Enhanced Solutions for the Block-Term Decomposition in Rank-$(L_r, L_r, 1)$ Terms

Liana Khamidullina, *Student Member, IEEE*, Gabriela Seidl, Ivan Alexeevich Podkurkov, Alexey Alexandrovich Korobkov, and Martin Haardt, *Fellow, IEEE*

***Abstract*—The block-term decompositions (BTD) represent tensors as a linear combination of low multilinear rank terms and can be explicitly related to the Canonical Polyadic decomposition (CPD). In this paper, we introduce the SECSI-BTD framework, which exploits the connection between two decompositions to estimate the block-terms of the rank-$(L_r, L_r, 1)$ BTD. The proposed SECSI-BTD algorithm includes the initial calculation of the factor estimates using the SEmi-algebraic framework for approximate Canonical polyadic decompositions via SImultaneous Matrix Diagonalizations (SECSI), followed by clustering and refinement procedures that return the appropriate rank-$(L_r, L_r, 1)$ BTD terms. Moreover, we introduce a new approach to estimate the multilinear rank structure of the tensor based on the HOSVD and k-means clustering. Since the proposed SECSI-BTD algorithm does not require a known rank structure but can still take advantage of the known ranks when available, it is more flexible than the existing techniques in the literature. Additionally, our algorithm does not require multiple initializations, and the simulation results show that it provides more accurate results and a better convergence behavior for an extensive range of SNRs.**

***Index Terms*—Block-term decomposition, tensor, SECSI, Canonical Polyadic, k-means, model order.**

I. Introduction

OWING to multidimensionality-related benefits, tensor-based techniques have become a primary tool for many signal processing applications. Compared to classical matrix factorizations, tensor decompositions possess more relaxed uniqueness conditions and preserve the data's original, often inherently multidimensional structure. This enables modeling a signal across multiple domains and facilitates the interpretation of results. Additionally, a variety of tensor factorizations allow choosing a technique that best suits a given task or data.

Manuscript received 21 July 2022; revised 22 February 2023; accepted 4 June 2023. Date of publication 3 July 2023; date of current version 20 July 2023. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yue M. Lu. This work was supported in part by the German Research Foundation (DFG) under Grant 402834619 (AdAMMM, HA 2239/14-1, and ZH 640/2-1), and in part by the Open Access Publication Fund of the Technische Universität Ilmenau. *(Corresponding author: Liana Khamidullina.)*

Liana Khamidullina, Gabriela Seidl, and Martin Haardt are with the Communications Research Laboratory, Ilmenau University of Technology, 98693 Ilmenau, Germany (e-mail: liana.khamidullina@tu-ilmenau.de; seidl.gabriel@googlemail.com; martin.haardt@tu-ilmenau.de).

Ivan Alexeevich Podkurkov and Alexey Alexandrovich Korobkov are with the Department of Radioelectronic and Telecommunications Systems, Kazan National Research Technical University named after A. N. Tupolev, 420111 Kazan, Russia (e-mail: podkiva@mail.ru; korobkov@inbox.ru).

Digital Object Identifier 10.1109/TSP.2023.3289730

Probably the most prominent and extensively used tensor decompositions are the Higher-Order Singular Value Decomposition (HOSVD) [1], [2] and the Canonical Polyadic Decomposition (CPD), sometimes also referred to as CANDECOMP or PARAFAC [3], [4], [5]. They have found their applications in a wide range of fields, including statistics, communications, localization, biomedical signal processing, source separation, and many others [5], [6]. Furthermore, both decompositions have several variations and extensions, and the block-term decomposition (BTD) can be considered one of them. Depending on the ranks of sub-blocks in the BTD, several types of this decomposition have been distinguished in the literature, for example, decomposition in rank-$(L, L, 1)$ terms, in rank-(L, M, N) terms, or in rank-$(L, M, \cdot)$ terms [7], [8]. In this article, our focus falls on the block-term decomposition in rank-$(L_r, L_r, 1)$ terms, which has gained increasing attention from researchers in the last decade.

Whereas CPD techniques are quite well understood in the literature (there are elegant uniqueness theorems, stable and powerful tools for their computation and model order estimation), block-term decompositions are still a subject of active research since they pose a more complex problem due to inherent ambiguities and weaker uniqueness properties. Interestingly, exactly these properties render the block-term decompositions attractive for some applications since their uniqueness conditions are less strict, and they can be applied under more general circumstances than the CPD. However, state-of-the-art BTD frameworks as, for example, Tensorlab [9], do not always provide a stable decomposition performance, implying that they are prone to erroneous estimation of the factors for a great range of possible model parameters (these can be traced back to suboptimal initialization of the BTD-algorithm). An additional challenge that arises before computing the BTD is the estimation of the number of BTD terms and the multilinear ranks. For instance, in Tensorlab solutions, the number of rank-$(L_r, L_r, 1)$ terms as well as the L_rs have to be known beforehand, which might not be the case for some applications.

Recently, block-term decompositions have received a lot of attention in different research areas. The authors in [7], [8], [10], [11], [12], [13] introduce the definitions, link to CPD, and the uniqueness conditions for different types of block-term decompositions as well as optimization-based and algebraic algorithms to estimate the block-term factors assuming that the rank structure is known beforehand. On the other hand, the authors in [14], [15], [16], [17], [18] present the hierarchical

iteratively reweighted least squares (HIRLS) and the alternating group lasso (AGL) algorithms that estimate both the ranks and the factors of rank-$(L_r, L_r, 1)$ BTDs. However, they do not exploit prior knowledge of ranks when available and require multiple initializations to ensure convergence. Another recent article on the multilinear rank decomposition investigates the conditions under which the decomposition in rank-$(L_r, L_r, 1)$ terms is unique and can be computed via an eigenvalue decomposition [19]. The BTD algorithm based on the group sparsity property of the loading matrices introduced in [20] also allows performing the decomposition and the model order estimation. Still, it is limited to a rank-$(L, L, 1)$ BTD with equal multilinear ranks.

Other research directions, which further exploit the BTD, include a wide variety of different applications and the extension to coupled decompositions. For instance, the uniqueness conditions and the algorithms for coupled CPD and BTD in rank-$(L_{r,n}, L_{r,n}, 1)$ terms are discussed in [21], [22], and structured data fusion by means of coupled tensor decompositions is presented in [23]. The authors in [24] use a coupled block simultaneous generalized Schur decomposition to calculate the coupled rank-$(L_m, L_n, \cdot)$ BTD. In contrast to the current work, in [25], we propose an algorithm to calculate the coupled rank-$(L_r, L_r, 1)$ BTD of multiple tensors with a common mode which often occurs in biomedical data applications [26]. Moreover, several authors have presented many other practical applications of the BTD, which include communications signal processing, radar, and image and graph analysis [27], [28], [29], [30], [31], [32], [33], [34]. The nonnegative rank-$(L_r, L_r, 1)$ and coupled rank-$(L_r, L_r, 1)$ decompositions with application to hyperspectral imagery and cartography are investigated in [35], [36], [37]. The authors in [38] propose an algorithm to compute the rank-(L, M, N) BTD of large streaming tensor datasets. Another way of looking at the CPD and BTD is discussed in [39], [40]. The authors view these decompositions as a special case of the Tucker decomposition and introduce the Krylov-Levenberg-Marquardt algorithm to compute it. Compared to the rank-$(L_r, L_r, 1)$ BTD studied in this article, the BTD in [39] is considered as a sum of Tucker tensors with a block diagonal core tensor which in turn requires prior knowledge of the block-structure (multilinear ranks).

Contributions: This article proposes a new approach to calculate the block-term decomposition in rank-$(L_r, L_r, 1)$ terms based on the SEmi-algebraic framework for approximate Canonical polyadic decompositions via SImultaneous Matrix Diagonalizations (SECSI) [41], which shows enhanced numerical stability even for low SNR scenarios. In contrast to the schemes in [14], [15], [16], [18], our algorithm does not require multiple initializations to ensure convergence. Moreover, compared to the algorithm in [9], the proposed approach can perform the decomposition even with an unknown rank structure. To this end, we introduce an extension of the LineAr Regression of Global Eigenvalues (LaRGE) scheme, originally designed for for the estimation of the CPD rank of a noise-corrupted low-rank tensor, to estimate the number of L_r terms. The article also shows how the L_rs can be estimated through k-means clustering. Compared to our work in [25], this contribution explores a general

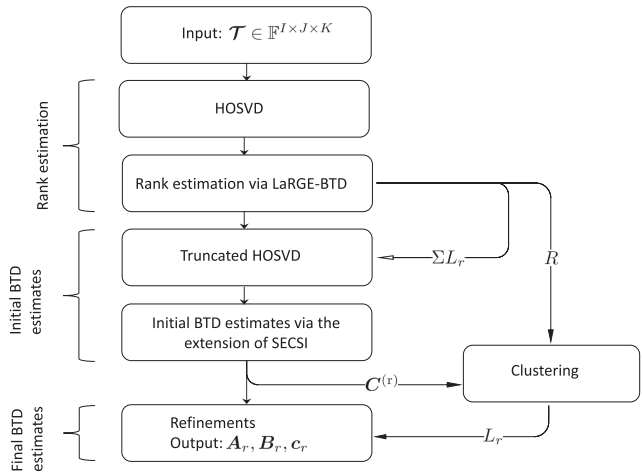


Fig. 1. Block diagram of the SECSI-BTD framework.

rank-$(L_r, L_r, 1)$ decomposition of a single three-dimensional tensor with possibly unknown block- and multilinear ranks. We provide a detailed description of the proposed approach, including the CPD to BTD transformation, the estimation of ranks, and the calculation of the BTD factors. Moreover, we conduct extensive and thorough synthetic data simulations to validate the proposed algorithm and compare it to the schemes from the literature. We refer to the proposed rank-$(L_r, L_r, 1)$ BTD framework as SECSI-BTD. It should be mentioned that it does not require a known rank structure but can still exploit prior knowledge of the model order when available. Altogether, the SECSI-BTD framework to compute an approximate BTD from noise-corrupted measurements is composed of three main blocks (Fig. 1):

- The model order estimation block based on the LineAr Regression of Global Eigenvalues (LaRGE) scheme to estimate the number of block-terms and the sum of multilinear ranks (can be skipped if the ranks are known beforehand);
- The computation of the initial estimates using the BTD extension of the SECSI framework and clustering, which includes the calculation of the initial rank-$(L_r, L_r, 1)$ decomposition factors via simultaneous matrix diagonalizations and clustering via the k-means algorithm;
- The refinement procedures that bring the initial estimates to the BTD form and return the final estimates of rank-$(L_r, L_r, 1)$ terms by employing ALS or NLS iterations [8], [13].

For notational simplicity, hereafter, by writing "BTD" we refer to BTD in rank-$(L_r, L_r, 1)$ terms.

Notation: Matrices and vectors are denoted by upper-case ($\boldsymbol{A}$) and lower-case ($\boldsymbol{a}$) bold-faced letters, respectively. Bold-faced calligraphic letters denote tensors ($\boldsymbol{\mathcal{A}}$). The superscripts $\{\cdot\}^{\mathrm{T}}$, $\{\cdot\}^{\mathrm{H}}$, $\{\cdot\}^{-1}$, and $\{\cdot\}^{+}$ denote the transpose, Hermitian transpose, matrix inverse, and Moore–Penrose pseudoinverse, respectively. Moreover, the mth column and the lth row of $\boldsymbol{A}$ is denoted as $\boldsymbol{A}_{(:,m)}$ and $\boldsymbol{A}_{(l,:)}$, respectively. The operator $\mathcal{D}\{\cdot\}$ denotes the construction of a diagonal matrix with diagonal elements being the entries of the input vector. We use $\otimes$, $\diamond$, and $\circ$ to denote the Kronecker, Khatri-Rao (column-wise Kronecker), and outer

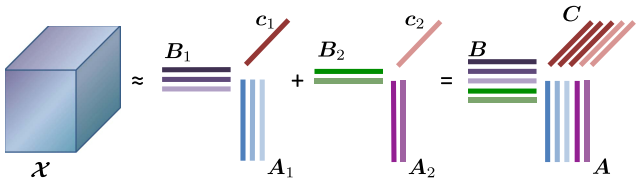


Fig. 2. Rank-$(L_r, L_r, 1)$ BTD with $L_1 = 3$, $L_2 = 2$, and $R = 2$ as a CPD with repeated columns in the 3-mode (matrix $\boldsymbol{C}$).

products, respectively. The n-mode unfolding of the tensor $\boldsymbol{\mathcal{A}}$ is denoted as $[\boldsymbol{\mathcal{A}}]_{(n)}$ (we use the reverse cyclical ordering of the columns [1]), and the n-mode product between a tensor $\boldsymbol{\mathcal{A}}$ and a matrix $\boldsymbol{A}$ is denoted as $\boldsymbol{\mathcal{A}} \times_n \boldsymbol{A}$ [1]. The operator $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. Additionally, we denote the Frobenius norm of a tensor $\boldsymbol{\mathcal{A}}$ by $\|\boldsymbol{\mathcal{A}}\|_{\mathrm{F}}$ defined as the square root of the sum of the squares of all its elements, or equally, as the Frobenius norm of an arbitrary n-mode unfolding of the tensor [5]. An identity matrix of size $d \times d$ is denoted as $\boldsymbol{I}_d$, whereas $\boldsymbol{\mathcal{I}}_{d,R}$ denotes a super-diagonal d-dimensional tensor of size $R \times R \times \cdots \times R$ with elements equal to one if all d indices are equal and zeros otherwise. A column vector of ones of size R is denoted as $\mathbf{1}_R$. The field of real ($\mathbb{R}$) or complex ($\mathbb{C}$) numbers is represented by $\mathbb{F}$. To ease the notation, we use ΣL_r to denote $\sum_{r=1}^{R} L_r$.

This article is organized as follows. Section II provides some preliminaries on the BTD. In Section III, we present an algorithm to estimate the model order of the rank-$(L_r, L_r, 1)$ BTD. Then, we introduce a simultaneous matrix diagonalization (SMD)-based technique to calculate the block-term factor matrices in Section IV. Section V presents the refinement procedures to construct the final estimates. The numerical results are shown in Section VI, and Section VII is devoted to conclusions.

II. Preliminaries: BTD vs. CPD

In this article, we exploit the connection between the CPD and BTD to estimate the block-terms of a rank-$(L_r, L_r, 1)$ decomposition. Even though their uniqueness properties differ, the BTD and CPD have much in common. The BTD can be seen as a special case of the CPD with collinearity in one of the factors, see Fig. 2. Then, the rank-$(L_r, L_r, 1)$ BTD of a three-dimensional noise-corrupted tensor $\boldsymbol{\mathcal{T}} \in \mathbb{F}^{I \times J \times K}$ can either be written in a BTD form [9]

$$\boldsymbol{\mathcal{T}} = \sum_{r=1}^{R} \left(\boldsymbol{A}_r \cdot \boldsymbol{B}_r^{\mathrm{T}}\right) \circ \boldsymbol{c}_r + \boldsymbol{\mathcal{N}}, \tag{1}$$

or in a CPD form

$$\boldsymbol{\mathcal{T}} = \sum_{r=1}^{R} \sum_{\ell=1}^{L_r} \boldsymbol{A}_{r,(:,\ell)} \circ \boldsymbol{B}_{r,(:,\ell)} \circ \boldsymbol{c}_r + \boldsymbol{\mathcal{N}} \tag{2}$$

$$= \boldsymbol{\mathcal{I}}_{3,\Sigma L_r} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}^{(\mathrm{r})} + \boldsymbol{\mathcal{N}}, \tag{3}$$

where R and L_r are the number of block-terms and the multilinear ranks, respectively. Moreover, $\boldsymbol{A}_r \in \mathbb{F}^{I \times L_r}$ and $\boldsymbol{B}_r \in \mathbb{F}^{J \times L_r}$ are the r-th submatrices of $\boldsymbol{A}$ and $\boldsymbol{B}$ so that $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_R] \in \mathbb{F}^{I \times \Sigma L_r}$ and $\boldsymbol{B} = [\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_R] \in \mathbb{F}^{J \times \Sigma L_r}$. The vectors $\boldsymbol{c}_r \in \mathbb{F}^{K \times 1}$, $r = \{1, \ldots, R\}$ are stacked in the matrix $\boldsymbol{C}$ so that $\boldsymbol{C} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_R] \in \mathbb{F}^{K \times R}$. The matrix

$$\boldsymbol{C}^{(\mathrm{r})} = [\boldsymbol{c}_1 \cdot \mathbf{1}_{L_1}^{\mathrm{T}}, \boldsymbol{c}_2 \cdot \mathbf{1}_{L_2}^{\mathrm{T}}, \ldots, \boldsymbol{c}_R \cdot \mathbf{1}_{L_R}^{\mathrm{T}}] \in \mathbb{F}^{K \times \Sigma L_r} \tag{4}$$

has repeated (or linearly dependent) columns. Then, the BTD in (1) and (2) can be viewed as CPD with linear dependencies in the third factor matrix or, similarly, the CPD can be considered as a BTD with all L_rs being equal to one. With repeated or colinear columns in $\boldsymbol{C}^{(\mathrm{r})}$, Kruskal's condition [42] for uniqueness is evidently not satisfied. On the other hand, it has been shown in [7], [19] that the BTD is more attractive in this regard and can be uniquely determined under more relaxed conditions than the CPD. At the same time, while the CPD is said to be essentially unique up to an arbitrary permutation and scaling of its rank-one terms, an additional matrix product ambiguity occurs when considering the essential uniqueness of the multilinear-rank terms in the BTD

$$\boldsymbol{A}_r \boldsymbol{B}_r^{\mathrm{T}} = (\boldsymbol{A}_r \boldsymbol{H}_r)\left(\boldsymbol{H}_r^{-1} \boldsymbol{B}_r^{\mathrm{T}}\right) = \boldsymbol{A}'_r \boldsymbol{B}'^{\mathrm{T}}_r, \tag{5}$$

where $\boldsymbol{H}_r$ is an arbitrary nonsingular matrix. Since this submatrix product ambiguity is hard to resolve, the BTD is often written in the following form

$$\boldsymbol{\mathcal{T}} = \sum_{r=1}^{R} \boldsymbol{E}_r \circ \boldsymbol{c}_r + \boldsymbol{\mathcal{N}}, \tag{6}$$

where $\boldsymbol{A}_r$ and $\boldsymbol{B}_r$ are merged into $\boldsymbol{E}_r \in \mathbb{F}^{I \times J}$ such that $\boldsymbol{E}_r = \boldsymbol{A}_r \boldsymbol{B}_r^{\mathrm{T}}$ [7].

In the next sections, we will introduce an approach to determine the number of terms R and the multilinear ranks L_r using the HOSVD and clustering, followed by an estimation of the BTD factors via SECSI [41].

III. HOSVD-Based Model Order Estimation

In the most common applications of the BTD, such as ECG and EEG-MEG data processing or source separation, the rank structure is usually not available beforehand. Some papers determine the ranks by trial and error, which is very straightforward but not efficient [26], [43]. The authors in [14], [15], [16], [18] propose approaches that compute the factors and ranks of the BTD jointly in an iterative way which is computationally not effective in cases when, for example, only the rank structure itself is a subject of interest. In this article, we show how to estimate the model order of noise-corrupted BTD tensors by separating the signal and noise subspaces based on an extension of the linear regression of global eigenvalues (LaRGE) approach in [44].

To this end, let us consider a noise-corrupted three-dimensional measurement tensor $\boldsymbol{\mathcal{T}}$ given by $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{T}}_0 + \boldsymbol{\mathcal{N}} \in \mathbb{F}^{I \times J \times K}$, where $\boldsymbol{\mathcal{T}}_0 = \sum_{r=1}^{R} (\boldsymbol{A}_r \cdot \boldsymbol{B}_r^{\mathrm{T}}) \circ \boldsymbol{c}_r \in \mathbb{F}^{I \times J \times K}$ is the noiseless BTD structured data, and $\boldsymbol{\mathcal{N}} \in \mathbb{F}^{I \times J \times K}$ is an additive noise tensor. Whereas the rank of the first two modes of this rank-$(L_r, L_r, 1)$ block-term decomposition is assumed to be equal to ΣL_r, the rank in the 3-mode is assumed to be R. Then, the HOSVD of $\boldsymbol{\mathcal{T}}$ is given by

$$\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3, \tag{7}$$

where $\boldsymbol{\mathcal{S}} \in \mathbb{F}^{I\times J\times K}$ is a core tensor, and $\boldsymbol{U}_1 \in \mathbb{F}^{I\times I}$, $\boldsymbol{U}_2 \in \mathbb{F}^{J\times J}$, and $\boldsymbol{U}_3 \in \mathbb{F}^{K\times K}$ contain the left singular vectors of the n-mode unfoldings of $\boldsymbol{\mathcal{X}}$ computed from $[\boldsymbol{\mathcal{T}}]_{(n)} = \boldsymbol{U}_n \boldsymbol{\Sigma}_n \boldsymbol{V}_n^{\mathrm{H}}$, $n \in \{1, 2, 3\}$, where $\boldsymbol{\Sigma}_n$ contains the n-mode singular values $\sigma_i^{(n)}$ on its main diagonal. The n-mode singular values $\sigma_i^{(n)}$ are related to the n-mode eigenvalues $\lambda_i^{(n)}$ of $[\boldsymbol{\mathcal{T}}]_{(n)}[\boldsymbol{\mathcal{T}}]_{(n)}^{\mathrm{H}}$ through $\lambda_i^{(n)} = (\sigma_i^{(n)})^2$. Moreover, the eigenvalues can be computed as $\mathrm{diag}(\lambda_1^{(n)}, \ldots, \lambda_Q^{(n)}) = [\boldsymbol{\mathcal{S}}]_{(n)} \cdot [\boldsymbol{\mathcal{S}}]_{(n)}^{\mathrm{H}}$, where $Q \in \{I, J, K\}$ and $[\boldsymbol{\mathcal{S}}]_{(n)}$ is the n-mode unfolding of the core tensor $\boldsymbol{\mathcal{S}}$ in (7).

The main concept of LaRGE is based on the fact that (asymptotically) the noise eigenvalues have an exponential decay and, therefore, unlike the signal eigenvalues, can be approximated by a straight line on a logarithmic scale. Hence, the point where the linear regression fails will indicate the detection of the smallest signal eigenvalue and, accordingly, the rank. In the original LaRGE algorithm for the estimation of the CPD rank, the authors perform the linear regression on the so-called *global eigenvalues* [45]. The ith global eigenvalue is equal to the product of the ith n-mode eigenvalues, where $i = 1, \ldots, M$, $M = \min\{I, J, K\}$. The assumption that all factors in the CPD model have the same rank/column dimension allows taking into account the eigenvalues from all modes and leads to more reliable estimates. On the other hand, as it follows from the definition of the multilinear rank decompositions, not all the n-mode ranks have to be equal. If this asymmetric rank structure is not taken into account, the global eigenvalues might be heavily affected by the rank deficiencies in the third mode. Therefore, we split the BTD rank structure estimation procedure into two runs to accommodate this. During the first run, the ΣL_r are estimated based on the *semi-global eigenvalues* from the 1-mode and the 2-mode

$$\tilde{\lambda}_i^{(1,2)} = \left(\sigma_i^{(1)}\right)^2 \cdot \left(\sigma_i^{(2)}\right)^2, \tag{8}$$

and during the second run, only the 3-mode is used

$$\tilde{\lambda}_i^{(3)} = \left(\sigma_i^{(3)}\right)^2. \tag{9}$$

The linear regression scheme can be applied to $\lambda_i^{(1,2)} = \ln \tilde{\lambda}_i^{(1,2)}$ and $\lambda_i^{(3)} = \ln \tilde{\lambda}_i^{(3)}$ separately to estimate ΣL_r and R as follows. Starting from the smallest eigenvalue λ_M, find a prediction $\hat{\lambda}_i$ of the next eigenvalue on a logarithmic scale[1] subject to

$$\min_{a_1, a_2} \left(\sum_{i=M}^{M-k} \left(\hat{\lambda}_i - \lambda_i\right)^2 \right), \tag{10}$$

where $\hat{\lambda}_i = a_1 i + a_2$ is the smallest eigenvalue, $k = \{1, \ldots, M-1\}$ is the step index, and M is equal to $\min\{I, J\}$ when estimating ΣL_r and to K when estimating R. For each step k calculate the relative prediction error as follows

$$\delta_{M-k} = \frac{\lambda_{M-k} - \hat{\lambda}_{M-k}}{\left|\hat{\lambda}_{M-k}\right|} = \frac{\Delta_{M-k}}{\left|\hat{\lambda}_{M-k}\right|}. \tag{11}$$

Next, calculate the standard deviation of the approximation error as

$$\sigma_{M-k} = \sqrt{\frac{1}{k} \sum_{i=M}^{M-k} \left(\Delta_i - \frac{1}{k} \sum_{i=M}^{M-k} \Delta_i \right)^2}, \tag{12}$$

where the second quantity in parentheses denotes the mean value of the absolute prediction error Δ_i. Then, compute the ratio between the relative prediction error and the standard deviation of the approximation errors in the previous steps as

$$\mathrm{PESDR}_k = \frac{\delta_{M-k}}{\sigma_{M-k-1}}. \tag{13}$$

The first signal eigenvalue is detected when the Prediction Error to Standard Deviation Ratio (PESDR) exceeds the predefined threshold ρ for the first time. The number of signal eigenvalues corresponds to an estimate of ΣL_r when regression is performed on the 1-mode and 2-mode eigenvalues, and to an estimate of R when the 3-mode eigenvalues are used. The estimate of ΣL_r is further employed for the computation of initial estimates via SECSI, while the estimate of R is used for the clustering of the BTD terms.

We refer to the proposed model order estimation algorithm (for the BTD) as LaRGE-BTD, whose main steps are summarized in Algorithm 1. Moreover, to avoid estimation errors appearing due to a relatively small difference between the rank and the smallest dimension of the tensor, LaRGE with a penalty function (LaRGE PF) can be employed [46]. The penalty function ensures that the value of σ_{M-k} exceeds a certain threshold ε, which allows reducing the outliers that may lead to wrong estimates. For more details on the LaRGE algorithm, we refer the reader to [44].

The numerical results of the LaRGE-BTD estimation performance are demonstrated in Fig. 3. For the simulations, 1000 complex-valued tensors were constructed according to the model in (1) with factor matrix entries drawn from a zero-mean uncorrelated Gaussian distribution with variance σ_n^2. Accordingly, the SNR is defined as $1/\sigma_n^2$. Fig. 3(a) depicts the percentage of correct rank estimates as a function of the threshold ρ. As it can be observed, different thresholds should be used for the estimation of the number of block terms and for the sum of the multilinear ranks. According to the simulation results, the recommended threshold ρ for R and ΣL_r are between $1.5 - 2$ and $0.4 - 0.5$, respectively. The second simulation in Fig. 3(b) shows the performance of LaRGE-BTD as a function of the SNR ($\rho_R = 1.5$ and $\rho_{\Sigma L_r} = 0.5$). As it can be seen, the estimation of the number of block terms R is more reliable. This can be explained by the fact that the difference between ΣL_r and the dimensions of a tensor is smaller than the difference between R and the dimensions of a tensor.

[1] Since the further rank estimation steps are the same for both ΣL_r and R, for notational simplicity, we skip the superscripts (1,2) and (3) in λ_i (λ_i corresponds to $\lambda_i^{(1,2)}$ when estimating ΣL_r, and to $\lambda_i^{(3)}$ when estimating R) and M.

Algorithm 1: LaRGE-BTD: ΣL_r (case 1) and R (case 2).

Require: 3-way tensor $\mathcal{T} \in \mathbb{F}^{I \times J \times K}$
$\sigma_i^{(n)} \leftarrow$ HOSVD($\mathcal{T}$)
Case 1: $M = \min\{I, J\}$, Case 2: $M = K$
for $i = 1, \ldots, M$ **do**
  Case 1: $\tilde{\lambda}_i = (\sigma_i^{(1)})^2 \cdot (\sigma_i^{(2)})^2$; **Case 2:** $\tilde{\lambda}_i = (\sigma_i^{(3)})^2$
  $\lambda_i = \ln \tilde{\lambda}_i$
end for
for $k = 1, \ldots, M-1$ **do**
  prediction $\hat{\lambda}_{M-k}$
  $\delta_{M-k} = \frac{\lambda_{M-k} - \hat{\lambda}_{M-k}}{|\hat{\lambda}_{M-k}|} = \frac{\Delta_{M-k}}{|\hat{\lambda}_{M-k}|}$
  $\sigma_{M-k} = \sqrt{\frac{1}{k}\sum_{i=M}^{M-k}(\Delta_i - \frac{1}{k}\sum_{i=M}^{M-k}\Delta_i)^2}$
  $\mathrm{PESDR}_k = \frac{\delta_{M-k}}{\sigma_{M-k-1}}$
  if $(\mathrm{PESDR}_{k-1} < \rho) \wedge (\mathrm{PESDR}_k \geq \rho)$ **then**
    $\mathrm{Rank} = M - k$
    break
  end if
end for

IV. Extensions of the SECSI Framework for Block-Term Decomposition

In contrast to previous simultaneous matrix diagonalization (SMD)-based approaches [47], the SECSI framework exploits the tensor structure of the CPD to construct not only one but the full set of possible SMDs. By solving all SMDs, multiple estimates of the factor matrices can be obtained, and strategies have been presented to choose the best estimate in a subsequent step [41].

In this contribution, we introduce an extension of SECSI that enables the computation of initial estimates of the rank-$(L_r, L_r, 1)$ block-term decomposition. The new extension includes the following major enhancements and contributions:

- An introduction of a new heuristic that reduces the number of SMDs to be computed, thereby decreasing the computational load.
- An estimation of the multilinear ranks L_r using clustering.
- A befitting partitioning of the columns of the estimated matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ into multilinear rank submatrices.
- A design of highly reliable initial BTD estimates that guarantees the convergence of the refinement procedures and does not require multiple initializations.

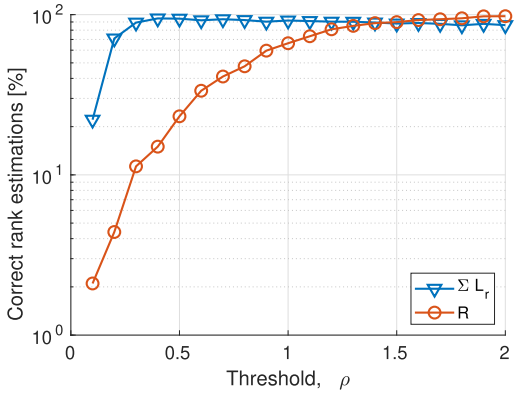
A. Brief Overview of the SECSI Framework

The SECSI framework has originally been designed to decompose a tensor into the sum of rank-1 terms. Before presenting its extension to a multilinear rank decomposition, let us review its main steps.

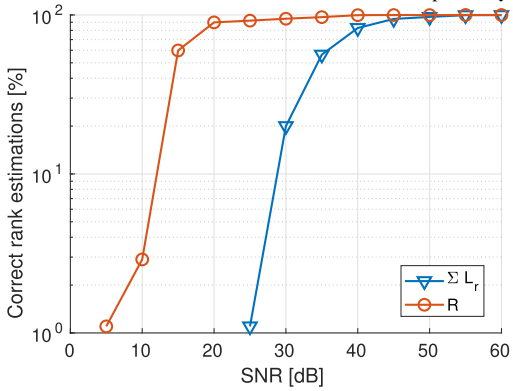
1) Truncated HOSVD: Let $\mathcal{T}_0 \in \mathbb{F}^{I \times J \times K}$ be a noise-free tensor with a given (or estimated) rank d whose CPD is given by

$$\mathcal{T}_0 = \mathcal{I}_{3,d} \times_1 \boldsymbol{F}_1 \times_2 \boldsymbol{F}_2 \times_3 \boldsymbol{F}_3, \tag{14}$$

where $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, and $\boldsymbol{F}_3$ are the corresponding factor matrices.



(a) Correct rank estimations vs. Threshold. SNR = 20 dB and SNR = 45 dB for R and L_r simulations, respectively.



(b) Correct rank estimations vs. SNR.

Fig. 3. Performance of LaRGE-BTD. Tensor of size $40 \times 40 \times 40$ with $R = 3$, $L_r = [3, 3, 3]$. Results are averaged over 1000 Monte-Carlo trials.

The first step of SECSI is the computation of a truncated HOSVD[2] of $\mathcal{T}$ to generate a truncated core tensor $\boldsymbol{\mathcal{S}}^{[s]} \in \mathbb{F}^{d \times d \times d}$ and a set of truncated unitary matrices $\boldsymbol{U}_1^{[s]} \in \mathbb{F}^{I \times d}$, $\boldsymbol{U}_2^{[s]} \in \mathbb{F}^{J \times d}$, and $\boldsymbol{U}_3^{[s]} \in \mathbb{F}^{K \times d}$ that span the column space of the corresponding n-mode unfolding of $\mathcal{T}_0$

$$\mathcal{T}_0 = \boldsymbol{\mathcal{S}}^{[s]} \times_1 \boldsymbol{U}_1^{[s]} \times_2 \boldsymbol{U}_2^{[s]} \times_3 \boldsymbol{U}_3^{[s]}. \tag{15}$$

Only the first d singular values and their corresponding left singular vectors are chosen to take into account the given (or estimated) CPD rank. The columns of each of the unitary matrices span the same vector space as the CPD factor matrices in the same mode. Thus, there exist three non-singular $d \times d$ transform matrices[3] $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ that diagonalize the core tensor $\boldsymbol{\mathcal{S}}^{[s]}$

[2]If the HOSVD has been already calculated during the LaRGE-based rank estimation step (Section III), then it can be further truncated in the first step of SECSI without a need to calculate it again. This will significantly reduce the computational cost.

[3]The assumption of the existence of the three non-singular transform matrices holds when the CPD is not rank deficient in any of the three modes, i.e., all factor matrices have full column rank. If the CP decomposition is rank deficient in one mode, the corresponding transform matrix and SMDs do not exist anymore. However, if the remaining two modes are not rank deficient, we can still construct two SMDs (left-hand and right-hand sides) that do not contain the transform matrix corresponding to the rank-deficient mode. Therefore, the SECSI framework is applicable if at least two of the tensor modes are not rank deficient.

and provide a connection between the two sets of matrices

$$\boldsymbol{F}_1 = \boldsymbol{U}_1^{[s]} \cdot \boldsymbol{T}_1, \quad \boldsymbol{F}_2 = \boldsymbol{U}_2^{[s]} \cdot \boldsymbol{T}_2, \quad \boldsymbol{F}_3 = \boldsymbol{U}_3^{[s]} \cdot \boldsymbol{T}_3. \quad (16)$$

In the next step, these transformation matrices are estimated (in several different ways).

2) Simultaneous Matrix Diagonalization of the Core Tensor's Slices: It has been shown in [41] that the n-mode slices of the core tensor $\boldsymbol{\mathcal{S}}$ have a direct relation to the columns of the n-mode's factor matrix and the transformation matrices of the other two modes from (16). For example, if $\boldsymbol{\mathcal{S}}^{(3)}_{(:,:,k)}$ is the k-th 3-mode slice of the modified core tensor $\boldsymbol{\mathcal{S}}^{(3)} = \boldsymbol{\mathcal{S}}^{[s]} \times_3 \boldsymbol{U}_3^{[s]} \in \mathbb{F}^{d \times d \times K}$, and $\hat{\boldsymbol{c}}_k$ is the k-th row of $\boldsymbol{F}_3$, then

$$\mathcal{D}\{\hat{\boldsymbol{c}}_k\} = \boldsymbol{T}_1^{-1} \cdot \boldsymbol{\mathcal{S}}^{(3)}_{(:,:,k)} \cdot (\boldsymbol{T}_2^{\mathrm{T}})^{-1}. \quad (17)$$

Equation (17) corresponds to a non-symmetric SMD [41]. It can be converted into a symmetric one by multiplying (17) by the inverse of an arbitrary pivoting slice $p \in \{1, \ldots, K\}$ from either the right-hand side (rhs)

$$\boldsymbol{\mathcal{S}}^{(3)\mathrm{rhs}}_{(:,:,k)} = \boldsymbol{\mathcal{S}}^{(3)}_{(:,:,k)} \cdot (\boldsymbol{\mathcal{S}}^{(3)}_{(:,:,p)})^{-1} \quad (18)$$

$$= \boldsymbol{T}_1 \cdot \mathcal{D}\{\hat{\boldsymbol{c}}_k\} \cdot \mathcal{D}\{\hat{\boldsymbol{c}}_p\}^{-1} \cdot \boldsymbol{T}_1^{-1}, \quad (19)$$

or the left-hand side (lhs)

$$\boldsymbol{\mathcal{S}}^{(3)\mathrm{lhs}}_{(:,:,k)} = ((\boldsymbol{\mathcal{S}}^{(3)}_{(:,:,p)})^{-1} \cdot \boldsymbol{\mathcal{S}}^{(3)}_{(:,:,k)})^{\mathrm{T}} \quad (20)$$

$$= \boldsymbol{T}_2 \cdot \mathcal{D}\{\hat{\boldsymbol{c}}_k\} \cdot \mathcal{D}\{\hat{\boldsymbol{c}}_p\}^{-1} \cdot \boldsymbol{T}_2^{-1}. \quad (21)$$

The slice with the smallest condition number[4] is considered as a clever choice for a pivot. As it can be seen from (19) and (21), the transform matrices $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ can be estimated by simultaneous diagonalization of all slices of $\boldsymbol{\mathcal{S}}^{(3)\mathrm{rhs}}$ and $\boldsymbol{\mathcal{S}}^{(3)\mathrm{lhs}}$, respectively. Moreover, from the diagonal elements of the jointly diagonalized matrices, estimates of $\boldsymbol{F}_3$ can be obtained from the right-hand side SMD (19) or the left-hand side SMD (21). The ambiguity that is brought to each column by $\mathcal{D}\{\hat{\boldsymbol{c}}_p\}^{-1}$ lies within the scaling ambiguity that is inherent for any CPD and can therefore be ignored. The current implementation of SECSI employs the SMD algorithm described in [49], which builds up the transformation matrix iteratively out of alternating shear matrices and unitary transform matrices. The acquired $\boldsymbol{T}_1$ ($\boldsymbol{T}_2$) obtained from this SMD can be used to estimate $\boldsymbol{F}_1$ ($\boldsymbol{F}_2$) according to (16). After two of the three factor matrix estimates have been found, a least squares solution for the last factor can be computed from one of the following equations

$$\begin{aligned} \boldsymbol{F}_1 &= [\boldsymbol{\mathcal{T}}_0]_{(1)} \cdot (\boldsymbol{F}_2 \diamond \boldsymbol{F}_3)^{\mathrm{T}+}, \\ \boldsymbol{F}_2 &= [\boldsymbol{\mathcal{T}}_0]_{(2)} \cdot (\boldsymbol{F}_3 \diamond \boldsymbol{F}_1)^{\mathrm{T}+}, \\ \boldsymbol{F}_3 &= [\boldsymbol{\mathcal{T}}_0]_{(3)} \cdot (\boldsymbol{F}_1 \diamond \boldsymbol{F}_2)^{\mathrm{T}+}. \end{aligned} \quad (22)$$

The similarity between this problem and the generalized eigenvalue decomposition (GEVD) approach that is used to initialize the Tensorlab algorithm should be mentioned here. Whereas both algorithms are essentially pencil-based, the approach in [50] uses only two slices of the tensor to compute a generalized eigenvalue decomposition and, subsequently, estimate the factor matrices. In contrast, the SECSI framework uses an SMD-based approach that examines different modes and, instead of considering only one subpencil, takes all the slices of the tensor into account.

The authors in [51] study the performances of the pencil-based algorithms for the CPD and state that "for every pencil-based algorithm, there exists an open set of the rank r tensors in $\mathbb{R}^{n_1 \times n_2 \times n_3}$ for which it is unstable" and the instability is caused by a significant difference between the condition number of a tensor rank decomposition [52] in $\mathbb{R}^{n_1 \times n_2 \times n_3}$ and $\mathbb{R}^{n_1 \times n_2 \times 2}$: the expected condition number of a tensor rank decomposition for $n_1 \times n_2 \times 2$ tensors is much larger on the average than the tensor condition number[5] for $n_1 \times n_2 \times n_3$ tensors. The article also states that "as n_3 increases, very large condition numbers become increasingly unlikely". Consequently, the fact that we consider the simultaneous matrix diagonalization of tensors of size $d \times d \times K$ ((18)–(20)), according to the aforementioned article, greatly increases the probability of a low tensor condition number, and consequently, leads to more stable solutions.

Another CPD algorithm that improves the accuracy of the GEVD-based solutions is proposed in [53]. The authors introduce the GESD approach that exploits not one but many subpencils of the tensor to find the generalized eigenvectors and eigenspaces that correspond to sufficiently well-separated generalized eigenvalues. Two of the three factor matrices are then obtained by combining information from the different subpencils. The GESD algorithm outperforms the GEVD, especially for the correlated factors case, and therefore its extension to block-term decompositions might be promising. Compared to the GESD, SECSI does not consider the subpencils of the tensor but all the slices jointly. Moreover, in the non-degenerate CPD cases, the algorithm allows computing the SMDs in different modes (not only in the 3-mode) to get the best solution.

3) Choice of the Final Estimate: As mentioned above, the three factor matrices can either be retrieved by the left-hand side estimate or the right-hand side estimate, which enables a choice out of two sets of estimated factor matrices for each mode, resulting in 6 sets of factor matrix estimates for a tensor with three dimensions. The best accuracy in terms of the reconstruction error can be achieved by considering all combinations of estimated factor matrices $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, and $\boldsymbol{F}_3$ (from all SMDs) and choosing the combination with the smallest reconstruction error as final estimates ("Best Matching (BM)" approach). Since, in this case, all the combinations, including the factor matrices from different SMDs have to be checked, this solution is computationally quite expensive. On the positive side, the SECSI framework offers several heuristics that allow controlling the trade-off between performance and complexity. Therefore, the final estimates of the framework can be chosen according to predefined heuristics, which are based on different selection criteria such as reconstruction error, condition number,

[4]The condition number of a matrix $\boldsymbol{A}$ is defined as the ratio of its largest singular value to the smallest singular value and quantifies the sensitivity of the $\boldsymbol{Ax} = \boldsymbol{b}$ problem to the changes in $\boldsymbol{A}$ [48].

[5]Here, by "condition number" and "tensor condition number" we refer to "condition number of tensor rank decomposition" [51], [52].

or SMD residuals. The SECSI framework features the following heuristics to choose from [41]:

- *REC PS (reconstruction error - paired solutions)*: Instead of enabling factor matrix combinations from different SMDs as in the "BM" approach, only the combinations that originate from the same SMD (*"paired solutions"*) are evaluated. As an accuracy measure, this heuristic uses the *reconstruction error* between the tensor reconstructed from the estimated loading matrices and the original data tensor. The set of matrices that provides the smallest reconstruction error is picked as the final output.
- *RES* (*SMD residuals*): Out of all estimates, the set of factor matrices that originate from the "best" SMD is used, which is the SMD whose transformation matrix $\boldsymbol{T}$ provides the best diagonalization of the n-mode core tensor slices, thus leaving the smallest *residual error*. The residual error is defined as the average Frobenius norm of the off-diagonal elements of the tensor slice pencils after the estimated transformation matrices $\boldsymbol{T}$ and $\boldsymbol{T}^{-1}$ are applied to it.
- *CON PS (condition number - paired solutions)*: Instead of solving all possible SMD problems, only the two SMDs in the mode where the pivot slice has the best *condition number* are computed. The final solution is selected from the SMD (left-hand side or right-hand side) that yields the lower reconstruction error. Subsequently, the two solutions are compared in terms of their reconstruction error, and the one which yields the lower reconstruction error is returned as the final solution. The combinations between the estimates from the two different SMDs are not considered ("paired solutions"). This heuristic is the fastest since not all SMDs have to be solved.

These heuristics allow reducing computational complexity by offering an excellent trade-off between the complexity and accuracy of the solution. Since the smallest possible reconstruction error is achieved by the "best matching" approach, we consider it as a benchmark solution that determines the achievable reconstruction error in the SECSI framework.

The proposed BTD extension of SECSI uses new BTD heuristics which exploit the prior knowledge of the block-term structure of the tensor to reduce the computational load of SECSI.

B. Extension to Block-Term Decomposition

SECSI is capable of computing an approximate CPD even in so-called "degenerate" cases, meaning that one factor matrix has proportional (or highly correlated) columns or if one dimension of the tensor is smaller than the CP-rank of the tensor. In these cases, a non-singular transformation matrix for the corresponding mode does not exist. However, to be able to generate an estimate for the CPD factor matrices of a three-way array, at least two of the three transformation matrices have to exist and have to be non-singular, so that (19) and (21) can be obtained. If a BTD scenario is viewed as a special case of the CPD, one of the CPD factor matrices must have repeated columns, leading to a degenerate case. A CPD approximation is still possible under these circumstances, although it is only essentially unique (see (5) in Section II). Note that the existence of the transformation matrices coincides with one of the sufficient conditions for essential uniqueness defined in [7].

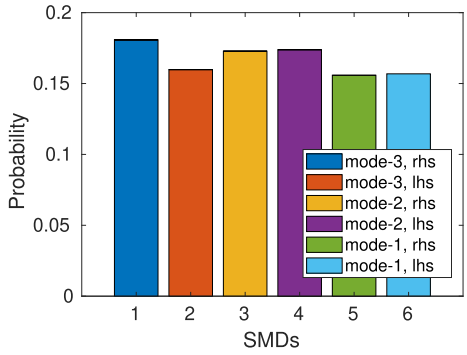
Reducing the Computational Load of the SMDs: High numerical stability of the estimates (which makes them favorable) obtained via SECSI comes with computational cost. If we consider the BTD as a special case of the CPD, we assume that one of the factors has linearly dependent columns. This can be exploited to decrease the computational load of the SECSI framework. For most of the heuristics, all six possible SMD have to be calculated in order to choose the best estimate. This is reasonable in case of the CPD because, apart from the different dimensions of the tensor for each of the modes, the CPD problem is a symmetrical one and the factor matrices often feature similar mathematical properties. In the case of a BTD however, the problem becomes asymmetrical due to the specific rank structure, which is also one of the reasons why two of the factor matrices are only unique up to the product of their submatrices. Moreover, this asymmetry affects the choice of the estimates in the last step of SECSI as follows.

If one of the modes is rank deficient (assume the third mode in case of the rank-$(L_r, L_r, 1)$ BTD), the corresponding transform matrix and the SMDs where this transform matrix is present do not exist anymore. However, we can still construct the two SMDs (left-hand side and right-hand side, equations (19) and (21)) in the 3-mode. Then, the 3-mode factor matrix can be directly obtained from the K diagonal matrices, and the remaining factors are estimated from the transform matrix and the least squares fit. This procedure will result in two sets of estimates for every factor matrix.

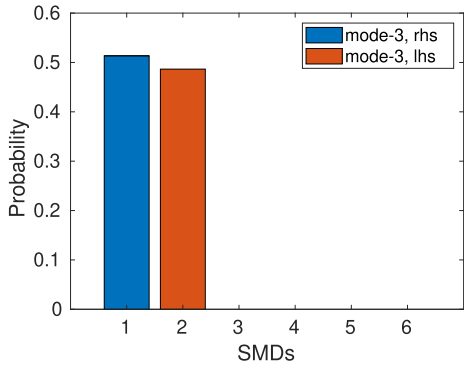
The following numerical experiments demonstrate the probabilities with which the final solutions were chosen from particular SMDs when applying SECSI on CPD- and BTD-model tensors. First, we generate three factor matrices with CP rank ΣL_r and random complex Gaussian-distributed entries according to the model in (14) to build up a CPD tensor. After that, a noise tensor with a small power is added to the tensor. Then, we decompose the noise-distorted tensor via the SECSI framework with the "REC PS" heuristic and track which of the 6 sets of estimated factor matrices is used as the final output. As the histogram in Fig. 4(a) shows, there are no preferred modes in the CPD scenario, and the final estimates can originate from all SMDs with a non-zero probability.

The second simulation is conducted in the same way as the one described above, but the tensor is now constructed from block-terms according to (1) and with the multilinear rank-$(L_r, L_r, 1)$ structure. Whereas the first and second mode factor matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ have ΣL_r distinct columns, the third mode factor matrix $\boldsymbol{C}^{(\mathrm{r})}$ has repeated columns as in (3). The results for the second simulation are shown in Fig. 4(b). It can be seen that for the BTD-constructed tensor in all cases one of the estimates with the diagonalizations in the 3-mode is used, whereas no such tendency can be seen for the CPD-constructed tensor.

These simulations demonstrate the advantages of the utilization of different mode SMDs in the original SECSI framework for the CPD. In a given application, we might not know beforehand in which mode the rank deficiency will occur. Most



(a) Canonical polyadic decomposition



(b) Block-term decomposition

Fig. 4. Histogram of paired solutions chosen by SECSI ("REC PS") after comparing their reconstruction errors. "lhs" and "rhs" refer to the left-hand side and right-hand side transformation matrix estimate, respectively. $\mathcal{T} \in \mathbb{R}^{20\times30\times40}$, SNR $= 50$ dB. Results are averaged over 1000 Monte-Carlo trials.

of the GEVD-based algorithms, by default, are applied to the 3-mode slices and fail to obtain an accurate solution when the rank-deficiencies occur in the other modes. SESCI, by contrast, constructs all SMDs, and allows choosing the solution with the best performance (in terms of the chosen selection criterion)

Nevertheless, the computation of the SMDs in all modes increases the computational time. Therefore, we have added new heuristics to the SECSI-framework: "REC PS BT" (*reconstruction error - paired solutions - block-terms*) and "RES BT" (*residual error - block-terms*). Similarly to the CPD-SECSI heuristics described in Section IV-A3, they choose the final estimates based on the reconstruction error and SMD residual error, respectively. In contrast, to the original heuristics, they only consider the SMDs in the 3-mode. The "REC PS BT" heuristic solves both right-hand and left-hand side SMDs and picks the estimates from the SMD that provides the smallest reconstruction error. The "RES BT" heuristic computes the final estimates only from one SMD that provides the best diagonalization in terms of the residual error. This lowers the computation time of the SMDs up to a factor of three (depending on which heuristic is used as a reference).

Since in the BTD case, we only consider "paired solutions", i.e, the final estimates always originate from the same SMD problem, the scaling and permutation ambiguities of the factor matrices are consistent and do not affect the reconstruction error.

Restoring the Submatrix Structure: As it can be observed from (1) and (3), the BTD can be transformed into a CPD by repeating the r-th column of $\boldsymbol{C}$ in (1) L_r times. Whereas the transformation of $\boldsymbol{C}$ from the BTD into $\boldsymbol{C}^{(\mathrm{r})}$ in the CPD is therefore rather trivial (if the submatrix structure of $\boldsymbol{A}$ and $\boldsymbol{B}$ is assumed to be known), the reverse operation is a more challenging task. After performing an approximate CPD on the noise-corrupted BTD tensor, the estimated matrix $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ will not perfectly match the structure of $\boldsymbol{C}^{(\mathrm{r})}$ in (4). The columns in $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ will be arbitrarily permuted and scaled, and the repeated columns in $\boldsymbol{C}^{(\mathrm{r})}$ will not be explicitly equal in $\hat{\boldsymbol{C}}^{(\mathrm{r})}$. Therefore, one has to find an efficient way to reduce the number of columns of the estimated matrix $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ to R columns to obtain $\hat{\boldsymbol{C}}$ without clipping off valuable information.

First, let us consider the output of the SECSI-framework. In the noiseless case, the columns of $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ are repetitions of the columns of $\hat{\boldsymbol{C}}$ in a random order. For the computation of $\hat{\boldsymbol{C}}$ from the CPD approximation $\hat{\boldsymbol{C}}^{(\mathrm{r})}$, the submatrix structure of $\boldsymbol{A}$ and $\boldsymbol{B}$ has to be restored. This means that the columns which belong to the same $(L_r, L_r, 1)$-term need to be adjacent to each other. In the noiseless case, this can be done by rearranging the columns of $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ so that equal columns are adjacent and by rearranging the columns of $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{B}}$ in the same way. If the tensor is disturbed by noise, however, the repeated columns of $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ become more distinct from each other, bringing up the need for a more sophisticated approach. The k-means clustering, a commonly used technique in data analysis, shows to be quite effective for this task since it only requires the number of clusters to be known beforehand [54]. It is a simple, iterative algorithm that tries to find a set of cluster centers so that the summed squared distance of all elements to their nearest cluster center is minimized. In the BTD context, each cluster belongs to a different $(L_r, L_r, 1)$ block term, and if there were no clustering errors, the r-th cluster will contain L_r columns. In our simulations, we use the k-means++ algorithm with a distance measure defined as a cosine of an angle between two unit-norm vectors ($\cos\theta = \frac{\mathrm{Re}\{\boldsymbol{c}_i^{\mathrm{H}}\boldsymbol{c}_j\}}{\|\boldsymbol{c}_i\|\|\boldsymbol{c}_j\|}$). After the columns of $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ have been grouped into clusters, a permutation matrix $\boldsymbol{P}$ can be retrieved that rearranges the columns of $\hat{\boldsymbol{C}}^{(\mathrm{r})}$ so that all columns that belong to the same cluster are adjacent to each other. By multiplying $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{B}}$ by the same permutation matrix, their submatrix structure can be restored. The k-means clustering algorithm is also used in the GEVD initialization of the Tensorlab BTD algorithm [9]. It can be observed that in some cases, the k-means fails to generate clusters of the desired size from the GEVD estimates. In that event, the Tensorlab algorithm switches to random initialization. As discussed in Section IV-A2, these instabilities might be caused by the fact that the expected tensor condition number of the selected two-slices subtensor might be very high on the average, compared to the expected tensor condition number of the more-than-two slices tensor. On the contrary, SECSI exploits the information of all available tensor slices which stabilizes its performance. In cases when the L_rs are not available, we utilize k-means to estimate the multilinear ranks. But if the multilinear ranks are known, we

Algorithm 2: SECSI-BTD framework for rank-$(L_r, L_r, 1)$ BTD with rank estimation.

Require: 3-way tensor $\boldsymbol{\mathcal{T}}$

1: $\hat{R}, \Sigma\hat{L}_r \leftarrow \text{LaRGE}_{\text{BT}}(\boldsymbol{\mathcal{T}})$ ▷*Ranks est.*

2: $\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{C}}^{(\text{r})} \leftarrow \text{SECSI}_{\text{BT}}(\boldsymbol{\mathcal{T}}, \Sigma\hat{L}_r)$ ▷*Factors in CPD form*

3: $\boldsymbol{P}, \hat{L}_r\text{s} \leftarrow k\text{-means}(\hat{\boldsymbol{C}}, R)$ ▷*Clustering*

4: $\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}^{(\text{r})} \leftarrow \text{reordering}(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{C}}^{(\text{r})}, \hat{L}_r, \boldsymbol{P})$

5: $\tilde{\boldsymbol{A}}_r, \tilde{\boldsymbol{B}}_r, \tilde{\boldsymbol{c}}_r \leftarrow \text{refinement}$ ▷$\boldsymbol{C}$ *in BT form, ALS/NLS*

employ the constrained k-means to guarantee the appropriate cluster sizes [55]. Another approach to calculate L_r based on Symmetric Joint Block Diagonalization (S-JBD) is proposed in [19]. However, it still requires prior knowledge of R and ΣL_r.

V. Refinements

After clustering, the columns of $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$, and $\hat{\boldsymbol{C}}^{(\text{r})}$ are sorted in a way that the columns that belong to the same submatrix are adjacent to each other. We denote the permuted matrices as $\tilde{\boldsymbol{A}}$, $\tilde{\boldsymbol{B}}$, and $\tilde{\boldsymbol{C}}^{(\text{r})}$. Next, the following refinement steps can be applied to generate the final block term estimates as in (1). At this stage, the matrix $\tilde{\boldsymbol{C}}^{(\text{r})}$ still contains the colinear columns ordered according to R clusters and needs to be brought to the BTD form. In other words, the colinear columns have to be reduced. To perform the reduction effectively, i.e., without cutting off any data, we first refine the matrices $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{B}}$ in an ALS fashion as follows

$$\tilde{\boldsymbol{A}} = [\boldsymbol{\mathcal{T}}]_{(1)} \left(\tilde{\boldsymbol{B}} \diamond \tilde{\boldsymbol{C}}^{(\text{r})}\right)^{\text{T}+}, \tag{23}$$

$$\tilde{\boldsymbol{B}} = [\boldsymbol{\mathcal{T}}]_{(2)} \left(\tilde{\boldsymbol{C}}^{(\text{r})} \diamond \tilde{\boldsymbol{A}}\right)^{\text{T}+}. \tag{24}$$

This step incorporates the multiple estimates of the columns of $\tilde{\boldsymbol{C}}^{(\text{r})}$ into $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{B}}$. Then, an estimate of $\boldsymbol{C} \in \mathbb{F}^{K \times R}$ can be computed using equation for $\tilde{\boldsymbol{C}}$ from the ALS-based scheme for the BTD in [8]

$$\tilde{\boldsymbol{C}} = [\boldsymbol{\mathcal{T}}]_{(3)} \left[(\tilde{\boldsymbol{A}}_1 \diamond \tilde{\boldsymbol{B}}_1) \cdot \boldsymbol{1}_{L_1}, \ldots, (\tilde{\boldsymbol{A}}_R \diamond \tilde{\boldsymbol{B}}_R) \cdot \boldsymbol{1}_{L_R}\right]^{\text{T}+}. \tag{25}$$

Next, to finalize the factor matrix estimation, a suitable refinement scheme can be applied (either using nonlinear least squares (NLS) or alternating least squares (ALS) [8], [13]). In the BTD-ALS procedure, the factor matrices $\tilde{\boldsymbol{A}}$ and $\tilde{\boldsymbol{B}}$ are updated as

$$\tilde{\boldsymbol{A}} = [\boldsymbol{\mathcal{T}}]_{(1)} \left(\tilde{\boldsymbol{B}} \diamond_{\text{sm}} \tilde{\boldsymbol{C}}\right)^{\text{T}+}, \tag{26}$$

$$\tilde{\boldsymbol{B}} = [\boldsymbol{\mathcal{T}}]_{(2)} \left(\tilde{\boldsymbol{C}} \diamond_{\text{sm}} \tilde{\boldsymbol{A}}\right)^{\text{T}+}, \tag{27}$$

where $\diamond_{\text{sm}}$ denotes the submatrix-wise Khatri-Rao product defined for two matrices $\boldsymbol{X} = [\boldsymbol{X}_1, \ldots, \boldsymbol{X}_R] \in \mathbb{F}^{I \times \Sigma L_r}$ and $\boldsymbol{Y} = [\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_R] \in \mathbb{F}^{J \times \Sigma L_r}$ as $\boldsymbol{X} \diamond_{\text{sm}} \boldsymbol{Y} = [\boldsymbol{X}_1 \otimes \boldsymbol{Y}_1, \ldots, \boldsymbol{X}_R \otimes \boldsymbol{Y}_R] \in \mathbb{C}^{IJ \times \Sigma L_r^2}$. The matrix $\tilde{\boldsymbol{C}}$ is updated as in (25). The NLS/ALS schemes are run until one of the stopping criteria is met (the relative change between two successive iterations is small, the maximum number of iterations is reached, or the change in objective function value relative to the tensor norm is less than a specified tolerance).

The proposed SECSI-BTD framework to compute the rank-$(L_r, L_r, 1)$ BTD with rank estimation is summarized in Algorithm 2. For tensors with a known rank structure, step 1 can be skipped.

VI. Numerical Results

In this section, we conduct simulations with synthetically generated data to assess the performance of the proposed algorithms and compare it to the algorithms from the Tensorlab toolbox [9], [13] for computing an approximate rank-$(L_r, L_r, 1)$ decomposition.

The tensors for simulations are constructed according to the rank-$(L_r, L_r, 1)$ BTD model in (1) where the factor matrices have been drawn from a zero mean circularly symmetric complex Gaussian (ZMCSCG) distribution with unit variance, and the noise tensor $\boldsymbol{\mathcal{N}}$ have been formed from ZMCSCG entries with variance σ_n^2. Accordingly, the SNR is defined as $1/\sigma_n^2$.

We use two different accuracy measures, a relative squared reconstruction error (SRE) and a relative squared factor error (SFE), to evaluate the accuracy of an estimated BTD. The SRE is defined as

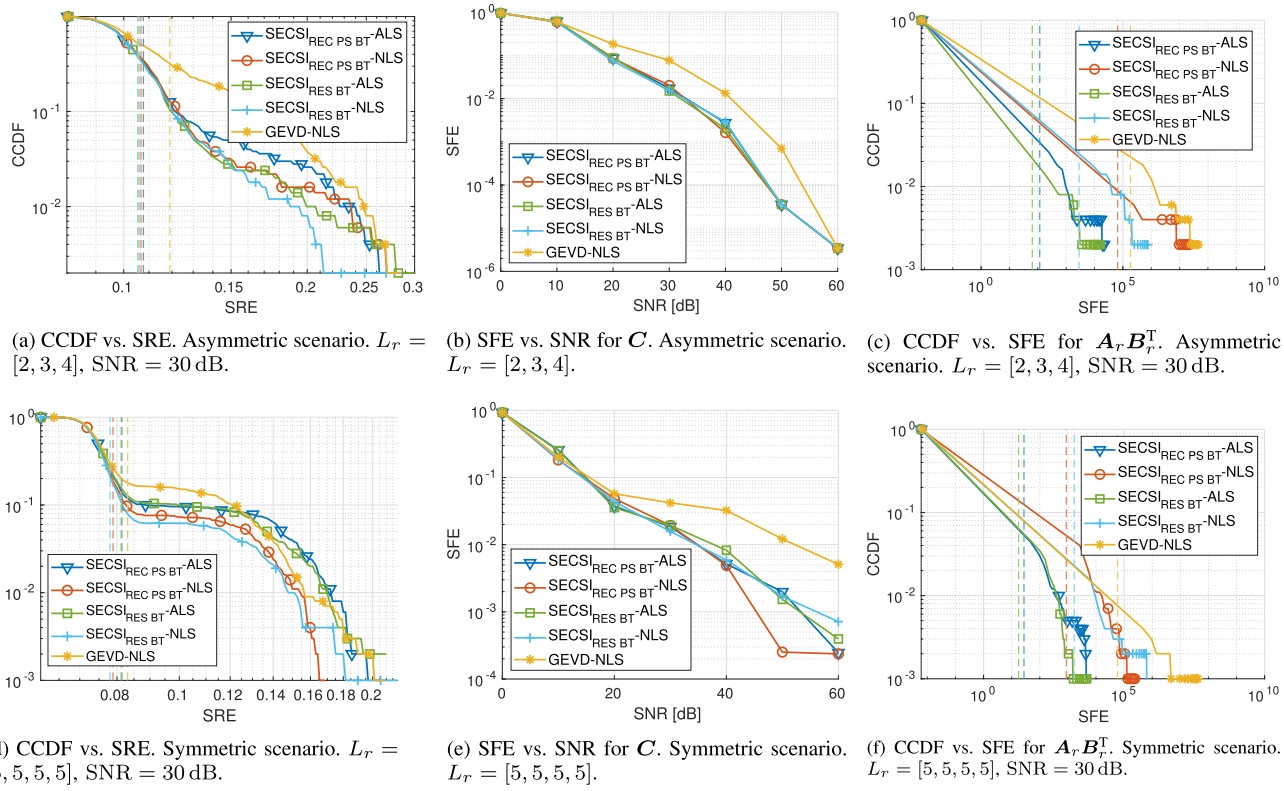
$$\text{SRE} = \frac{\left\|\hat{\boldsymbol{\mathcal{T}}} - \boldsymbol{\mathcal{T}}_0\right\|_{\text{F}}^2}{\|\boldsymbol{\mathcal{T}}_0\|_{\text{F}}^2}, \tag{28}$$

where $\boldsymbol{\mathcal{T}}_0$ is the original noise-free tensor, and $\hat{\boldsymbol{\mathcal{T}}}$ is a tensor reconstructed from the estimated BTD factors as $\hat{\boldsymbol{\mathcal{T}}} = \sum_{r=1}^{R} (\tilde{\boldsymbol{A}}_r \cdot \tilde{\boldsymbol{B}}_r^{\text{T}}) \circ \tilde{\boldsymbol{c}}_r$. The SFE for the matrix $\boldsymbol{C}$ is defined as

$$\text{SFE} = \frac{\left\|\tilde{\boldsymbol{C}} \cdot \boldsymbol{P}_c - \boldsymbol{C}\right\|_{\text{F}}^2}{\|\boldsymbol{C}\|_{\text{F}}^2}, \tag{29}$$

where the matrix $\boldsymbol{P}_c$ corrects the permutation and scaling ambiguity that is inherent in the estimation of the factor matrices. These ambiguities are resolved as follows. First, the columns of both $\tilde{\boldsymbol{C}}$ and $\boldsymbol{C}$ are normalized to unit norm. The permutation is corrected by rearranging the columns in $\tilde{\boldsymbol{C}}$ so that the inner product $\tilde{\boldsymbol{c}}_r^{\text{H}} \cdot \boldsymbol{c}_r$ between the estimated and original factor columns is maximized. Then, the scaling is resolved by multiplying the estimated columns by their scalar projection $\tilde{\boldsymbol{c}}_r^{\text{H}} \cdot \boldsymbol{c}_r$ in the direction of the original vectors $\boldsymbol{c}_r$. Whereas the matrix $\boldsymbol{C}$ can be estimated up to scaling and column permutation ambiguities, the multilinear factor matrices $\boldsymbol{A}_r$ and $\boldsymbol{B}_r$ are only unique up to the products of their submatrices due to the submatrix product ambiguities show in (5). Therefore, when evaluating the performance for the matrices $\boldsymbol{A}_r$ and $\boldsymbol{B}_r$, in case of the SFE calculations we consider the products $\boldsymbol{A}_r \boldsymbol{B}_r^{\text{T}}$ as follows

$$\text{SFE}(\boldsymbol{A}\boldsymbol{B}^{\text{T}}) = \mathbb{E}\left\{\frac{\left\|\tilde{\boldsymbol{A}}_r \tilde{\boldsymbol{B}}_r^{\text{T}} - \boldsymbol{A}_r \boldsymbol{B}_r^{\text{T}}\right\|_{\text{F}}^2}{\|\boldsymbol{A}_r \boldsymbol{B}_r^{\text{T}}\|_{\text{F}}}\right\}, \tag{30}$$



(a) CCDF vs. SRE. Asymmetric scenario. $L_r = [2, 3, 4]$, SNR $= 30$ dB.

(b) SFE vs. SNR for $\boldsymbol{C}$. Asymmetric scenario. $L_r = [2, 3, 4]$.

(c) CCDF vs. SFE for $\boldsymbol{A}_r\boldsymbol{B}_r^{\mathrm{T}}$. Asymmetric scenario. $L_r = [2, 3, 4]$, SNR $= 30$ dB.

(d) CCDF vs. SRE. Symmetric scenario. $L_r = [5, 5, 5, 5]$, SNR $= 30$ dB.

(e) SFE vs. SNR for $\boldsymbol{C}$. Symmetric scenario. $L_r = [5, 5, 5, 5]$.

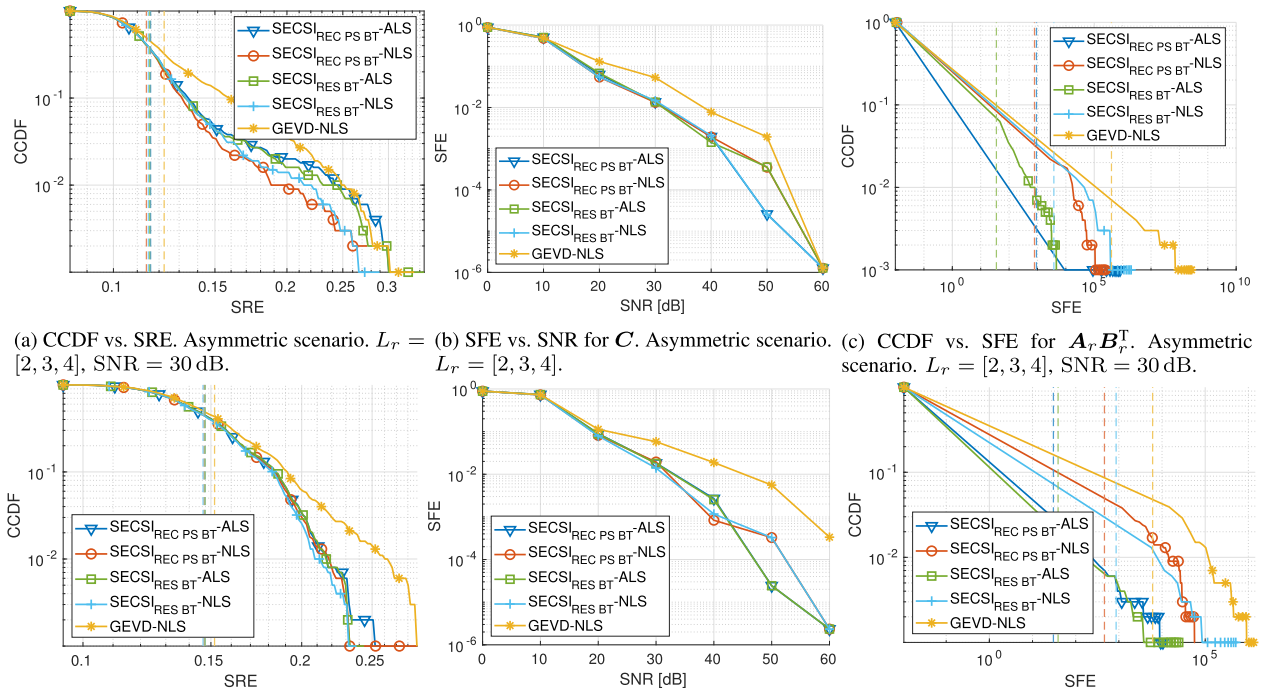
(f) CCDF vs. SFE for $\boldsymbol{A}_r\boldsymbol{B}_r^{\mathrm{T}}$. Symmetric scenario. $L_r = [5, 5, 5, 5]$, SNR $= 30$ dB.

Fig. 5. $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{20\times30\times40}$ with asymmetric and symmetric multilinear rank structures. Averaged over 1000 Monte-Carlo trials.

where $\boldsymbol{A}_r$ and $\boldsymbol{B}_r$ are the original factor matrices, and $\tilde{\boldsymbol{A}}_r$ and $\tilde{\boldsymbol{B}}_r$ are the estimated factors after resolving the permutation and scaling ambiguities.

We compare the estimation performances of the proposed SECSI-BTD algorithm with the performance of the NLS algorithm with GEVD initialization from [13], which has been commonly used in many BTD applications [26], [43], [56], [57]. For the proposed SECSI-BTD approach, we examine both the ALS and NLS refinement procedures. ALS stops when the change in the reconstruction error between two successive iterations is less than a specified threshold (10^{-6}) or the maximum number of iterations (30) is reached. For the NLS refinement, we use the nonlinear least squares procedure based on Gauss-Newton with dogleg trust region from Tensorlab [9] with its default settings. The algorithm stops when one of the following criteria is met: the maximum number of iterations has been reached (200), the change in the objective function value relative to the norm of the tensor is less than a specified tolerance (10^{-12}), or the ratio of the step size relative to the norm of the current iterate is less than a specified tolerance (10^{-6}). To ensure a fair comparison, the rank structure is assumed to be known for all algorithms.

In the simulations, we consider four scenarios with asymmetric and symmetric multilinear rank structures. In the first two scenarios, the tensors are of size $(20 \times 30 \times 40)$ with the rank structures $R = 3$, $L_r = [2, 3, 4]$ and $R = 4$, $L_r = [5, 5, 5, 5]$. In the other two scenarios, the tensors have a smaller 3-mode dimension and are of size $(30 \times 40 \times 15)$ with $L_r = [2, 3, 4]$ and $L_r = [2, 2, 2]$. The SRE and SFE performances for $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{20\times30\times40}$ and $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{30\times40\times15}$ are shown in Figs. 5 and 6, respectively. The blue curves denote the proposed SECSI-BTD approach with the "REC PS BT" heuristic and ALS refinement, red lines correspond to SECSI-BTD with "REC PS BT" and NLS refinement, the green and the light blue curves denote the SECSI-BTD approach with the "RES" heuristic and ALS or NLS refinements, respectively. The NLS algorithm with GEVD-based initialization from [13] is denoted with yellow color. Figs. 5(a), (d), 6(a), and (d) show the complementary cumulative distribution functions (CCDF) with respect to the reconstruction errors for all scenarios (for better visual representation without overwhelming the figures, we plot only 10 out of 1000 markers on the CCDF plots). The vertical dashed lines represent the mean of the errors for each algorithm. The factor reconstruction errors with respect to SNRs for the matrix $\boldsymbol{C}$ in different scenarios are shown in Figs. 5(b), (e), 6(b), and (e). As it can be observed, the proposed algorithms outperform the GEVD-based scheme in a large range of SNRs for all scenarios. In a scenario with $L_r = [5, 5, 5, 5]$ (difficult scenario, since $\Sigma L_r = I = 20$) the SECSI-BTD schemes with the NLS refinement show a better convergence behavior than the other algorithms. The CCDFs with respect to SFE for the matrix products $\boldsymbol{A}_r\boldsymbol{B}_r^{\mathrm{T}}$ are presented in Figs. 5(c), (f), 6(c), and (f). As can be seen from the CCDF plots, the GEVD scheme results in more outliers, and the proposed algorithms provide better initialization and, consequently, more stable performance in all scenarios. The explanations of the increased robustness of the SMD-based initialization are discussed in Section IV-A2.



(a) CCDF vs. SRE. Asymmetric scenario. $L_r = [2, 3, 4]$, SNR $= 30$ dB.

(b) SFE vs. SNR for $\boldsymbol{\mathcal{C}}$. Asymmetric scenario. $L_r = [2, 3, 4]$.

(c) CCDF vs. SFE for $\boldsymbol{A}_r\boldsymbol{B}_r^{\mathrm{T}}$. Asymmetric scenario. $L_r = [2, 3, 4]$, SNR $= 30$ dB.

(d) CCDF vs. SRE. Symmetric scenario. $L_r = [2, 2, 2]$, SNR $= 30$ dB.

(e) SFE vs. SNR for $\boldsymbol{\mathcal{C}}$. Symmetric scenario. $L_r = [2, 2, 2]$.

(f) CCDF vs. SFE for $\boldsymbol{A}_r\boldsymbol{B}_r^{\mathrm{T}}$. Symmetric scenario. $L_r = [2, 2, 2]$, SNR $= 30$ dB.

Fig. 6. $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{30\times40\times15}$ with asymmetric and symmetric multilinear rank structures. Averaged over 1000 Monte-Carlo trials.

TABLE I
AVERAGE RUN TIME OF THE ALGORITHMS IN DIFFERENT SCENARIOS [SEC]

Scenario	SECSI-BTD-ALS ("REC PS BT")	SECSI-BTD-ALS ("RES BT")	SECSI-BTD-NLS ("REC PS BT")	SECSI-BTD-NLS ("RES BT")	GEVD-NLS
$30 \times 40 \times 15$ $L_r = [2, 2, 2]$	0.5216	0.5184	1.4900	1.4752	1.3346
$30 \times 40 \times 15$ $L_r = [2, 3, 4]$	1.1037	1.2001	1.9106	2.2164	1.4910
$20 \times 30 \times 40$ $L_r = [5, 5, 5, 5]$	29.4994	27.9834	32.5071	30.4271	2.0646
$20 \times 30 \times 40$ $L_r = [2, 3, 4]$	3.2022	3.2068	4.1025	3.9924	1.6720

The average run times of the algorithms for SNR $= 20$ dB are shown in Table I (the algorithm performances were evaluated on an Intel Xeon Gold 6342 CPU 2.80 GHz machine running Linux CentOS 7, kernel 3.10.0-1160.el7.x86_64 and MATLAB R2020b 64-bit). We can observe that for the scenario with $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{30\times40\times15}$ and $L_r = [2, 2, 2]$, the SECSI-BTD with ALS refinement is almost three times faster than the GEVD-NLS and SECSI-NLS algorithms, which show similar time performance. In case of the scenario with $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{30\times40\times15}$ and $L_r = [2, 3, 4]$, the algorithms show a comparable time performance with the SECSI-BT-ALS being slightly faster on average. For the scenarios with $\boldsymbol{\mathcal{T}} \in \mathbb{C}^{20\times30\times40}$ the proposed algorithms are slower than the GEVD-NLS solutions. However, considering that the SECSI-BTD schemes provide more accurate and reliable error performance results, they can be an appealing solution for applications where accuracy is more important. Moreover, SECSI-BTD is still faster than the GEVD-based approaches for the cases where the dimension in the rank-deficient mode is smaller than in other modes. These differences in the computational time performances are explained by the fact that the rank-deficient mode determines the dimensionality of the SMD problem: it increases the number of matrices to be jointly diagonalized. Similary, the time increase in the third scenario is related to the increased sizes of simultaneously diagonalized matrices ($\Sigma L_r = 20$). However, for these scenarios, the complexity might still be decreased by considering a more efficient diagonalization algorithm than in [49].

VII. CONCLUSION

In this article, we have exploited the connection between the Canonical Polyadic and the rank-$(L_r, L_r, 1)$ block-term decompositions and presented the SECSI-BTD framework to compute an approximate rank-$(L_r, L_r, 1)$ BTD. We have divided the proposed algorithm into three main blocks, which include the rank structure estimation, the initial estimation of

the BTD factors, and a final refinement procedure. For the rank structure estimation, we have introduced an extension of the LaRGE technique for CPD model order estimation to estimate the number of blocks and the sum of L_rs in the BTD. Moreover, a procedure that uses clustering for the estimation of multilinear ranks has been presented. Furthermore, we have shown how the Semi-algebraic framework for approximate CPD via Simultaneous Matrix Diagonalizations (SECSI) can be employed for the computation of the initial BTD factors. Additionally, new heuristics have been added to the original algorithm to reduce the computational time and make the estimation more efficient. In the last block of the algorithm, we have presented the clustering and refinement procedures that return the final rank-$(L_r, L_r, 1)$ decomposition estimates. The simulation results have shown that the proposed SECSI-BTD algorithm outperforms the state-of-the-art techniques in terms of accuracy and robustness, especially in the context of factor reconstruction errors. The run time simulations show that the time complexity depends on the ranks and dimensionality of the rank-deficient mode, and the algorithm is faster than the state-of-the-art schemes for some scenarios. Moreover, our algorithm does not require multiple initializations or a known rank structure. However, it can still take advantage of the known ranks when available, which makes it more flexible than the existing techniques in the literature.

ACKNOWLEDGMENT

We sincerely thank the anonymous reviewers for their constructive comments, which greatly enhanced the quality of the article.

REFERENCES

[1] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000, doi: 10.1137/S0895479896305696.

[2] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, pp. 279–311, 1966.

[3] J. D. Carroll and J. J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.

[4] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-model factor analysis," *UCLA Work. Papers Phonetics*, vol. 16, pp. 1–84, Dec. 1970.

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

[6] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[7] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms—Part II: Definitions and uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1033–1066, 2008.

[8] L. De Lathauwer and D. Nion, "Decompositions of a higher-order tensor in block terms—Part III: Alternating least squares algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1067–1083, 2008.

[9] N. Vervliet, O. Debals, L. Sorber, M. van Barel, and L. De Lathauwer, "Tensorlab 3.0," Mar. 2016. [Online]. Available: https://www.tensorlab.net/

[10] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms–Part I: Lemmas for partitioned matrices," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1022–1032, 2008, doi: 10.1137/060661685.

[11] D. Nion and L. De Lathauwer, "A link between the decomposition of a third-order tensor in rank-(L,L,1) terms and joint block diagonalization," in *Proc. IEEE 3rd Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2009, pp. 89–92.

[12] L. de Lathauwer, "Blind separation of exponential polynomials and the decomposition of a tensor in Rank-$(L_r, L_r, 1)$ terms," *SIAM J. Matrix Anal. Appl.*, vol. 32, no. 4, pp. 1451–1474, 2011.

[13] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in Rank-$(L_r, L_r, 1)$ terms, and a new generalization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 695–720, 2013, doi: 10.1137/120868323.

[14] A. A. Rontogiannis, E. Kofidis, and P. V. Giampouras, "Block-term tensor decomposition: Model selection and computation," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 3, pp. 464–475, Apr. 2021.

[15] A. A. Rontogiannis, E. Kofidis, and P. V. Giampouras, "Block-term tensor decomposition: Model selection and computation," in *Proc. IEEE 28th Eur. Signal Process. Conf.*, 2021, pp. 1976–1980.

[16] A. A. Rontogiannis, P. V. Giampouras, and E. Kofidis, "Rank-revealing block-term decomposition for tensor completion," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 2915–2919.

[17] P. V. Giampouras, A. A. Rontogiannis, and E. Kofidis, "Block-term tensor decomposition model selection and computation: The bayesian way," *IEEE Trans. Signal Process.*, vol. 70, pp. 1704–1717, 2022.

[18] J. H. d. M. Goulart, P. M. R. de Oliveira, R. C. Farias, V. Zarzoso, and P. Comon, "Alternating group lasso for block-term tensor decomposition and application to ECG source separation," *IEEE Trans. Signal Process.*, vol. 68, pp. 2682–2696, 2020.

[19] I. Domanov and L. De Lathauwer, "On uniqueness and computation of the decomposition of a tensor into multilinear Rank-$(L_r, L_r, 1)$ terms," *SIAM J. Matrix Anal. Appl.*, vol. 41, no. 2, pp. 747–803, 2020.

[20] X. Han, L. Albera, A. Kachenoura, H. Shu, and L. Senhadji, "Block term decomposition with rank estimation using group sparsity," in *Proc. IEEE 7th Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2017, pp. 1–5.

[21] M. Sø rensen and L. Lathauwer, "Coupled canonical polyadic decompositions and (coupled) decompositions in multilinear rank-$(L_{r,n}, L_{r,n}, 1)$ terms-Part I: Uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 36, pp. 496–522, 2015.

[22] M. Sørensen, I. Domanov, and L. Lathauwer, "Coupled canonical polyadic decompositions and (coupled) decompositions in multilinear rank- $(L_{r,n}, L_{r,n}, 1)$ terms–Part II: Algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 36, pp. 1015–1045, Jan. 2015.

[23] L. Sorber, M. Van Barel, and L. De Lathauwer, "Structured data fusion," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 586–600, Jun. 2015.

[24] X.-F. Gong, Q.-H. Lin, O. Debals, N. Vervliet, and L. De Lathauwer, "Coupled rank-$(L_m, L_n, \bullet)$ block term decomposition by coupled block simultaneous generalized schur decomposition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 2554–2558.

[25] I. Safiullin, L. Khamidullina, A. A. Korobkov, and M. Haardt, "Enhanced computation of the coupled block-term decomposition in multilinear rank terms," in *Proc. IEEE 12th Sensor Array Multichannel Signal Process. Workshop*, 2022, pp. 400–404.

[26] Y. Cheng, M. Riesmeyer, J. Haueisen, and M. Haardt, "Using the multilinear Rank-$(L_r, L_r, 1)$ decomposition for the detection of the 200 hz band activity in somatosensory evoked magnetic fields and somatosensory evoked electrical potentials," *IEEE Access*, vol. 9, pp. 106232–106244, 2021.

[27] L. N. Ribeiro, A. R. Hidalgo-Muñoz, and V. Zarzoso, "Atrial signal extraction in atrial fibrillation electrocardiograms using a tensor decomposition approach," in *Proc. IEEE 37th Annu. Int. Conf. Eng. Med. Biol. Soc.*, 2015, pp. 6987–6990.

[28] P. Oliveira and V. Zarzoso, "Block term decomposition of ECG recordings for atrial fibrillation analysis: Temporal and inter-patient variability," *J. Commun. Inf. Syst.*, vol. 34, no. 1, pp. 111–119, 2019.

[29] L. N. Ribeiro, A. L. F. de Almeida, and V. Zarzoso, "Enhanced block term decomposition for atrial activity extraction in atrial fibrillation ECG," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop*, 2016, pp. 1–5.

[30] V. Zarzoso, "Parameter estimation in block term decomposition for noninvasive atrial fibrillation analysis," in *Proc. IEEE 7th Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2017, pp. 1–5.

[31] B. Hunyadi et al., "Block term decomposition for modelling epileptic seizures," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, pp. 1–19, 2014.

[32] L. De Lathauwer and A. de Baynast, "Blind deconvolution of DS-CDMA signals by means of decomposition in Rank-(1, l., l) terms," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1562–1571, Apr. 2008.

[33] J.-X. Yang, X.-F. Gong, H. Li, Y.-G. Xu, and Z.-W. Liu, "Using coupled multilinear rank-$(L, L, 1)$ block term decomposition in multi-static-multi-pulse MIMO radar to localize targets," in *Proc. Adv. Neural Netw.–ISNN*, H. Lu, H. Tang, and Z. Wang Eds. 2019, pp. 565–574.
[34] E. Gujral, R. Pasricha, and E. Papalexakis, "Beyond Rank-1: Discovering rich community structure in multi-aspect graphs," in *Proc. Web Conf.*, 2020, pp. 452–462.
[35] G. Zhang, X. Fu, K. Huang, and J. Wang, "Hyperspectral super-resolution: A coupled nonnegative block-term tensor decomposition approach," in *Proc. IEEE 8th Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, 2019, pp. 470–474.
[36] F. Xiong, Y. Qian, J. Zhou, and Y. Y. Tang, "Hyperspectral unmixing via total variation regularized nonnegativetensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2341–2357, Apr. 2019.
[37] G. Zhang, X. Fu, J. Wang, X.-L. Zhao, and M. Hong, "Spectrum cartography via coupled block-term tensor decomposition," *IEEE Trans. Signal Process.*, vol. 68, pp. 3660–3675, 2020.
[38] E. Gujral and E. E. Papalexakis, "OnlineBTD: Streaming algorithms to track the block term decomposition of large tensors," in *Proc. IEEE 7th Int. Conf. Data Sci. Adv. Analytics*, 2020, pp. 168–177.
[39] P. Tichavský, A.-H. Phan, and A. Cichocki, "Sensitivity in tensor decomposition," *IEEE Signal Process. Lett.*, vol. 26, no. 11, pp. 1653–1657, Nov. 2019.
[40] P. Tichavský, A.-H. Phan, and A. Cichocki, "Krylov-Levenberg-Marquardt algorithm for structured tucker tensor decompositions," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 3, pp. 550–559, Apr. 2021.
[41] F. Roemer and M. Haardt, "A semi-algebraic framework for approximate CP decompositions via simultaneous matrix diagonalizations (SECSI)," *Elsevier Signal Process.*, vol. 93, no. 9, pp. 2722–2738, Sep. 2013.
[42] A. Stegeman and N. D. Sidiropoulos, "On Kruskal's uniqueness condition for the candecomp/parafac decomposition," *Linear Algebra Appl.*, vol. 420, no. 2/3, pp. 540–552, 2007.
[43] M. de Vos et al., "Canonical decomposition of ictal scalp EEG reliably detects the seizure onset zone," *NeuroImage*, vol. 37, no. 3, pp. 844–854, 2007.
[44] A. A. Korobkov, M. K. Diugurova, J. Haueisen, and M. Haardt, "Multi-dimensional model order estimation using LineAr regression of global eigenvalues (LaRGE) with applications to EEG and MEG recordings," in *Proc. IEEE 28th Eur. Signal Process. Conf.*, 2021, pp. 1005–1009.
[45] J. P. C. L. da Costa, M. Haardt, F. Roemer, and G. Del Galdo, "Enhanced model order estimation using higher-order arrays," in *Proc. IEEE Conf. Rec. 41st Asilomar Conf. Signals, Syst. Comput.*, 2007, pp. 412–416.
[46] A. A. Korobkov, M. K. Diugurova, J. Haueisen, and M. Haardt, "Robust multi-dimensional model order estimation using LineAr regression of global eigenvalues (LaRGE)," *IEEE Trans. Signal Process.*, vol. 70, pp. 5751–5764, 2022.
[47] L. Lathauwer, "A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization," *SIAM J. Matrix Anal. Appl.*, vol. 28, no. 3, pp. 642–666, Jan. 2006.
[48] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Baltimore, MA, USA: Johns Hopkins Univ. Press, 2013.
[49] T. Fu and X. Gao, "Simultaneous diagonalization with similarity transformation for non-defective matrices," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. Proc.*, 2006, pp. 1137–1140.
[50] I. Domanov and L. De Lathauwer, "Canonical polyadic decomposition of third-order tensors: Relaxed uniqueness conditions and algebraic algorithm," *Linear Algebra Appl.*, vol. 513, pp. 342–375, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S002437951630492X
[51] C. Beltrán, P. Breiding, and N. Vannieuwenhoven, "Pencil-based algorithms for tensor rank decomposition are not stable," *SIAM J. Matrix Anal. Appl.*, vol. 40, no. 2, pp. 739–773, 2019, doi: 10.1137/18M1200531.
[52] P. Breiding and N. Vannieuwenhoven, "The condition number of join decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 39, no. 1, pp. 287–309, 2018, doi: 10.1137/17M1142880.
[53] E. Evert, M. Vandecappelle, and L. De Lathauwer, "A recursive eigenspace computation for the canonical polyadic decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 43, no. 1, pp. 274–300, 2022, doi: 10.1137/21M1423026.
[54] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.
[55] S. Zhu, D. Wang, and T. Li, "Data clustering with size constraints," *Knowl.-Based Syst.*, vol. 23, no. 8, pp. 883–889, 2010.
[56] J. Chen, J. Qiu, and C. Ahn, "Construction worker's awkward posture recognition through supervised motion tensor decomposition," *Automat. Construction*, vol. 77, pp. 67–81, 2017.
[57] Y. Shen, B. Baingana, and G. B. Giannakis, "Tensor decompositions for identifying directed graph topologies and tracking dynamic networks," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3675–3687, Jul. 2017.

Liana Khamidullina (Student Member, IEEE) received the bachelor's degree in telecommunications from Kazan National Research Technical University, Kazan, Russia, in 2016, and the M.Sc. degree in communications and signal processing from the German Russian Institute of Advanced Technologies, Kazan, organized between Kazan National Research Technical University, and Ilmenau University of Technology, Ilmenau, Germany, as a double degree program, in 2018. She is currently working toward the Ph.D. degree in communications and signal processing with the Ilmenau University of Technology. Since 2019, she has been a Research assistant with the Communications Research Laboratory, Ilmenau University of Technology. Her research interests include tensor-based signal processing, multi-user MIMO precoding, radar, direction of arrival estimation, and biomedical signal processing.

Gabriela Seidl received the bachelor's degree in electrical and information engineering from the Ilmenau University of Technology, Ilmenau, Germany, in 2021. She is currently working toward the Educational degree with the University of Leipzig, Leipzig, Germany, to become a Teacher of maths and physics (Gymnasium school form). During her studies, she was a Student Assistant with the IMMS Institute for Microelectronics and Mechatronics Systems, Ilmenau, and the Communications Research Laboratory, Ilmenau University of Technology. Her research interests include tensor-based signal processing and near-field source localization techniques with Ilmenau University of Technology. She changed careers after completing her bachelor's thesis. Her thesis discussed extending the SECSI-Framework for canonical polyadic decompositions (CPDs) to block-term decompositions. Her further research interests include science communication for adolescents, gamification of school classes, and modern school concepts.

Ivan Alexeevich Podkurkov received the B.E. degree in telecommunications from Kazan National Research Technical University - KAI (KNRTU-KAI, former Kazan Aviation Institute), Kazan, Russia, in 2014, and the M.Sc. degree in communications and signal processing from the German Russian Institute of Advanced Technologies (GRIAT), Kazan, organized between KNRTU-KAI, Russia, and TU Ilmenau (TUIL), Ilmenau, Germany, as a double degree program, in 2016. He is currently the Deputy Head of Electronics Department, ELEPS Ltd, Kazan. Then, he proceeded his research on a Double Ph.D. Program organised between the Department of Radio Electronic and Telecommunication Systems, KNRTU-KAI and Communications Research Laboratory, TUIL. From 2014 to 2016, he was an Engineer, and then a part-time Teaching Assistant with the Department of Radio Electronic and Telecommunication Systems of the Radio Electronic and Telecommunication Institute, KNRTU-KAI, from 2016 to 2018. From 2016 to 2021, he was an Electrical Engineer with ELEPS Ltd. and then since 2021, has been the Deputy Head of Electronics Department. His research interests include radar, direction of arrival estimation, statistical and tensor-based signal processing, and advanced non-Gaussian interference models.

Alexey Alexandrovich Korobkov received the diploma-engineer (M.S.) and doctor-engineer (Ph.D.) degrees in electrical engineering from Kazan National Research Technical University n.a. A.N. Tupolev-KAI (KNRTU-KAI), Kazan, Russia, in 1998 and 2002, respectively. In 2001, he was a Teaching Assistant with the Department of Radio Electronics and Telecommunication Systems, Institute for Radio-Electronics, Photonics and Digital Technologies, KNRTU-KAI, where since 2003, he has been an Associate Professor. His research interests include biomedical radio-electronic systems, statistical, array, and tensor-based signal processing, high-resolution parameter estimation, and advanced non-Gaussian interference models.

Martin Haardt (Fellow, IEEE) received the diplom-ingenieur (M.S.) degree in electrical engineering from the Ruhr-University Bochum, Germany, in 1991 and the doktor-ingenieur (Ph.D.) degree from Munich University of Technology, Germany, in 1996. During his diploma studies he also studied as a visiting scholar at Purdue University, USA. Since 2001, he has been a Full Professor with the Department of Electrical Engineering and Information Technology and the Head of the Communications Research Laboratory, Ilmenau University of Technology, Ilmenau, Germany. In 1997, he joined Siemens Mobile Networks, Munich, where he was responsible for strategic research for third generation mobile radio systems. From 1998 to 2001, he was the Director for International Projects and University Cooperations in the mobile infrastructure business of Siemens, Munich, where his work focused on mobile communications beyond the third generation. His research interests include wireless communications, array signal processing, high-resolution parameter estimation, and tensor-based signal processing. He was the recipient of the 2009 Best Paper Award from the IEEE Signal Processing Society, Vodafone Innovations Award for outstanding research in mobile communications, ITG best paper award from the Association of Electrical Engineering, Electronics, and Information Technology (VDE), and Rohde & Schwarz Outstanding Dissertation Award. Prof. Haardt has been the Senior Editor of the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING since 2019, was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING during 2002–2006 and 2011–2015, IEEE SIGNAL PROCESSING LETTERSduring 2006–2010, *Research Letters in Signal Processing* during 2007–2009, *EURASIP Signal Processing Journal* during 2011–2014, and the Guest Editor of the *EURASIP Journal on Wireless Communications and Networking*. Since 2009, he has been an Associate Editor for the *Hindawi Journal of Electrical and Computer Engineering*. From 2011 to 2019, he was an elected Member of the Sensor Array and Multichannel Technical Committee of the IEEE Signal Processing Society, where he was the Vice Chair during 2015–2016, Chair during 2017–2018, and has been the Past Chair since 2019. Since 2020, he has been an elected Member of the Signal Processing Theory and Methods Technical Committee of the IEEE Signal Processing Society. Moreover, he was the Technical Co-Chair of PIMRC 2005 in Berlin, Germany, ISWCS 2010 in York, U.K., European Wireless 2014 in Barcelona, Spain, and Asilomar Conference on Signals, Systems, and Computers 2018, USA, and the General Co-Chair of WSA 2013 in Stuttgart, Germany, ISWCS 2013 in Ilmenau, Germany, CAMSAP 2013 in Saint Martin, French Antilles, WSA 2015 in Ilmenau, SAM 2016 in Rio de Janeiro, Brazil, CAMSAP 2017 in Curacao, Dutch Antilles, SAM 2020 in Hangzhou, China, Asilomar Conference on Signals, Systems, and Computers 2021, USA, and CAMSAP 2023 in Costa Rica.