A parallel sparse regularization method for structured multilinear low-rank tensor decomposition

Gerta Kushe*,Yang Yang†, Christian Steffens‡, Marius Pesavento*
*Communication Systems Group, Technische Universität Darmstadt, Germany
†Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg
‡Telespazio-VEGA Deutschland, Darmstadt, Germany
{kushe, pesavento}@nt.tu-darmstadt.de, yang.yang@uni.lu, christian.steffens@telespazio-vega.de

***Abstract*—In this paper we consider the structured multilinear low-rank tensor decomposition problem where group sparsity is enforced using nuclear norm regularization. We adopt the recently proposed sequential convex approximation approach to develop an optimization algorithm suitable for implementation on modern parallel hardware architectures. An existing optimization algorithm for this non-convex and non-differentable optimization problem relies on a lifting approach. For large problem dimensions the lifting procedure is, however, inefficient as it drastically increases the number of optimization variables. Our proposed algorithm does not require lifting and directly operates on the original parameters space. We demonstrate the performance gains in terms of convergence speed of the proposed sparse tensor decomposition method for the example of two dimensional harmonic retrieval.**

I. Introduction

Multidimensional harmonic retrieval and more generally structured multilinear low-rank tensor decomposition is a problem that is fundamental in many important applications including radar signal processing, direction-of-arrival estimation, MIMO channel sounding and nuclear magnetic resonance tomography. Recently, multidimensional harmonic retrieval has been successfully applied in the context of channel estimation in massive MIMO systems [1], [2], [3], [4], [5]. Traditionally, subspace based methods are used to solve the two- and multidimensional harmonic retrieval problem [6], [7]. Subspace methods achieve a satisfactory estimation performance at low computational complexity. The MUSIC algorithm presented in [8] can be used to estimate the frequencies of a superposition of multiple harmonic signals. However, unlike the one-dimensional harmonic retrieval problem for which the computationally efficient root-procedure can be applied, in multidimenisional harmonic retrieval a computationally demanding spectral search is required. In [9], [10], [11] search free methods for two- and multidimensional harmonic retrieval have been proposed that exploit the multiple shift-invariance structure of the measurement tensor. In [12] a polynomial rooting technique for 2D harmonic retrieval has been propose.

This work was supported by the EXPRESS project within the DFG priority program CoSIP (DFG-SPP 1798).

Recently, sparse recovery techniques came into the focus of parameter estimation, which exhibit the superresolution property [13]. In comparison to the subspace based methods, sparse recovery methods show a good performance even though for a small number of snapshots and do not rely on the signal covariance matrix estimation. However, sparse recovery methods require a fine discretization of the parameter grid, in order to achieve a high resolution and to avoid the basis mismatch [14]. For uniform linear sampling schemes gridless sparse recovery methods based on atomic norm minimization have been proposed in [15], [16] and [17]. In [3] a sparse recovery method based on nuclear norm minimization is presented which is also suitable for irregular sampling schemes. Recently, customized iterative algorithm have been proposed to efficiently solve the corresponding sparse regularization problems without the need of using general purpose interior point solvers. The block coordinate decent method (BCD) [18] has, e.g., been applied in [19] in the context of row-sparse signal recovery from multiple measurements. In [3] the successive convex approximation (SCA) algorithm has been adapted for rank-sparse recovery in the context of two-dimensional harmonic retrieval. The drawback of this method is that a lifting procedure had to be employed to circumvent bilinear product terms in the optimization. This significantly increases the number of optimization variables resulting in prohibitive computational complexity and large memory demands in large scale problems.

In this paper we propose an alternative SCA algorithm for structured sparse multilinear low-rank tensor decomposition which avoids these difficulties and which is therefore suitable for large scale optimization. The algorithm is fully parallel, involves only simple operations and is therefore well suited for implementation on parallel hardware architectures and graphical processing units (GPUs).

II. Signal Model

We consider the two-dimensional (2D) harmonic retrieval problem where a superposition of P source signals with 2D frequencies $\{(\theta_p, \phi_p)\}$, $(p = 1, \ldots, P)$ is recorded over L snapshots. Let, $\boldsymbol{a}(\theta) = [1, e^{-\mathrm{j}\pi\theta}, \ldots, e^{-\mathrm{j}\pi(M-1)\theta}]^{\mathrm{T}} \in \mathbb{C}^M$

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

and $\boldsymbol{b}(\phi) = [1, e^{-\mathrm{j}\pi\phi}, \ldots, e^{-\mathrm{j}\pi(N-1)\phi}]^{\mathrm{T}} \in \mathbb{C}^N$ denote the frequency responses along the first and the second sampling axes with sample dimensions M and N, respectively for $\theta = [-1, 1]$, $\phi = [-1, 1]$. The measurements form a three-dimensional tensor

$$\boldsymbol{\mathcal{Y}} = \sum_{p=1}^{P} \boldsymbol{a}_p \circ \boldsymbol{b}_p \circ \boldsymbol{h}_p + \boldsymbol{\mathcal{W}} \quad \in \mathbb{C}^{M\times N\times L} \tag{1}$$

where, $h_p(\ell)$ denotes the complex valued source amplitudes of the p-th harmonic in snapshot ℓ, $(\ell = 1, \ldots, L)$, $\boldsymbol{h}_p = [h_p(1), \ldots, h_p(L)]^{\mathrm{T}} \in \mathbb{C}^L$ is the corresponding amplitude vector of source ℓ, $\boldsymbol{a} \circ \boldsymbol{b} \circ \boldsymbol{h} \in \mathbb{C}^{M\times N\times L}$ denotes the outer tensor product of vectors $\boldsymbol{a} \in \mathbb{C}^M$, $\boldsymbol{b} \in \mathbb{C}^N$, and $\boldsymbol{h} \in \mathbb{C}^L$, i.e $[\boldsymbol{a} \circ \boldsymbol{b} \circ \boldsymbol{h}]_{mn\ell} = a_m b_n h_\ell$ (see also [20]) and $\boldsymbol{\mathcal{W}}$ denotes zero-mean additive white circularly i.i.d. Gaussian measurement noise of variance σ^2. Furthermore, we define the frequency response matrices $\boldsymbol{A} = [\boldsymbol{a}(\theta_1), \ldots, \boldsymbol{a}(\theta_P)]$ and $\boldsymbol{B} = [\boldsymbol{b}(\phi_1), \ldots, \boldsymbol{b}(\phi_P)]$ and the source amplitude matrix $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_P]$.

In this paper, we focus for simplicity of notation on the two-dimensional harmonic retrieval problem. We remark that the tensor model (1) can be extended to higher dimensions by incorporating additional sampling axes, e.g., for the multi-dimensional harmonic retrieval problem.

Let $\boldsymbol{\mathcal{Y}}^{(k)}$ denote the k-th order unfolding of tensor $\boldsymbol{\mathcal{Y}}$ in (1) [21]. Then the first, second and third order unfolding can be written, respectively, as

$$\boldsymbol{\mathcal{Y}}^{(1)} = \boldsymbol{A}\ (\boldsymbol{H} \odot \boldsymbol{B})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(1)} \quad \in \mathbb{C}^{M\times NL} \tag{2}$$

$$\boldsymbol{\mathcal{Y}}^{(2)} = \boldsymbol{B}\ (\boldsymbol{H} \odot \boldsymbol{A})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(2)} \quad \in \mathbb{C}^{N\times ML} \tag{3}$$

$$\boldsymbol{\mathcal{Y}}^{(3)} = \boldsymbol{H}\ (\boldsymbol{B} \odot \boldsymbol{A})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(3)} \quad \in \mathbb{C}^{L\times MN}, \tag{4}$$

where $\odot$ denotes the Khatri-Rao product and $\boldsymbol{\mathcal{W}}^{(k)}$ is the k-th unfolding of the noise tensor [21].

III. Sparse Recovery

In this work we consider a sparse recovery approach for the multilinear low-rank factorization model in (1) where we exploit the parametric structure of the frequency response vectors $\boldsymbol{a}(\theta)$ and $\boldsymbol{b}(\phi)$ along the first and the second sampling axes, respectively. In this approach a frequency response matrix $\boldsymbol{B}^{\mathrm{d}}$ is formed by discretizing the second frequency parameter in Q candidate values $\{\phi_q^{\mathrm{d}}\}_{q=1}^{Q}$ over the frequency interval of interest for $Q \gg P$. The dictionary matrix is given as, $\boldsymbol{B}^{\mathrm{d}} = [\boldsymbol{b}(\phi_1^{\mathrm{d}}), \cdots, \boldsymbol{b}(\phi_Q^{\mathrm{d}})] \in \mathbb{C}^{N\times Q}$. Assume for simplicity that the true frequencies ϕ_p, $(p = 1, \ldots, P)$ lie on the parameter grid of candidate frequencies. Correspondingly, we can define the column sparse matrices $\boldsymbol{A}^{\mathrm{c}} = [\boldsymbol{a}_1^{\mathrm{c}}, \cdots, \boldsymbol{a}_Q^{\mathrm{c}}] \in \mathbb{C}^{M\times Q}$ and $\boldsymbol{H}^{\mathrm{c}} = [\boldsymbol{h}_1^{\mathrm{c}}, \cdots, \boldsymbol{h}_Q^{\mathrm{c}}] \in \mathbb{C}^{L\times Q}$ such that if the q-th candidate frequency ϕ_q^{d} is equal to the frequency ϕ_p of the p-th signal, then column $\boldsymbol{a}_q^{\mathrm{c}}$ is equal to the steering vector of the p-th signal and is identical to the zero vector, otherwise. Hence, define

$$\boldsymbol{a}_q^{\mathrm{c}} = \begin{cases} \boldsymbol{a}(\theta_p), & \text{if } \phi_q^{\mathrm{d}} = \phi_p; \\ \boldsymbol{0}_M, & \text{otherwise.} \end{cases} \tag{5}$$

Similarly, define

$$\boldsymbol{h}_q^{\mathrm{c}} = \begin{cases} \boldsymbol{h}_q, & \text{if } \phi_q^{\mathrm{d}} = \phi_p; \\ \boldsymbol{0}_L, & \text{otherwise.} \end{cases} \tag{6}$$

With these definitions we can, e.g., write (1) equivalently as

$$\boldsymbol{\mathcal{Y}}^{(1)} = \boldsymbol{A}^{\mathrm{c}}\ (\boldsymbol{H}^{\mathrm{c}} \odot \boldsymbol{B}^{\mathrm{d}})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(1)} \quad \in \mathbb{C}^{M\times NL} \tag{7}$$

$$\boldsymbol{\mathcal{Y}}^{(2)} = \boldsymbol{B}^{\mathrm{d}}\ (\boldsymbol{H}^{\mathrm{c}} \odot \boldsymbol{A}^{\mathrm{c}})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(2)} \quad \in \mathbb{C}^{N\times ML} \tag{8}$$

$$\boldsymbol{\mathcal{Y}}^{(3)} = \boldsymbol{H}^{\mathrm{c}}\ (\boldsymbol{B}^{\mathrm{d}} \odot \boldsymbol{A}^{\mathrm{c}})^{\mathrm{T}} + \boldsymbol{\mathcal{W}}^{(3)} \quad \in \mathbb{C}^{L\times MN}. \tag{9}$$

Based on the sparse representation model (8) the multilinear low-rank factorization problem can be expressed as a nuclear norm based convex optimization problem, as given in [3], where a lifting approach is used.

$$\min_{\boldsymbol{G}} \frac{1}{2} \left\| \boldsymbol{\mathcal{Y}}^{(2)} - \boldsymbol{B}^{\mathrm{d}} \boldsymbol{G}^{\mathrm{T}} \right\|_{\mathrm{F}}^2 + \lambda \sum_{q=1}^{Q} \left\| \boldsymbol{G}_q \right\|_*, \tag{10}$$

$\left\| \boldsymbol{X} \right\|_* = \mathrm{Tr}(\sqrt{\boldsymbol{X}^{\mathrm{H}}\boldsymbol{X}})$ denotes the nuclear norm of a matrix $\boldsymbol{X}$. Denote the matrix $\boldsymbol{G} = (\boldsymbol{H}^{\mathrm{c}} \odot \boldsymbol{A}^{\mathrm{c}})$ and the block variables $\boldsymbol{g}_q = (\boldsymbol{h}_q \odot \boldsymbol{a}_q) \in \mathbb{C}^{ML\times 1}$, $(q = 1, \ldots, Q)$. To avoid the bi-linear product terms take the nuclear norm of submatrix $\boldsymbol{G}_q = \mathrm{unvec}(\boldsymbol{g}_q) \in \mathbb{C}^{M\times L}$, where $\mathrm{unvec}(\cdot)$ is defined such that $\mathrm{vec}\,(\boldsymbol{G})_q = \boldsymbol{g}_q$. The first summand in (10) represents the data fidelity term, while the second summand is added as a sparsity inducing regularization term to enforce column sparse solutions corresponding to (5) and (6). However, the lifting procedure significantly increases the number of optimization variables from $Q(M+L)$ in the original formulation to QML in the lifted formulation resulting in significant overhead. Based on the SCA framework introduced in [22], in this paper we directly address the nuclear norm based problem without lifting. In order to avoid the difficulties arising from the non-differentiable and the non-decomposable nuclear norm regularization term presented in (10), we use the property that, the nuclear norm of a low-rank matrix $\boldsymbol{X}$ can be written as the following optimization problem [23]:

$$\left\| \boldsymbol{X} \right\|_* = \min_{(\boldsymbol{X},\boldsymbol{Q},\boldsymbol{P})} \frac{1}{2} (\|\boldsymbol{Q}\|_{\mathrm{F}}^2 + \|\boldsymbol{P}\|_{\mathrm{F}}^2) \quad \text{s.t.} \quad \boldsymbol{X} = \boldsymbol{Q}\boldsymbol{P}^{\mathrm{T}}, \tag{11}$$

where $\boldsymbol{X} \in \mathbb{C}^{M\times L}$, $\boldsymbol{Q} \in \mathbb{C}^{M\times R}$ and $\boldsymbol{P} \in \mathbb{C}^{L\times R}$ for dimension R usually much smaller than M and L and larger than the rank of $\boldsymbol{X}$. Note that the minimizers $\boldsymbol{Q}^\star$ and $\boldsymbol{P}^\star$ of (11) satisfy $\|\boldsymbol{Q}^\star\|_{\mathrm{F}}^2 = \|\boldsymbol{P}^\star\|_{\mathrm{F}}^2$. Inserting property (11) in the problem in (10) and making use of (7)-(9) yields the following equivalent problem

$$(\boldsymbol{A}^\star, \boldsymbol{H}^\star) = \underset{(\boldsymbol{A},\boldsymbol{H})}{\arg\min}\ f(\boldsymbol{A}, \boldsymbol{H}) \tag{12}$$

where,

$$\begin{aligned} f(\boldsymbol{A}, \boldsymbol{H}) &= \left\| \boldsymbol{\mathcal{Y}}^{(1)} - \boldsymbol{A}(\boldsymbol{H} \odot \boldsymbol{B}^{\mathrm{d}})^{\mathrm{T}} \right\|_{\mathrm{F}}^2 + \lambda(\|\boldsymbol{A}\|_{\mathrm{F}}^2 + \|\boldsymbol{H}\|_{\mathrm{F}}^2) \\ &= \left\| \boldsymbol{\mathcal{Y}}^{(3)} - \boldsymbol{H}(\boldsymbol{B}^{\mathrm{d}} \odot \boldsymbol{A})^{\mathrm{T}} \right\|_{\mathrm{F}}^2 + \lambda(\|\boldsymbol{A}\|_{\mathrm{F}}^2 + \|\boldsymbol{H}\|_{\mathrm{F}}^2). \end{aligned} \tag{13}$$

Although the problem (12) is nonconvex, every stationary point of the problem is an optimal solution of the nuclear norm based problem presented in (10) under certain regularity conditions given in [22].

IV. Successive convex approximation algorithm

There are various ways to solve the nonconvex non-differentiable problem in (10) or the equivalent smooth reformulations in (12). In [24] a semidefinite programming approach is used and in [25] the block coordinate method is presented. In this work, we use the successive convex approximation (SCA) framework proposed in [22], which fully supports massive parallel processing and implementation on parallel hardware architectures. Instead of solving problem (12) directly the idea of the SCA framework is to solve a sequence of approximate problems, where the approximate problem is much easier to solve than the original problem (12), e.g., in parallel and based on closed form expressions. The approximate problems are obtained by approximating in iteration t the original objective function $f(\boldsymbol{A},\boldsymbol{H})$ in (13) at the current point $(\boldsymbol{A}^t,\boldsymbol{H}^t)$ by a properly designed decomposable approximate function $\tilde{f}\big((\boldsymbol{A},\boldsymbol{H});(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)$. In order to guarantee convergence to a stationary point of the original problem, the approximate functions need to satisfy certain regularity conditions as stated in [22, Assumptions *(A1)-(A5)*]. Particularly, the approximate function $\tilde{f}\big((\boldsymbol{A},\boldsymbol{H});(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)$ must be convex in $(\boldsymbol{A},\boldsymbol{H})$ and continuously differentiable for given $(\boldsymbol{A}^t,\boldsymbol{H}^t)$, continuous in $(\boldsymbol{A}^t,\boldsymbol{H}^t)$ for given $(\boldsymbol{A},\boldsymbol{H})$ and the gradients of the approximate function must coincide with the gradient of the original function at $(\boldsymbol{A}^t,\boldsymbol{H}^t)$ (see [22] for details). From the optimal solution of the approximate problem a descent direction of the original problem is obtained, which is only true if we are not already at a stationary point. The variable update is performed with a stepsize computed from the exact or successive line search [22]. Note that although the problem (12) is not jointly convex in $\boldsymbol{A}$ and $\boldsymbol{H}$, it is individually convex in either of the variables if the other variable is fixed. Let us introduce the following general partitions $\boldsymbol{A}=[\boldsymbol{A}_1,\boldsymbol{A}_2,\dots,\boldsymbol{A}_J]$, $\boldsymbol{B}^{\mathrm{d}}=[\boldsymbol{B}^{\mathrm{d}}_1,\boldsymbol{B}^{\mathrm{d}}_2,\dots,\boldsymbol{B}^{\mathrm{d}}_J]$ and $\boldsymbol{H}=[\boldsymbol{H}_1,\boldsymbol{H}_2,\dots,\boldsymbol{H}_J]$ for J denoting the number of submatrix partitions. Denote, the submatrices $\boldsymbol{A}^t_{-j}=[\boldsymbol{A}^t_1,\cdots,\boldsymbol{A}^t_{j-1},\boldsymbol{A}^t_{j+1},\cdots,\boldsymbol{A}^t_J]$, $\boldsymbol{B}^{\mathrm{d}}_{-j}=[\boldsymbol{B}^{\mathrm{d}}_1,\cdots,\boldsymbol{B}^{\mathrm{d}}_{j-1},\boldsymbol{B}^{\mathrm{d}}_{j+1},\cdots,\boldsymbol{B}^{\mathrm{d}}_J]$ and $\boldsymbol{H}^t_{-j}=[\boldsymbol{H}^t_1,\cdots,\boldsymbol{H}^t_{j-1},\boldsymbol{H}^t_{j+1},\cdots,\boldsymbol{H}^t_J]$, then at given point $(\boldsymbol{A}^t,\boldsymbol{H}^t)$ in iteration t, the approximate problem comprises of minimizing the approximate function,

$$\tilde{f}\big((\boldsymbol{A},\boldsymbol{H});(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)=\tilde{f}_{\boldsymbol{A}}\big(\boldsymbol{A};(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)+\tilde{f}_{\boldsymbol{H}}\big(\boldsymbol{H};(\boldsymbol{H}^t,\boldsymbol{A}^t)\big) \tag{14}$$

where,

$$\begin{aligned}\tilde{f}_{\boldsymbol{A}}\big(\boldsymbol{A};(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)&=\sum_{j=1}^{J}\tilde{f}_{\boldsymbol{A}_j}\big(\boldsymbol{A}_j;(\boldsymbol{H}^t,\boldsymbol{A}^t_{-j})\big)\\ \tilde{f}_{\boldsymbol{H}}\big(\boldsymbol{H};(\boldsymbol{H}^t,\boldsymbol{A}^t)\big)&=\sum_{j=1}^{J}\tilde{f}_{\boldsymbol{H}_j}\big(\boldsymbol{H}_j;(\boldsymbol{H}^t_{-j},\boldsymbol{A}^t)\big)\end{aligned} \tag{15}$$

and,

$$\begin{aligned}\tilde{f}_{\boldsymbol{A}_j}\big(\boldsymbol{A}_j;(\boldsymbol{H}^t,\boldsymbol{A}^t_{-j})\big)&=f\big(\boldsymbol{A}_j;(\boldsymbol{H}^t,\boldsymbol{A}^t_{-j})\big) \qquad (16)\\ &=\Big\|\boldsymbol{\mathcal{Y}}^{(1)}_{-j}-\boldsymbol{A}_j(\boldsymbol{H}^t_j\odot\boldsymbol{B}^{\mathrm{d}}_j)^{\mathrm{T}}\Big\|_{\mathrm{F}}^2+\lambda\|\boldsymbol{A}_j\|_{\mathrm{F}}^2\\ \tilde{f}_{\boldsymbol{H}_j}\big(\boldsymbol{H}_j;(\boldsymbol{H}^t_{-j},\boldsymbol{A}^t)\big)&=f\big(\boldsymbol{H}_j;(\boldsymbol{H}^t_{-j},\boldsymbol{A}^t)\big) \qquad (17)\\ &=\Big\|\boldsymbol{\mathcal{Y}}^{(3)}_{-j}-\boldsymbol{H}_j(\boldsymbol{B}^{\mathrm{d}}_j\odot\boldsymbol{A}^t_j)^{\mathrm{T}}\Big\|_{\mathrm{F}}^2+\lambda\|\boldsymbol{H}_j\|_{\mathrm{F}}^2\end{aligned}$$

for $\boldsymbol{\mathcal{Y}}^{(1)}_{-j}=\boldsymbol{\mathcal{Y}}^{(1)}-\boldsymbol{A}^t_{-j}(\boldsymbol{H}^t_{-j}\odot\boldsymbol{B}^{\mathrm{d}}_{-j})^{\mathrm{T}}$ and $\boldsymbol{\mathcal{Y}}^{(3)}_{-j}=\boldsymbol{\mathcal{Y}}^{(3)}-\boldsymbol{H}^t_{-j}(\boldsymbol{B}^{\mathrm{d}}_{-j}\odot\boldsymbol{A}^t_{-j})^{\mathrm{T}}$. Denote the solution of the approximate problem

$$(\mathbb{B}_{\boldsymbol{A}}(\boldsymbol{A}^t,\boldsymbol{H}^t),\mathbb{B}_{\boldsymbol{H}}(\boldsymbol{A}^t,\boldsymbol{H}^t))=\underset{(\boldsymbol{A},\boldsymbol{H})}{\arg\min}\;\tilde{f}\big((\boldsymbol{A},\boldsymbol{H});(\boldsymbol{A}^t,\boldsymbol{H}^t)\big) \tag{18}$$

as, $\mathbb{B}_{\boldsymbol{A}}(\boldsymbol{A}^t,\boldsymbol{H}^t)=[\mathbb{B}_{\boldsymbol{A}_1}(\boldsymbol{A}^t_{-1},\boldsymbol{H}^t),\cdots,\mathbb{B}_{\boldsymbol{A}_J}(\boldsymbol{A}^t_{-J},\boldsymbol{H}^t)]$ and $\mathbb{B}_{\boldsymbol{H}}(\boldsymbol{A}^t,\boldsymbol{H}^t)=[\mathbb{B}_{\boldsymbol{H}_1}(\boldsymbol{A}^t,\boldsymbol{H}^t_{-1}),\cdots,\mathbb{B}_{\boldsymbol{H}_J}(\boldsymbol{A}^t,\boldsymbol{H}^t_{-J})]$, then using (14)-(17) the approximate problem decomposes into parallel subproblems, that can be solved efficiently. For each subproblem closed form solutions are obtained as,

$$\begin{aligned}\mathbb{B}_{\boldsymbol{A}_j}(\boldsymbol{A}^t_{-j},\boldsymbol{H}^t)&=\underset{\boldsymbol{A}_j}{\arg\min}\;\tilde{f}_{\boldsymbol{A}_j}\big(\boldsymbol{A}_j;(\boldsymbol{A}^t_{-j},\boldsymbol{H}^t)\big) \qquad (19)\\ &=\boldsymbol{\mathcal{Y}}^{(1)}_{-j}(\boldsymbol{H}^t_j\odot\boldsymbol{B}^{\mathrm{d}}_j)^*\big((\boldsymbol{H}^t_j\odot\boldsymbol{B}^{\mathrm{d}}_j)^{\mathrm{T}}(\boldsymbol{H}^t_j\odot\boldsymbol{B}^{\mathrm{d}}_j)^*+\lambda\boldsymbol{I}\big)^{-1}\\ \mathbb{B}_{\boldsymbol{H}_j}(\boldsymbol{H}^t_{-j},\boldsymbol{A}^t)&=\underset{\boldsymbol{H}_j}{\arg\min}\;\tilde{f}_{\boldsymbol{H}_j}\big(\boldsymbol{H}_j;(\boldsymbol{A}^t,\boldsymbol{H}^t_{-j})\big) \qquad (20)\\ &=\boldsymbol{\mathcal{Y}}^{(3)}_{-j}(\boldsymbol{B}^{\mathrm{d}}_j\odot\boldsymbol{A}^t_j)^*\big((\boldsymbol{B}^{\mathrm{d}}_j\odot\boldsymbol{A}^t_j)^{\mathrm{T}}(\boldsymbol{B}^{\mathrm{d}}_j\odot\boldsymbol{A}^t_j)^*+\lambda\boldsymbol{I}\big)^{-1}.\end{aligned}$$

Then according to the SCA framework in [22] $(\mathbb{B}\boldsymbol{A}^t-\boldsymbol{A}^t,\mathbb{B}\boldsymbol{H}^t-\boldsymbol{H}^t)$ is a decent direction for problem (12) at point $(\boldsymbol{A}^t,\boldsymbol{H}^t)$, if $(\boldsymbol{A}^t,\boldsymbol{H}^t)$ is not already a stationary point of the problem. The variable update in iteration t is carried out according to

$$\boldsymbol{A}^{t+1}=\boldsymbol{A}^t+\gamma(\mathbb{B}\boldsymbol{A}^t-\boldsymbol{A}^t) \tag{21}$$

$$\boldsymbol{H}^{t+1}=\boldsymbol{H}^t+\gamma(\mathbb{B}\boldsymbol{H}^t-\boldsymbol{H}^t), \tag{22}$$

where $\gamma\in(0,1]$ is the stepsize selected by the exact line search scheme, i.e., by inserting (21)-(22) in (13) and determining the stepsize that yields the largest decrease of the objective function. Define $\Delta\boldsymbol{A}^t=\mathbb{B}\boldsymbol{A}^t-\boldsymbol{A}^t$ and $\Delta\boldsymbol{H}^t=\mathbb{B}\boldsymbol{H}^t-\boldsymbol{H}^t$. The exact line search corresponds to minimizing the function:

$$f(\gamma)=\Big\|\boldsymbol{\mathcal{Y}}^{(1)}-(\boldsymbol{A}^t+\gamma\Delta\boldsymbol{A}^t)\big((\boldsymbol{H}^t+\gamma\Delta\boldsymbol{H}^t)\odot\boldsymbol{B}^{\mathrm{d}}\big)^{\mathrm{T}}\Big\|_{\mathrm{F}}^2+\lambda\big\|(\boldsymbol{A}^t+\gamma\Delta\boldsymbol{A}^t)\big\|_{\mathrm{F}}^2+\lambda\,\big\|(\boldsymbol{H}^t+\gamma\Delta\boldsymbol{H}^t)\big\|_{\mathrm{F}}^2 \tag{23}$$

over the interval (0,1]. Hence the optimal stepsize is obtained by minimizing a four order polynomial:

$$\begin{aligned}\gamma^t&=\underset{0\le\gamma\le1}{\arg\min}\;f(\gamma)\\ &=\underset{0\le\gamma\le1}{\arg\min}\{a\gamma^4+b\gamma^3+c\gamma^2+d\gamma^1+e\gamma^0\}\end{aligned} \tag{24}$$

with polynomial coefficients given

$$a = \text{Tr}\{\boldsymbol{M}_3\boldsymbol{M}_3^{\text{H}}\} = \|\boldsymbol{M}_3\|_{\text{F}}^2 \tag{25}$$

$$b = \text{Tr}\{\boldsymbol{M}_2\boldsymbol{M}_3^{\text{H}} + \boldsymbol{M}_3\boldsymbol{M}_2^{\text{H}}\} \tag{26}$$

$$c = \text{Tr}\{\boldsymbol{M}_2\boldsymbol{M}_2^{\text{H}} + \boldsymbol{M}_3\boldsymbol{M}_1^{\text{H}} + \boldsymbol{M}_1\boldsymbol{M}_3^{\text{H}}\} + \lambda \left\|\Delta\boldsymbol{A}^t\right\|_{\text{F}}^2 + \lambda \left\|\Delta\boldsymbol{H}^t\right\|_{\text{F}}^2 \tag{27}$$

$$d = \text{Tr}\{\boldsymbol{M}_2\boldsymbol{M}_1^{\text{H}} + \boldsymbol{M}_1\boldsymbol{M}_2^{\text{H}}\} + \lambda\text{Tr}\{\boldsymbol{A}^t\Delta\boldsymbol{A}^{t\text{H}} + \Delta\boldsymbol{A}^t\boldsymbol{A}^{t\text{H}}\} + \lambda\text{Tr}\{\boldsymbol{H}^t\Delta\boldsymbol{H}^{t\text{H}} + \Delta\boldsymbol{H}^t\boldsymbol{H}^{t\text{H}}\}, \tag{28}$$

where $\boldsymbol{M}_1 = \boldsymbol{\mathcal{Y}}^{(1)} - \boldsymbol{A}^t(\boldsymbol{H}^t \odot \boldsymbol{B}^{\text{d}})^{\text{T}}$, $\boldsymbol{M}_2 = -\Delta\boldsymbol{A}^t(\boldsymbol{H}^t \odot \boldsymbol{B}^{\text{d}})^{\text{T}} - \boldsymbol{A}^t(\Delta\boldsymbol{H}^t \odot \boldsymbol{B}^{\text{d}})^{\text{T}}$ and $\boldsymbol{M}_3 = -\Delta\boldsymbol{A}^t(\Delta\boldsymbol{H}^t \odot \boldsymbol{B}^{\text{d}})^{\text{T}}$. Finding the minimum in (23) is equivalent to finding the nonnegative real root of a third order polynomial.

The solution of a third order polynomial is a set of three roots, where at least one of the roots is real-valued. Thus, in order to solve (24) we determine the set of real-valued roots in the closed interval $\gamma = [0, 1]$. Then, we evaluate the polynomial in (24) for all the roots in the set and the stepsize is the root that yields the minimal function value.
The SCA Algorithm is summarized in Algorithm 1:

Algorithm 1 The proposed successive convex approximation framework corresponding to problem (13) for some small precision constant ϵ.

INIT: $t = 0$, $\boldsymbol{A}^0$ and $\boldsymbol{H}^0$ non-zero and fixed.
S1: Compute $\mathbb{B}\boldsymbol{A}^t$ and $\mathbb{B}\boldsymbol{H}^t$ according to (19) and (20).
S2: Determine the stepsize γ^t according to (24)
S3: Update $\boldsymbol{A}^{t+1}$ and $\boldsymbol{H}^{t+1}$ according to (21) and (22)
S4: If $\left\|\boldsymbol{A}^{t+1} - \boldsymbol{A}^t\right\|_{\text{F}} \leq \epsilon$ and $\left\|\boldsymbol{H}^{t+1} - \boldsymbol{H}^t\right\|_{\text{F}} \leq \epsilon$
STOP: otherwise $t \leftarrow t + 1$ go to **S1**

V. Numerical Results

In this section, we perform numerical tests for the proposed Algorithm 1 for solving problem (12) and compare the algorithm to the lifting based SCA algorithm proposed in [3] and other state-of-the-art methods for two-dimensional harmonic retrieval. We consider the signal model in (1) with $M = 4$, $N = 16$ and $L = 3$ and a superposition of $P = 5$ source signals with the two-dimensional frequencies chosen as $(\phi_1, \theta_1) = (0.0213, 0.423)$, $(\phi_2, \theta_2) = (0.1538, 0.688)$, $(\phi_3, \theta_3) = (0.2463, -0.082)$, $(\phi_4, \theta_4) = (0.4462, -0.517)$ and $(\phi_5, \theta_5) = (0.6275, -0.264)$. The corresponding complex valued amplitudes h_p have unit power with uniform random phases. The signal-to-noise ratio is defined as $\text{SNR} = 1/\sigma^2$ and for our tests we assume $\text{SNR} = 5\text{dB}$. The number of updated submatrices per iteration is $J = 16$.

In the first numerical test, we compare the convergence speed of the proposed SCA method for the problem reformulation (12) with the lifting based SCA algorithm proposed in [3] for problem (10). We display the normalized reconstruction error defined as $\left\|\hat{\boldsymbol{X}} - \boldsymbol{X}^t\right\|_{\text{F}} / \left\|\hat{\boldsymbol{X}}\right\|_{\text{F}}$, where $\hat{\boldsymbol{X}}$ denotes the solution of the algorithm and $\boldsymbol{X}^t$ denotes the approximate solution in the t-th iteration. For both algorithms, we consider one parallel update of all the blocks as an iteration. For the first experiment we select a regularization parameter $\lambda = \lambda_0/8$ where $\lambda_0 = \max_q \left\|(\boldsymbol{B}_q^d \otimes \boldsymbol{I}_M)^{\text{H}}\boldsymbol{\mathcal{Y}}^{(3)^{\text{T}}}\right\|_2$ as suggested in [3]. The grid consists of $Q = 160$ uniformly discretized points. As can be seen in Fig. 1 the proposed algorithm outperforms the nuclear norm based SCA algorithm in [3] in terms of convergence speed. Moreover, the effect of the sampling dimension M on the convergence speed is shown. It can be seen that as the sampling dimension increases, both algorithms show a slightly improved performance in terms of the convergence speed.

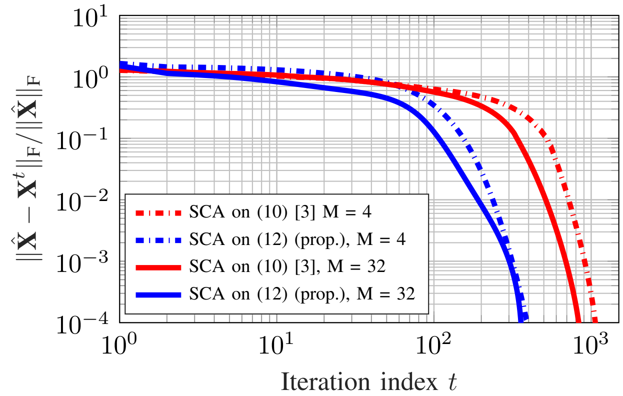

Figure 1. Convergence speed for different number of receive antennas.

For performance evaluation of the root-mean-square estimation error (RMSE) of the 2D harmonic retrieval, we compare in Figs. 2 a) and b) the RMSE for the proposed method to that of the subspace based 2D-MUSIC method [2], the 2D-Root-RARE estimator [9], the sparse recovery method space-alternating orthogonal matching pursuit (SA-OMP) [26], the sparse recovery method nuclear norm based SCA algorithm in [3] and the corresponding Cramer-Rao bound [8]. The RMSEs are averaged over 100 Monte Carlo runs to estimate the frequencies ϕ_p and θ_p for $p = 1, \ldots, \hat{P}$ which are resolved according to [3]. The frequencies $\{\phi\}_{p=1}^{\hat{P}}$ are estimated from the column support of the matrix $\boldsymbol{H}^t$ and $\{\theta\}_{p=1}^{\hat{P}}$ are estimated from the non-zero columns in $\boldsymbol{A}^t$ using e.g. the 1D MUSIC method. Moreover, adaptive grid refinement is used for the sparse recovery methods [27]. For the MUSIC and Root-RARE methods we have used spatial smoothing and forward-backward averaging in order to avoid a rank deficient sample covariance matrix due to the low number of snapshots. The sparse recovery based method use the same regularization parameter that is chosen such that the recovered number of source signals is equal to the true number of sources $P = 5$. Among all the compared methods, the SA-OMP and MUSIC method perform a joint parameter estimation. SA-OMP shows a bias in the high SNR regime due to its greedy nature. The performance of our proposed algorithm is comparable to the SCA algorithm in [3] and the MUSIC method. However, the advantage of our proposed method lies in the reduced computational complexity and the convergence speed.

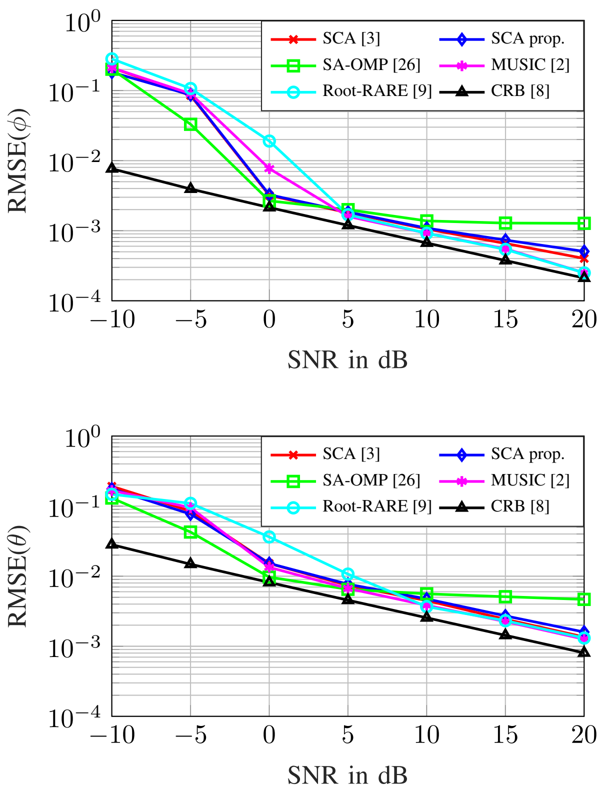

Figure 2. a) RMSE for ϕ frequency. b) RMSE for θ.

VI. Conclusion

In this paper, we have proposed an algorithm based on the recently proposed successive convex approximation framework of [22] applied to the for structured multilinear low-rank tensor decomposition problem. The algorithm exhibits full parallelization, reduced computational complexity and faster convergence as the state-of-the-art method for this problem. Unlike the algorithm in [3], the proposed algorithm does not require a lifting procedure, which increased the number of optimization variables. Simulation results confirm that the estimation performance of the proposed SCA algorithm is similar to the state-of-the arts and close to the Cramer-Rao performance bound.

References

[1] M. C. Vanderveen, A. J. Van der Veen, and A. Paulraj, "Estimation of multipath parameters in wireless communications," *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 682–690, March 1998.

[2] J. Li, J. Conan, and S. Pierre, "Joint estimation of channel parameters for MIMO communication systems," in *2nd International Symposium on Wireless Communication Systems*, 2005, pp. 22–26.

[3] C. Steffens, Y. Yang, and M. Pesavento, "Multidimensional sparse recovery for MIMO channel parameter estimation," in *24th European Signal Processing Conference (EUSIPCO)*, Aug 2016, pp. 66–70.

[4] Y. Zhang, D. Wang, J. Wang, and X. You, "Channel estimation for massive MIMO-OFDM systems by tracking the joint angle-delay subspace," *IEEE Access*, vol. 4, no. 10, pp. 166–179, 2016.

[5] R. Shafin, L. Liu, Y. Li, A. Wang, and J. Zhang, "Angle and delay estimation for 3-D massive MIMO/FD-MIMO systems based on parametric channel modeling," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 5370–5383, Aug 2017.

[6] A. B. Gershman, M. Rübsamen, and M. Pesavento, "One- and two-dimensional direction-of-arrival estimation: An overview of search-free techniques," *Signal Processing*, vol. 90, no. 5, pp. 1338 – 1349, 2010, Special Section on Statistical Signal and Array Processing.

[7] M. Haardt, M. Pesavento, F. Roemer, and M. N. E. Korso, "Chapter 15 - Subspace methods and exploitation of special array structures," in *Academic Press Library in Signal Processing: Volume 3*, ser. Academic Press Library in Signal Processing, A. M. Zoubir, M. Viberg, R. Chellappa, and S. Theodoridis, Eds. Elsevier, 2014, vol. 3, pp. 651 – 717.

[8] C. P. Mathews and M. D. Zoltowski, "Eigenstructure techniques for 2-D angle estimation with uniform circular arrays," *IEEE Transactions on signal processing*, vol. 42, no. 9, pp. 2395–2407, 1994.

[9] M. Pesavento, C. F. Mecklenbräuker, and J. F. Böhme, "Multidimensional rank reduction estimator for parametric MIMO channel models," *EURASIP Journal on Applied Signal Processing*, vol. 2004, pp. 1354–1363, 2004.

[10] M. Pesavento, "Fast algorithms for multidimensional harmonic retrieval," Dissertation, Ruhr-Universität Bochum, Germany, Feb. 2005.

[11] M. D. Zoltowski, M. Haardt, and C. P. Mathews, "Closed-form 2-D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT," *IEEE Transactions on Signal Processing*, vol. 44, no. 2, pp. 316–328, Feb 1996.

[12] G. F. Hatke and K. W. Forsythe, "A class of polynomial rooting algorithms for joint azimuth/elevation estimation using multidimensional arrays," in *28th Asilomar Conference on Signals, Systems and Computers*, vol. 1, Oct 1994, pp. 694–699.

[13] D. L. Donoho, "Superresolution via sparsity constraints," *SIAM journal on mathematical analysis*, vol. 23, no. 5, pp. 1309–1331, 1992.

[14] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to basis mismatch in compressed sensing," *IEEE Transactions on Signal Processing*, vol. 59, no. 5, pp. 2182–2195, 2011.

[15] Y. Chi and Y. Chen, "Compressive two-dimensional harmonic retrieval via atomic norm minimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 4, pp. 1030–1042, Feb 2015.

[16] Z. Yang, L. Xie, and P. Stoica, "Vandermonde decomposition of multilevel toeplitz matrices with application to multidimensional super-resolution," *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3685–3701, June 2016.

[17] Z. Tian, Z. Zhang, and Y. Wang, "Low-complexity optimization for two-dimensional direction-of-arrival estimation via decoupled atomic norm minimization," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 3071–3075.

[18] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of optimization theory and applications*, vol. 109, no. 3, pp. 475–494, 2001.

[19] C. Steffens, M. Pesavento, and M. E. Pfetsch, "A compact formulation for the $\ell_{2,1}$ mixed-norm minimization problem," *IEEE Transactions on Signal Processing*, vol. 66, no. 6, pp. 1483–1497, March 2018.

[20] E. Acar and B. Yener, "Unsupervised multiway data analysis: A literature survey," *IEEE Transactions on Knowledge and Data engineering*, vol. 21, no. 1, pp. 6–20, 2009.

[21] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[22] Y. Yang, M. Pesavento, S. Chatzinotas, and B. Ottersten, "Successive convex approximation algorithms for sparse signal estimation with nonconvex regularizations," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1286–1302, Dec 2018.

[23] S. Burer and R. D. Monteiro, "A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization," *Mathematical Programming*, vol. 95, no. 2, pp. 329–357, 2003.

[24] M. Fazel, H. Hindi, S. P. Boyd *et al.*, "A rank minimization heuristic with application to minimum order system approximation," in *Proceedings of the American control conference*, vol. 6, 2001, pp. 4734–4739.

[25] C. Steffens, P. Parvazi, and M. Pesavento, "Direction finding and array calibration based on sparse reconstruction in partly calibrated arrays," in *IEEE 8th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 2014, pp. 21–24.

[26] S.-H. Byun, W. Seong, and S.-M. Kim, "Sparse underwater acoustic channel parameter estimation using a wideband receiver array," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 4, pp. 718–729, 2013.

[27] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Transactions on signal processing*, vol. 53, no. 8, pp. 3010–3022, 2005.