TENSOR-STRUCTURE STRUCTURED LEAST SQUARES (TS-SLS) TO IMPROVE THE PERFORMANCE OF MULTI-DIMENSIONAL ESPRIT-TYPE ALGORITHMS

Florian Roemer and Martin Haardt

Ilmenau University of Technology, Communications Research Laboratory
P.O. Box 100565, D-98684 Ilmenau, Germany
florian.roemer@ieee.org, martin.haardt@tu-ilmenau.de

***Abstract* — Multidimensional ESPRIT-type parameter estimation algorithms obtain their frequency estimates from the solution of sets of highly structured equations (the shift invariance equations). The Structured Least Squares (SLS) algorithm is known as an efficient method to obtain these solutions since the inherent structure is explicitly taken into account.**

In this contribution we show that if the underlying R-dimensional signals are represented by tensors, this structure can be exploited even further. In addition to an improved signal subspace estimate, the SLS algorithm is modified to directly exploit the tensor structure of the signal subspace obtained through the higher order SVD. The resulting algorithm which we term Tensor-Structure SLS offers a superior performance compared to existing approaches in critical cases, e.g., if there are highly correlated sources or a small number of available snapshots.

***Keywords**: Direction of arrival estimation, Multidimensional signal processing, Parameter estimation, Array signal processing*

1. INTRODUCTION

High resolution parameter estimation from R-dimensional signals is a task that is required for a variety of applications. Its studies have given rise to many efficient algorithms. A prominent class among these are ESPRIT-type algorithms, e.g., standard ESPRIT [11], Unitary ESPRIT [5] or its R-D extensions [6].

One of the key steps in all ESPRIT-type methods is the solution of the highly structured invariance equations. In addition to the simple Least Squares (LS) solution, extensions known as Total Least Squares (TLS, [3]) and Structured Least Squares (SLS, [4]) have been proposed. Since the latter takes the structure of the invariance equations into account, it is superior to LS and TLS methods.

In most approaches known to date, the underlying R-dimensional signals are stored in matrices. This representation does not account for the grid structure inherent in the data. A more natural approach is to use tensors for these signals which leads to Tensor-ESPRIT-type algorithms [10], e.g., the R-D Unitary Tensor-ESPRIT algorithm. In contrast to existing tensor approaches using PARAFAC [12] we focus on the direct analogy to the matrix case and use the corresponding extensions to the concepts of the SVD (i.e., the higher order SVD [1]) and the low-rank approximation [2].

In this contribution the tensor data model and extensions to ESPRIT-type algorithms for tensors are reviewed. We then extend the idea of SLS to tensors. First, it is shown how SLS can be expressed in the analogous tensor notation. Then we develop the Tensor-Structure Structured Least Squares (TS-SLS) algorithm and its R-D extension which both do not only exploit the structure of the invariance equations but additionally the particular structure in the signal subspace obtained through the higher order SVD. The performance of the algorithms is compared through computer simulations at the end of this paper.

The authors gratefully acknowledge the partial support of the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under contract no. HA 2239/1-1.

2. TENSOR AND MATRIX NOTATION

In order to facilitate the distinction between scalars, matrices, and tensors, the following representations are used: Scalars are denoted as italic letters ($a, b, \ldots, A, B, \ldots, \alpha, \beta, \ldots$), vectors as lower-case bold-face letters ($\boldsymbol{a}, \boldsymbol{b}, \ldots$), matrices as bold-face capitals ($\boldsymbol{A}, \boldsymbol{B}, \ldots$), and tensors are written as bold-face calligraphic letters ($\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}, \ldots$). We use the superscripts $^T, ^H, ^{-1}, ^+$ for transposition, Hermitian transposition, matrix inversion, and the Moore-Penrose pseudo inverse of matrices and * for complex conjugation, respectively. Moreover the Kronecker product between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is denoted by $\boldsymbol{A} \otimes \boldsymbol{B}$.

The tensor operations we use are consistent with [1]: **The scalar product** of two tensors $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ is symbolized by $\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle$ and computed by summing the element-wise product of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}^*$ over all indices. This definition allows us to define the higher-order norm of a tensor $\boldsymbol{\mathcal{A}}$ as $||\boldsymbol{\mathcal{A}}||_{\mathrm{H}} \doteq \sqrt{\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}} \rangle}$, similarly to the Frobenius norm of a matrix $\boldsymbol{A}$ denoted as $||\boldsymbol{A}||_{\mathrm{F}}$.

The n-mode product: The product of a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ and a matrix $\boldsymbol{U} \in \mathbb{C}^{J_n \times I_n}$ along the n-th mode is denoted as $\boldsymbol{\mathcal{A}} \times_n \boldsymbol{U} \in \mathbb{C}^{I_1 \times I_2 \ldots \times J_n \ldots \times I_N}$. It is obtained by multiplying all n-mode vectors of $\boldsymbol{\mathcal{A}}$ from the left-hand side by the matrix $\boldsymbol{U}$.

The HOSVD: The higher order SVD of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ is given by

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \ldots \times_N \boldsymbol{U}_N, \tag{1}$$

where $\boldsymbol{\mathcal{S}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ is the core-tensor which satisfies the all-orthogonality conditions [1] and $\boldsymbol{U}_n \in \mathbb{C}^{I_n \times I_n}$, $n = 1, 2, \ldots, N$ are the unitary matrices of n-mode singular vectors. The notation $[\boldsymbol{\mathcal{A}} \sqcup_n \boldsymbol{\mathcal{B}}]$ denotes the concatenation of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ along the n-th mode. Moreover, a matrix unfolding of the tensor $\boldsymbol{\mathcal{A}}$ along the n-th mode is symbolized by $\boldsymbol{A}_{(n)}$ or $[\boldsymbol{\mathcal{A}}]_{(n)}$ and can be understood as a matrix containing all the n-mode vectors of the tensor $\boldsymbol{\mathcal{A}}$. The order of the columns is chosen in accordance with [1].

In extension to [1] we define the vec-operator on a tensor $\boldsymbol{\mathcal{A}}$ as $\mathrm{vec}\{\boldsymbol{\mathcal{A}}\} = \boldsymbol{a}$, where the n-th component of $\boldsymbol{a}$ is given by $i_1 + I_1 \cdot (i_2 - 1) + I_2 \cdot I_1 \cdot (i_3 - 1) + \ldots + I_N \cdot \ldots \cdot I_2 \cdot I_1 \cdot (i_N - 1)$, $i_p = 1, 2, \ldots, I_p$, $p = 1, 2, \ldots, N$. Then there exist unique permutation matrices $\boldsymbol{P}^{(n)}$ that rearrange the elements in $\mathrm{vec}\left\{[\boldsymbol{\mathcal{A}}]_{(n)}\right\}$ such that they are equal to $\mathrm{vec}\{\boldsymbol{\mathcal{A}}\}$, i.e.,

$$\text{vec}\left\{\boldsymbol{\mathcal{A}}\right\} = \boldsymbol{P}^{(n)} \cdot \text{vec}\left\{\left[\boldsymbol{\mathcal{A}}\right]_{(n)}\right\} \ \forall \boldsymbol{\mathcal{A}}. \tag{2}$$

These permutation matrices represent a direct extension of the concept of commutation matrices known from the matrix case [9].

3. DATA MODEL

The methods presented in this paper are applicable to R-D harmonic retrieval problems, which can efficiently be formulated in terms of tensor equations, as first demonstrated in [10].

The measurement data is modeled as a superposition of d undamped exponentials from narrowband sources, sampled on an R-dimensional grid of $M = M_1 \times M_2 \ldots \times M_R$ sensors and observed at N subsequent time instants. The superposition of d undamped exponentials is described by

$$\begin{aligned} x_{m_1,m_2,\ldots,m_R,n} &= \sum_{i=1}^{d}\left(\prod_{r=1}^{R} e^{j\cdot(m_r-1)\cdot\mu_i^{(r)}}\right)\cdot s_i(n), \\ &\quad + n_{m_1,m_2,\ldots,m_R,n} \end{aligned} \tag{3}$$

with $m_r = 1, 2, \ldots, M_r$, spatial frequencies $\mu_i^{(r)}$, symbols $s_i(n)$ and noise samples $n_{m_1,m_2,\ldots,m_R,n}$ which are assumed to be i.i.d. zero mean circularly symmetric complex Gaussian random variables. In the classical matrix approach [6] equation (3) is converted into a matrix equation by stacking the dimensions into a measurement matrix $\boldsymbol{X} \in \mathbb{C}^{M\times N}$ which can then be modeled as

$$\boldsymbol{X} = \boldsymbol{A}\cdot\boldsymbol{S} + \boldsymbol{N}. \tag{4}$$

Here, $\boldsymbol{A} \in \mathbb{C}^{M\times d}$ represents the R-D array steering matrix, $\boldsymbol{S} \in \mathbb{C}^{d\times N}$ contains the complex symbols and the matrix $\boldsymbol{N}$ the noise samples. Since this stacking does not capture the lattice structure inherent in the measurement data, we define a measurement tensor $\boldsymbol{\mathcal{Y}} \in \mathbb{C}^{M_1\times M_2\ldots\times M_R\times N}$ which contains the samples from the measurement process and can be modeled through

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{A}} \times_{R+1} \boldsymbol{S}^T + \boldsymbol{\mathcal{N}}, \tag{5}$$

where $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{M_1\times M_2\ldots\times M_R\times d}$ now represents the array steering tensor and $\boldsymbol{\mathcal{N}} \in \mathbb{C}^{M_1\times M_2\ldots\times M_R\times d}$ the noise tensor. Note that $[\boldsymbol{\mathcal{A}}]_{(R+1)} = \boldsymbol{A}^T$ and $[\boldsymbol{\mathcal{Y}}]_{(R+1)} = \boldsymbol{X}^T$.

4. TENSOR SHIFT INVARIANCE EQUATIONS

The tensor data model allows us to express the shift invariance equations in tensor notation in the following fashion

$$\begin{aligned} \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Phi}^{(r)} &= \boldsymbol{\mathcal{A}} \times_r \boldsymbol{J}_2^{(r)}, \quad r = 1, 2, \ldots, R, \\ \boldsymbol{\Phi}^{(r)} &= \text{diag}\left\{\left[e^{j\mu_1^{(r)}}, \ \ldots, \ e^{j\mu_d^{(r)}}\right]\right\}, \end{aligned} \tag{6}$$

where $\mu_i^{(r)}$ denotes the spatial frequency of the i-th wavefront in the r-th mode and $\boldsymbol{J}_n^{(r)} \in \mathbb{R}^{M_r^{(\text{sel})}\times M_r}$, $n = 1, 2$ represent the selection matrices for the r-th mode, $r = 1, 2, \ldots, R$ that select $M_r^{(\text{sel})}$ out of M_r elements.

Now we introduce the "economy size" version of the HOSVD of the measurement tensor $\boldsymbol{\mathcal{Y}}$ given by

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{S}}^{[s]} \times_1 \boldsymbol{U}_1^{[s]} \times_2 \boldsymbol{U}_2^{[s]} \ldots \times_{R+1} \boldsymbol{U}_{R+1}^{[s]}, \tag{7}$$

where $\boldsymbol{U}_r^{[s]} \in \mathbb{C}^{M_r\times p_r}$, $r = 1, 2, \ldots, R$, $\boldsymbol{U}_{R+1}^{[s]} \in \mathbb{C}^{N\times d}$, $\boldsymbol{\mathcal{S}}^{[s]} \in \mathbb{C}^{p_1\times p_2\ldots\times p_R\times d}$, and $p_r = \min(M_r, d)$ for $N \geq d$ (if $N < d$ spatial smoothing has to be applied to increase N [10]). Equation (7) holds exactly in the absence of noise. For the noisy tensor (7) represents a low-rank approximation. It can be obtained by truncating the r-th mode of the core tensor to p_r elements and the singular vector matrices $\boldsymbol{U}_r$ to p_r columns for $r = 1, 2, \ldots, R$ (for $r = R+1$ truncate to d elements). However, in [2] it is shown that the best rank-$(p_1, p_2, \ldots, p_R, d)$ approximation in the least squares sense can only be computed through an iterative procedure. Nevertheless, the parameter estimation results obtained by the two methods are very similar as long as the SNR is not too low.

As an improvement of the basis for the estimated signal subspace $\boldsymbol{U}_s \in \mathbb{C}^{M\times d}$ known from the matrix approach [4] we define a tensor $\boldsymbol{\mathcal{U}}^{[s]} \in \mathbb{C}^{M_1\times M_2\ldots\times M_R\times d}$ through

$$\boldsymbol{\mathcal{U}}^{[s]} = \boldsymbol{\mathcal{S}}^{[s]} \times_1 \boldsymbol{U}_1^{[s]} \times_2 \boldsymbol{U}_2^{[s]} \ldots \times_R \boldsymbol{U}_R^{[s]}. \tag{8}$$

It can easily be shown that in the absence of noise $\boldsymbol{\mathcal{U}}^{[s]}$ and $\boldsymbol{\mathcal{A}}$ are related through a non-singular $d\times d$ transform matrix $\boldsymbol{T}$, such that

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}}^{[s]} \times_{R+1} \boldsymbol{T}, \tag{9}$$

which implies that the r-spaces of $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{U}}^{[s]}$ are equal for $r = 1, 2, \ldots, R$. With additive noise, (9) holds approximately. Using (9) in (6) we obtain the shift invariance equations in terms of the signal subspace given by

$$\boldsymbol{\mathcal{U}}^{[s]} \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Psi}^{(r)} \approx \boldsymbol{\mathcal{U}}^{[s]} \times_r \boldsymbol{J}_2^{(r)}, \quad r = 1, 2, \ldots, R, \tag{10}$$

where $\boldsymbol{\Psi}^{(r)}$ has the same eigenvalues as $\boldsymbol{\Phi}^{(r)}$. The LS solution to (10) is given by [10]

$$\begin{aligned} \boldsymbol{\Psi}^{(r)T} &= \left(\widetilde{\boldsymbol{J}}_1^{(r)} \cdot \left[\boldsymbol{\mathcal{U}}^{[s]}\right]_{(R+1)}^T\right)^+ \cdot \widetilde{\boldsymbol{J}}_2^{(r)} \cdot \left[\boldsymbol{\mathcal{U}}^{[s]}\right]_{(R+1)}^T \\ \widetilde{\boldsymbol{J}}_i^{(r)} &= \boldsymbol{I}_{\prod_{q=1}^{r-1} M_q} \otimes \boldsymbol{J}_i^{(r)} \otimes \boldsymbol{I}_{\prod_{q=r+1}^{R} M_q}, \quad i = 1, 2. \end{aligned} \tag{11}$$

Obviously, this solution is similar to the least squares solution in the matrix approach except for the fact that the estimated basis for the signal subspace $\boldsymbol{U}_s$ is replaced by $\left[\boldsymbol{\mathcal{U}}^{[s]}\right]_{(R+1)}^T$. We can therefore conceive the latter as an improved signal subspace estimate. The improvement results from the fact that in the tensor case we take the special structure of the R-dimensional lattice into account while computing a low-rank approximation based on the HOSVD of the measurement tensor. This allows us to "denoise" the measurements more efficiently.

The concepts of forward-backward averaging and the mapping of centro-Hermitian matrices to real-valued matrices [8] can also be extended to tensors [10]. This leads to the real-valued shift invariance equations given by

$$\begin{aligned} &\boldsymbol{\mathcal{E}}^{[s]} \times_r \boldsymbol{K}_1^{(r)} \times_{R+1} \boldsymbol{\Upsilon}^{(r)} = \boldsymbol{\mathcal{E}}^{[s]} \times_r \boldsymbol{K}_2^{(r)}, \text{ where} \\ &\boldsymbol{K}_1^{(r)} = 2\cdot\text{Re}\left\{\boldsymbol{Q}_{M_r^{(\text{sel})}}^H \cdot \boldsymbol{J}_2^{(r)} \cdot \boldsymbol{Q}_{M_r}\right\} \text{ and} \\ &\boldsymbol{K}_2^{(r)} = 2\cdot\text{Im}\left\{\boldsymbol{Q}_{M_r^{(\text{sel})}}^H \cdot \boldsymbol{J}_2^{(r)} \cdot \boldsymbol{Q}_{M_r}\right\}, \ r = 1, 2, \ldots, R. \end{aligned} \tag{12}$$

Here, $\boldsymbol{Q}_p$ are the well-known $p\times p$ left-$\boldsymbol{\Pi}$-real matrices which satisfy $\boldsymbol{\Pi}\cdot\boldsymbol{Q}_p^* = \boldsymbol{Q}_p$ and the eigenvalues $\lambda_i^{(r)}$ of the matrices $\boldsymbol{\Upsilon}^{(r)}$ are related to the spatial frequencies through $\mu_i^{(r)} = 2\cdot\arctan(\lambda_i^{(r)})$. Moreover, $\boldsymbol{\mathcal{E}}^{[s]}$ is the multidimensional extension of the real-valued basis for the signal subspace $\boldsymbol{E}_s$ from the matrix approach. It is obtained through the HOSVD of the transformed measurement tensor $\varphi(\boldsymbol{\mathcal{Z}})$ in the same manner as $\boldsymbol{\mathcal{U}}^{[s]}$ is obtained from $\boldsymbol{\mathcal{Y}}$, where $\varphi(\boldsymbol{\mathcal{Z}})$ is given by

$$\begin{aligned} \varphi(\boldsymbol{\mathcal{Z}}) &= \boldsymbol{\mathcal{Z}} \times_1 \boldsymbol{Q}_{M_1}^H \times_2 \boldsymbol{Q}_{M_2}^H \ldots \times_R \boldsymbol{Q}_{M_R}^H \times_{R+1} \boldsymbol{Q}_{2N}^H, \text{ and} \\ \boldsymbol{\mathcal{Z}} &= \left[\boldsymbol{\mathcal{Y}} \sqcup_{R+1} \boldsymbol{\mathcal{Y}}^* \times_1 \boldsymbol{\Pi}_{M_1} \times_2 \boldsymbol{\Pi}_{M_2} \ldots \times_{R+1} \boldsymbol{\Pi}_N\right]. \end{aligned}$$

5. TENSOR-STRUCTURE SLS

The structured least squares algorithm developed in [4] represents an efficient way to solve the invariance equations since their inherent structure is explicitly taken into account. In this section we use the complex-valued shift invariance equations from (10). However, the same methods apply to the solutions of (12) as well.

The key idea of SLS is to take into account the fact that the signal subspace is not known without error. Therefore, a perturbation term $\Delta \boldsymbol{U}_s$ for the matrix $\boldsymbol{U}_s$ is modeled and the least squares solution to the invariance equations with respect to the unknown matrix $\boldsymbol{\Psi}^{(r)}$ and the perturbation term $\Delta \boldsymbol{U}_s$ is computed. This concept can directly be extended to (10) by defining a tensor-perturbation term for $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ which results in the following optimization problem

$$\min_{\boldsymbol{\Psi}^{(r)}, \Delta\boldsymbol{\mathcal{U}}^{[s]}} \left\{ \left|\left| \boldsymbol{\mathcal{R}}^{(r)} \right|\right|_{\mathrm{H}}^2 + \kappa^2 \cdot \left|\left| \Delta\boldsymbol{\mathcal{U}}^{[s]} \right|\right|_{\mathrm{H}}^2 \right\} \tag{13}$$

$$\boldsymbol{\mathcal{R}}^{(r)} = \left(\boldsymbol{\mathcal{U}}^{[s]} + \Delta\boldsymbol{\mathcal{U}}^{[s]}\right) \times_r \boldsymbol{J}_1^{(r)} \times_{R+1} \boldsymbol{\Psi}^{(r)} - \left(\boldsymbol{\mathcal{U}}^{[s]} + \Delta\boldsymbol{\mathcal{U}}^{[s]}\right) \times_r \boldsymbol{J}_2^{(r)},$$

where $\boldsymbol{\mathcal{R}}^{(r)}$ is termed the residual tensor, and the second term in the cost function represents a regularization term.

It turns out that the solution to this tensor-valued SLS problem results in an algorithm performing essentially the same computations as the corresponding matrix SLS algorithm. As in the tensor formulation for LS [10] the only difference is that from the tensor approach we have an improved signal subspace estimate given by $\left[\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}\right]_{(R+1)}^T$. If we use this matrix in the matrix SLS algorithm we obtain the same result as by using $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ in the tensor version. In other words, (13) represents an equivalent formulation for the SLS problem using tensors.

In contrast to (13), the TS-SLS algorithm additionally exploits the structure of the signal subspace obtained through the HOSVD. The key idea of this method is that instead of modeling one perturbation term for the entire tensor $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ we model a perturbation term for each of the components it was computed from. Although the algorithm can be developed for R dimensions, for simplicity of presentation we assume $R = 2$ and focus on the first shift invariance equation ($r = 1$). Moreover, the superscript $^{[\mathrm{s}]}$ is dropped, so that $\boldsymbol{\mathcal{U}}^{[\mathrm{s}]}$ becomes $\boldsymbol{\mathcal{U}}$, $\boldsymbol{\mathcal{S}}^{[\mathrm{s}]}$ becomes $\boldsymbol{\mathcal{S}}$, and $\boldsymbol{U}_r^{[\mathrm{s}]}$ becomes $\boldsymbol{U}_r$. Using these simplifications we get $\boldsymbol{\mathcal{U}} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2$ which leads to three perturbation terms: $\Delta\boldsymbol{\mathcal{S}}$, $\Delta\boldsymbol{U}_1$, and $\Delta\boldsymbol{U}_2$. The optimization problem can then be formulated in the following fashion

$$\begin{aligned}\min_{\boldsymbol{\Psi}^{(1)}, \Delta\boldsymbol{\mathcal{S}}, \Delta\boldsymbol{U}_1, \Delta\boldsymbol{U}_2} & \left\{ \left|\left| \boldsymbol{\mathcal{R}}^{(1)} \right|\right|_{\mathrm{H}}^2 + \kappa_1^2 \left|\left|\Delta\boldsymbol{U}_1\right|\right|_{\mathrm{F}}^2 \right. \\ & \left. + \kappa_2^2 \left|\left|\Delta\boldsymbol{U}_2\right|\right|_{\mathrm{F}}^2 + \kappa_3^2 \left|\left|\Delta\boldsymbol{\mathcal{S}}\right|\right|_{\mathrm{H}}^2 \right\}, \end{aligned} \tag{14}$$

$$\begin{aligned}\boldsymbol{\mathcal{R}}^{(1)} = {} & (\boldsymbol{\mathcal{S}} + \Delta\boldsymbol{\mathcal{S}}) \times_1 \left(\boldsymbol{J}_1^{(1)} \left[\boldsymbol{U}_1 + \Delta\boldsymbol{U}_1\right]\right) \times_2 \left[\boldsymbol{U}_2 + \Delta\boldsymbol{U}_2\right] \times_3 \boldsymbol{\Psi}^{(1)} \\ & - (\boldsymbol{\mathcal{S}} + \Delta\boldsymbol{\mathcal{S}}) \times_1 \left(\boldsymbol{J}_2^{(1)} \left[\boldsymbol{U}_1 + \Delta\boldsymbol{U}_1\right]\right) \times_2 \left[\boldsymbol{U}_2 + \Delta\boldsymbol{U}_2\right], \end{aligned}$$

where $\boldsymbol{\mathcal{R}}^{(1)}$ is the residual tensor and we now have three regularization terms, one for each perturbation term we defined. The weighting factors κ_r are chosen in a manner similar to [4].

This nonlinear optimization problem can be solved by local linearization in an iterative fashion. As a starting point we use the LS solution for the matrix $\boldsymbol{\Psi}^{(1)}$ and zeros for all the perturbation terms. The update equations in the k-th step are now given by

$$\begin{aligned} \boldsymbol{\Psi}_{k+1}^{(1)} &= \boldsymbol{\Psi}_k^{(1)} + \Delta\boldsymbol{\Psi}_k^{(1)} \\ \Delta\boldsymbol{\mathcal{S}}_{k+1} &= \Delta\boldsymbol{\mathcal{S}}_k + \Delta\Delta\boldsymbol{\mathcal{S}}_k \\ \Delta\boldsymbol{U}_{1,k+1} &= \Delta\boldsymbol{U}_{1,k} + \Delta\Delta\boldsymbol{U}_{1,k} \\ \Delta\boldsymbol{U}_{2,k+1} &= \Delta\boldsymbol{U}_{2,k} + \Delta\Delta\boldsymbol{U}_{2,k} \end{aligned}$$

$$\begin{bmatrix} \mathrm{vec}\left\{\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix} = -\boldsymbol{F}^{+} \cdot \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k^{(1)}\right\} \\ \kappa_1 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{1,k}\right\} \\ \kappa_2 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{2,k}\right\} \\ \kappa_3 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix}$$

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1^{(1)} & \boldsymbol{F}_2^{(1)} & \boldsymbol{F}_3^{(1)} & \boldsymbol{F}_4^{(1)} \\ \boldsymbol{0} & \kappa_1 \cdot \boldsymbol{I}_{M_1 \cdot p_1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \kappa_2 \cdot \boldsymbol{I}_{M_2 \cdot p_2} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \kappa_3 \cdot \boldsymbol{I}_{p_1 \cdot p_2 \cdot d} \end{bmatrix}$$

$$\boldsymbol{F}_1^{(1)} = \boldsymbol{P}^{(3)} \cdot \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{J}_1^{(1)} \boldsymbol{U}_{1,k} \times_2 \boldsymbol{U}_{2,k} \right]_{(3)}^T \otimes \boldsymbol{I}_d \right)$$

$$\begin{aligned}\boldsymbol{F}_2^{(1)} = \boldsymbol{P}^{(1)} \cdot \Bigg(& \left(\left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(1)} \right]_{(1)}^T \otimes \boldsymbol{J}_1^{(1)} \right) \\ & - \left(\left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{U}_{2,k} \right]_{(1)}^T \otimes \boldsymbol{J}_2^{(1)} \right) \Bigg) \end{aligned}$$

$$\begin{aligned}\boldsymbol{F}_3^{(1)} = \boldsymbol{P}^{(2)} \cdot \Bigg(& \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{J}_1^{(1)} \boldsymbol{U}_{1,k} \times_3 \boldsymbol{\Psi}_k^{(1)} \right]_{(2)}^T \otimes \boldsymbol{I}_{M_2} \right) \\ & - \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{J}_2^{(1)} \boldsymbol{U}_{1,k} \right]_{(2)}^T \otimes \boldsymbol{I}_{M_2} \right) \Bigg) \end{aligned}$$

$$\boldsymbol{F}_4^{(1)} = \boldsymbol{\Psi}_k^{(1)} \otimes \boldsymbol{U}_{2,k} \otimes \boldsymbol{J}_1^{(1)} \boldsymbol{U}_{1,k} - \boldsymbol{I}_d \otimes \boldsymbol{U}_{2,k} \otimes \boldsymbol{J}_2^{(1)} \boldsymbol{U}_{1,k}$$

$$\boldsymbol{\mathcal{S}}_k = \boldsymbol{\mathcal{S}} + \Delta\boldsymbol{\mathcal{S}}_k, \quad \boldsymbol{U}_{n,k} = \boldsymbol{U}_n + \Delta\boldsymbol{U}_{n,k},\ n = 1, 2,$$

i.e., in each iteration step we compute updates to the perturbation terms as well as the unknown matrix $\boldsymbol{\Psi}^{(1)}$ by multiplying the pseudo-inverse of the matrix $\boldsymbol{F}$ with a vector computed from the previous perturbation terms and the current residual tensor (for an efficient implementation, the QR decomposition can be used). These updates are then applied to the optimization variables for the next iteration step. In the update equations $\boldsymbol{P}^{(n)},\ n = 1, 2, 3$ refers to the permutation matrices introduced in Section 2. For $r = 2$, the first and second mode are consistently exchanged

$$\boldsymbol{F}_1^{(2)} = \boldsymbol{P}^{(3)} \cdot \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{U}_{1,k} \times_2 \boldsymbol{J}_1^{(2)} \boldsymbol{U}_{2,k} \right]_{(3)}^T \otimes \boldsymbol{I}_d \right)$$

$$\begin{aligned}\boldsymbol{F}_2^{(2)} = \boldsymbol{P}^{(1)} \cdot \Bigg(& \left(\left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{J}_1^{(2)} \boldsymbol{U}_{2,k} \times_3 \boldsymbol{\Psi}_k^{(2)} \right]_{(1)}^T \otimes \boldsymbol{I}_{M_2} \right) \\ & - \left(\left[\boldsymbol{\mathcal{S}}_k \times_2 \boldsymbol{J}_2^{(2)} \boldsymbol{U}_{2,k} \right]_{(1)}^T \otimes \boldsymbol{I}_{M_2} \right) \Bigg) \end{aligned}$$

$$\begin{aligned}\boldsymbol{F}_3^{(2)} = \boldsymbol{P}^{(2)} \cdot \Bigg(& \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{U}_{1,k} \times_3 \boldsymbol{\Psi}_k^{(2)} \right]_{(2)}^T \otimes \boldsymbol{J}_1^{(2)} \right) \\ & - \left(\left[\boldsymbol{\mathcal{S}}_k \times_1 \boldsymbol{U}_{1,k} \right]_{(2)}^T \otimes \boldsymbol{J}_2^{(2)} \right) \Bigg) \end{aligned}$$

$$\boldsymbol{F}_4^{(2)} = \boldsymbol{\Psi}_k^{(2)} \otimes \boldsymbol{J}_1^{(2)} \boldsymbol{U}_{2,k} \otimes \boldsymbol{U}_{1,k} - \boldsymbol{I}_d \otimes \boldsymbol{J}_2^{(2)} \boldsymbol{U}_{2,k} \otimes \boldsymbol{U}_{1,k}.$$

Similarly to the matrix SLS algorithm, TS-SLS can be modified to solve the R shift invariance equations jointly which leads to the TS-RD-SLS algorithm. In this case we model the perturbation terms to be equal for all the shift invariance equations. The optimization variables are therefore the perturbation terms for the core tensor and the singular vector matrices $\boldsymbol{U}_r,\ r = 1, 2, \ldots, R$ and additionally the matrices $\boldsymbol{\Psi}^{(r)},\ r = 1, 2, \ldots, R$. The cost function for the TS-RD-SLS algorithm is given by

$$\sum_{r=1}^{R} \left|\left| \boldsymbol{\mathcal{R}}^{(r)} \right|\right|_{\mathrm{H}}^2 + \sum_{r=1}^{R} \kappa_r^2 \left|\left|\Delta\boldsymbol{U}_r\right|\right|_{\mathrm{F}}^2 + \kappa_{R+1}^2 \left|\left|\Delta\boldsymbol{\mathcal{S}}\right|\right|_{\mathrm{H}}^2, \tag{15}$$

where $\boldsymbol{\mathcal{R}}^{(r)}$ is the residual tensor obtained by subtracting the right-hand side of the r-th shift invariance equation from its left-hand side. The iterative solution is very similar to the 1-D case. The extended update equation takes the following form ($R = 2$ taken as an example)

$$\begin{bmatrix} \mathrm{vec}\left\{\Delta\boldsymbol{\Psi}_k^{(1)}\right\} \\ \mathrm{vec}\left\{\Delta\boldsymbol{\Psi}_k^{(2)}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{1,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{U}_{2,k}\right\} \\ \mathrm{vec}\left\{\Delta\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix} = -\boldsymbol{F}^{+} \cdot \begin{bmatrix} \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k^{(1)}\right\} \\ \mathrm{vec}\left\{\boldsymbol{\mathcal{R}}_k^{(2)}\right\} \\ \kappa_1 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{1,k}\right\} \\ \kappa_2 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{U}_{2,k}\right\} \\ \kappa_3 \cdot \mathrm{vec}\left\{\Delta\boldsymbol{\mathcal{S}}_k\right\} \end{bmatrix}$$

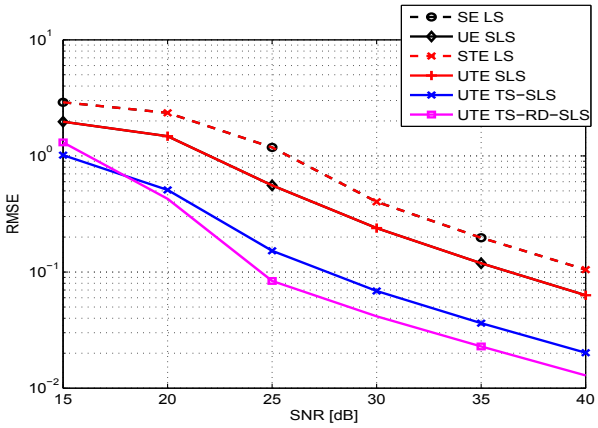

Fig. 1. RMSE vs. SNR for $d = 3$ correlated sources on a 3×3 URA, $N = 10$, $\boldsymbol{\mu}_1 = [1, -1]^T$, $\boldsymbol{\mu}_2 = [0, 1]^T$, $\boldsymbol{\mu}_3 = [-1, 0]^T$.

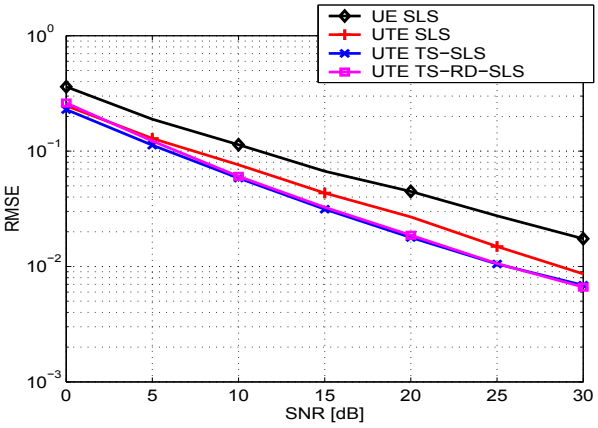

Fig. 2. RMSE vs. SNR for $d = 2$ sources on a 5×7 URA, single snapshot ($N = 1$), $\boldsymbol{\mu}_1 = [1, -1]^T$, $\boldsymbol{\mu}_2 = [0, 1]^T$.

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1^{(1)} & \mathbf{0} & \boldsymbol{F}_2^{(1)} & \boldsymbol{F}_3^{(1)} & \boldsymbol{F}_4^{(1)} \\ \mathbf{0} & \boldsymbol{F}_1^{(2)} & \boldsymbol{F}_2^{(2)} & \boldsymbol{F}_3^{(2)} & \boldsymbol{F}_4^{(2)} \\ \mathbf{0} & \mathbf{0} & \kappa_1 \cdot \boldsymbol{I}_{M_1 \cdot p_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \kappa_2 \cdot \boldsymbol{I}_{M_2 \cdot p_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \kappa_3 \cdot \boldsymbol{I}_{p_1 \cdot p_2 \cdot d} \end{bmatrix}.$$

After solving the shift invariance equations, the eigenvalues of $\boldsymbol{\Psi}^{(r)}$ can be computed jointly in order to achieve the correct pairing of the spatial frequencies to the sources. This can be achieved by a joint diagonalization scheme, e.g., the Simultaneous Schur Decomposition [6].

6. SIMULATION RESULTS

In this section, the performance of the algorithms is compared through computer simulations. In the first simulation in Figure 1, $d = 3$ sources with complex Gaussian distributed amplitudes and a strong correlation of $\rho = 0.9999$ were captured by a 3×3 uniform rectangular array collecting $N = 10$ subsequent snapshots for various values of the SNR at the receiver. The top two curves show standard ESPRIT (SE) and its tensor version (STE) with LS (see [10]). There is no improvement since for $d \geq \max\{M_1, M_2\}$ the improved signal subspace estimate is the same as in the matrix approach [10]. This is also the case for the comparison of Unitary ESPRIT (UE) with its tensor version (UTE), both with SLS. The two curves below show the RMS estimation error of UTE with TS-SLS and TS-RD-SLS, respectively, which provide a significant gain.

For the simulation shown in Figure 2, a 5×7 URA with $N = 1$ snapshot was used for $d = 2$ sources emitting BPSK symbols. Here we can observe the effect of the improved signal subspace estimate in going from UE to UTE and the additional gain in using TS-SLS or TS-RD-SLS.

In general, TS-SLS and TS-RD-SLS are beneficial in critical scenarios, e.g., if the number of snapshots is small and/or if the sources are highly correlated.

7. CONCLUSIONS

In this contribution, a novel approach to solve the shift invariance equations of ESPRIT-type algorithms in tensor notation is discussed. In addition to taking into account the structure of the shift invariance equations, the TS-SLS algorithm exploits the special structure of the signal subspace obtained through the HOSVD. It therefore provides gains in addition to the ones obtained by the tensor signal subspace estimation step.

TS-SLS can readily be generalized to TS-RD-SLS which solves the R shift invariance equations jointly. A version of TS-RD-SLS incorporating the modifications to SLS discussed in [7] was also implemented but in our simulations it showed no advantage over TS-RD-SLS.

TS-SLS provides an improved performance especially in critical cases, e.g., for highly correlated sources or in cases where the number of snapshots is small. This is still true in scenarios where the improved signal subspace estimate is the same as the subspace estimate from the matrix approach, i.e., if the number of wavefronts exceeds the number of sensors in all the modes. This behavior was also demonstrated through computer simulations.

REFERENCES

[1] L. de Lathauwer, B. de Moor, and J. Vanderwalle. A multilinear singular value decomposition. *Siam J. Matrix Anal. Appl.*, 21(4), 2000.

[2] L. de Lathauwer, B. de Moor, and J. Vanderwalle. On the best rank-1 and rank-$(r_1, r_2, \ldots, r_n)$ approximation of higher-order tensors. *Siam J. Matrix Anal. Appl.*, 21(4), 2000.

[3] G. H. Golub and C. F. Van Loan. An analysis of the total least squares problem. *Siam J. Numer. Anal.*, 17:883–893, 1980.

[4] M. Haardt. Structured least squares to improve the performance of ESPRIT-type algorithms. *IEEE Transactions on Signal Processing*, 45(3):792–799, March 1997.

[5] M. Haardt and J. A. Nossek. Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden. *IEEE Trans. Signal Processing*, 43:1232–1242, 1995.

[6] M. Haardt and J. A. Nossek. Simultaneous schur decomposition of several non-symmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems. *IEEE Trans. Signal Processing*, 46:161–169, January 1998.

[7] M.R.J.A.E. Kwakkernaat, Y.L.C. Jong, J.C. Bultitude, and M.H.A.J. Herben. Improved structured least squares for the application of Unitary ESPRIT to cross arrays. *IEEE Signal Processing Letters*, 13(6):349–352, June 2006.

[8] A. Lee. Centrohermitian and skew-centrohermitian matrices. *Linear algebra and its applications*, 29:205–210, 1980.

[9] J. R. Magnus and H. Neudecker. *Matrix differential calculus with applications in statistics and econometrics*. John Wiley and Sons, 1995.

[10] F. Roemer, M. Haardt, and G. Del Galdo. Higher order SVD based subspace estimation to improve multi-dimensional parameter estimation algorithms. In *Proc. 40th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, November 2006.

[11] R. Roy, A. Paulraj, and T. Kailath. ESPRIT - a subspace rotation approach to estimation of parameters of cisoids in noise. *IEEE Trans. Acoust., Speech, Signal Process.*, AASP-34:1340–1342, October 1986.

[12] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis. Parallel factor analysis in sensor array processing. *IEEE Transactions on Signal Processing*, 48(8):2377–2388, August 2000.