8530	3.4300	9.6250e−1	6.3400e−1	5.4150e−1
NNM-ESPRIT	4.6655	3.8330	5.7500e−1	4.9950e−1	4.2250e−1
SS-ESPRIT	1.8065	1.3785	1.1120	1.1450	1.1720
CPU time	−5 dB	0 dB	5 dB	10 dB	15 dB
SAM-CBCMA	6.4338e−2	1.1487e−1	9.0203e−2	8.1604e−2	8.0568e−2
SAA-CBCMA	7.8952e−2	1.4470e−1	1.3355e−1	1.3179e−1	1.3171e−1
JLASSO	1.0321	1.0308	9.7341e−1	1.0235	9.8989e−1
GLS-ESPRIT	6.5929e−1	6.8463e−1	7.4989e−1	7.7167e−1	8.3656e−1
ANM-ESPRIT	9.6501e−1	1.0085	1.1442	1.1335	1.1316
NNM-ESPRIT	8.9919	9.4239	1.3658e+1	1.3382e+1	1.3484e+1
SS-ESPRIT	7.0312e−4	8.9063e−4	9.6094e−4	9.0625e−4	7.3438e−4
Model order	-5 dB	0 dB	5 dB	10 dB	15 dB
GLS-ESPRIT	$\frac{8.8113}{1.5290e+1}$	$\frac{5.0543}{1.6770e+1}$	$\frac{2.9477}{1.6464e+1}$	$\frac{8.8081e-1}{1.5208e+1}$	$\frac{4.1628e-1}{1.5099e+1}$
ANM-ESPRIT	$\frac{8.7452}{1.6398e+1}$	$\frac{4.7126}{1.7976e+1}$	$\frac{1.5421}{1.5919e+1}$	$\frac{7.0750e-1}{1.5633e+1}$	$\frac{5.8704e-1}{1.5540e+1}$
NNM-ESPRIT	$\frac{8.7183}{1.6112e+1}$	$\frac{4.5394}{1.8471e+1}$	$\frac{1.2403}{1.5522e+1}$	$\frac{5.7976e-1}{1.5493e+1}$	$\frac{5.2209e-1}{1.5419e+1}$
SS-ESPRIT	$\frac{5.6644}{1.1100e+1}$	$\frac{3.9258}{1.3020e+1}$	$\frac{2.5124}{1.3659e+1}$	$\frac{2.1537}{1.3374e+1}$	$\frac{2.1570}{1.3242e+1}$

spectra of SAA-CBCMA and SAM-CBCMA are also similar to the ground truth while JLASSO, ANM-ESPRIT, and SS-ESPRIT yield relatively inaccurate spectra.

Remark 2:. Although our methods and the ANM method using the semidefinite programming are derived from the same criterion, our methods may perform worse than the ANM method, as shown in Fig. 4, because of their own limitations requiring, in order to ensure that (6) is valid and so is the problem (9), that many snapshots[11] be retained and that multiple sources be uncorrelated. On the other hand, the ANM approach is free from such limitations, and guarantees the unique optimal solution in atomic decomposed form by exploiting the Hermitian Toeplitz structure of the data covariance matrix when SNR goes to infinity, the number of sensors is sufficiently large, and the DOAs of plane waves are sufficiently different from each other. Further discussion is found in [31,47].

Figs. 6–11 depict the convergence behaviors of the SAM-CBCMA and SAA-CBCMA algorithms. As a convergence metric, we use δ_k defined as the l_2 norm of the difference between the estimated pairs (i.e., estimated coordinate blocks) before and after kth update. We observe from the figures a common trend that the curves of δ_k become flat at about 10^{-6}, which is the pre-set tolerance for estimating u_k, $k = 1, \dots, N$. This implies that up to the predefined tolerance, each algorithm converges to a stationary point of the cost function. In Figs. 6–8 that show the convergence behavior of SAM-CBCMA, strong peaks that are caused by merge process are observed after the metric δ_k becomes small, i.e., when the current update calculation makes a small perturbation of an estimated pair. Recall that once the merge step is performed, all the pairs satisfying the merge condition but one are forced to have zero $\hat{x}_k$. Accordingly, the difference between the current δ_k and δ_{k-1} can be significant and consequently strong peaks can appear in such cases. This is mainly observed not only in Fig. 8 where $K = 15$ but also in Fig. 7 which deals with only two targets. The latter takes place because the two targets are close to each other and, therefore, the Fourier responses (approximately sinc functions) induced by two similar u_i and u_j, $i \neq j$ have near-overlapping sidelobes as well as near-overlapping mainlobes. x_l corresponding to these sidelobes are eliminated by thresholding in the update equation (13) or by merging. Therefore, in this case, the merge process needs to be performed more frequently. Figs. 9–11 show the convergence behavior of SAA-CBCMA, displaying that fluctuations caused by activated atoms are observed in early update calculations. Note that δ_k has a relatively large value right after an atom is activated, and then becomes stabilized. So, we can infer from the number of the crests of fluctuations (at early update calculations) how many atoms are activated. As expected, the 15-target case shown in Fig. 11 exhibits the most crests.

[11] It is worthy to note that given finite snapshots, there always exists a gap between $\mathbf{R_y}$ in (6) and the sample covariance matrix (even in absence of noise).



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 5. Power spectrum for each algorithm over 20 Monte Carlo runs. The ground truth with power [1, 0.9, 0.5. 0.5] and DOA [0.45, 0.41, -0.1, -0.2]. For comparison, we set the SNR to 10 dB and plot only the detected peaks for SAA-CBCMA, SAM-CBCMA, and JLASSO, and the peaks with perfect model order selection for the other ones.

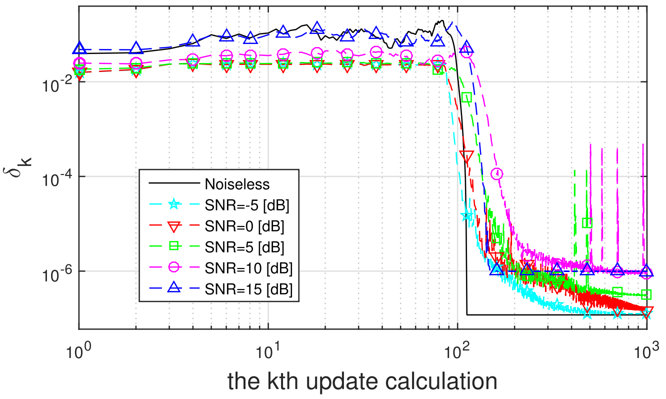


Fig. 6. Convergence of the SAM-CBCMA in the $K = 1$ target case.

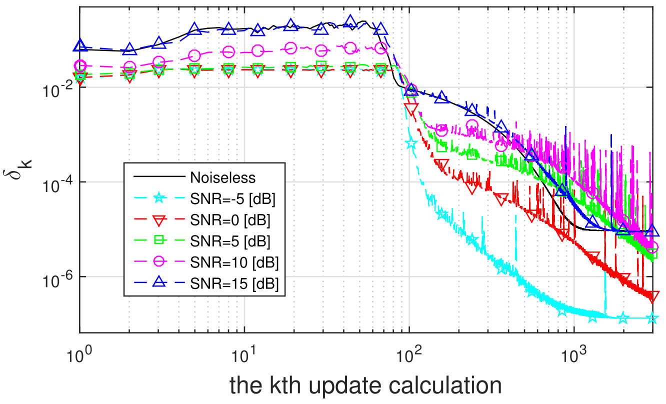


Fig. 7. Convergence of the SAM-CBCMA in the $K = 2$ target case.

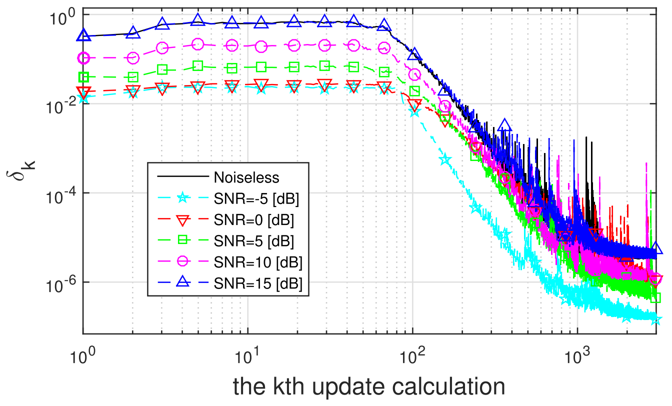


Fig. 8. Convergence of the SAM-CBCMA in the $K = 15$ target case.

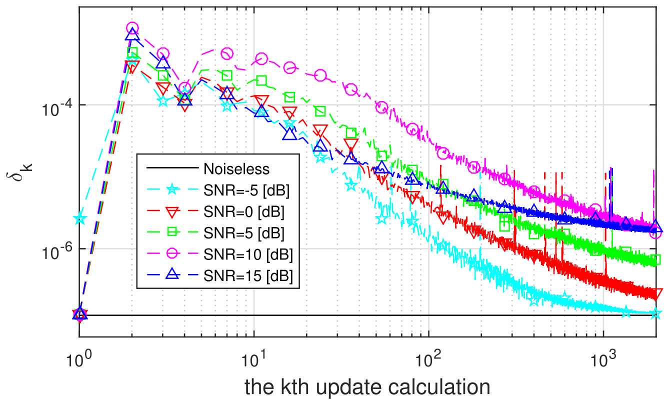


Fig. 9. Convergence of the SAA-CBCMA in the $K = 1$ target case.

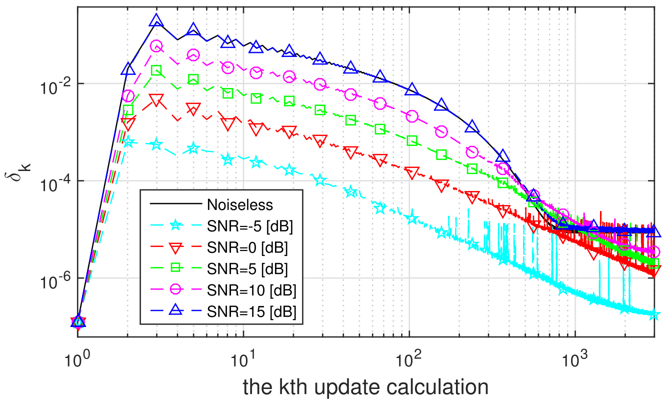


Fig. 10. Convergence of the SAA-CBCMA in the $K = 2$ target case.

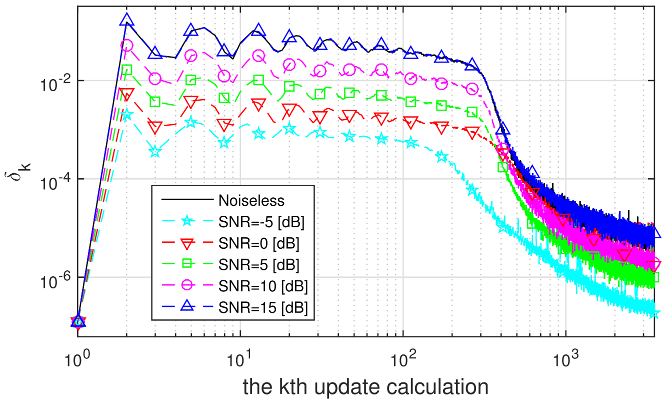


Fig. 11. Convergence of the SAA-CBCMA in the $K = 15$ target case.

5. Conclusions

We proposed two algorithms, SAM-CBCMA and SAA-CBCMA, for solving nonnegative gridless compressive sensing problems that arise in DOA estimation with co-prime arrays. The convergence analysis shows that the proposed algorithms are capable of finding a stationary point satisfying the KKT condition of atomic norm minimization. We numerically compared the performance of the proposed methods with those of JLASSO, GLS-ESPRIT, ANM-ESPRIT, NNM-ESPRIT, and SS-ESPRIT in terms of several criteria. The simulation results demonstrate that SAM-CBCMA and SAA-CBCMA compare favorably with prior methods in terms of computational efficiency, detectability, resolvability, and the accuracy of the estimates. Moreover, the convergence behavior of the proposed methods is tested numerically. SAM-CBCMA and SAA-CBCMA may be used in a broader range of applications including hyperspectral unmixing [48] and object recognition [49].

Acknowledgements

The authors would like to thank the editor and the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. They are also grateful to Dr. Ashkan Panahi for providing his Matlab code of the algorithm [33].

References

[1] A. Wang, L. Liu, J. Zhang, Low complexity direction of arrival (DOA) estimation for 2d massive MIMO systems, in: Globecom Workshops (GC Wkshps), 2012 IEEE, IEEE, 2012, pp. 703–707.
[2] M. Jin, G. Liao, J. Li, Joint DOD and DOA estimation for bistatic MIMOradar, Signal Process. 89 (2) (2009) 244–251.
[3] K.T. Wong, M.D. Zoltowski, Extended-aperture underwater acoustic multisource azimuth/elevation direction-finding using uniformly but sparsely spaced vector hydrophones, IEEE J. Ocean. Eng. 22 (4) (1997) 659–672.
[4] P. Palanisamy, N. Kalyanasundaram, P. Swetha, Two-dimensional doa estimation of coherent signals using acoustic vector sensor array, Signal Process. 92 (1) (2012) 19–28.
[5] S. Miron, N. Le Bihan, J.I. Mars, Vector-sensor music for polarized seismic sources localization, EURASIP J. Adv. Signal Process. 2005 (1) (2005) 74–84.
[6] J. Li, P. Stoica, L. Xu, W. Roberts, On parameter identifiability of MIMO radar, IEEE Signal Process. Lett. 14 (12) (2007) 968–971.
[7] A. Moffet, Minimum-redundancy linear arrays, IEEE Trans.AntennasPropag. 16 (2) (1968) 172–175.
[8] R.T. Hoctor, S.A. Kassam, The unifying role of the coarray in aperture synthesis for coherent and incoherent imaging, Proc. IEEE 78 (4) (1990) 735–752.
[9] S. Lang, G. Duckworth, J. McClellan, Array design for MEM and MLM array processing, in: Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'81., 6, IEEE, 1981, pp. 145–148.
[10] P. Pal, P. Vaidyanathan, Nested arrays: a novel approach to array processing with enhanced degrees of freedom, IEEE Trans. Signal Process. 58 (8) (2010) 4167–4181.
[11] P.P. Vaidyanathan, P. Pal, Sparse sensing with co-prime samplers and arrays, IEEE Trans. Signal Process. 59 (2) (2011) 573–586.
[12] P. Pal, P.P. Vaidyanathan, Coprime sampling and the music algorithm, in: Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE, IEEE, 2011, pp. 289–294.
[13] S. Qin, Y.D. Zhang, M.G. Amin, Generalized coprime array configurations for direction-of-arrival estimation, IEEE Trans. Signal Process. 63 (6) (2015) 1377–1390.
[14] Z.-Q. He, Z.-P. Shi, L. Huang, Covariance sparsity-aware doa estimation for nonuniform noise, Digit. Signal Process. 28 (2014) 75–81.
[15] Z. Tan, Y.C. Eldar, A. Nehorai, Direction of arrival estimation using co-prime arrays: a super resolution viewpoint, IEEE Trans. Signal Process. 62 (21) (2014) 5565–5576.
[16] Z. Tan, A. Nehorai, Sparse direction of arrival estimation using co-prime arrays with off-grid targets, IEEE Signal Process. Lett. 21 (1) (2014) 26–29.
[17] D.L. Donoho, J. Tanner, Sparse nonnegative solution of underdetermined linear equations by linear programming, Proc. Natl. Acad. Sci. U.S.A. 102 (27) (2005) 9446–9451.
[18] A.M. Bruckstein, M. Elad, M. Zibulevsky, On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations, IEEE Trans. Inf. Theory 54 (11) (2008) 4813–4820.
[19] D. Chen, R.J. Plemmons, Nonnegativity constraints in numerical analysis, in: The Birth of Numerical Analysis, 10, 2009, pp. 109–140.
[20] S. Foucart, D. Koslicki, Sparse recovery by means of nonnegative least squares, IEEE Signal Process. Lett. 21 (4) (2014) 498–502.
[21] Y. Chi, L.L. Scharf, A. Pezeshki, A.R. Calderbank, Sensitivity to basis mismatch in compressed sensing, IEEE Trans. Signal Process. 59 (5) (2011) 2182–2195.
[22] G. Tang, B.N. Bhaskar, P. Shah, B. Recht, Compressed sensing off the grid, IEEE Trans.Inf.Theory 59 (11) (2013) 7465–7490.
[23] E.J. Candès, C. Fernandez-Granda, Towards a mathematical theory of super-resolution, Commun. Pure Appl. Math. 67 (6) (2014) 906–956.
[24] E.J. Candès, C. Fernandez-Granda, Super-resolution from noisy data, J. Fourier Anal. Appl. 19 (6) (2013) 1229–1254.
[25] Z. Yang, L. Xie, C. Zhang, Off-grid direction of arrival estimation using sparse Bayesian inference, IEEE Trans. Signal Process. 61 (1) (2013) 38–43.
[26] X. Wu, W.-P. Zhu, J. Yan, Direction of arrival estimation for off-grid signals based on sparse Bayesian learning, IEEE Sens. J. 16 (7) (2016) 2004–2016.
[27] Y. Li, Y. Chi, Off-the-grid line spectrum denoising and estimation with multiple measurement vectors, IEEE Trans. Signal Process. 64 (5) (2016) 1257–1269.
[28] Y. Chen, Y. Chi, Spectral compressed sensing via structured matrix completion, in: International Conference on Machine Learning, 2013, pp. 414–422.
[29] Z. Yang, L. Xie, C. Zhang, A discretization-free sparse and parametric approach for linear array signal processing, IEEE Trans. Signal Process. 62 (19) (2014) 4959–4973.
[30] Z. Yang, L. Xie, On gridless sparse methods for multi-snapshot direction of arrival estimation, Circuits, Syst. Signal Process. (2016) 1–15.
[31] Z. Yang, J. Li, P. Stoica, L. Xie, Sparse methods for direction-of-arrival estimation, arXiv preprint arXiv:1609.09596 (2016).
[32] P. Stoica, R.L. Moses, et al., Spectral Analysis of Signals, 452, Pearson Prentice Hall Upper Saddle River, NJ, 2005.
[33] A. Panahi, M. Viberg, B. Hassibi, A numerical implementation of gridless compressed sensing, in: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on, IEEE, 2015, pp. 3342–3346.
[34] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, SIAM Rev. 43 (1) (2001) 129–159.
[35] T.T. Wu, K. Lange, Coordinate descent algorithms for lasso penalized regression, Ann. Appl. Stat. (2008) 224–244.
[36] X. Li, T. Zhao, R. Arora, H. Liu, M. Hong, An improved convergence analysis of cyclic block coordinate descent-type methods for strongly convex minimization, in: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016, pp. 491–499.
[37] P. Tseng, Convergence of a block coordinate descent method for non-differentiable minimization, J.Optim.TheoryAppl. 109 (3) (2001) 475–494.
[38] Z. Yang, L. Xie, Enhancing sparsity and resolution via re-weighted atomic norm minimization, IEEE Trans. Signal Process. 64 (4) (2016) 995–1006.

[39] H. Yang, J. Chun, H. Vikalo, Nonnegative gridless compressive sensing for co-prime arrays, in: Signal and Information Processing (GlobalSIP), 2016 IEEE Global Conference on, IEEE, 2016, pp. 1364–1367.

[40] C.L. Lawson, R.J. Hanson, Solving least squares problems, SIAM, 1995.

[41] W.F. Trench, Introduction to Real Analysis, Pearson Education, Upper Saddle River, New Jersey, 2003 (free online version 2013).

[42] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[43] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, in: Foundations and Trends® in Machine Learning, 3(1), 2011, pp. 1–122.

[44] F. Cong, A.K. Nandi, Z. He, A. Cichocki, T. Ristaniemi, Fast and effective model order selection method to determine the number of sources in a linear transformation model, in: Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European, IEEE, 2012, pp. 1870–1874.

[45] Z. He, A. Cichocki, S. Xie, K. Choi, Detecting the number of clusters in n-way probabilistic clustering, IEEE Trans. Pattern Anal. Mach.Intell. 32 (11) (2010) 2006–2021.

[46] B. Ottersten, P. Stoica, R. Roy, Covariance matching estimation techniques for array signal processing applications, Digital Signal Process. 8 (3) (1998) 185–210.

[47] Z. Yang, L. Xie, Exact joint sparse frequency recovery via optimization methods, IEEE Trans. Signal Process. 64 (19) (2014) 5145–5157.

[48] M.-D. Iordache, J.M. Bioucas-Dias, A. Plaza, Sparse unmixing of hyperspectral data, IEEE Trans. Geosci. Remote Sens. 49 (6) (2011) 2014–2039.

[49] R. He, W.-S. Zheng, B.-G. Hu, X.-W. Kong, Two-stage nonnegative sparse representation for large-scale face recognition, IEEE Trans.Neural Netw.Learn.Syst. 24 (1) (2013) 35–46.