

Bayesian Algorithms for Kronecker-Structured Sparse Vector Recovery With Application to IRS-MIMO Channel Estimation

He, Yanbin; Joseph, Geethu

DOI

[10.1109/TSP.2024.3416543](https://doi.org/10.1109/TSP.2024.3416543)

Publication date

2025

Document Version

Final published version

Published in

IEEE Transactions on Signal Processing

Citation (APA)

He, Y., & Joseph, G. (2025). Bayesian Algorithms for Kronecker-Structured Sparse Vector Recovery With Application to IRS-MIMO Channel Estimation. *IEEE Transactions on Signal Processing*, 73, 142-157. <https://doi.org/10.1109/TSP.2024.3416543>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

<https://www.openaccess.nl/en/you-share-we-take-care>

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Bayesian Algorithms for Kronecker-Structured Sparse Vector Recovery With Application to IRS-MIMO Channel Estimation

Yanbin He , *Graduate Student Member, IEEE* and Geethu Joseph , *Senior Member, IEEE*

Abstract—We study the sparse recovery problem with an underdetermined linear system characterized by a Kronecker-structured dictionary and a Kronecker-supported sparse vector. We cast this problem into the sparse Bayesian learning (SBL) framework and rely on the expectation-maximization method for a solution. To this end, we model the Kronecker-structured support with a hierarchical Gaussian prior distribution parameterized by a Kronecker-structured hyperparameter, leading to a non-convex optimization problem. The optimization problem is solved using the alternating minimization (AM) method and a singular value decomposition (SVD)-based method, resulting in two algorithms. Further, we analytically guarantee that the AM-based method converges to the stationary point of the SBL cost function. The SVD-based method, though it adopts approximations, is empirically shown to be more efficient and accurate. We then apply our algorithm to estimate the uplink wireless channel in an intelligent reflecting surface-aided MIMO system and extend the AM-based algorithm to address block sparsity in the channel. We also study the SBL cost to show that the minima of the cost function are achieved at sparse solutions and that incorporating the Kronecker structure reduces the number of local minima of the SBL cost function. Our numerical results demonstrate the effectiveness of our algorithms compared to the state-of-the-art.

Index Terms—Compressed sensing, sparse Bayesian learning, block sparsity, alternating minimization, singular value decomposition, Kronecker product, convergence analysis, IRS-aided MIMO channel estimation.

I. INTRODUCTION

THE basis expansion model (BEM) is widely used in various fields, such as image processing [2], [3], [4], [5] and wireless communications [6], [7], [8], to obtain flexible linear representations for non-linear functions. It is achieved by

approximating non-linear functions as linear combinations of simple *basis functions* [9]. BEM comprises a linear model with a *dictionary* of basis functions and *coefficients* associated with the dictionary. This paradigm, united with compressed sensing (CS), can be employed to estimate the unknown parameters of non-linear functions. The idea is to sample the unknown parameter over a range to construct known functions that comprise an over-complete dictionary. Here, only a few basis functions corresponding to the true parameters are activated, resulting in sparse coefficients. Thus, CS techniques can be leveraged to reconstruct the sparse coefficients from the linear model. Further, many signal representation problems in wireless communications [1], [10], [11], [12] and image processing [13], [14] applications use multidimensional BEM [15], [16], [17], [18], [19]. Multidimensional BEMs are typically associated with Kronecker-structured dictionaries, i.e., the Kronecker product of multiple factor dictionary matrices. Each factor dictionary represents a different dimension and acts as the overcomplete basis for the respective dimension. The inherent nature of the Kronecker product calculates all possible combinations of the parameter values across the factor dictionaries. As a result, the combined dictionary captures every conceivable combination of basis vectors (values of the unknown parameters) across multiple dimensions. We investigate the CS problem of solving a high-dimensional underdetermined linear system of equations characterized by a multidimensional Kronecker dictionary.

Moreover, in many cases, the sparse coefficient can be a sparse vector with Kronecker-structured support. Since in the multidimensional basis expansion model, there are only a few true values among all possible parameters, the coefficient vector is sparse in each dimension. Then, the support of the sparse coefficient vector is the Kronecker product of several low-dimensional support vectors. Some motivating applications are in wireless communications [1], [11], [12] and image processing [13], [14], [15], [16], [19]. As an example, consider the channel estimation problem for the intelligent reflecting surfaces (IRS)-aided wireless communication systems [1]. The BEM for the unknown channel matrix is constructed by sampling pre-defined spatial angle grids and forming a dictionary using the corresponding steering vectors. Then, the channel matrix can be represented using a three-dimensional sparse BEM coefficient vector where the three dimensions (mode) are the angle-of-departure (AoD), angle-of-arrival (AoA), and

Received 26 July 2023; revised 12 April 2024; accepted 9 June 2024. Date of publication 19 June 2024; date of current version 27 December 2024. An earlier version of this paper was presented at the IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP), 2023 [DOI: 10.1109/ICASSP49357.2023.10095932]. The associate editor coordinating the review of this article and approving it for publication was Samet Oymak. (*Corresponding author: Geethu Joseph.*)

The authors are with the Signal Processing Systems Group, Electrical Engineering, Mathematics, and Computer Science Faculty, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: y.he-1@tudelft.nl; g.joseph@tudelft.nl).

Digital Object Identifier 10.1109/TSP.2024.3416543

difference of the AoA and AoD at the IRS. Furthermore, different combinations of AoDs and AoAs naturally elicit the Kronecker product, leading to a Kronecker-structured dictionary and sparse coefficients. Motivated by such multi-parameter estimation problems, we consider the following Kronecker-structured linear inversion problem,

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{y} \in \mathbb{C}^{\bar{M} \times 1}$ is the noisy measurement, $\mathbf{H} \in \mathbb{C}^{\bar{M} \times \bar{N}}$ is the Kronecker-structured dictionary with $\bar{M} < \bar{N}$, $\mathbf{x} \in \mathbb{C}^{\bar{N} \times 1}$ is the unknown sparse BEM coefficient vector, and \mathbf{n} is the measurement noise. Specifically,

$$\mathbf{H} = \mathbf{H}_1 \otimes \mathbf{H}_2 \otimes \cdots \otimes \mathbf{H}_I = \otimes_{i=1}^I \mathbf{H}_i, \quad (2)$$

where $\mathbf{H}_i \in \mathbb{C}^{M_i \times N_i}$ with $\prod_{i=1}^I M_i = \bar{M}$, and $\prod_{i=1}^I N_i = \bar{N}$. Operation \otimes is the Kronecker product. The sparse vector \mathbf{x} has Kronecker-structured support, i.e., the indices of the non-zero elements of the sparse vector \mathbf{x} can be represented as

$$\text{supp}(\mathbf{x}) = \otimes_{i=1}^I \text{supp}(\mathbf{x}_i), \quad (3)$$

where $\text{supp}(\mathbf{x})$ is the (binary) support vector indicating the nonzero entries of \mathbf{x} . We note that \mathbf{x} need not be represented as a Kronecker product of lower dimensional sparse vectors, as the nonzero entries need not be Kronecker-structured support. We only require that the support follows a Kronecker pattern. We aim to reconstruct the sparse vector \mathbf{x} given the Kronecker-structured dictionary \mathbf{H} , noisy measurement \mathbf{y} by exploiting its Kronecker-structured support.

Various signal processing techniques have been proposed to reconstruct the sparse vector \mathbf{x} from the linear measurement \mathbf{y} and Kronecker-structured dictionary \mathbf{H} . [16] designed a greedy method called Kronecker-orthogonal matching pursuit (KOMP) to generalize the traditional OMP for multidimensional signals. Like OMP, the KOMP algorithm has low complexity but requires hand-tuning of a sensitive stopping threshold. Recently, parameter tuning-free approaches based on sparse Bayesian learning (SBL) were studied [11]. SBL is known to have superior performance [20] and flexibility to incorporate several additional structures along with sparsity [21], [22], [23], [24]. The Kronecker structure can also be incorporated into the SBL framework. [13] introduced the SBL approach for the Kronecker structure in (2) with $I = 3$, using tensor-wise hyperparameters. The linear inversion problem with $I = 2$ was considered in [25]. This framework was later generalized to the I -dimensional tensor and applied to wireless communications [11], [12]. However, the derivation of the SBL algorithm in [11], [12] relied on several approximations, leading to a suboptimal recovery accuracy. Hence, we seek novel Bayesian algorithms exploiting the Kronecker and sparse structures to improve reconstruction accuracy and efficiency.

Furthermore, the existing Bayesian algorithms exploiting the Kronecker structure [11], [12] lack theoretical guarantees on their convergence and performance, and these properties are only demonstrated empirically. The study on convergence is limited because the convergence guarantee of the classic SBL using the expectation-maximization (EM) algorithm cannot be

trivially extended for these algorithms. [11], [12] claimed that the solution to the underlying optimization problem should be obtained at the stationary point of the cost function. However, it is unclear whether the stationary point can be reached due to its iterative nature and approximations. Similarly, the performance improvement due to incorporating the Kronecker structure into the SBL has been shown in [11] without any theoretical justification. Given a lack of theoretical analysis of the existing algorithms and analysis of the SBL cost function, we make progress on these problems by analyzing our new Bayesian algorithms.

Our contributions: We devote this paper to the algorithm development and convergence analysis of algorithms for the Kronecker-structured sparse recovery problem in (1). Our main contributions are as follows:

- *Algorithm development:* We present our new Bayesian recovery algorithms in Sec. II. We first present two novel SBL algorithms to recover sparse vector \mathbf{x} with Kronecker-structured support, called KroSBL. The first KroSBL algorithm, based on alternating minimization (AM), solves the underlying optimization problem of the SBL algorithm using the AM procedure. The second KroSBL algorithm, based on singular value decomposition (SVD), is faster and uses a simple approximation to obtain the SBL algorithm.
- *Application:* We apply our problem to a prototypical application of IRS-aided MIMO channel estimation in Sec. IV. Besides the sparsity with Kronecker-structured support, the BEM representation of IRS-cascaded channels also exhibits block sparsity where the nonzero entries occur in clusters. To handle this additional structure, we extend our AM-based algorithm for Kronecker-structured block sparsity based on non-negative least squares.
- *Convergence guarantee:* We derive convergence guarantees for the AM-based algorithm in Sec. III-A. We establish that *i)* the AM procedure can attain the stationary point of the cost function in the M-step, and *ii)* the AM-based algorithm is guaranteed to converge to the stationary point of the KroSBL cost function in the noisy setting. These results address the gap in existing works by providing theoretical guarantees, which were previously lacking.
- *Cost function analysis:* We examine the local minima of the KroSBL cost function in Sec. III-B. Beyond the structure that the sparse vector \mathbf{x} has a Kronecker-structured support, we assume the sparse vector in (1) satisfies $\mathbf{x} = \otimes_{i=1}^I \mathbf{x}_i$. Then, we prove that all local minima are sparse in the noiseless case. Besides, we demonstrate that incorporating the Kronecker structure in the hyperparameter vector as the KroSBL cost function can significantly reduce the local minima under the unique representation property (URP) assumption for \mathbf{H} .
- *Numerical Results:* We assess our schemes in two scenarios in Sec. V. Firstly, we study the sparse recovery performance of the presented schemes against algorithms in the literature to demonstrate their superior recovery accuracy and run time. Secondly, we conduct a case study on channel estimation for IRS-aided systems with our

approach and illustrate that incorporating block sparsity in our scheme can further enhance performance.

Overall, we present two algorithms for the sparse signal recovery problem in BEM with multiple unknown parameters. The first algorithm, AM-KroSBL, enjoys solid theoretical guarantees, while the other algorithm, SVD-KroSBL, is computationally light and practically more relevant.

Notation: Boldface small letters denote vectors, and boldface capital letters denote matrices. The symbols $[x]_i$, $[\mathbf{X}]_i$, and $[\mathbf{X}]_{ij}$ represent the i -th entry of vector \mathbf{x} , the i -th column of matrix \mathbf{X} , and the entry on the i -th row and j -th column of matrix \mathbf{X} , respectively. We denote the all-one vector with length N as $\mathbf{1}_N$. The symbol $\|\cdot\|_p$ denotes the vector ℓ_p norm and $\|\cdot\|_F$ is the matrix Frobenius norm. We use $\mathbf{x} > 0$ (or $\mathbf{x} \geq 0$) to denote that all the entries in \mathbf{x} are positive (or non-negative). Depending on the argument, the vector element-wise inversion or matrix inversion is denoted as $(\cdot)^{-1}$. If the argument is a vector, operator $\text{diag}(\cdot)$ returns a diagonal matrix with the argument along the diagonal, and it returns a vector of its diagonal entries if the argument is a matrix. The symbols $(\cdot)^T$, $(\cdot)^*$, $(\cdot)^H$, $|\cdot|$, and $(\cdot)^\dagger$ are the matrix operations of transpose, conjugate, conjugate transpose, determinant, and pseudo-inverse, respectively. Also, \otimes and \odot represent the Kronecker product and the Khatri-Rao products, respectively. The matrix \mathbf{I}_N denotes the identity matrix of size $N \times N$. We use $\mathcal{CN}(\mathbf{a}, \mathbf{B})$ to denote the complex Gaussian distribution with mean \mathbf{a} and covariance \mathbf{B} . The set of real, complex matrices of size $M \times N$ is represented by $\mathbb{R}^{M \times N}$ and $\mathbb{C}^{M \times N}$, respectively.

II. KRONECKER-STRUCTURED SPARSE BAYESIAN LEARNING

We study the model in (1) with additive Gaussian noise \mathbf{n} following $\mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I}_M)$. For simplicity, we assume that the noise variance σ^2 is known and $N_i = N$ for $i = 1, 2, \dots, I$. We relax the knowledge of noise variance in Sec. II-C. This section presents new recovery algorithms to solve for \mathbf{x} from (1), exploiting the Kronecker structure in the support.

Inspired by the SBL framework [20], we impose a fictitious sparsity promoting zero-mean Gaussian prior on \mathbf{x} with an unknown covariance matrix $\mathbf{\Gamma} \in \mathbb{R}^{N^I \times N^I}$. We construct the covariance matrix as $\mathbf{\Gamma} = \text{diag}(\boldsymbol{\gamma})$ with $\boldsymbol{\gamma} = \otimes_{i=1}^I \boldsymbol{\gamma}_i$ and $\boldsymbol{\gamma}_i \in \mathbb{R}^{N \times 1}$. We impose $\boldsymbol{\gamma} = \otimes_{i=1}^I \boldsymbol{\gamma}_i$ to mimic the Kronecker-structured support. Specifically,

$$p(\mathbf{x}; \{\boldsymbol{\gamma}_i\}) = \mathcal{CN}(\mathbf{0}, \mathbf{\Gamma}), \quad (4)$$

where $\{\boldsymbol{\gamma}_i\}$ is the simplified notation for $\{\boldsymbol{\gamma}_i\}_{i=1}^I$ used henceforth. Then, we turn to the type-II ML estimation, i.e., we first estimate the hyperparameters $\{\boldsymbol{\gamma}_i\}$, based on which the MAP estimate of \mathbf{x} obtained as $\arg \max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}; \{\boldsymbol{\gamma}_i\})$ [26]. The ML estimates of $\{\boldsymbol{\gamma}_i\}$ are obtained by minimizing the negative log-likelihood, i.e., the KroSBL cost function is given by

$$\mathcal{L}(\{\boldsymbol{\gamma}_i\}) = -\log p(\mathbf{y}; \{\boldsymbol{\gamma}_i\}) = \log |\boldsymbol{\Sigma}_{\mathbf{y}}| + \mathbf{y}^H \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{y}, \quad (5)$$

where $\boldsymbol{\Sigma}_{\mathbf{y}} = \sigma^2 \mathbf{I}_M + \mathbf{H} \mathbf{\Gamma} \mathbf{H}^H$ [20]. We note that $\boldsymbol{\gamma} = \otimes_{i=1}^I \boldsymbol{\gamma}_i = \otimes_{i=1}^I \alpha_i \boldsymbol{\gamma}_i$ for any $\alpha_i > 0$ when $\prod_{i=1}^I \alpha_i = 1$. Thus, if $\{\boldsymbol{\gamma}_i\}$ maximizes (5), then $\{\alpha_i \boldsymbol{\gamma}_i\}$ also achieves the

maximum for any $\alpha_i > 0$ with $\prod_i \alpha_i = 1$. To eliminate this scaling ambiguity, we normalize the hyperparameter vectors, i.e., we impose the constraint $\|\boldsymbol{\gamma}_i\|_2 = 1$ for $i = 1, 2, \dots, I-1$. So the ML problem to estimate $\{\boldsymbol{\gamma}_i\}$ reduces to

$$\min_{\{\boldsymbol{\gamma}_i\} \in \mathcal{C}} \mathcal{L}(\{\boldsymbol{\gamma}_i\}), \quad (6)$$

where we define the constraint set

$$\mathcal{C} = \left\{ \{\boldsymbol{\gamma}_i\} \mid \boldsymbol{\gamma}_i \geq 0, i = 1, 2, \dots, I, \|\boldsymbol{\gamma}_i\|_2 = 1, \forall i \neq I \right\}. \quad (7)$$

The problem in (6) does not admit a closed-form solution. Thus, we resort to the standard EM algorithm [20], [27] for an iterative solution. Specifically, the r th iteration of EM is

$$\mathbf{E}\text{-step: Compute } \mathbb{E}_{\mathbf{x}|\mathbf{y}; \boldsymbol{\gamma}^{(r)}} \{ \log [p(\mathbf{y}, \mathbf{x}; \{\boldsymbol{\gamma}_i\})] \}, \quad (8)$$

$$\mathbf{M}\text{-step: } \{\boldsymbol{\gamma}_i\}^{(r+1)} = \arg \max_{\{\boldsymbol{\gamma}_i\} \in \mathcal{C}_+} \mathbb{E}_{\mathbf{x}|\mathbf{y}; \boldsymbol{\gamma}^{(r)}} \{ \log [p(\mathbf{y}, \mathbf{x}; \{\boldsymbol{\gamma}_i\})] \}, \quad (9)$$

where $\boldsymbol{\gamma}^{(r)} = \otimes_{i=1}^I \boldsymbol{\gamma}_i^{(r)}$ is the estimate in the r th iteration and

$$\mathcal{C}_+ = \left\{ \{\boldsymbol{\gamma}_i\} \in \mathcal{C} \mid \boldsymbol{\gamma}_i > 0, i = 1, 2, \dots, I \right\}. \quad (10)$$

Here, we restrict the feasible set in (9) to \mathcal{C}_+ instead of \mathcal{C} in (6) to avoid degenerate distributions. We also initialize the algorithm with $\{\boldsymbol{\gamma}_i\}^{(1)} \in \mathcal{C}_+$, for example, $\forall i = 1, \dots, I-1$, $\boldsymbol{\gamma}_i^{(1)} = \frac{1}{\sqrt{N}} \mathbf{1}$, and $\boldsymbol{\gamma}_I^{(1)} = \sqrt{N} \mathbf{1}$. Further, using straightforward algebraic simplifications [28], we can reduce (9) to

$$\{\boldsymbol{\gamma}_i\}^{(r+1)} = \arg \min_{\{\boldsymbol{\gamma}_i\} \in \mathcal{C}_+} \log |\text{diag}(\boldsymbol{\gamma})| + (\mathbf{d}^{(r)})^T \boldsymbol{\gamma}^{-1}, \quad (11)$$

where we define

$$\mathbf{d}^{(r)} = \text{diag}(\boldsymbol{\Sigma}_{\mathbf{x}} + \boldsymbol{\mu}_{\mathbf{x}} \boldsymbol{\mu}_{\mathbf{x}}^H). \quad (12)$$

Here, $\boldsymbol{\mu}_{\mathbf{x}}$ and $\boldsymbol{\Sigma}_{\mathbf{x}}$, which depend on $\boldsymbol{\gamma}^{(r)}$, are the mean and variance of conditional distribution $p(\mathbf{x}|\mathbf{y}; \boldsymbol{\gamma}^{(r)})$, respectively,

$$\boldsymbol{\mu}_{\mathbf{x}} = \sigma^{-2} \boldsymbol{\Sigma}_{\mathbf{x}} \mathbf{H}^H \mathbf{y}, \quad \boldsymbol{\Sigma}_{\mathbf{x}} = \left[\sigma^{-2} \mathbf{H}^H \mathbf{H} + \text{diag}(\boldsymbol{\gamma}^{(r)})^{-1} \right]^{-1}. \quad (13)$$

Here the dictionary matrix \mathbf{H} is Kronecker-structured, i.e., $\mathbf{H} = \otimes_{i=1}^I \mathbf{H}_i$. This special structure can be used to reduce the computational complexity of computing $\boldsymbol{\Sigma}_{\mathbf{x}}$ in (13) [1]. Let $Q(\{\boldsymbol{\gamma}_i\}|\boldsymbol{\gamma}^{(r)}) = \log |\text{diag}(\boldsymbol{\gamma})| + (\mathbf{d}^{(r)})^T \boldsymbol{\gamma}^{-1}$, which depends on $\boldsymbol{\gamma}^{(r)}$ via $\mathbf{d}^{(r)}$. Then, the M-step is written as

$$\min_{\{\boldsymbol{\gamma}_i\}} Q(\{\boldsymbol{\gamma}_i\}|\boldsymbol{\gamma}^{(r)}) \quad \text{s.t. } \boldsymbol{\gamma} = \otimes_{i=1}^I \boldsymbol{\gamma}_i, \{\boldsymbol{\gamma}_i\} \in \mathcal{C}_+. \quad (14)$$

The solution to (14) without the constraint $\boldsymbol{\gamma} = \otimes_{i=1}^I \boldsymbol{\gamma}_i$ is straightforward [20]. However, since we adopt a Kronecker-structured prior (4), the Kronecker constraint poses a challenge to derive a closed-form solution. To solve (14) with the

Algorithm 1: AM-KroSBL

Data: Measurements \mathbf{y} , matrix \mathbf{H} , noise power σ^2

1 **Parameters:** Threshold ϵ and ϵ_{AM}

2 **Initialization:** set $r = 1$, $\{\gamma_i\}^{(0)} = \mathbf{0}$, $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$

3 **while** $\|\otimes_{i=1}^I \gamma_i^{(r)} - \otimes_{i=1}^I \gamma_i^{(r-1)}\|_2 > \epsilon$ **do**

4 Compute $\mathbf{d}^{(r)}$ using (12) and (13)

5 Set $t = 1$, $\{\gamma_i\}^{(r,1)} = \{\gamma_i\}^{(r)}$, $\{\gamma_i\}^{(r,0)} = \mathbf{0}$

6 **while** $\|\otimes_{i=1}^I \gamma_i^{(r,t)} - \otimes_{i=1}^I \gamma_i^{(r,t-1)}\|_2 > \epsilon_{\text{AM}}$ **do**

7 Compute $\{\gamma_i\}^{(r,t+1)}$ using (15) and (16)

8 Update AM iteration index $t \leftarrow t + 1$

9 **end**

10 $\{\gamma_i\}^{(r+1)} = \{\gamma_i\}^{(r,t)}$

11 Update iteration index $r \leftarrow r + 1$

12 **end**

Result: Output $\mathbf{x} = \boldsymbol{\mu}_{\mathbf{x}}$ using (13)

Kronecker constraint, two distinct ways are presented: AM-based and SVD-based approaches.

A. AM-Based KroSBL (AM-KroSBL)

AM-KroSBL solves (14) by alternately updating one hyperparameter vector while keeping the others fixed. We first compute the gradient of cost function¹ $Q(\{\gamma_i\})$ with respect to $\{\gamma_i\}$ and set it to zero, leading to

$$\tilde{\gamma}_i = N^{-I+1} \left[(\otimes_{j=1}^{i-1} (\tilde{\gamma}_j)^{-1}) \otimes \mathbf{I}_N \otimes (\otimes_{j=i+1}^I (\gamma_j^{(r,t)})^{-1}) \right]^T \mathbf{d}^{(r)}, \quad (15)$$

for $i = 1, 2, \dots, I$, with $\gamma_i^{(r,t)}$ is the estimate in the t th iteration of AM and the r th iteration of EM. To avoid the scaling ambiguity, in each iteration t , we project the hyperparameter vectors to \mathcal{C}_+ as

$$\gamma_i^{(r,t+1)} = \begin{cases} \frac{\tilde{\gamma}_i}{\|\tilde{\gamma}_i\|_2} & \text{for } i = 1, 2, \dots, I-1 \\ \prod_{j=1}^{I-1} \|\tilde{\gamma}_j\|_2 \tilde{\gamma}_I & \text{for } i = I. \end{cases} \quad (16)$$

The projection (16) does not change the cost function value Q as $\otimes_{i=1}^I \gamma_i^{(r,t+1)} = \otimes_{i=1}^I \tilde{\gamma}_i$. The steps are summarized in Algorithm 1. The AM-KroSBL is guaranteed to improve the cost function given by (11) after every iteration. However, due to its iterative nature, it is computationally inefficient. Thus, we next present a non-iterative method to solve (14) using SVD.

B. SVD-Based KroSBL (SVD-KroSBL)

This method solves (14) without the constraint $\gamma = \otimes_{i=1}^I \gamma_i$ first and then projects the solution to the constraint set. We note from [20] that

$$\arg \min_{\boldsymbol{\gamma}} \log |\text{diag}(\boldsymbol{\gamma})| + (\mathbf{d}^{(r)})^T \boldsymbol{\gamma}^{-1} = \mathbf{d}^{(r)}. \quad (17)$$

¹We interchangeably use $Q(\{\gamma_i\}|\boldsymbol{\gamma}^{(r)})$, $Q(\{\gamma_i\})$, and Q in the paper.

Algorithm 2: SVD-KroSBL

Data: Measurements \mathbf{y} , matrix \mathbf{H} , noise power σ^2

1 **Parameters:** Threshold ϵ

2 **Initialization:** set $r = 1$, $\{\gamma_i\}^{(0)} = \mathbf{0}$, $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$

3 **while** $\|\otimes_{i=1}^I \gamma_i^{(r)} - \otimes_{i=1}^I \gamma_i^{(r-1)}\|_2 > \epsilon$ **do**

4 Compute $\mathbf{d}^{(r)}$ using (12) and (13)

5 **for** $i = 1, \dots, I-1$ **do**

6 Solve (18) for $\gamma_i^{(r+1)}$

7 **end**

8 Update iteration index: $r \leftarrow r + 1$

9 **end**

Result: Output $\mathbf{x} = \boldsymbol{\mu}_{\mathbf{x}}$ using (13)

To project the above solution to the constraint set, we solve for $\{\gamma_i\}$ that minimizes $\|\mathbf{d}^{(r)} - \otimes_{i=1}^I \gamma_i\|_2$. We further approximate this optimization problem as $(I-1)$ rank-1 approximations:

$$\gamma_i^{(r+1)} = \arg \min_{\gamma_i: \|\gamma_i\|_2=1, \tilde{\gamma}_i \in \mathbb{R}^{N(I-i)}} \|\tilde{\gamma}_{i-1} - \gamma_i \otimes \tilde{\gamma}_i\|_2, \quad \forall i = 1, 2, \dots, I-1, \quad (18)$$

where $\tilde{\gamma}_0 = \mathbf{d}^{(r)}$ and $\tilde{\gamma}_{I-1} = \gamma_I$. Since $\gamma_i \otimes \tilde{\gamma}_i = \text{vec}(\tilde{\gamma}_i \gamma_i^T)$, we rearrange $\tilde{\gamma}_{i-1} \in \mathbb{R}^{N(I-i+1) \times 1}$ into a matrix $\mathbf{D}^{(i)} \in \mathbb{R}^{N(I-i) \times N}$ such that $\text{vec}(\mathbf{D}^{(i)}) = \tilde{\gamma}_{i-1}$. Thus, we rewrite (18) as

$$\gamma_i^{(r+1)} = \arg \min_{\gamma_i: \|\gamma_i\|_2=1, \tilde{\gamma}_i \in \mathbb{R}^{N(I-1)}} \|\mathbf{D}^{(i)} - \tilde{\gamma}_i \gamma_i^T\|_{\text{F}}, \quad (19)$$

where $\|\cdot\|_{\text{F}}$ is the matrix Frobenius norm. As matrix $\tilde{\gamma}_i \gamma_i^T$ is a rank-1 matrix, solving problem (19) is equivalent to finding the rank-1 approximation of $\mathbf{D}^{(i)}$. The optimal solution to $\gamma_i^{(r+1)}$ is the first right singular vector of $\mathbf{D}^{(i)}$, satisfying constraint $\|\gamma_i\|_2 = 1$. So, the above problem can be solved using the SVD algorithm. This approach is summarized in Algorithm 2.

C. Noise Variance Estimation

The derivations so far assume the knowledge of the noise variance. However, in practical scenarios, the noise variance is not always provided and needs to be estimated. To compute the noise variance, we have added the noise variance as a hyperparameter of the EM algorithm and derived the corresponding update equations. The resulting M-step optimization problem is separable in $\boldsymbol{\gamma}$ and σ^2 , and it leads to an additional noise variance step in the EM iteration without changing the update equation for $\boldsymbol{\gamma}$. Specifically, in the r th EM iteration, we have

$$\begin{aligned} & (\sigma^2)^{(r+1)} \\ &= \frac{\|\mathbf{y} - \mathbf{H}\boldsymbol{\mu}_{\mathbf{x}}\|_2^2 + (\sigma^2)^{(r)} [N^I - \text{tr}(\boldsymbol{\Sigma}_{\mathbf{x}} \text{diag}(\boldsymbol{\gamma}^{(r)})^{-1})]}{\bar{M}}, \end{aligned} \quad (20)$$

and we replace the known noise variance σ^2 in (13) with the estimated value $(\sigma^2)^{(r)}$.

III. THEORETICAL ANALYSIS OF KROSBL

This section focuses on the theoretical analysis of KroSBL. We discuss the convergence guarantee for AM-KroSBL in Sec. III-A. Then, we present our results on the values to which the algorithm converges by studying the local minima of the KroSBL cost function in (5).

A. Convergence Property of AM-KroSBL

The convergence of AM-KroSBL is established using the properties of the EM algorithm, which is well studied in [29]. It is known that under certain conditions, the EM algorithm guarantees convergence to stationary points of \mathcal{L} . Nonetheless, the guarantees of the EM algorithm in KroSBL depend on the convergence of the AM algorithm (inner loop). So, this section answers two questions: *What are the convergence properties of the AM algorithm? Do the properties of AM guarantee the convergence of AM-KroSBL?* The first question is answered by Proposition 1, serving as a cornerstone to the answer to the second question via Theorem 1. We first introduce a lemma that supports the main results.

Lemma 1: Consider the AM algorithm that solves the M-step optimization problem of the r th EM iteration of AM-KroSBL (Algorithm 1) given by (14), for a fixed iteration index $r > 0$. If $\mathbf{d}^{(r)} > 0$ in (12), then the cost function sequence $Q(\{\gamma_i\}^{(r,t)})_{t=1}^{\infty}$ generated by the AM algorithm is non-increasing and convergent.

Proof: The non-increasing nature of sequence $Q(\{\gamma_i\}^{(r,t)})_{t=1}^{\infty}$ is because the AM algorithm in every iteration optimizes one hyperparameter vector while keeping the others fixed, i.e.,

$$\begin{aligned} Q(\{\gamma_i\}^{(r,t)}) &\geq Q(\tilde{\gamma}_1, \{\gamma_i^{(r,t)}\}_{i=2}^I) \geq Q(\{\tilde{\gamma}_i\}_{i=1}^I, \{\gamma_i^{(r,t)}\}_{i=3}^I) \\ &\geq Q(\{\tilde{\gamma}_i\}) = Q(\{\gamma_i\}^{(r,t+1)}), \end{aligned} \quad (21)$$

where the last step follows because the projection step in (16) does not change the cost function value. Further, when $\mathbf{d}^{(r)} > 0$, we have

$$Q(\{\gamma_i\}) = \sum_{j=1}^{N^I} \log[\gamma]_j + [\mathbf{d}^{(r)}]_j [\gamma]_j^{-1} \geq \sum_{j=1}^{N^I} \log[\mathbf{d}^{(r)}]_j + N^I. \quad (22)$$

The non-increasing sequence $Q(\{\gamma_i\}^{(r,t)})_{t=1}^{\infty}$ is bounded from below and thus convergent. \square

Although the sequence of the cost function $Q(\{\gamma_i\}^{(r,t)})_{t=1}^{\infty}$ converges, this does not indicate any convergence property of its iterates, i.e., $\{\gamma_i\}^{(r,t)}_{t=1}^{\infty}$ [30]. The following proposition uses the above lemma to show that the iterates of the AM algorithm converge.

Proposition 1: [AM algorithm convergence] Consider the AM algorithm that solves the M-step optimization problem of the r th EM iteration of AM-KroSBL (Algorithm 1) given by (14), for a fixed iteration index $r > 0$. If $\mathbf{d}^{(r)} > 0$ in (12), then the sequence $\{\gamma_i\}^{(r,t)}_{t=1}^{\infty}$ converges to the set of stationary points of the M-step cost function $Q(\{\gamma_i\})$ in \mathcal{C}_+ .

Proof: See Appendix A. \square

It is interesting to note that the projection step (16) is coupled for $i = I$, because $\gamma_I = \prod_{j=1}^{I-1} \|\tilde{\gamma}_j\|_2 \tilde{\gamma}_I$ involves computing the ℓ_2 norm of $\tilde{\gamma}_i$ for $\forall i = 1, \dots, I-1$. However, this coupled projection step does not affect the convergence of the sequence $\{\gamma_i\}^{(r,t)}_{t=1}^{\infty}$. The reason is that according to Zangwill's convergence theorem [31] used to prove our results, regardless of how to produce the sequence $\{\gamma_i\}^{(r,t)}_{t=1}^{\infty}$, if $\{\gamma_i\}^{(r,t)}_{t=1}^{\infty}$ satisfies the required properties (see Appendix A), the sequence converges to the stationary point. The AM procedure with projection is a way of producing such a sequence and possesses the required properties. Nonetheless, we note that Proposition 1 only guarantees that the AM algorithm converges to a stationary point, which is not necessarily a global minimum. However, the following result establishes that the convergence of AM to a stationary point is sufficient to ensure the convergence of AM-KroSBL.

Theorem 1: Consider the model in (1) with the assumptions *i)* the noise variance $\sigma^2 > 0$, *ii)* there exists $\epsilon > 0$ such that the dictionary satisfies $\|[\mathbf{H}]_i\|_2 > \epsilon$, for $i = 1, 2, \dots, N^I$, and *iii)* the starting point of AM-KroSBL $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$. Then, the sequence $\{\gamma_i\}^{(r)}_{r=1}^{\infty}$ generated by AM-KroSBL (Algorithm 1) converges to the set of the stationary points of its cost function \mathcal{L} given by (5).

Proof: See Appendix B. \square

We note that the assumptions of Theorem 1 are realistic. Regarding assumption *i)*, noise cannot be avoided in practice. It is also common to assume $\sigma^2 > 0$ in the SBL analysis since it is difficult to discuss the behavior of EM-based SBL in the noiseless case [30], [32]. However, the noiseless case can be seen as a limiting case of $\sigma^2 \rightarrow 0$. Intuitively, we can set σ^2 as a very small but nonzero value and drive this value arbitrarily to zero. Thus, asymptotically, the convergence property should also hold in the noiseless case. The assumption *ii)* on the dictionary holds when \mathbf{H} has no zero columns. If the norm of a column in \mathbf{H} is zero, indicating that all its elements are zero, then that column does not contribute to the measurement and can be removed.

Assumption *iii)* the starting point of AM-KroSBL $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$ is also easy to satisfy. It can be generated by projecting any $\{\gamma_i\} > 0$ using (16). Furthermore, it is necessary to have $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$ since the starting point should be feasible, and more importantly, $\{\gamma_i\}^{(1)} > 0$. Suppose that we initialize the n^* th entry of $\gamma_{i^*}^{(1)}$ with zero for some i^* and n^* , then then all the N^{I-1} entries in $\gamma^{(1)} = \otimes_{i=1}^I \gamma_i^{(1)}$ involving $[\gamma_{i^*}]_{n^*}$ are zeros. Let \mathcal{M} be the set of indices in $\otimes_{i=1}^I \gamma_i^{(1)}$ being zero. Then, the submatrix $[\Sigma_x]_{\mathcal{M}}$ is zero because from (13),

$$\Sigma_x = \Gamma^{(1)} - \Gamma^{(1)} \mathbf{H}^H \left(\sigma^2 \mathbf{I}_{N^I} + \mathbf{H} \Gamma^{(1)} \mathbf{H}^H \right)^{-1} \mathbf{H} \Gamma^{(1)}, \quad (23)$$

where the rows of $\Gamma^{(1)} = \text{diag}(\otimes_{i=1}^I \gamma_i^{(1)})$ indexed by \mathcal{M} are zero. Consequently, $[\mathbf{d}^{(1)}]_{\mathcal{M}}$ is also zero due to the following relation from (12) and (13),

$$\mathbf{d}^{(1)} = \text{diag} \left(\Sigma_x \left[\mathbf{I}_{N^I} + \sigma^{-4} \mathbf{H}^H \mathbf{y} \mathbf{y}^H \mathbf{H} \Sigma_x \right] \right). \quad (24)$$

Further, if $[\mathbf{d}^{(1)}]_{\mathcal{M}}$ is zero, then the second term in Q of (11) involving $[\gamma_{i^*}]_{n^*}$ vanishes, making the optimization problem (14) separable in γ_{i^*} and leading to $\arg \min \log[\gamma_{i^*}]_{n^*}$. This optimization drives $[\gamma_{i^*}]_{n^*} = 0$. This indicates that the entries of the hyperparameter $\{\gamma_{i^*}\}^{(r)}$ within the index set \mathcal{M} will remain zero throughout the EM iteration for $r = 1, 2, \dots$. Therefore, initializing $\{\gamma_{i^*}\}^{(1)}$ with zero intrinsically assumes that the subvector $[\mathbf{x}]_{\mathcal{M}}$ cannot be non-zero from the beginning since the associated hyperparameters $[\boldsymbol{\gamma}]_{\mathcal{M}}$ remain zero by initialization, which can be erroneous. Further, initialization with $\{\gamma_{i^*}\}^{(1)} > 0$ is common in the EM-based SBL algorithm [30], [33]. Since the M-step naturally involves element-wise inversion of the hyperparameter $\boldsymbol{\gamma}$. Hence, we assume that $\{\gamma_{i^*}\}^{(1)} \in \mathcal{C}_+$.

Furthermore, Proposition 1 suggests that $\{\gamma_{i^*}\}^{(r)} \in \mathcal{C}_+$ and thus $\{\gamma_{i^*}\}^{(r)} > 0$, which seems to contradict the expected sparsity of the estimates $\{\gamma_{i^*}\}^{(r)}$. However, $\{\gamma_{i^*}\}^{(r)} > 0$ only holds under the assumption $\mathbf{d}^{(r)} > 0$ and the sequence $\{\mathbf{d}^{(r)}\}_{r=1}^{\infty}$ belongs to an open set $\{\mathbf{d} | \mathbf{d} > 0\}$. From our experiments, we observe that the sequence converges to $\mathbf{d}^{(\infty)}$ that belongs to the boundary of the open set, leading to sparse $\{\gamma_{i^*}\} \in \mathcal{C} \setminus \mathcal{C}_+$. Intuitively, this behavior can be viewed as follows. If the n^* th entry of $\gamma_{i^*}^{(r)}$ goes to zero for some i^* and n^* , then, the submatrix $[\boldsymbol{\Sigma}_{\mathbf{x}}]_{\mathcal{M}}$ goes to zero because of (23) and (24) in the r th EM iteration. Conversely, suppose that $[\mathbf{d}^{(r)}]_{\mathcal{M}}$ goes to zero, then, the second term in Q of (11) involving $[\gamma_{i^*}]_{n^*}$ becomes $\log[\gamma_{i^*}]_{n^*}$ and drives $[\gamma_{i^*}]_{n^*}$ to zero. Thus, a sparse $\mathbf{d}^{(r)}$ encourages a sparse $\boldsymbol{\gamma}^{(r)}$ and vice versa, leading to a sparse convergent point $\{\gamma_{i^*}\}^{(\infty)} \in \mathcal{C} \setminus \mathcal{C}_+$.

B. Local Minima of KroSBL Cost Function

Having studied the convergence properties of the algorithm, we now look at the properties of the limit points. Unlike the previous section, the results of this section assume that the sparse vector is also Kronecker-structured, i.e., $\mathbf{x} = \otimes_{i=1}^I \mathbf{x}_i$. The first result of the section, Theorem 2, proves that all local minima $\{\gamma_{i^*}\}$ of the KroSBL cost function \mathcal{L} in (5) are sparse. Subsequently, in Theorem 3, we derive an upper bound on the number of local minima of the KroSBL cost function.

Theorem 2: In the noiseless setting, every local minimum of \mathcal{L} is achieved at a sparse solution $\{\gamma_{i^*}\}$, that is, $\|\gamma_{i^*}\|_0 \leq M_i$ for $i = 1, 2, \dots, I$, if the sparse vector is Kronecker-structured, i.e., $\mathbf{x} = \otimes_{i=1}^I \mathbf{x}_i$, for some $\mathbf{x}_i \in \mathbb{C}^N$.

Proof: See Appendix C. \square

Theorem 2 indicates that the local minimum is sparse, not because some hyperparameter vectors ($\boldsymbol{\gamma}_i$'s) are dense while others are sparse, leading to a sparse Kronecker product. Instead, it implies each $\boldsymbol{\gamma}_i$ generated by KroSBL is sparse, following our Kronecker-structured support model.

Now we discuss an upper bound for the number of local minima of \mathcal{L} . For this, we use the concept of unique representation property (URP). The matrix \mathbf{H} is said to satisfy URP if any subset of \bar{M} columns of \mathbf{H} is linearly independent [20]. If the dictionary \mathbf{H} satisfies the URP, the number of local minima of \mathcal{L} in (5) without the Kronecker-structured support constraint \mathcal{C}

(the classic SBL algorithm) is [20]

$$\mathcal{N}_{\text{SBL}} \leq \binom{N^I}{\bar{M}} - \sum_{p=1}^P \binom{N^I - D_p}{\bar{M} - D_p} + P \leq \binom{N^I}{\bar{M}}, \quad (25)$$

where D_p is ℓ_0 -norm of the p th degenerate sparse solution of the SBL cost function, and P is the number of sparse solution. When we impose the Kronecker-structured support constraint, the upper bound of the number of local minima decreases, as discussed next.

Theorem 3: Consider the model in (1) and assume that i) the noise variance $\sigma^2 = 0$, ii) \mathbf{H} satisfies the URP, and iii) there exist P_i degenerate sparse solutions $\mathbf{x}_{i,p}$, $p = 1, 2, \dots, P_i$ such that $\mathbf{y} = (\otimes_{i=1}^I \mathbf{H}_i)(\otimes_{i=1}^I \mathbf{x}_{i,p})$ and $\|\mathbf{x}_{i,p}\|_0 = D_{i,p} < M_i$. Then, the number of distinct local minima of the KroSBL cost \mathcal{L} in \mathcal{C} , denoted as \mathcal{N} , satisfies

$$\mathcal{N} \leq \prod_{i=1}^I \left(\binom{N}{M_i} - \sum_{p=1}^{P_i} \binom{N - D_{i,p}}{M_i - D_{i,p}} + P_i \right) \leq \prod_{i=1}^I \binom{N}{M_i}. \quad (26)$$

Proof: See Appendix D. \square

Theorem 3 and the result for classical SBL uses the same assumption, \mathbf{H} satisfies URP, to derive an upper bound for the distinct number of local minima. However, our result shows that \mathcal{N} is dominated by $\prod_{i=1}^I \binom{N}{M_i}$ while \mathcal{N}_{SBL} is dominated by $\binom{N^I}{\bar{M}} > \prod_{i=1}^I \binom{N}{M_i}$. Thus, incorporating the Kronecker structure can greatly diminish the solution space, explaining the better reconstruction performance of the KroSBL.

IV. APPLICATION AND EXTENSION

In this section, we discuss the application of our algorithm to channel estimation in an IRS-assisted MIMO system.

A. Cascaded Channel Estimation for IRS-Aided MIMO

Consider an uplink MIMO millimeter-wave/terahertz band system with a T -antenna transmitter mobile station (MS), an R -antenna receiver base station (BS), and an L -element uniform linear array IRS. Let $\boldsymbol{\Phi}_{\text{MS}} \in \mathbb{C}^{L \times T}$ and $\boldsymbol{\Phi}_{\text{BS}} \in \mathbb{C}^{R \times L}$ denote the MS-IRS and IRS-BS (narrowband) channels, respectively,

$$\boldsymbol{\Phi}_{\text{MS}} = \sum_{p=1}^{P_{\text{MS}}} \sqrt{\frac{LT}{P_{\text{MS}}}} \beta_{\text{MS},p} \mathbf{a}_L(\phi_{\text{MS},p}) \mathbf{a}_T(\alpha_{\text{MS}})^H, \quad (27)$$

$$\boldsymbol{\Phi}_{\text{BS}} = \sum_{p=1}^{P_{\text{BS}}} \sqrt{\frac{RL}{P_{\text{BS}}}} \beta_{\text{BS},p} \mathbf{a}_R(\alpha_{\text{BS},p}) \mathbf{a}_L(\phi_{\text{BS}})^H, \quad (28)$$

where P_{MS} and P_{BS} are the number of rays. Also, for any integer Q and angle ψ , steering vector $\mathbf{a}_Q(\psi) \in \mathbb{C}^{Q \times 1}$ with half-wavelength spacing is

$$\mathbf{a}_Q(\psi) = \frac{1}{\sqrt{Q}} [1 \quad e^{j\pi \cos \psi} \dots e^{j\pi(Q-1) \cos \psi}]^T. \quad (29)$$

The angles $\phi_{\text{MS},p}$, α_{MS} , $\alpha_{\text{BS},p}$, and ϕ_{BS} denote the p th AoA of the IRS, and the p th AoD of the MS, the p th AoA of the BS, and the AoD of the IRS, respectively (see Fig. 1). Then, the cascaded MS-IRS-BS channel is then given by $\boldsymbol{\Phi}_{\text{BS}} \text{diag}(\boldsymbol{\theta}) \boldsymbol{\Phi}_{\text{MS}}$

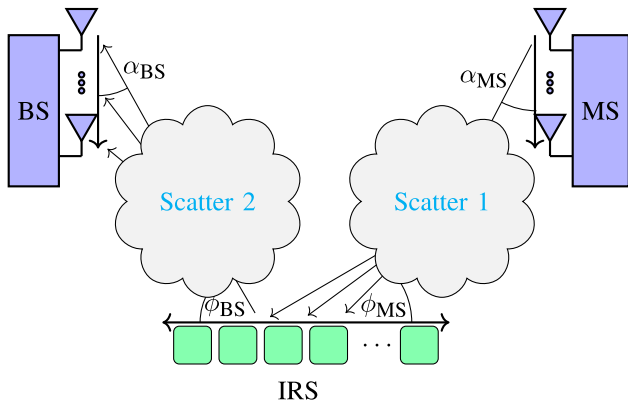


Fig. 1. An illustration of AoAs and AoDs in an IRS-aided channel.

for a given IRS configuration $\theta \in \mathbb{C}^{L \times 1}$. Here, the i th entry of θ represents the gain and phase shift due to the i th IRS element. We aim to estimate the cascaded channel $\Phi_{\text{BS}} \text{diag}(\theta) \Phi_{\text{MS}}$ for any θ .

To estimate the channel, we send pilot symbols over K time slots over which Φ_{MS} and Φ_{BS} is assumed to be constant. We choose $K_I < K$ IRS configurations, and for each configuration, we transmit pilot data $\mathbf{G} \in \mathbb{C}^{T \times K_P}$ over K_P time slots such that $K = K_I K_P$. Hence, the received signal $\mathbf{Y}_k \in \mathbb{C}^{R \times K_P}$ corresponding to the k th configuration θ_k is [34], [35]

$$\mathbf{Y}_k = \Phi_{\text{BS}} \text{diag}(\theta_k) \Phi_{\text{MS}} \mathbf{G} + \mathbf{N}_k, \quad (30)$$

where $\mathbf{N}_k \in \mathbb{C}^{R \times K_P}$ is the additive white Gaussian noise with zero mean and variance σ^2 . To estimate the cascaded channel, we exploit angular sparsity in the channel matrices Φ_{MS} and Φ_{BS} . For this, we apply the BEM by sampling the angular domain using a set of N grid angles $\{\psi_n\}_{n=1}^N$ such that $\cos(\psi_n) = 2n/N - 1$ [36]. Then, (27) and (28) reduce to

$$\Phi_{\text{BS}} = \mathbf{A}_R \mathbf{x}_R \mathbf{x}_{L,d}^H \mathbf{A}_L^H \quad \text{and} \quad \Phi_{\text{MS}} = \mathbf{A}_L \mathbf{x}_{L,a} \mathbf{x}_T^H \mathbf{A}_T^H, \quad (31)$$

where for any integer $Q > 0$, using (29), we define

$$\mathbf{A}_Q = [\mathbf{a}_Q(\psi_1) \quad \mathbf{a}_Q(\psi_2) \quad \dots \quad \mathbf{a}_Q(\psi_N)] \in \mathbb{C}^{Q \times N}. \quad (32)$$

Also, $\mathbf{x}_R, \mathbf{x}_{L,d}, \mathbf{x}_{L,a}, \mathbf{x}_T \in \mathbb{C}^{N \times 1}$ are the unknown sparse channel representations corresponding to the AoAs/AoD of the channel. Substituting (31) into (30), and vectorizing the received signal $\{\mathbf{Y}_k\}_{k=1}^{K_I}$ followed by some algebraic simplifications (see [1] for details), we arrive at

$$\mathbf{y} = (\tilde{\mathbf{H}}_L \otimes \mathbf{H}_T \otimes \mathbf{H}_R) (\mathbf{x}_{L,a} \otimes \mathbf{x}_{L,d}^* \otimes \mathbf{x}_T^* \otimes \mathbf{x}_R) + \mathbf{n} \in \mathbb{C}^{RK \times 1}, \quad (33)$$

where $\tilde{\mathbf{H}}_L = \Theta^T (\mathbf{A}_L^T \odot \mathbf{A}_L^H)^T$, $\mathbf{H}_T = \mathbf{X}^T \mathbf{A}_T^*$, and $\mathbf{H}_R = \mathbf{A}_R$, with \odot denoting the Khatri-Rao products. Also, $\Theta \in \mathbb{C}^{L \times K_I}$ collects all the IRS configurations $\{\theta_k\}_{k=1}^{K_I}$ as columns. Since only the first N columns of $\tilde{\mathbf{H}}_L$ are distinct [37], we remove the redundant columns to reduce the dimension of the representation resulting in

$$\mathbf{y} = (\mathbf{H}_L \otimes \mathbf{H}_T \otimes \mathbf{H}_R) \mathbf{x} + \mathbf{n} = \mathbf{H} \mathbf{x} + \mathbf{n} \in \mathbb{C}^{RK \times 1}, \quad (34)$$

where $\mathbf{H}_L \in \mathbb{C}^{K_I \times N}$ is formed by the first N columns of $\tilde{\mathbf{H}}_L$ and $\mathbf{H} = \mathbf{H}_L \otimes \mathbf{H}_T \otimes \mathbf{H}_R \in \mathbb{C}^{RK \times N^3}$. Here, (34) is equivalent to (1) with $I = 3$, $M_1 = K_I$, $M_2 = K_P$, $M_3 = R$, and hence $M = RK$. Also, we define $\mathbf{x} = \mathbf{x}_L \otimes \mathbf{x}_T^* \otimes \mathbf{x}_R \in \mathbb{C}^{N^3 \times 1}$ with $\mathbf{x}_L \in \mathbb{C}^{N \times 1}$ being the scaled version of the first N entries of $\mathbf{x}_{L,a} \otimes \mathbf{x}_{L,d}^*$. Hence, (34) translates the channel estimation problem into the sparse recovery problem in (1) with unknown vector \mathbf{x} . Now we can apply our algorithms (AM-KroSBL and SVD-KroSBL) with $I = 3$ to reconstruct the Kronecker-structured sparse channel vector \mathbf{x} .

B. Extension of AM-KroSBL for Block Sparsity

In the IRS-aided channel model, the scatterers lead to spreading AoAs (see Fig. 1), causing clustered non-zero BEM coefficients. In our model, sparse vectors \mathbf{x}_L and \mathbf{x}_R , containing the BEM coefficients of the AoAs of IRS and BS, possess block sparsity structures with unknown block boundaries.

To tackle block sparsity, we draw inspiration from the PC-SBL algorithm [21] and impose a prior on each entry of the sparse vector, which not only depends on its hyperparameter but also the hyperparameters of its neighbors. This method connects the sparsity patterns of the adjacent entries, promoting block sparsity. We assume that γ_1 exhibits block sparsity for ease of exposition. However, this idea can be readily extended when multiple hyperparameter vectors possess block sparsity. We adopt the prior on \mathbf{x} with hyperparameters $\{\gamma_i\}$ as $p(\mathbf{x} | \{\gamma_i\}) = \mathcal{CN}(\mathbf{0}, \hat{\gamma})$, where $\hat{\gamma} = \hat{\gamma}_1 \otimes (\otimes_{i=2}^I \gamma_i)$ and

$$\hat{\gamma}_1 = \mathbf{C}_\beta \gamma_1, \quad (35)$$

where $\mathbf{C}_\beta \in \mathbb{R}^{N \times N}$ is a tridiagonal Toeplitz matrix with ones along its diagonal and β along its first sub- and super-diagonals [38]. The parameter $\beta > 0$ is the pattern-coupled coefficient. Using the new prior, the mean and variance of conditional distribution in the r th EM iteration are modified by replacing $\gamma^{(r)}$ with $\hat{\gamma}^{(r)}$ in (13). Thus, the optimization problem in the M-step is

$$\min_{\{\gamma_i\}} \log |\text{diag}(\hat{\gamma})| + (\mathbf{d}^{(r)})^T \hat{\gamma}^{-1} \quad \text{s.t.} \quad \{\hat{\gamma}_1, \{\gamma_i\}_{i=2}^I\} \in \mathcal{C}_+. \quad (36)$$

We solve (36) using an iterative algorithm similar in spirit to AM-KroSBL by setting the gradient of the cost function with respect to all hyperparameter vectors to zero. Here, the update for $\{\gamma_i\}_{i=2}^I$ is given by (15). However, due to the entanglement of hyperparameters in $\hat{\gamma}_1$, the update for $\gamma_1^{(r,t+1)}$ is

$$\begin{aligned} N^{I-1} \mathbf{C}_\beta (\hat{\gamma}_1^{(r,t+1)})^{-1} &= \mathbf{C}_\beta \text{diag} \left((\hat{\gamma}_1^{(r,t+1)})^{-2} \right) \\ &\quad \times \left(\mathbf{I}_N \otimes (\otimes_{i=2}^I (\gamma_i^{(r,t)})^{-1}) \right)^T \mathbf{d}^{(r)}. \end{aligned} \quad (37)$$

Solving (37) is not trivial due to the matrix-vector multiplication on both sides. However, if \mathbf{C}_β is invertible, we can simplify (37) using (35) as

$$\begin{aligned} \mathbf{C}_\beta \gamma_1^{(r,t+1)} &= \hat{\gamma}_1^{(r,t+1)} \\ &= N^{-I+1} (\mathbf{I}_N \otimes (\otimes_{i=2}^I (\gamma_i^{(r,t)})^{-1}))^T \mathbf{d}^{(r)}. \end{aligned} \quad (38)$$

Algorithm 3: PC-KroSBL

Data: Measurements \mathbf{y} , matrix \mathbf{H} , noise power σ^2

1 Parameters: Threshold ϵ and ϵ_{AM}

2 Initialization: set $r = 1$, $\{\gamma_i\}^{(0)} = \mathbf{0}$, $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$

3 while $\|\hat{\gamma}^{(r)} - \hat{\gamma}^{(r-1)}\|_2 > \epsilon$ **do**

4 Compute $\mathbf{d}^{(r)}$ using (12) and (13) with $\gamma^{(r)} = \hat{\gamma}^{(r)}$

5 Set $t = 1$, $\{\gamma_i\}^{(r,1)} = \{\gamma_i\}^{(r)} \in \mathcal{C}_+$, $\{\gamma_i\}^{(r,0)} = \mathbf{0}$

6 **while** $\|\hat{\gamma}^{(r,t)} - \hat{\gamma}^{(r,t-1)}\|_2 > \epsilon_{\text{AM}}$ **do**

7 Compute $\gamma_1^{(r,t+1)}$ using (39)

8 Compute $\tilde{\gamma}_1 = \gamma_1^{(r,t+1)}$ using (35)

9 Compute $\{\gamma_i\}^{(r,t+1)}$ using (15) and (16)

10 Update AM iteration index $t \leftarrow t + 1$

11 **end**

12 $\{\gamma_i\}^{(r+1)} = \{\gamma_i\}^{(r,t)}$

13 Update iteration index $r \leftarrow r + 1$

14 **end**

Result: Output $\mathbf{x} = \boldsymbol{\mu}_x$ using (13)

Also, we notice that the n -th eigenvalue of \mathbf{C}_β is $\lambda_n = 1 + 2\beta \cos(\frac{n\pi}{N+1})$ for $n = 1, 2, \dots, N$ [38]. Thus, we choose $\beta \neq -(2 \cos(\frac{n\pi}{N+1}))^{-1}$ so that \mathbf{C}_β is non-singular. With this choice, we update γ_1 along with the constraint $\gamma_1^{(r,t+1)} > 0$ with

$$\min_{\gamma_1^{(r,t+1)} \geq 0} \|\mathbf{C}_\beta \gamma_1^{(r,t+1)} - N^{-I+1} (\mathbf{I}_N \otimes (\otimes_{i=2}^I (\gamma_i^{(r,t)})^{-1}))^T \mathbf{d}^{(r)}\|_2. \quad (39)$$

The resulting algorithm, namely PC-KroSBL, is summarized in Algorithm 3.

V. PERFORMANCE EVALUATION

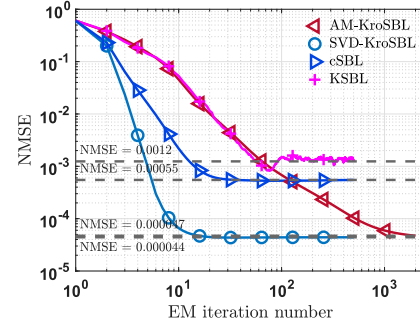
We conduct numerical experiments to investigate the efficacy of our algorithms for sparse vector recovery. We evaluate the recovery performance of AM-KroSBL and SVD-KroSBL by comparing them with three benchmarking algorithms: the classic SBL (cSBL) [20], KSBL [11], and KOMP [16].

We choose $I = 3$ i.e., $\mathbf{H} = \otimes_{i=1}^3 \mathbf{H}_i$ and the sparse vector $\mathbf{x} = \otimes_{i=1}^3 \mathbf{x}_i$ where $\mathbf{x}_i \in \mathbb{R}^{15 \times 1}$. The entries of $\mathbf{x}_i \in \mathbb{R}^{15 \times 1}$, $\mathbf{H}_1 \in \mathbb{R}^{M \times 15}$, $\mathbf{H}_2 \in \mathbb{R}^{12 \times 15}$, and $\mathbf{H}_3 \in \mathbb{R}^{15 \times 15}$ are drawn from $\mathcal{N}(0, 1)$, where $M = \{6, 8, 10, 12, 14\}$, called measurement level, controls the total number of measurements $\bar{M} = 180M$ (or the under-sampling ratio $\bar{M}/N^I = 180M/3375$). The sparsity level for each \mathbf{x}_i is $S = \{2, 3, 4, 5, 6\}$, and the support is generated uniformly at random. The zero-mean additive white Gaussian measurement noise level is decided by the signal-to-noise ratio $\text{SNR} (\text{dB}) = 10 \log_{10} \mathbb{E}\{\|\mathbf{H}\mathbf{x}\|_2^2 / \|\mathbf{n}\|_2^2\}$ and takes values from $\{5, 10, 15, 20, 25, 30\}$.

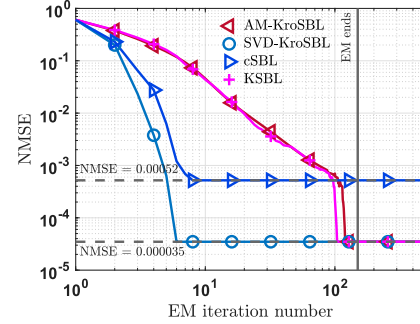
We use three metrics for the assessment: NMSE, support recovery rate (SRR), and run time. Here, we define

$$\text{NMSE} = \mathbb{E} \left\{ \frac{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}{\|\mathbf{x}\|_2^2} \right\}, \quad (40)$$

$$\text{SRR} = \frac{|\text{supp}(\hat{\mathbf{x}}) \cap \text{supp}(\mathbf{x})|}{|\text{supp}(\hat{\mathbf{x}}) \cup \text{supp}(\mathbf{x})|}, \quad (41)$$



(a) NMSE without pruning step



(b) NMSE with pruning step

Fig. 2. Convergence plots of AM-, SVD-KroSBL, KSBL, and cSBL with measurement level $M = 14$, sparsity $S = 4$, and $\text{SNR} = 30\text{dB}$.

where \mathbf{x} is the ground truth and $\hat{\mathbf{x}}$ is the reconstructed vector. We set the maximum EM iteration to 150 EM for the SBL-based methods. We also implement the complexity reduction technique described in [1] for AM- and SVD-KroSBL and the technique in [11] for KSBL. The pruning is also included to prune small hyperparameters for all SBL-based methods [39]. We average over two hundred independent realizations. Our observations from the results, summarized in Figs. 2 to 7 and Tables I and II, are as follows. Our code is available here.

A. Convergence Illustration

Fig. 2 demonstrates the convergence property of cSBL, KSBL, AM-, and SVD-KroSBL *with and without pruning*. We include the convergence without pruning because our theoretical analysis does not account for pruning. First, we look at Fig. 2. We note that our AM-KroSBL and SVD-KroSBL lead to lower NMSE because they incorporate the Kronecker structure, avoiding unwanted local minimum compared with cSBL. However, KSBL, despite incorporating the same prior knowledge, results in a higher NMSE than AM- and SVD-KroSBL. Interestingly, KSBL initially has a similar NMSE as AM-KroSBL, as both schemes solve (14) using the gradient-based iterative method. However, KSBL is trapped in a local minimum after a few iterations because it uses a loose approximation and does not constrain its hyperparameters $\{\gamma_i\}$ in \mathcal{C} , i.e., does not normalize the hyperparameters. Our experiments show that the entries of some γ_i 's of KSBL upon convergence become very small while others become very large. Although the Kronecker product of $\{\gamma_i\}$ of both algorithms are the same initially, small

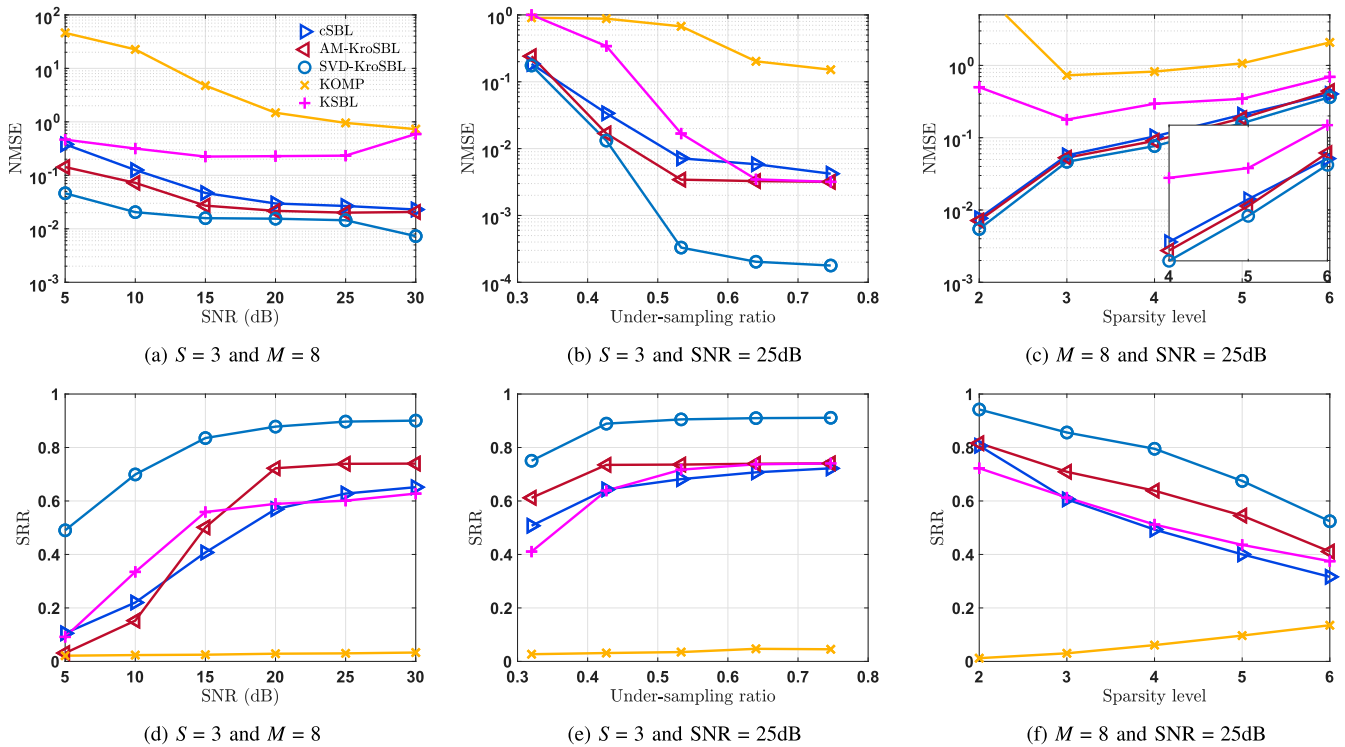


Fig. 3. NMSE and SRR performance of different algorithms as functions of SNR, under-sampling ratio, and sparsity level S . Unless otherwise mentioned in the plots, measurement level $M = 8$, sparsity level $S = 3$, and SNR = 25 dB.

γ_i values unstabilize KSBL. This numerical instability reduces the estimation accuracy and leads to a local minimum with high NMSE. To mitigate this issue, pruning small components is useful, helping KSBL to converge numerically, as shown in Fig. 2(b). Pruning also accelerates other schemes. However, NMSE performance is sensitive to the pruning threshold, which is only empirically chosen. A larger threshold leads to faster convergence but is at the risk of eliminating true components.

B. Comparison With the State-of-the-Art

Fig. 3(a) and 3(b) show that the performance of all the algorithms improves with SNR and the number of measurements $\bar{M} = 180M$ (or equivalently under-sampling ratio $\bar{M}/N^I = 180M/3375$), except for KSBL in Fig. 3(a) due to over-pruning. Similarly, Fig. 3(c) shows that increasing the sparsity level hinders the reconstruction of all schemes, as the number of measurements is unchanged. Further, our SVD-KroSBL outperforms all the other algorithms, both in terms of NMSE and SRR. The next best-performing algorithm is AM-KroSBL in most cases. The exception is in Fig. 3(d), where KSBL and cSBL have higher SRR than the AM-based when the SNR is low. This behavior is because the pruning step of the AM-based fails to eliminate small components outside the true support. However, the nonzero entries are recovered accurately, indicated by low NMSEs. Finally, KSBL is expected to perform better than

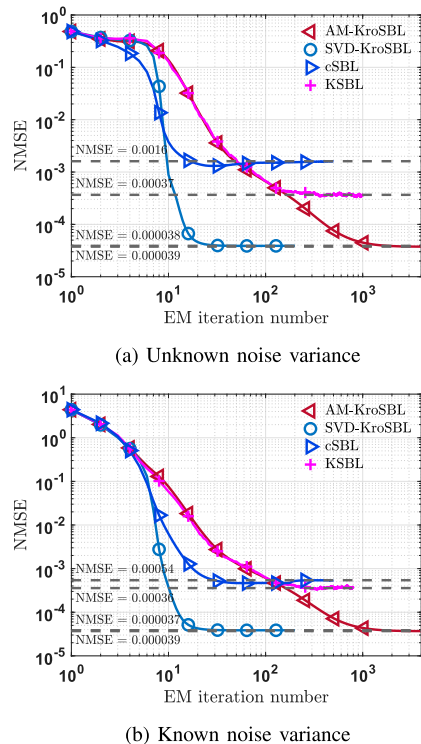


Fig. 4. Convergence performance of AM-, SVD-KroSBL, KSBL, and cSBL when the nonzero entries of sparse vector are not Kronecker-structured with measurement level $M = 14$, sparsity $S = 4$, and SNR = 30dB.

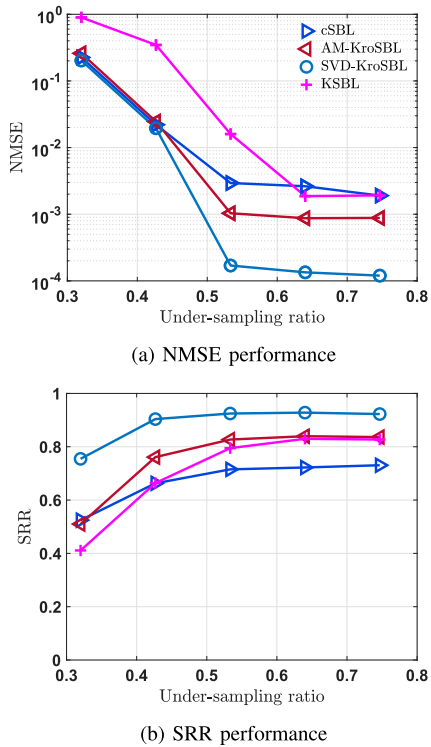


Fig. 5. NMSE and SRR performance of different algorithms as functions of under-sampling ratio when noise variance is unknown with sparsity level $S = 3$ and SNR = 25 dB.

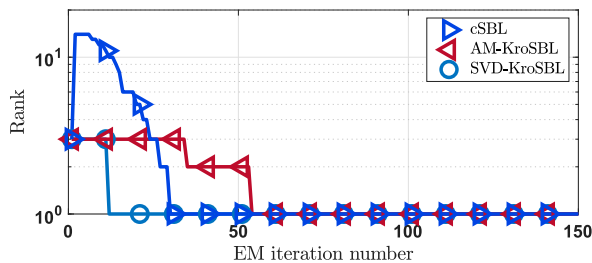


Fig. 6. The rank of reorganized $\mathbf{d}^{(r)}$ generated by different schemes in the noiseless setting with measurement level $M = 14$ and sparsity level $S = 4$.

cSBL. However, this is not true in the low-measurement regime due to its approximations, seen in Fig. 3(b).

Table I compares the run time of different schemes. KOMP has the lowest run time due to its greedy nature but suffers from high NMSE and low SRR, as shown in Fig. 3. SVD-KroSBL has the lowest run time among the remaining algorithms. It is faster as it is non-iterative and takes fewer EM iterations to converge (see Fig. 2), and it also uses the complexity reduction technique [1] in the E-step. We also observe that AM-KroSBL outperforms the cSBL and has a run time similar to KSBL. Although cSBL takes fewer EM iterations, each EM iteration of AM-KroSBL is faster than cSBL due to the complexity reduction [1]. Further, unlike KSBL and cSBL, AM-KroSBL has an iterative M-step, but the inner loop converges quickly, making it faster than KSBL.

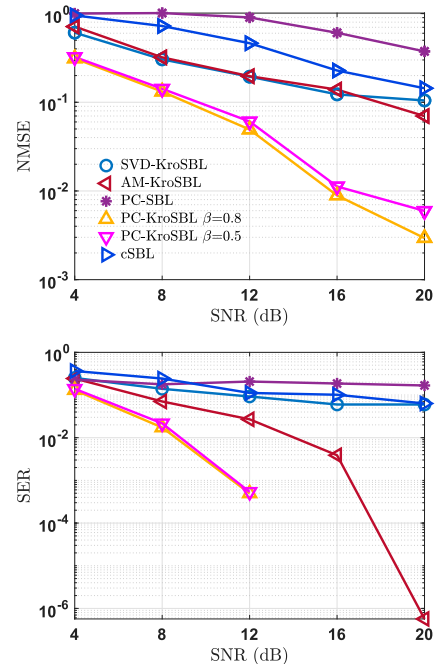


Fig. 7. NMSE of IRS-aided channel estimation and SER of different algorithms as functions of SNR.

TABLE I
COMPUTATION TIME (IN SECONDS) COMPARISON WITH SNR = 25dB AND MEASUREMENT LEVEL $M = 8$ (UNDERLINED TEXT FOR THE BEST AND BOLDFACE FOR THE SECOND BEST)

Sparsity Level	$S = 2$	$S = 3$	$S = 4$	$S = 5$	$S = 6$
AM-KroSBL	8.379	7.181	6.550	6.224	5.944
SVD-KroSBL	2.164	1.759	1.605	1.557	1.691
KSBL	11.318	10.313	9.669	9.041	8.659
cSBL	15.574	14.077	14.385	15.891	19.738
KOMP	<u>0.745</u>	<u>0.785</u>	<u>0.787</u>	<u>0.782</u>	<u>0.777</u>

TABLE II
COMPUTATION TIME (IN SECONDS) COMPARISON FOR THE IRS-AIDED MIMO CHANNEL ESTIMATION WITH $K_I = 6$ AND $K_P = 3$ (UNDERLINED TEXT FOR THE BEST AND BOLDFACE FOR THE SECOND BEST)

SNR	4dB	8dB	12dB	16dB	20dB
PC-KroSBL $\beta = 0.8$	<u>26.871</u>	14.392	9.821	7.979	7.380
PC-KroSBL $\beta = 0.5$	27.904	13.675	8.771	6.670	<u>5.441</u>
AM-KroSBL	44.480	19.672	11.973	12.533	11.915
SVD-KroSBL	30.194	<u>10.403</u>	<u>6.358</u>	<u>5.677</u>	5.746
PC-SBL	88.245	84.397	55.798	43.115	46.446
cSBL	62.844	22.455	14.478	11.258	9.008

C. Recovery of Sparse Vector Without Kronecker-Structured Non-Zero Entries

In this section, we present simulation results on recovering sparse vectors with Kronecker-structured support but not Kronecker-structured nonzero values. To this end, we first construct $I = 3$ binary support vectors \mathbf{b}_i , $i = 1, 2, 3$, containing $S = 4$ ones, and obtain the support vector of \mathbf{x} as $\mathbf{b} = \otimes_{i=1}^I \mathbf{b}_i$. The nonzero entries of \mathbf{x} are drawn from a normal distribution

$\mathcal{N}(0, 0.05)$. The results are summarized in Fig. 4, showing the results for the cases of known and unknown noise variance. Comparing Fig. 4 with Fig. 2, we can see that both our AM- and SVD-KroSBL work well with lower reconstruction errors, indicating that our algorithms still converge to a better local minimum than the other competing algorithms when recovering sparse vector \mathbf{x} with Kronecker-structured support.

D. Extension to Unknown Noise Variance Case

To evaluate our KroSBL algorithms when the noise variance is unknown, we incorporate the noise variance estimation technique (20) in the M-step. The simulation results are shown in Fig. 5. We do not include KOMP since Fig. 3 has already illustrated its poor performance. We observe that with more measurements, all algorithms can better estimate \mathbf{x} . Compared to Fig. 3(b) and 3(e), not providing the noise variance does not noticeably affect the overall NMSE and SRR performance of cSBL, AM-, and SVD-KroSBL. However, the performance of KSBL degrades both in NMSE and SRR, possibly due to the approximation adopted in its derivation.

Interestingly, we observe that AM-KroSBL yields better NMSE and SRR performance with the noise variance estimation step. The lower estimation error can be caused because the optimal value of the noise variance, which corresponds to the best performance, does not necessarily have to be the true noise variance. A similar trend was also observed in [39], demonstrating that the true noise variance can possibly lead to suboptimal recovery performance. Furthermore, different SBL algorithms can have different optimal noise variance values, so changes for SVD-KroSBL and cSBL have different magnitudes [39].

We note that noise estimation slightly increases the runtime of SVD-KroSBL. This behavior is intuitive, as the noise variance estimation is computationally light. Moreover, from Fig. 4(a), we observe that for the unknown noise variance case, the NMSE curve initially remains flat before gradually decreasing. This behavior is more pronounced for AM-KroSBL than for SVD-KroSBL, leading to a double computation time for AM-KroSBL when the noise variance is unprovided.

E. Comparison of SVD-KroSBL and AM-KroSBL

We observe from Fig. 3 that the SVD-KroSBL algorithm outperforms AM-KroSBL and cSBL with sufficient measurements regardless of its approximations. Here, we give the intuition behind the better performance of the SVD-KroSBL. In KroSBL, the M-step solves (14). SVD-KroSBL approximates (14) by first identifying (17) and then solving (18). Suppose $\mathbf{d}^{(r)} = \otimes_{i=1}^I \mathbf{d}_i^{(r)}$ for some nonnegative vectors $\mathbf{d}_i^{(r)} \in \mathbb{R}^{N \times 1}$. Then the optimal solution to (14) is attained at $\tilde{\gamma}_i = \mathbf{d}_i^{(r)} / \|\mathbf{d}_i^{(r)}\|$ for $i = 1, 2, \dots, I - 1$, and $\tilde{\gamma}_I = \prod_{i=1}^{I-1} \|\mathbf{d}_i^{(r)}\| \mathbf{d}_I^{(r)}$. Since (17) attains the optimal value at $\tilde{\gamma} = \mathbf{d}^{(r)}$ that satisfies the constraints of (14), $\tilde{\gamma} = \mathbf{d}^{(r)}$ is also optimal for (14). Hence, the SVD-based is exact when $\mathbf{d}^{(r)}$ is Kronecker-structured. Further, we empirically verified the rank of rearranged/devectorized $\mathbf{d}^{(r)}$

when $I = 3$, where rank = 1 indicates $\mathbf{d}^{(r)} = \otimes_{i=1}^3 \mathbf{d}_i^{(r)}$. Without noise, the SVD-based method drives $\mathbf{d}^{(r)}$ to this structure within a few EM iterations and retains this structure, as shown in Fig. 6. Therefore, after a few EM iterations, the SVD-based method solves the M-step exactly and outperforms the iterative first-order optimization in the AM-KroSBL algorithm. Furthermore, Fig. 6 also indicates that SVD-based converges to a Kronecker-structured $\mathbf{d}^{(r)}$ faster than AM-KroSBL and cSBL. Thus, SVD can be viewed as a stronger imposition of the Kronecker structure, accelerating EM than the others.

F. IRS-Aided Channel Estimation

In this subsection, we present the observation from the results obtained when our PC-KroSBL is applied to the IRS-aided MIMO channel estimation, as discussed in Sec. IV. We choose the number of BS antennas $R = 16$, the number of MS antennas $T = 6$, and the number of IRS elements $L = 256$. Each entry of the IRS configurations $\{\theta_k\}_{k=1}^{K_I}$ is uniformly drawn from $\{-1/\sqrt{L}, 1/\sqrt{L}\}$ with $K_I = 6$ and $K_P = 3$. The number of grid angles is $N = 18$, and all AoDs/AoAs are drawn from the grids. We assume the angle spreads over three grid points, leading to three consecutive non-zeros in sparse vectors \mathbf{x}_L and \mathbf{x}_R . For this, we choose one grid point uniformly at random and add the selected grid and its two neighboring grids to the support. The channel gains $\beta_{MS,p}$ and $\beta_{BS,p}$ defined in Sec. IV are drawn from $\mathcal{CN}(0, 1)$ [40]. The performance of our PC-KroSBL is compared with PC-SBL [21] and cSBL [20]. The pattern-coupled coefficient is 0.9 in the PC-SBL algorithm. We choose $\beta = 0.5$ and $\beta = 0.8$. The performance metrics are NMSE, symbol error rate (SER), and run time. Here, channel estimation NMSE is given as

$$\frac{1}{K_I} \sum_{k=1}^{K_I} \frac{\|\Phi_{BS} \text{diag}(\theta_k) \tilde{\Phi}_{MS} - \tilde{\Phi}_{BS} \text{diag}(\theta_k) \tilde{\Phi}_{MS}\|_F^2}{\|\Phi_{BS} \text{diag}(\theta_k) \tilde{\Phi}_{MS}\|_F^2},$$

with $\tilde{\Phi}_{BS} \text{diag}(\theta_k) \tilde{\Phi}_{MS}$ being the reconstructed channel. SER is computed over data transmission containing 10^5 uncoded QPSK symbols. The results are averaged over fifty independent channel realizations.

We focus on the low measurement regime with the under-sampling ratio $\bar{M}/\bar{N} = RK/N^3 \approx 5\%$. Fig. 7 shows that PC-KroSBL outperforms AM- and SVD-KroSBL that do not account for block sparsity in terms of NMSE and SER. Comparing the performance of PC-KroSBL for different values of β , we infer that $\beta = 0.8$ achieves better accuracy and lower error than $\beta = 0.5$ case. Another interesting observation is that AM-KroSBL has higher NMSE but lower SER than SVD-KroSBL when $\text{SNR} = \{4, 8, 12, 16\}$. This is because SVD-KroSBL fails to identify correct AoAs and AoDs for some channel realizations, resulting in severe channel estimation errors. In such cases, SER is close to one, significantly affecting the overall SER performance of SVD-KroSBL. However, the AM-KroSBL does not completely fail even in the low measurement scenario and works better with increasing SNR. Moreover, PC-KroSBL can consistently and accurately estimate the channel and ensure low SER with limited measurements.

Further, PC-SBL fails in all SNR scenarios because it does not exploit the Kronecker structure, and hence, it needs more measurements than KroSBL algorithms for a graceful recovery. Also, while both sparse vectors \mathbf{x}_L and \mathbf{x}_R exhibit block sparsity, PC-SBL can only exploit the block sparsity of the $\mathbf{x}_L \otimes \mathbf{x}_R$. In contrast, PC-KroSBL can utilize the block sparsity of both \mathbf{x}_L and \mathbf{x}_R , which is an added advantage of our algorithm. Finally, unlike PC-SBL, the cSBL algorithm involves no approximation, which is why it performs better than PC-SBL in the low measurement regime. However, our KroSBL outperforms cSBL, in which no prior knowledge is incorporated.

Table II includes the computation time of different schemes. PC-KroSBL has a comparable run time to SVD-KroSBL and is faster than PC-SBL and cSBL despite its iterative M-step (inner loop). This is due to the complexity reduction technique [1] and the fast convergence of the inner loop.

VI. CONCLUSION

In this paper, we solved the Kronecker-structured linear inversion problem using the SBL framework with the EM procedure. To encourage the Kronecker-supported sparse vector, we designed a prior distribution parameterized by a Kronecker-structured hyperparameter vector and derived two Bayesian algorithms: an iterative AM-based and non-iterative SVD-based algorithm. The AM-based algorithm is theoretically guaranteed to converge to the stationary point of the SBL cost function, while the SVD-based algorithm is faster and has better reconstruction accuracy. We applied our algorithm to the IRS-aided channel estimation for MIMO systems and extended the framework to the block sparsity case. We also analyzed the local minima of the SBL cost function. The analysis of SVD-KroSBL and devising real-time sparse recovery algorithms exploiting the Kronecker structure with reduced complexity can be exciting topics for future work.

APPENDIX A PROOF OF PROPOSITION 1

The proof uses Zangwill's Convergence Theorem A [31] and the following proposition to arrive at the desired results:

Proposition 2: Let $f(\mathbf{x}) : \mathcal{X} \rightarrow \mathbb{R}$ be a continuous function. If

$$f(\mathbf{x}) \rightarrow +\infty \text{ as } \mathcal{X} \ni \mathbf{x} \rightarrow \text{bd}(\mathcal{X}) \text{ or } \|\mathbf{x}\| \rightarrow +\infty, \quad (42)$$

where $\text{bd}(\mathcal{X})$ denotes the boundary of \mathcal{X} , then any sublevel set $\mathcal{S} : \{\mathbf{x} | f(\mathbf{x}) \leq c\}$ is compact.

Proof: This proof is adopted from [41]. A set is compact if it is closed and bounded [41]. We start with the proof of boundedness. Suppose \mathcal{S} is unbounded, then there must exist a sequence $\{\mathbf{x}_n\} \subset \mathcal{S}$ with $\|\mathbf{x}_n\| \rightarrow +\infty$. Then, (42) indicates $f(\mathbf{x}_n) \rightarrow +\infty$. Therefore, there exists an integer $n_0 > 0$ such that $f(\mathbf{x}_n) > c$, for $n > n_0$, which is a contradiction to the assumption $\{\mathbf{x}_n\} \subset \mathcal{S}$. So \mathcal{S} is bounded.

We next complete the proof by establishing the closedness of \mathcal{S} . Suppose \mathcal{S} is not a closed set. Then, there exists at least one sequence $\{\mathbf{x}_n \in \mathcal{S}\}$ converging to $\bar{\mathbf{x}} \notin \mathcal{S}$, which implies either

$\bar{\mathbf{x}} \in \mathcal{X}$ with $f(\bar{\mathbf{x}}) > c$, or $\bar{\mathbf{x}} \in \text{bd}(\mathcal{X})$. However, $\bar{\mathbf{x}} \in \text{bd}(\mathcal{X})$ further implies that $f(\mathbf{x}_n) \rightarrow +\infty$ from (42), which is a contradiction. If $\bar{\mathbf{x}} \in \mathcal{X}$ with $f(\bar{\mathbf{x}}) > c$, then there exists a neighbourhood \mathcal{V}_0 of $\bar{\mathbf{x}}$ such that $f(\mathbf{x}) > c$ for any $\mathbf{x} \in \mathcal{V}_0$. The existence of \mathcal{V}_0 is guaranteed because $f(\mathbf{x})$ is continuous at $\bar{\mathbf{x}}$. However, because $\{\mathbf{x}_n \in \mathcal{S}\}$ converges to $\bar{\mathbf{x}} \notin \mathcal{S}$, there exists an integer n_0 such that $\mathbf{x}_n \in \mathcal{V}_0$, for $n > n_0$. Hence, $f(\mathbf{x}_n) > c$, leading to a contradiction to the assumption that $\{\mathbf{x}_n\} \subset \mathcal{S}$. Thus, \mathcal{S} contains all its limit points, proving that \mathcal{S} is closed and the proof is complete. \square

Next, we prove the proposition using the above results. We let the sequence $\{\gamma_i\}_{i=1}^{\infty}$ be generated by the AM in Algorithm 1 with the starting point $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$, where we omit the EM index r for brevity. We also define the mapping from $\{\gamma_i\}^{(t)}$ to $\{\gamma_i\}^{(t+1)}$ in Algorithm 1 as a function $M(\cdot)$, i.e., $M(\{\gamma_i\}^{(t)}) = \{\gamma_i\}^{(t+1)}$. We prove the result using the Zangwill's convergence theorem [31]. Suppose the following conditions from the Zangwill's convergence theorem hold,

- 1) If the sequence $\{\gamma_i\}_{i=1}^{\infty}$ is in a compact set $\mathcal{S} \subset \mathcal{C}_+$.
- 2) If $\{\gamma_i\}^{(t)}$ is not a stationary point of $Q(\{\gamma_i\})$,

$$Q(\{\gamma_i\}^{(t)}) > Q(\{\gamma_i\}^{(t+1)}). \quad (43)$$

- 3) If $\{\gamma_i\}^{(t)}$ is a stationary point of $Q(\{\gamma_i\})$, the AM step terminates or

$$Q(\{\gamma_i\}^{(t)}) \geq Q(\{\gamma_i\}^{(t+1)}). \quad (44)$$

- 4) Function $Q(\{\gamma_i\})$ is continuous in $\{\gamma_i\}$ and $M(\cdot)$ is continuous at $\{\gamma_i\}^{(t)}$ if $\{\gamma_i\}^{(t)}$ is not a stationary point.

Then, Zangwill's theorem [31] guarantees that the AM algorithm stops at a stationary point of $Q(\{\gamma_i\})$. Consequently, in the remainder of the proof, we verify Conditions 1-4.

We begin with Condition 1. From Lemma 1, we deduce that $\{\gamma_i\}_{i=1}^{\infty} \subset \mathcal{S}$ where \mathcal{S} is the sublevel set of $Q(\{\gamma_i\}^{(0)})$. So to check Condition 1, we prove that \mathcal{S} is compact. To this end, we establish that \mathcal{S} is compact using the following result. Invoking Proposition 2, it is enough to verify that $Q(\{\gamma_i\})$ satisfies (42) on its domain \mathcal{C}_+ to ensure compactness of \mathcal{S} . For this, we notice that $Q(\{\gamma_i\})$ in (11) can be rewritten as

$$Q(\{\gamma_i\}) = \sum_{j=1}^{N^I} \log[\gamma]_j + [\mathbf{d}^{(r)}]_j [\gamma]_j^{-1} \quad (45)$$

$$= \log[\gamma]_{j_*} + [\mathbf{d}^{(r)}]_{j_*} [\gamma]_{j_*}^{-1} + \sum_{j \neq j_*} \log[\gamma]_j + [\mathbf{d}^{(r)}]_j [\gamma]_j^{-1}, \quad (46)$$

for any $j_* = 1, 2, \dots, N^I$. Since $\mathbf{d}^{(r)} > 0$ and the minimum value of function $f(x) = \log x + a/x$ is attained at $x = a$ and $f(x) \geq f(a) = \log a + 1$, for any $a > 0$, we get

$$Q(\{\gamma_i\}) \geq \log[\gamma]_{j_*} + [\mathbf{d}^{(r)}]_{j_*} [\gamma]_{j_*}^{-1} + \sum_{j \neq j_*} \left(\log[\mathbf{d}^{(r)}]_j + 1 \right). \quad (47)$$

The assumption $\mathbf{d}^{(r)} > 0$ also implies that the second term in the lower bound $\sum_{j \neq j_*} (\log[\mathbf{d}^{(r)}]_j + 1)$ is finite. Therefore, if $[\gamma]_{j_*} \rightarrow 0$ or $[\gamma]_{j_*} \rightarrow +\infty$, the lower bound on Q goes to $+\infty$, and so, $Q \rightarrow +\infty$.

Clearly, $Q \rightarrow +\infty$ when $\|\{\gamma_i\}\| = \left(\sum_{i=1}^I \|\gamma_i\|_2^2\right)^{1/2} \rightarrow +\infty$ because at least one entry γ_{j_*} of γ satisfies $[\gamma]_{j_*} \rightarrow +\infty$. Similarly, when $\mathcal{C}_+ \ni \{\gamma_i\} \rightarrow \text{bd}(\mathcal{C}_+)$, there exists an index i_* such that at least one entry of γ_{i_*} goes to zero. In that case, we can choose j_* such that

$$[\gamma]_{j_*} = \min_l [\gamma_{i_*}]_l \prod_{i \neq i_*} \|\gamma_i\|_\infty. \quad (48)$$

Then, $[\gamma]_{j_*} \rightarrow 0$ if $\prod_{i \neq i_*} \|\gamma_i\|_\infty$ is finite; otherwise $[\gamma]_{j_*} \rightarrow \infty$. In both cases, $Q \rightarrow +\infty$. Hence, (42) holds for $Q(\{\gamma_i\})$, and due to Proposition 2, \mathcal{S} is compact. Thus, Condition 1 is established.

We next examine Condition 2. A stationary point $\{\gamma_i^s\} \in \mathcal{C}_+$ of Q satisfies

$$\left. \frac{\partial Q(\{\gamma_i\}|\gamma^{(r)})}{\partial \{\gamma_i\}} \right|_{\{\gamma_i^s\}} = \mathbf{0}. \quad (49)$$

If $\{\gamma_i\}^{(t)} \in \mathcal{C}_+$ is not a stationary point of $Q(\{\gamma_i\})$, there exists at least one index $i_* \in \{1, 2, \dots, I\}$ such that

$$\begin{aligned} \gamma_{i_*}^{(t)} &\neq N^{-I+1} [(\otimes_{j=1}^{i_*-1} (\gamma_j^{(t)})^{-1}) \otimes \mathbf{I}_N \\ &\quad \otimes (\otimes_{j=i_*+1}^I (\gamma_j^{(t)})^{-1})]^\top \mathbf{d}^{(r)} \end{aligned} \quad (50)$$

$$\begin{aligned} \gamma_i^{(t)} &= N^{-I+1} [(\otimes_{j=1}^{i-1} (\gamma_j^{(t)})^{-1}) \otimes \mathbf{I}_N \\ &\quad \otimes (\otimes_{j=i+1}^I (\gamma_j^{(t)})^{-1})]^\top \mathbf{d}^{(r)}, \end{aligned} \quad (51)$$

for $i = 1, 2, \dots, i_* - 1$. Therefore, from the AM update (15),

$$\tilde{\gamma}_i = \gamma_i^{(t)}, \text{ for } i = 1, 2, \dots, i_* - 1. \quad (52)$$

Substituting this relation in (15) with $i = i_*$, we obtain

$$\begin{aligned} \tilde{\gamma}_{i_*} &= N^{-I+1} [(\otimes_{i=1}^{i_*-1} (\gamma_i^{(t)})^{-1}) \otimes \mathbf{I}_N \\ &\quad \otimes (\otimes_{i=i_*+1}^I (\gamma_i^{(t)})^{-1})]^\top \mathbf{d}^{(r)}. \end{aligned} \quad (53)$$

The AM update in (15) also guarantees

$$\tilde{\gamma}_{i_*} = \arg \min_{\gamma_{i_*}} Q \left(\{\tilde{\gamma}_i\}_{i=1}^{i_*-1}, \gamma_{i_*}, \{\gamma_i^{(t)}\}_{i=i_*+1}^I \right) \neq \gamma_{i_*}^{(t)}. \quad (54)$$

So using (52) and the above relation, we conclude

$$\begin{aligned} Q(\{\gamma_i\}^{(t)}) &= Q \left(\{\tilde{\gamma}_i\}_{i=1}^{i_*-1}, \gamma_{i_*}^{(t)}, \{\gamma_i^{(t)}\}_{i=i_*+1}^I \right) \\ &> Q \left(\{\tilde{\gamma}_i\}_{i=1}^{i_*-1}, \tilde{\gamma}_{i_*}, \{\gamma_i^{(t)}\}_{i=i_*+1}^I \right) \geq Q(\{\gamma_i\}^{(t+1)}), \end{aligned} \quad (55)$$

$$(56)$$

using arguments similar to (21). Thus, Condition 2 is verified.

Further, to check Condition 3, we note that if $\{\gamma_i\}^{(t)} \in \mathcal{C}_+$ is a stationary point, then it satisfies (51) for $i = 1, 2, \dots, I$. Then, $M(\{\gamma_i\}^{(t)}) = \{\gamma_i\}^{(t)}$ and $Q(\{\gamma_i\}^{(t)}) = Q(\{\gamma_i\}^{(t+1)})$. So, Condition 3 holds.

Finally, we note that $M(\cdot)$ and Q involves the operations $\det(\cdot)$, $\log(\cdot)$, and $(\cdot)^{-1}$ that are continuous on \mathcal{C}_+ . Hence, $M(\cdot)$ and $Q(\{\gamma_i\})$ are also continuous, satisfying Condition 4. Thus, the proof is complete.

APPENDIX B PROOF OF THEOREM 1

Theorem 1 is proven using the convergence guarantees for the generalized EM (GEM) [29, Theorem 6]. GEM is an iterative algorithm similar to EM where the numerically infeasible M-step is replaced with a point-to-set map such that for every iteration r ,

$$Q(\{\gamma_i\}^{(r)}|\{\gamma_i\}^{(r)}) \geq Q(\{\gamma_i\}^{(r+1)}|\{\gamma_i\}^{(r)}), \quad (57)$$

always holds for Q defined in (11). Unlike EM, the GEM algorithm does not require achieving the global minimum for the M-step optimization problem. We recall from Lemma 1 AM-KroSBL is a GEM algorithm. Now, suppose the following conditions of [29, Theorem 6] holds,

- 1) the domain of \mathcal{L} is a subset in \bar{N} -th dimensional Euclidean space $\mathbb{R}^{\bar{N}}$,
- 2) KroSBL cost function \mathcal{L} is continuous in its domain and differentiable in the interior of the domain,
- 3) the sublevel set of $\mathcal{L}(\{\gamma_i\}^{(1)})$ is compact for any initialization $\{\gamma_i\}^{(1)}$ with $\mathcal{L}(\{\gamma_i\}^{(1)}) < +\infty$,
- 4) the sequence $\{\gamma_i\}^{(r)}|_{r=1}^\infty$ generated by AM-KroSBL has the additional property $\nabla Q = \frac{\partial}{\partial \{\gamma_i\}^{(r+1)}} Q(\{\gamma_i\}^{(r+1)}|\gamma^{(r)}) = 0$,
- 5) ∇Q is continuous in both $\{\gamma_i\}^{(r+1)}$ and $\{\gamma_i\}^{(r)}$.

Then, Theorem 6 in [29] ensures that the sequence $\{\gamma_i\}^{(r)}|_{r=1}^\infty$ generated by AM-KroSBL converges to the stationary points of \mathcal{L} . Here, Condition 1 is trivially satisfied because the domain of \mathcal{L} , i.e., \mathcal{C} , is a subset of the \bar{N} -dimensional Euclidean space. Similarly, $\mathcal{L} = \log|\Sigma_{\mathbf{y}}| + \mathbf{y}^H \Sigma_{\mathbf{y}}^{-1} \mathbf{y}$ is a continuous function of $\{\gamma_i\}$ because matrix inversion and determinant are continuous in its entries [42, Theorems 5.19 and 5.20]. Further, the derivative of \mathcal{L} exists everywhere in \mathcal{C} , and thus, Condition 2 is also satisfied. So, in the remainder of the proof, we verify if Conditions 3-5 to establish the convergence guarantee of AM-KroSBL.

To verify Condition 3, we first show the compactness of the sublevel set of \mathcal{L} . Compactness is established via the coercivity of the KroSBL cost function \mathcal{L} since the sublevel sets of coercive functions are compact [41]. By definition [43], function \mathcal{L} is coercive if $\lim_{\|\{\gamma_i\}\| \rightarrow +\infty} \mathcal{L} = +\infty$, where $\|\{\gamma_i\}\| = \left(\sum_{i=1}^I \|\gamma_i\|_2^2\right)^{1/2}$. Further, since $\Sigma_{\mathbf{y}} = \sigma^2 \mathbf{I}_{\bar{M}} + \mathbf{H}\Gamma\mathbf{H}^H$ is positive-definite (PD) when $\sigma^2 > 0$, we have

$$\mathcal{L} = \log|\Sigma_{\mathbf{y}}| + \mathbf{y}^H \Sigma_{\mathbf{y}}^{-1} \mathbf{y} > \log|\Sigma_{\mathbf{y}}| = \sum_{j=1}^{\bar{M}} \log(\sigma^2 + \lambda_j), \quad (58)$$

where $\lambda_j \geq 0$ is the j th eigenvalue of $\mathbf{H}\Gamma\mathbf{H}^H$. So \mathcal{L} is coercive if

$$\lim_{\|\{\gamma_i\}\| \rightarrow +\infty} \sum_{j=1}^{\bar{M}} \log(\sigma^2 + \lambda_j) = +\infty, \quad (59)$$

which is true if at least one of the λ_j 's goes to infinity as $\|\{\gamma_i\}\|$ goes to $+\infty$. Moreover, from the boundedness assumption on

the norm of the dictionary columns, we derive

$$\sum_{j=1}^{\bar{M}} \lambda_j = \text{tr}(\mathbf{H}\mathbf{\Gamma}\mathbf{H}^H) = \sum_{i=1}^{N^I} [\gamma]_i \|\mathbf{H}_i\|_2^2 \quad (60)$$

$$\geq \epsilon^2 \sum_{i=1}^{N^I} [\gamma]_i \geq \epsilon^2 \|\boldsymbol{\gamma}\|_\infty. \quad (61)$$

So $\lim_{\|\{\gamma_i\}\| \rightarrow +\infty} \sum_{j=1}^{\bar{M}} \lambda_j = +\infty$, meaning at least one eigenvalue goes to $+\infty$, proving (59). Thus, KroSBL cost function \mathcal{L} is a coercive function on \mathcal{C} , and its sublevel set of the starting point $\{\gamma_i\}^{(1)}$ is compact. So, Condition 3 is true.

We next verify Condition 4, which requires the point $\{\gamma_i\}^{(r+1)}$ to be a stationary point of $Q(\{\gamma_i\}^{(r+1)}|\boldsymbol{\gamma}^{(r)})$. According to Proposition 1, it holds if $\mathbf{d}^{(r)} > 0$. So, we next show that $\mathbf{d}^{(r)} = \text{diag}(\boldsymbol{\Sigma}_x + \boldsymbol{\mu}_x \boldsymbol{\mu}_x^H) > 0$. Since $\text{diag}(\boldsymbol{\mu}_x \boldsymbol{\mu}_x^H) \geq 0$, it suffices to show that $\text{diag}(\boldsymbol{\Sigma}_x) > 0$. Also, since PD matrices have positive diagonal entries, from (13), it is enough to verify that $\boldsymbol{\Sigma}_x^{-1} = \sigma^{-2} \mathbf{H}^H \mathbf{H} + \text{diag}(\boldsymbol{\gamma}^{(r)})^{-1}$ is PD. So, $\mathbf{d}^{(r)} > 0$ if $\{\gamma_i\}^{(r)} > 0$ because $\sigma^{-2} \mathbf{H}^H \mathbf{H}$ is PSD [44]. From this observation, we prove the condition $\mathbf{d}^{(r)} > 0$ using induction. Since $\{\gamma_i\}^{(1)} \in \mathcal{C}_+$, we have $\mathbf{d}^{(1)} > 0$. Next, we assume that $\mathbf{d}^{(r)} > 0$, for some $r > 0$. From Proposition 1, $\{\gamma_i\}^{(r+1)}$ generated by the AM algorithm is bounded away from $\text{bd}(\mathcal{C}_+)$ when $\mathbf{d}^{(r)} > 0$. Thus, we conclude that $\{\gamma_i\}^{(r+1)} > 0$, which in turn implies $\mathbf{d}^{(r+1)} > 0$. Condition 4 holds in our case.

Finally, we show Condition 5 by first computing the gradient of Q with respect to γ_i as

$$\begin{aligned} \frac{\partial}{\partial \gamma_i} Q(\{\gamma_i\}|\boldsymbol{\gamma}^{(r)}) &= -N^{I-1} \gamma_i^{-1} + \text{diag}(\gamma_i)^{-2} \\ &\times \left[(\otimes_{l=1}^{i-1} (\gamma_l)^{-1}) \otimes \mathbf{I}_N \otimes (\otimes_{l=i+1}^I (\gamma_l^{(r,t)})^{-1}) \right]^T \mathbf{d}^{(r)}. \end{aligned} \quad (62)$$

Here, operators $(\cdot)^{-1}$ and $(\cdot)^{-2}$ are continuous in $(0, +\infty)$. Thus, the gradient is continuous in $\{\gamma_i\}$. Finally, in ∇Q , only $\mathbf{d}^{(r)}$ depends on $\{\gamma_i\}^{(r)}$ as in (13). Since the matrix inversion is continuous in its entries [42, Theorem 5.20], $\boldsymbol{\Sigma}_x$ and $\boldsymbol{\mu}_x$ are continuous in $\{\gamma_i\}^{(r)}$. Therefore, ∇Q is continuous in $\{\gamma_i\}^{(r)}$. Thus, Condition 5 is established, and the proof is complete.

APPENDIX C PROOF OF THEOREM 2

To prove the sparsity of local minima, we start with a few supporting lemmas.

Lemma 2: $\log |\boldsymbol{\Sigma}_y|$ is concave with respect to $\{\gamma_i\}$ in the noiseless case, i.e., $\sigma^2 = 0$.

Proof: When $\sigma^2 = 0$, $\boldsymbol{\Sigma}_y = \mathbf{H}\mathbf{\Gamma}\mathbf{H}^H = \otimes_{i=1}^I \mathbf{H}_i \boldsymbol{\Gamma}_i \mathbf{H}_i^H$. We have

$$\log |\boldsymbol{\Sigma}_y| = \log |\otimes_{i=1}^I \mathbf{H}_i \boldsymbol{\Gamma}_i \mathbf{H}_i^H| = \sum_{i=1}^I \left(\prod_{j \neq i} M_j \right) \log |\mathbf{H}_i \boldsymbol{\Gamma}_i \mathbf{H}_i^H|. \quad (63)$$

Since $\mathbf{H}_i \boldsymbol{\Gamma}_i \mathbf{H}_i^H$ is a PSD matrix and affine in γ_i , and function $\log |\cdot|$ is a concave function in the space of PSD matrices,

$\log |\mathbf{H}_i \boldsymbol{\Gamma}_i \mathbf{H}_i^H|$ is concave in γ_i . Thus, $\log |\boldsymbol{\Sigma}_y|$ is concave because the sum of concave functions is also concave. \square

Lemma 3: If $\{\gamma_i\}$ satisfies $\mathbf{b} = \mathbf{A}(\otimes_{i=1}^I \gamma_i)$, where

$$\mathbf{b} = \mathbf{y} - \sigma^2 \mathbf{u}; \quad \mathbf{A} = \mathbf{H} \text{diag}(\mathbf{H}^H \mathbf{u}), \quad (64)$$

and \mathbf{u} is any fixed vector such that $\mathbf{y}^H \mathbf{u} = C$, then $\mathbf{y}^H \boldsymbol{\Sigma}_y^{-1} \mathbf{y}$ is a constant C for any value of $\sigma^2 \geq 0$.

Proof: We combine $\mathbf{b} = \mathbf{A}(\otimes_{i=1}^I \gamma_i)$ and (64) to obtain

$$\mathbf{y} = \mathbf{A}(\otimes_{i=1}^I \gamma_i) + \sigma^2 \mathbf{u} = \mathbf{H} \text{diag}(\mathbf{H}^H \mathbf{u})(\otimes_{i=1}^I \gamma_i) + \sigma^2 \mathbf{u} \quad (65)$$

$$= \mathbf{H}\mathbf{\Gamma}\mathbf{H}^H \mathbf{u} + \sigma^2 \mathbf{u} = \boldsymbol{\Sigma}_y \mathbf{u}, \quad (66)$$

where $\boldsymbol{\Gamma} = \text{diag}(\otimes_{i=1}^I \gamma_i)$. Then, $\mathbf{y}^H \boldsymbol{\Sigma}_y^{-1} \mathbf{y} = \mathbf{y}^H \mathbf{u} = C$, for any value $\sigma^2 \geq 0$. \square

Lemma 4: Consider the set of linear equations with $\mathbf{t}_1, \mathbf{t}_2 \neq \mathbf{0}$,

$$(\boldsymbol{\Phi}_1 \otimes \boldsymbol{\Phi}_2)(\mathbf{w}_1 \otimes \mathbf{w}_2) = \mathbf{t}_1 \otimes \mathbf{t}_2. \quad (67)$$

Seeking \mathbf{w}_1 and \mathbf{w}_2 that satisfy (67) is equivalent to solving

$$\boldsymbol{\Phi}_1 \mathbf{w}_1 = \alpha \mathbf{t}_1; \quad \boldsymbol{\Phi}_2 \mathbf{w}_2 = \alpha^{-1} \mathbf{t}_2, \quad (68)$$

where α is any non-zero scalar.

Proof: We rewrite (67) as $(\boldsymbol{\Phi}_1 \mathbf{w}_1) \otimes (\boldsymbol{\Phi}_2 \mathbf{w}_2) = \mathbf{t}_1 \otimes \mathbf{t}_2$. Now, we can arrange this equation as $(\boldsymbol{\Phi}_2 \mathbf{w}_2)(\boldsymbol{\Phi}_1 \mathbf{w}_1)^T = \mathbf{t}_2 \mathbf{t}_1^T$, which is a rank-one matrix. Here, $\mathbf{t}_2 \mathbf{t}_1^T$ has at least one nonzero column since $\mathbf{t}_1, \mathbf{t}_2 \neq \mathbf{0}$, and every column of $\mathbf{t}_2 \mathbf{t}_1^T$ is a scaled version \mathbf{t}_2 . Therefore, the solution to the system of equations is given by (68), leading to the desired conclusion. \square

With all the mentioned lemmas, we present the proof of Theorem 2. First, we pose another optimization problem:

$$\min_{\{\gamma_i\} \geq 0} \log |\boldsymbol{\Sigma}_y| \text{ s.t. } \mathbf{A}(\otimes_{i=1}^I \gamma_i) = \mathbf{b}, \quad (69)$$

where $\mathbf{A} = \mathbf{H} \text{diag}(\mathbf{H}^H \mathbf{u})$ and $\mathbf{b} = \mathbf{y} - \sigma^2 \mathbf{u}$. As Lemma 3, the constraint $\mathbf{A}(\otimes_{i=1}^I \gamma_i) = \mathbf{b}$ holds the second term of \mathcal{L} constant, and we minimize the first term of \mathcal{L} , which is concave, over a bounded convex polytope. Then, any local minimum of (6) denoted by $\{\gamma_i^*\}$, must also be a local minimum of (69) with

$$\mathbf{y}^H \left(\sigma^2 \mathbf{I}_{\bar{M}} + \mathbf{H} \text{diag}(\otimes_{i=1}^I \gamma_i^*) \mathbf{H}^H \right)^{-1} \mathbf{y} = C^* = \mathbf{y}^H \mathbf{u}^*, \quad (70)$$

as long as there exists a vector \mathbf{u}^* satisfying $C^* = \mathbf{y}^H \mathbf{u}^*$ to construct \mathbf{A} and \mathbf{b} . A candidate for \mathbf{u}^* in the noiseless setting is

$$\mathbf{u}^* = C^* \frac{\mathbf{y}}{\|\mathbf{y}\|_2^2} = C^* \frac{\otimes_{i=1}^I \mathbf{H}_i \mathbf{x}_i}{\|\mathbf{y}\|_2^2}, \quad (71)$$

where we also use (1), (2) and the assumption that $\mathbf{x} = \otimes_i \mathbf{x}_i$. Thus, we obtain

$$\begin{aligned} \mathbf{A} &= \frac{C^*}{\|\mathbf{y}\|_2^2} \mathbf{H} \text{diag}(\mathbf{H}^H \otimes_{i=1}^I \mathbf{H}_i \mathbf{x}_i) \\ &= \frac{C^*}{\|\mathbf{y}\|_2^2} \otimes_{i=1}^I \mathbf{H}_i \text{diag}(\mathbf{H}_i^H \mathbf{H}_i \mathbf{x}_i). \end{aligned}$$

Similarly, we have

$$\mathbf{b} = \mathbf{y} = \otimes_{i=1}^I \mathbf{H}_i \mathbf{x}_i. \quad (72)$$

Thus, using Lemma 4 with the scaling factor $\alpha = 1$, the constraint $\mathbf{A}(\otimes_{i=1}^I \gamma_i) = \mathbf{b}$ can be written as

$$\mathbf{A}_i \gamma_i = \mathbf{b}_i, \forall i = 1, 2, \dots, I, \quad (73)$$

where $\mathbf{A}_i = (C^* / \|\mathbf{y}\|_2^2)^{1/I} \mathbf{H}_i \text{diag}(\mathbf{H}_i^H \mathbf{H}_i \mathbf{x}_i)$, and $\mathbf{b}_i = \mathbf{H}_i \mathbf{x}_i$. Problem (69) can be transformed into I separated problems as

$$\min_{\gamma_i} \log |\mathbf{H}_i \Gamma_i \mathbf{H}_i^H| \quad \text{s.t. } \mathbf{A}_i \gamma_i = \mathbf{y}_i, \gamma_i \geq 0, \quad (74)$$

for $i = 1, 2, \dots, I$. Any local minimum of \mathcal{L} , e.g., $\{\gamma_i\}^*$ is also a local minimum of (74). Furthermore, all local minima of (74) are achieved at extreme points, which are also basic feasible solutions with at most M_i non-zeros for each γ_i [45]. Thus, $\{\gamma_i^*\}$ is sparse when noise is absent.

APPENDIX D PROOF OF THEOREM 3

The proof is based on the following lemma:

Lemma 5: If $\mathbf{H} = \otimes_{i=1}^I \mathbf{H}_i$ satisfies URP, then \mathbf{H}_i for all i also satisfy URP.

Proof: Suppose we choose any M_i columns of \mathbf{H}_i indexed by \mathcal{M}_i , for $i = 1, 2, \dots, I$. Then, the Kronecker product of the corresponding submatrices, given by $\otimes_{i=1}^I [\mathbf{H}_i]_{\mathcal{M}_i} \in \mathbb{C}^{M \times M}$, is a submatrix of $\mathbf{H} = \otimes_{i=1}^I \mathbf{H}_i$. Since \mathbf{H} satisfies URP, any subset of M columns of \mathbf{H} are linearly independent, we get

$$\begin{aligned} \bar{M} = \text{rank}([\mathbf{H}]_{\otimes_{i=1}^I \mathcal{M}_i}) &= \prod_{i=1}^I \text{rank}([\mathbf{H}_i]_{\mathcal{M}_i}) \\ &\leq \prod_{i=1}^I M_i = \bar{M}. \end{aligned} \quad (75)$$

The last step follows because $\text{rank}([\mathbf{H}_i]_{\mathcal{M}_i}) \leq M_i$. So, (75) holds if and only if $\text{rank}([\mathbf{H}_i]_{\mathcal{M}_i}) = M_i$ for $i = 1, 2, \dots, I$. Hence, any subset of M_i columns of \mathbf{H}_i is linearly independent, and \mathbf{H}_i satisfies URP, for $i = 1, 2, \dots, I$. \square

Lemma 5 ensures that \mathbf{H}_i satisfies the URP for all i . Then, we next show that for any index set \mathcal{M}_i such that $|\mathcal{M}_i| = M_i$, if we restrict the nonzero values of γ_i to the set \mathcal{M}_i , there can only be one minimum for $\mathcal{L}(\{\gamma_i\}_{\mathcal{M}_i})$. For this, we note that

$$\begin{aligned} \mathcal{L}(\{\gamma_i\}_{\mathcal{M}_i}) &= \log \left| \otimes_{i=1}^I [\mathbf{H}_i]_{\mathcal{M}_i} [\Gamma_i]_{\mathcal{M}_i} [\mathbf{H}_i]_{\mathcal{M}_i}^H \right| \\ &\quad + \mathbf{y}^H \left(\otimes_{i=1}^I [\mathbf{H}_i]_{\mathcal{M}_i} [\Gamma_i]_{\mathcal{M}_i} [\mathbf{H}_i]_{\mathcal{M}_i}^H \right)^{-1} \mathbf{y} \\ &= \sum_{i=1}^I \left(\prod_{j \neq i} M_j \right) \log \left| [\mathbf{H}_i]_{\mathcal{M}_i} [\Gamma_i]_{\mathcal{M}_i} [\mathbf{H}_i]_{\mathcal{M}_i}^H \right| \\ &\quad + \mathbf{y}^H \otimes_{i=1}^I \left(([\mathbf{H}_i]_{\mathcal{M}_i}^H)^{-1} [\Gamma_i]_{\mathcal{M}_i}^{-1} [\mathbf{H}_i]_{\mathcal{M}_i} \right) \mathbf{y}. \end{aligned} \quad (77)$$

Here, the second term can be simplified using the assumption, $\mathbf{y} = \left(\otimes_{i=1}^I [\mathbf{H}_i]_{\mathcal{M}_i} \right) \left(\otimes_{i=1}^I [\mathbf{x}_i]_{\mathcal{M}_i} \right) = \otimes_{i=1}^I ([\mathbf{H}_i]_{\mathcal{M}_i} [\mathbf{x}_i]_{\mathcal{M}_i})$, as follows:

$$\begin{aligned} &\mathbf{y}^H \otimes_{i=1}^I \left([\mathbf{H}_i]_{\mathcal{M}_i} [\Gamma_i]_{\mathcal{M}_i} [\mathbf{H}_i]_{\mathcal{M}_i}^H \right)^{-1} \mathbf{y} \\ &= \otimes_{i=1}^I [\mathbf{x}_i]_{\mathcal{M}_i}^H [\Gamma_i]_{\mathcal{M}_i}^{-1} [\mathbf{x}_i]_{\mathcal{M}_i} = \prod_{i=1}^I [\mathbf{x}_i]_{\mathcal{M}_i}^H [\Gamma_i]_{\mathcal{M}_i}^{-1} [\mathbf{x}_i]_{\mathcal{M}_i}. \end{aligned} \quad (78)$$

Therefore, from (77), we arrive at

$$\begin{aligned} \mathcal{L}(\{\gamma_i\}_{\mathcal{M}_i}) &= \sum_{i=1}^I \prod_{j \neq i} M_j \log \left| [\mathbf{H}_i]_{\mathcal{M}_i} [\mathbf{H}_i]_{\mathcal{M}_i}^H \right| \\ &\quad + \sum_{i=1}^I \prod_{j \neq i} M_j \log \left| [\Gamma_i]_{\mathcal{M}_i} \right| \\ &\quad + \prod_{i=1}^I [\mathbf{x}_i]_{\mathcal{M}_i}^H [\Gamma_i]_{\mathcal{M}_i}^{-1} [\mathbf{x}_i]_{\mathcal{M}_i}. \end{aligned} \quad (79)$$

Setting the derivative of the above function with respect to γ_i to zero gives

$$\prod_{j \neq i} M_j [\gamma_i]_{\mathcal{M}_i} = [\mathbf{x}_i]_{\mathcal{M}_i}^2 \prod_{j \neq i} [\mathbf{x}_j]_{\mathcal{M}_j}^H [\Gamma_j]_{\mathcal{M}_j}^{-1} [\mathbf{x}_j]_{\mathcal{M}_j}, \quad (80)$$

for $i = 1, 2, \dots, I$. However, since $\{\gamma_i\} \in \mathcal{C}$, we obtain the unique minimum of $\mathcal{L}(\{\gamma_i\}_{\mathcal{M}_i})$ at

$$[\gamma_i]_{\mathcal{M}_i} = \begin{cases} \frac{[\mathbf{x}_i]_{\mathcal{M}_i}^2}{\|[\mathbf{x}_i]_{\mathcal{M}_i}\|^2}, & \text{if } i < I, \\ [\mathbf{x}_I]_{\mathcal{M}_I}^2 \prod_{j=1}^{I-1} \|[\mathbf{x}_j]_{\mathcal{M}_j}\|^2, & \text{if } i = I. \end{cases} \quad (81)$$

Therefore, every set of $\{\mathcal{M}_i\}_{i=1}^I$ corresponds to one unique local minimum, and counting for all possible index set combinations, we get $\mathcal{N} \leq \prod_{i=1}^I \binom{N}{M_i}$. Further, all index set combinations containing a degenerate solution share the same minimum [20]. Accounting for such repetitions, we derive (26). Thus, the proof is complete.

ACKNOWLEDGMENT

The authors thank Dr. David Wipf, Jianzhe Liang, Changheng Li, and Yanyan Hu for insightful discussions on the theoretical analysis and Sofia-Eirini Kotti for helping us with the coding.

REFERENCES

- [1] Y. He and G. Joseph, "Structure-aware sparse Bayesian learning-based channel estimation for intelligent reflecting surface-aided MIMO," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal, Process. (ICASSP)*, Jun. 2023, pp. 1–5.
- [2] L. He and L. Carin, "Exploiting structure in wavelet-based Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3488–3497, May 2009.
- [3] L. He, H. Chen, and L. Carin, "Tree-structured compressive sensing with variational Bayesian analysis," *IEEE Signal Process. Lett.*, vol. 17, no. 3, pp. 233–236, Dec. 2009.
- [4] X. Ma and G. B. Giannakis, "Maximum-diversity transmissions over doubly selective wireless channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1832–1840, Jun. 2003.
- [5] I. Barhumi, G. Leus, and M. Moonen, "MMSE estimation of basis expansion models for rapidly time-varying channels," in *Proc. EUSIPCO*, Sep. 2005, pp. 1–5.

- [6] G. B. Giannakis and C. Tepedelenlioglu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proc. IEEE*, vol. 86, no. 10, pp. 1969–1986, Oct. 1998.
- [7] J. Han, L. Zhang, Q. Zhang, and G. Leus, "Low-complexity equalization of orthogonal signal-division multiplexing in doubly-selective channels," *IEEE Trans. Signal Process.*, vol. 67, no. 4, pp. 915–929, Dec. 2018.
- [8] B. Zheng, C. You, W. Mei, and R. Zhang, "A survey on channel estimation and practical passive beamforming design for intelligent reflecting surface aided wireless communications," *IEEE Commun. Surv. Tutor.*, vol. 24, no. 2, pp. 1035–1071, Feb. 2022.
- [9] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, vol. 2. Springer, 2009.
- [10] D. C. Araújo, A. L. De Almeida, J. P. Da Costa, and R. T. de Sousa, "Tensor-based channel estimation for massive MIMO-OFDM systems," *IEEE Access*, vol. 7, pp. 42133–42147, 2019.
- [11] W.-C. Chang and Y. T. Su, "Sparse Bayesian learning based tensor dictionary learning and signal recovery with application to MIMO channel estimation," *IEEE J. Sel. Top. Signal Process.*, vol. 15, no. 3, pp. 847–859, Apr. 2021.
- [12] X. Xu, S. Zhang, F. Gao, and J. Wang, "Sparse Bayesian learning based channel extrapolation for RIS assisted MIMO-OFDM," *IEEE Trans. Commun.*, vol. 70, no. 8, pp. 5498–5513, Aug. 2022.
- [13] R. Zhao, Q. Wang, J. Fu, and L. Ren, "Exploiting block-sparsity for hyperspectral Kronecker compressive sensing: A tensor-based Bayesian method," *IEEE Trans. Image Process.*, vol. 29, pp. 1654–1668, 2020.
- [14] S. Yang, M. Wang, P. Li, L. Jin, B. Wu, and L. Jiao, "Compressive hyperspectral imaging via sparse tensor and nonlinear compressed sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 11, pp. 5943–5957, Jun. 2015.
- [15] C. F. Caiafa and A. Cichocki, "Block sparse representations of tensors using Kronecker bases," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal, Process. (ICASSP)*, Mar. 2012, pp. 2709–2712.
- [16] C. F. Caiafa and A. Cichocki, "Computing sparse representations of multidimensional signals using Kronecker bases," *Neural Comput.*, vol. 25, no. 1, pp. 186–220, Jan. 2013.
- [17] M. F. Duarte and R. G. Baraniuk, "Kronecker product matrices for compressive sensing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal, Process. (ICASSP)*, Mar. 2010, pp. 3650–3653.
- [18] M. F. Duarte and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 494–504, Aug. 2011.
- [19] C. F. Caiafa and A. Cichocki, "Multidimensional compressed sensing and their applications," *Wiley Interdiscip. Rev.: Data Min. Knowl. Discov.*, vol. 3, no. 6, pp. 355–380, Oct. 2013.
- [20] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.
- [21] J. Fang, Y. Shen, H. Li, and P. Wang, "Pattern-coupled sparse Bayesian learning for recovery of block-sparse signals," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 360–372, Nov. 2014.
- [22] L. Wang, L. Zhao, S. Rahardja, and G. Bi, "Alternative to extended block sparse Bayesian learning and its relation to pattern-coupled sparse Bayesian learning," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2759–2771, May 2018.
- [23] D. Prasanna and C. R. Murthy, "mmWave channel estimation via compressive covariance estimation: Role of sparsity and intra-vector correlation," *IEEE Trans. Signal Process.*, vol. 69, pp. 2356–2370, 2021.
- [24] X. Wu, S. Ma, X. Yang, and G. Yang, "Clustered sparse Bayesian learning based channel estimation for millimeter-wave massive MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 71, no. 12, pp. 12749–12764, Aug. 2022.
- [25] A. Băltouiu and B. Dumitrescu, "Sparse Bayesian learning algorithm for separable dictionaries," *Digit. Signal Process.*, vol. 111, Apr. 2021, Art. no. 102990.
- [26] K. Kreutz-Delgado, B. Rao, I. Fedorov, and S. Das, "Dictionaries in machine learning," in *Signal Process. Mach. Learn. Theory*, Elsevier, 2024, pp. 1073–1159.
- [27] Z. Zhang and B. D. Rao, "Extension of SBL algorithms for the recovery of block sparse signals with intra-block correlation," *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 2009–2015, Apr. 2013.
- [28] C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning*. Springer, 2006, vol. 4, no. 4.
- [29] C. J. Wu, "On the convergence properties of the EM algorithm," *Ann. Statist.*, pp. 95–103, vol. 11, no. 1, Mar. 1983.
- [30] G. Joseph and C. R. Murthy, "On the convergence of a Bayesian algorithm for joint dictionary learning and sparse recovery," *IEEE Trans. Signal Process.*, vol. 68, pp. 343–358, 2019.
- [31] W. I. Zangwill, *Nonlinear Programming: A Unified Approach*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1969.
- [32] S. Khanna and C. R. Murthy, "On the support recovery of jointly sparse Gaussian sources via sparse Bayesian learning," *IEEE Trans. Inf. Theory*, vol. 68, no. 11, pp. 7361–7378, Nov. 2022.
- [33] A. Lin, A. H. Song, B. Bilgic, and D. Ba, "Covariance-free sparse Bayesian learning," *IEEE Trans. Signal Process.*, vol. 70, pp. 3818–3831, 2022.
- [34] H. Guo and V. K. Lau, "Uplink cascaded channel estimation for intelligent reflecting surface assisted multiuser MISO systems," *IEEE Trans. Signal Process.*, vol. 70, pp. 3964–3977, 2022.
- [35] Y. Zhu, H. Guo, and V. K. Lau, "Bayesian channel estimation in multi-user massive MIMO with extremely large antenna array," *IEEE Trans. Signal Process.*, vol. 69, pp. 5463–5478, 2021.
- [36] Z. Mao, X. Liu, and M. Peng, "Channel estimation for intelligent reflecting surface assisted massive MIMO systems—A deep learning approach," *IEEE Commun. Lett.*, vol. 26, no. 4, pp. 798–802, Jan. 2022.
- [37] P. Wang, J. Fang, H. Duan, and H. Li, "Compressed channel estimation for intelligent reflecting surface-assisted millimeter wave systems," *IEEE Signal Process. Lett.*, vol. 27, pp. 905–909, 2020.
- [38] S. Noschese, L. Pasquini, and L. Reichel, "Tridiagonal Toeplitz matrices: Properties and novel applications," *Numer. Linear Algebra Appl.*, vol. 20, no. 2, pp. 302–326, Mar. 2013.
- [39] Z. Zhang and B. D. Rao, "Clarify some issues on the sparse Bayesian learning for sparse signal recovery," Tech. Rep., University of California, San Diego, CA, USA, 2011.
- [40] Y. Lin, S. Jin, M. Matthaiou, and X. You, "Channel estimation and user localization for IRS-assisted MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2320–2335, Apr. 2021.
- [41] G. C. Calafiore and L. El Ghaoui, *Optimization Models*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
- [42] J. R. Schott, *Matrix Analysis for Statistics*. Hoboken, NJ, USA: Wiley, 2016.
- [43] I. Fedorov, "Structured learning with scale mixture priors," Ph.D. dissertation, ECE Dept., UC San Diego, 2018.
- [44] E. Million, "The Hadamard product," *Course Notes*, vol. 3, no. 6, 2007.
- [45] D. G. Luenberger et al., *Linear and Nonlinear Programming*, vol. 2. Springer, 1984.