Block-Term Tensor Decomposition: Model Selection and Computation

Athanasios A. Rontogiannis, *Member, IEEE*, Eleftherios Kofidis, *Member, IEEE*, and Paris V. Giampouras

***Abstract*—The so-called block-term decomposition (BTD) tensor model has been recently receiving increasing attention due to its enhanced ability of representing systems and signals that are composed of *blocks* of rank higher than one, a scenario encountered in numerous and diverse applications. Its uniqueness and approximation have thus been thoroughly studied. Nevertheless, the challenging problem of estimating the BTD model structure, namely the number of block terms and their individual ranks, has only recently started to attract significant attention. In this paper, a novel method of BTD model selection and computation is proposed, based on the idea of imposing column sparsity *jointly* on the factors and in a *hierarchical* manner and estimating the ranks as the numbers of factor columns of non-negligible magnitude. Following a block successive upper bound minimization (BSUM) approach for the proposed optimization problem is shown to result in an alternating hierarchical iteratively reweighted least squares (HIRLS) algorithm, which is fast converging and enjoys high computational efficiency, as it relies in its iterations on small-sized sub-problems with closed-form solutions. Simulation results for both synthetic examples and a hyper-spectral image denoising application are reported, which demonstrate the superiority of the proposed scheme over the state-of-the-art in terms of success rate in rank estimation as well as computation time and rate of convergence while attaining a comparable tensor approximation performance.**

***Index Terms*—Alternating group lasso (AGL), alternating least squares (ALS), block coordinate descent (BCD), block successive upper bound minimization (BSUM), block-term tensor decomposition (BTD), hierarchical iterative reweighted least squares (HIRLS), rank, tensor.**

I. Introduction

BLOCK-TERM Decomposition (BTD) was introduced in [1] as a tensor model that combines the Canonical Polyadic Decomposition (CPD) and the Tucker decomposition (TD), in the sense that it decomposes a tensor in a sum of tensors that have low multilinear rank (instead of rank one as in CPD[1]).

Manuscript received July 1, 2020; revised November 23, 2020; accepted January 8, 2021. Date of publication January 13, 2021; date of current version March 29, 2021. The work of Paris V. Giampouras was supported by the European Union Horizon 2020 Marie Skłodowska-Curie Global Fellowship program: HyPPOCRATES-H2020-MSCA-IF-2018 under Grant Agreement 844290. The guest editor coordinating the review of this manuscript and approving it for publication was Prof. Lina J. Karam. *(Corresponding author: Eleftherios Kofidis.)*

Athanasios A. Rontogiannis is with the Institute for Astronomy, Astrophysics, Space Applications and Remote Sensing, National Observatory of Athens, Athens, 15236 Penteli, Greece (e-mail: tronto@noa.gr).

Eleftherios Kofidis is with the Department of Statistics and Insurance Science, University of Piraeus, 18534 Piraeus, Greece, and also with the Computer Technology Institute & Press "Diophantus" (CTI), 26504 Patras, Greece (e-mail: kofidis@unipi.gr).

Paris V. Giampouras is with the Mathematical Institute for Data Science, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: parisg@jhu.edu).

Digital Object Identifier 10.1109/JSTSP.2021.3051488

[1]Note that a rank-1 tensor is also a rank-$(1, 1, \ldots, 1)$ tensor.

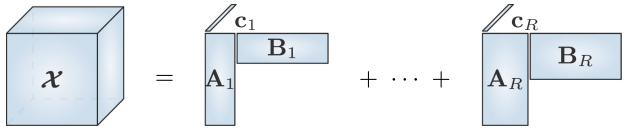


Fig. 1. Rank-$(L_r, L_r, 1)$ block-term decomposition.

In other words, BTD is a sum of TDs (block terms). Hence a BTD can be seen as a *constrained* TD, with its core tensor being block diagonal (see [1, Fig. 2.3]). Given the sum-of-TDs structure of BTD and in view of the fact that CPD is also a constrained TD [2], BTD can also be seen as a *constrained* CPD having factors with (some) collinear columns [1]. In a way, BTD lies between the two extremes (in terms of core tensor structure), CPD and TD, and it is interesting to recall the related remark made in [1], namely that ""the" rank of a higher-order tensor is actually a combination of the two aspects: one should specify the number of blocks *and* their size." Accurately and efficiently estimating these numbers for a given tensor is the main subject of this work.

Although [1] introduced BTD as a sum of R rank-(L_r, M_r, N_r) terms $(r = 1, 2, \ldots, R)$ in general, the special case of rank-$(L_r, L_r, 1)$ BTD has attracted a lot more of attention, because of both its more frequent occurrence in applications and the existence of more concrete and easier to check uniqueness conditions. This paper will also focus on this special yet very popular BTD model. Consider a 3rd-order tensor, $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I\times J\times K}$. Then its rank-$(L_r, L_r, 1)$ decomposition is written as

$$\boldsymbol{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{E}_r \circ \mathbf{c}_r, \tag{1}$$

where $\mathbf{E}_r$ is an $I \times J$ matrix of rank L_r, $\mathbf{c}_r$ is a nonzero column K-vector and $\circ$ denotes outer product. Clearly, $\mathbf{E}_r$ can be written as a matrix product $\mathbf{A}_r\mathbf{B}_r^{\mathrm{T}}$ with the matrices $\mathbf{A}_r \in \mathbb{C}^{I\times L_r}$ and $\mathbf{B}_r \in \mathbb{C}^{J\times L_r}$ being of full column rank, L_r. A schematic diagram of the rank-$(L_r, L_r, 1)$ BTD is shown in Fig. 1.

BTD has found applications in communications (e.g., [3], [4]), neuro- and anatomical imaging [5]–[8], electrocardiography (ECG) (e.g., [9]–[11]), hyper-spectral imaging (HSI) [12]–[15], community detection in networks [16], spectrum cartography [17], and electron microscopy [18], among others. Recently it has also been proposed as a compact model of neural networks in modern machine learning applications [19], [20]. The application of BTD in blind source separation (BSS) was first considered in [21] and later presented in more detail in [22],

giving rise to the so-called Block Component Analysis (BCA) approach. The underlying idea is that BTD can better represent components (sources) of a variable complexity (hence rank), while CPD-based BSS[2] restricts the sources to have rank one.[3]

The uniqueness of BTD was studied in [1], also for the general rank-(L_r, M_r, N_r) case. Essential uniqueness for the rank-$(L_r, L_r, 1)$ BTD of eq. (1) means that the only indeterminacies are the order of the R terms and a scaling of the $\mathbf{E}_r$ matrix with a counter-scaling of the vector $\mathbf{c}_r$. The most popular (though not the only one) uniqueness theorem for this case states that a *sufficient* uniqueness condition is that the partitioned matrices $\mathbf{A} \triangleq [\,\mathbf{A}_1\ \mathbf{A}_2\ \cdots\ \mathbf{A}_R\,]$ and $\mathbf{B} \triangleq [\,\mathbf{B}_1\ \mathbf{B}_2\ \cdots\ \mathbf{B}_R\,]$ are of full column rank and $\mathbf{C} \triangleq [\,\mathbf{c}_1\ \mathbf{c}_2\ \cdots\ \mathbf{c}_R\,]$ does not have any collinear columns [1, Theorem 4.1]. The generic version of the requirement for full column rank of $\mathbf{A}, \mathbf{B}$ is that $\min(I, J) \geq \sum_{r=1}^{R} L_r$, which can easily be met in applications where R and L_r are small. It should however be noted that this is not a necessary condition as our simulation results also demonstrate.

Alternating least squares (ALS) was extended to the computation of a tensor BTD in [23]. In that same work, it was also shown (and demonstrated through an example) that degeneracy can also occur for BTD.[4] In the noise-free case, and as shown in [1, Theorem 4.1], the BTD can be also computed with the aid of a generalized eigenvalue decomposition (GEVD), provided the above uniqueness condition is satisfied. Recently, algebraic solution methods that are free from this limitation have been also reported [25]–[27]. In the presence of noise, these solutions can serve to initialize the ALS iterations [23]. ALS with the appropriate modifications to incorporate the non-negativity constraint was used in [12] for non-negative BTD of hyper-spectral imagery. Non-alternating (all-at-once) computation approaches, including gradient descent and nonlinear least squares, were followed in [28] and the resulting methods are implemented in Tensorlab [29]. Additional methods of BTD computation include ALS regularized through ℓ_2 norms of its factors (to avoid over-fitting [17] or to enforce low rank [13]) or through proximal point modifications [30], deflation-based [31], variable projection using Riemannian gradient for rank-(L_r, M_r, N_r) BTD with factors of orthonormal columns [32], tensor block diagonalization [33], solving the equivalent matrix factorization problem with one of the factors constrained to have low-rank rows [34], and computing an appropriately constrained coupled CPD [27].

In most of the BTD methods mentioned above, R and L_r, $r = 1, 2, \ldots, R$ are assumed known (and it is commonly assumed that all L_r are all equal to L, for simplicity). In fact, in practice, this is a challenging question on its own. Unless external information is given (such as in a telecommunications [22] or in a HSI unmixing application with given or estimated ground truth [12]), there is no way to know these values a priori. An observation that is common in all known BCA applications is that the separation performance does not strongly depend on the particular values of the L_r ranks [6], [12], [17]. In fact, as it was also observed in [5], [6], the method is robust to overestimation of L_r (although, as observed in [8], performance of BTD-based classifiers may considerably vary with L_r). Nonetheless, one should try not to set L_r to a very high value. The reason is that, in addition to increasing the computational complexity, setting L_r too high may hinder interpretation of the results through letting noise/artifact sources interfere with the desired sources [5]. This holds for R as well, although its choice is known to be more crucial to the obtained performance. For example, setting R too high in [5] results in source splitting (also referred to as over-factoring [35]), thus compromising the separation and interpretation of the components.

A. Related Work

Model order selection techniques for BTD can be dictated by corresponding CPD techniques, as reviewed in [5, Section 4], including clustering similar CPD components (e.g., [33]). Schemes of multilinear rank estimation (largely based on matrix rank estimation and/or extensions of one-dimensional information-theoretic criteria) are also relevant in view of the constrained TD structure of BTD [7], [36]–[39]. In the absence of noise, the model rank parameters can be computed as a by-product of recently reported algebraic BTD methods [25], [26]. Thus, in the non-iterative method of [26], and in the (almost) noise-free case, these are estimated (with the aid of singular value decompositions (SVDs)) from a joint block matrix diagonalization problem. For noisier tensors, R and $\sum_r L_r$ are assumed known.

Model order selection can also be application-specific. For example, L_r's are estimated in [9] as the auto-regressive (AR) orders of the sources in ECG analysis, with R assumed known. In [6], and in the functional magnetic resonance imaging (fMRI) context, L_r is estimated as the number of statistically significant (bearing useful information) columns of $\mathbf{A}_r, \mathbf{B}_r$. [13] relies on the subspace-based method of [40] for estimating the number R of spectral signatures in BTD-based HSI de-noising.

Alternative techniques rely on sparsity arguments for model selection. A greedy scheme, inspired from a sparse coding viewpoint, is proposed in [41], for more general tensor decompositions [42] including BTD as a special case. Instead of building the model incrementally, however, one can follow the reverse way of starting from a rank *overestimate* and arrive at the true rank(s) by eliminating negligible components, aided in this task by appropriate regularization. Such an approach is followed in [35], [43], where the constrained CPD formulation of BTD is taken advantage of to first estimate R and then L_r's assumed all equal, before computing the model factors in (1). In each case, a regularization term is added to the tensor approximation cost, which is composed of mixed norms of the factor matrices and serves as upper bound on the tensor nuclear norm thus promoting column sparsity of the factors and hence low rank. The augmented Lagrangian method is adopted for the computations.

Nevertheless, as demonstrated in [44], [45] for the CPD case, the problems of model rank estimation and approximation of

[2]Also referred to as Canonical Polyadic Analysis (CPA) [22].

[3]An intuitively pleasant way to describe this difference is to say that, while CPA decomposes the data into "atoms," BCA decomposes it into "molecules" [22].

[4]That a best BTD approximation of given ranks may not exist for a real-valued tensor was later shown in [24]. This is not the case for tensors in $\mathbb{C}$, however (cf. [24] and references therein).

factors can be addressed *jointly*, with significant gains in both accuracy and complexity (of particular interest for big data applications). This idea is proposed in [46] for the rank-$(L_r, L_r, 1)$ BTD model with not necessarily all equal block-term ranks L_r. A regularization term consisting of the sum of the mixed $\ell_{1,2}$ norms of the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ is added to the squared error of the tensor approximation, namely

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \frac{1}{2} \left\| \boldsymbol{\mathcal{Y}} - \sum_{r=1}^{R} \mathbf{A}_r \mathbf{B}_r^{\mathrm{T}} \circ \mathbf{c}_r \right\|_{\mathrm{F}}^2 + \gamma(\|\mathbf{A}\|_{1,2} + \|\mathbf{B}\|_{1,2} + \|\mathbf{C}\|_{1,2}), \quad (2)$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm, $\|\cdot\|_{1,2}$ denotes the mixed $\ell_{1,2}$ norm (defined as the ℓ_1 norm of the ℓ_2 norms of the matrix columns[5]), and γ is the regularization parameter weighing the regularization term over the data fidelity term. This sparsity-inducing regularization helps promoting low rank for the BTD factors and hence estimating R (as the number of non-zero columns of $\mathbf{C}$) and L_r's (as the number of non-zero columns of the rth blocks of $\mathbf{A}, \mathbf{B}$ that correspond to non-zero columns of $\mathbf{C}$). For the solution of (2), a proximal term is first added in [46] and then a block coordinate descent (BCD) approach is taken, leading to a regularized version of the ALS procedure of [23] that will be referred to henceforth as the BTD alternating group lasso (BTD-AGL) algorithm.

B. Our Contribution

The approach we propose in this paper also falls in the previous category. Yet, it has a number of very important new features, inherited from our earlier work on factorization-based low-rank approximation of matrices [47], [48], [49], from which it draws inspiration. In [49], the sum of reweighted Frobenius norms of the factors of the data matrix is used as regularization and, in particular, a diagonal weighting, jointly depending on the factors, is proposed, naturally leading to an iteratively reweighted least squares (IRLS) [50] solution approach, with fast convergence and low complexity. Here we generalize that idea in the BTD problem. The regularization of [49] is employed, in two levels: first, combining the reweighted norms of $\mathbf{A}$ and $\mathbf{B}$, and second, coupling these with the reweighted norm of $\mathbf{C}$. This two-level coupling naturally matches the structure of the model in (1), making explicit the different roles of $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$, in contrast to previous related works [35], [46] that miss to exploit this relation. Furthermore, due to this fact, the regularization proposed here has a stronger sparsity promoting action compared with previous works. Applying majorization with appropriate upper bounds and a BCD approach results in an alternating hierarchical IRLS (HIRLS) algorithm that manages to both reveal the ranks and compute the BTD factors at a high convergence rate and low computational cost. Notably, iterations involve closed-form updates that contain only matrix-matrix multiplications, which can be efficiently implemented on most modern computer systems and are easily parallelizable. The complexity can be reduced even more by eliminating negligible columns (column pruning) in the course of the iterations (as in [49] for the low-rank matrix factorization problem). Simulation results for synthetic examples are reported, which demonstrate the superiority of the proposed scheme over the state-of-the-art in terms of success rate in rank estimation as well as computation time and rate of convergence. An HSI de-noising example is also studied, which shows the proposed method to attain a performance comparable with that of BTD-AGL, albeit at a much shorter run-time.

A short version of the present work can be found in our recent conference paper [51]. Additions to that version in the present paper include a much more detailed presentation of the related literature, and of the proposed approach and method, as well as derivations, convergence proofs, and complexity estimates that are deferred to the appendices. More extensive simulation results, including results from a de-noising application, are also reported here.

C. Organization of the Paper

The rest of this paper is organized as follows. The adopted notation is described in the following subsection. The problem is mathematically stated in Section II, where the proposed regularization approach is also detailed. The proposed solution method is developed and presented in Section III. Section IV reports and discusses the simulation results. Conclusions are drawn and future work plans are outlined in Section V. Derivations and proofs are deferred to the appendices.

D. Notation

Lower- and upper-case bold letters are used to denote vectors and matrices, respectively. Higher-order tensors are denoted by upper-case bold calligraphic letters. For a tensor $\boldsymbol{\mathcal{X}}$, $\mathbf{X}_{(n)}$ stands for its mode-n unfolding. $*$ stands for the Hadamard product and $\otimes$ for the Kronecker product. The Khatri-Rao product is denoted by $\odot$ in its general (partition-wise) version and by $\odot_{\mathrm{c}}$ in its column-wise version. $\circ$ denotes the outer product. The superscript $^{\mathrm{T}}$ stands for transposition. The identity matrix of order N and the all ones $M \times N$ matrix are respectively denoted by $\mathbf{I}_N$ and $\mathbf{1}_{M\times N}$. $\mathbf{1}_N$ stands for $\mathbf{1}_{N\times 1}$. The row vectorization and the trace of a matrix $\mathbf{X}$ are denoted by $\mathrm{vec}(\mathbf{X})$ and $\mathrm{tr}(\mathbf{X})$, respectively. $\nabla_{\mathbf{X}}$ stands for the gradient operator with respect to (w.r.t) $\mathbf{X}$. $\mathrm{diag}(\mathbf{x})$ is the diagonal matrix with the vector $\mathbf{x}$ on its main diagonal. The Euclidean vector norm and the Frobenius matrix and tensor norms are denoted by $\|\cdot\|_2$ and $\|\cdot\|_{\mathrm{F}}$, respectively. The mixed 1, 2 ($\ell_{1,2}$) norm of a matrix $\mathbf{X} = [\,\mathbf{x}_1 \cdots \mathbf{x}_N\,]$ is defined as $\sum_{n=1}^{N} \|\mathbf{x}_i\|_2$. $\mathbb{C}$ is the field of complex numbers.

II. Problem Statement and Proposed Approach

Given an $I \times J \times K$ tensor $\boldsymbol{\mathcal{Y}}$, its best (in the least squares sense) rank-$(L_r, L_r, 1)$ approximation is sought for, namely

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} f(\mathbf{A},\mathbf{B},\mathbf{C}) \triangleq \frac{1}{2} \left\| \boldsymbol{\mathcal{Y}} - \sum_{r=1}^{R} \mathbf{A}_r \mathbf{B}_r^{\mathrm{T}} \circ \mathbf{c}_r \right\|_{\mathrm{F}}^2, \quad (3)$$

where the matrices $\mathbf{A}_r = [\,\mathbf{a}_{r1}\ \mathbf{a}_{r2} \cdots \mathbf{a}_{rL_r}\,] \in \mathbb{C}^{I\times L_r}$, $\mathbf{B}_r = [\,\mathbf{b}_{r1}\ \mathbf{b}_{r2} \cdots \mathbf{b}_{rL_r}\,] \in \mathbb{C}^{J\times L_r}$, $\mathbf{C} \in \mathbb{C}^{K\times R}$, and the ranks R and L_r, $r = 1, 2, \ldots, R$ are *a-priori* unknown. In terms of its

[5] In [46] this is referred to as the $\ell_{2,1}$ norm.

mode unfoldings $\mathbf{X}_{(1)} \in \mathbb{C}^{I\times JK}$, $\mathbf{X}_{(2)} \in \mathbb{C}^{J\times IK}$ and $\mathbf{X}_{(3)} \in \mathbb{C}^{K\times IJ}$, the tensor $\boldsymbol{\mathcal{X}} \triangleq \sum_{r=1}^{R} \mathbf{A}_r\mathbf{B}_r^{\mathrm{T}} \circ \mathbf{c}_r$ can be written as [1]

$$\mathbf{X}_{(1)}^{\mathrm{T}} = (\mathbf{B} \odot \mathbf{C})\mathbf{A}^{\mathrm{T}}, \tag{4}$$

$$\mathbf{X}_{(2)}^{\mathrm{T}} = (\mathbf{C} \odot \mathbf{A})\mathbf{B}^{\mathrm{T}}, \tag{5}$$

$$\mathbf{X}_{(3)}^{\mathrm{T}} = \left[(\mathbf{A}_1 \odot_{\mathrm{c}} \mathbf{B}_1)\mathbf{1}_{L_1} \cdots (\mathbf{A}_R \odot_{\mathrm{c}} \mathbf{B}_R)\mathbf{1}_{L_R}\right]\mathbf{C}^{\mathrm{T}}. \tag{6}$$

These expressions can be used in alternatingly solving for $\mathbf{A}, \mathbf{B}, \mathbf{C}$, respectively.

The regularization-based approach adds terms to the objective function above with the aim of imposing constraints on the sought factors, as in (2) for example. However, in contrast to (2), where all BTD factors are treated in the same manner, the regularizer proposed in this paper perfectly matches the structure of the BTD model, offering increased flexibility via a suitable joint block and column sparsity promoting mechanism of a hierarchical nature. The proposed modification to (3) can be stated as

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} f(\mathbf{A},\mathbf{B},\mathbf{C}) + \lambda\|\mathbf{F}(\mathbf{A},\mathbf{B},\mathbf{C})\|_{1,2}, \tag{7}$$

where $\lambda > 0$ is a parameter to be selected. Regularization is performed with the aid of the $\ell_{1,2}$ norm of the $2 \times R$ matrix $\mathbf{F}(\mathbf{A},\mathbf{B},\mathbf{C})$, constructed as follows. Let $\mathbf{G} \triangleq [\,\mathbf{A}^{\mathrm{T}}\ \mathbf{B}^{\mathrm{T}}\,]^{\mathrm{T}}$ be the $(I+J)\times\sum_{r=1}^{R} L_r$ matrix resulting from stacking the factors $\mathbf{A}$ and $\mathbf{B}$ and $\mathbf{G}_r \triangleq [\,\mathbf{A}_r^{\mathrm{T}}\ \mathbf{B}_r^{\mathrm{T}}\,]^{\mathrm{T}}$ denote its rth $(I+J)\times L_r$ block. The matrix $\mathbf{F}(\mathbf{A},\mathbf{B},\mathbf{C})$ is defined as

$$\mathbf{F}(\mathbf{A},\mathbf{B},\mathbf{C}) \triangleq \begin{bmatrix} \|\mathbf{G}_1\|_{1,2} & \|\mathbf{G}_2\|_{1,2} & \cdots & \|\mathbf{G}_R\|_{1,2} \\ \|\mathbf{c}_1\|_2 & \|\mathbf{c}_2\|_2 & \cdots & \|\mathbf{c}_R\|_2 \end{bmatrix}. \tag{8}$$

The minimization of the $\ell_{1,2}$ norm of a vector or matrix subject to a data proximity criterion has been widely utilized in the literature for enforcing group sparsity in vector/matrix recovery problems [52]. This property of the $\ell_{1,2}$ norm was exploited in our earlier work [49], [53] for model order selection in low-rank matrix factorization applications. In the present work, we extend that idea to the BTD problem by employing a two-level hierarchical $\ell_{1,2}$ norm-based regularization scheme. At the upper level, the $\ell_{1,2}$ norm of the matrix $\mathbf{F}(\mathbf{A},\mathbf{B},\mathbf{C})$ above promotes the elimination of whole blocks of $\mathbf{A}$ and $\mathbf{B}$ (which are tied together by the mixed norms $\|\mathbf{G}_r\|_{1,2}$, $r = 1, 2, \ldots, R$) and the corresponding columns of $\mathbf{C}$. At the lower level, the $\ell_{1,2}$ norms $\|\mathbf{G}_r\|_{1,2}$ induce column sparsity to the "surviving" blocks of $\mathbf{A}, \mathbf{B}$. Hence, we have the flexibility to overestimate the ranks R and L_r, $r = 1, 2, \ldots, R$ as $R = R_{\mathrm{ini}}$ and $L_r = L_{\mathrm{ini}}$ in the unknown BTD model, since this regularization can reduce them towards their actual values with a proper selection of the regularization parameter λ. The problem in (7) may be solved using a block successive upper bound minimization (BSUM) approach [54], as described in the next section. As explained in Appendix D, the resulting algorithm is an alternating hierarchical IRLS scheme, referred to henceforth as *BTD-HIRLS*.

III. Proposed Method

First, we rewrite the minimization problem (7) more explicitly in terms of the BTD factors $\mathbf{A}, \mathbf{B}$, and $\mathbf{C}$ as

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \frac{1}{2}\left\|\boldsymbol{\mathcal{Y}} - \sum_{r=1}^{R} \mathbf{A}_r\mathbf{B}_r^{\mathrm{T}} \circ \mathbf{c}_r\right\|_{\mathrm{F}}^2 + \lambda\sum_{r=1}^{R}\sqrt{\left(\sum\nolimits_{l=1}^{L}\sqrt{\|\mathbf{a}_{rl}\|_2^2 + \|\mathbf{b}_{rl}\|_2^2 + \eta^2}\right)^2 + \|\mathbf{c}_r\|_2^2 + \eta^2}, \tag{9}$$

where η^2 is a very small positive constant that ensures smoothness and R and L here stand for the initial *(over)estimates* of the model rank parameters. It can be shown that the objective function in (9) is convex w.r.t. each one of the factors $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ separately but not w.r.t. all of them. Moreover, due to the regularization term, it is non-separable w.r.t to each one of the matrix factors. As a result, minimizing the objective function in (9) alternatingly w.r.t. the BTD factors (i.e., in a BCD way with blocks the matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$) would not lead to closed-form solutions, which are desirable in an iterative algorithm. Capitalizing on our previous work on low-rank matrix factorization [49], we curb that problem by following a BSUM approach for the objective function in (9). The idea is that at each iteration of the BSUM scheme the BTD factors can be computed in *closed form* by minimizing appropriate upper bound functions of their initial objectives. Provided that these functions satisfy certain conditions [54], the BSUM procedure is guaranteed to converge to stationary points of the objective function of the original minimization problem.

To be more specific, using the mode-1 unfolding of $\boldsymbol{\mathcal{Y}}$ in (9) and $\boldsymbol{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{A}_r\mathbf{B}_r^{\mathrm{T}} \circ \mathbf{c}_r$ (cf. (4)), the objective function w.r.t. $\mathbf{A}$ at iteration k may be expressed as follows

$$f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}^k,\mathbf{C}^k) = \frac{1}{2}\left\|\mathbf{Y}_{(1)}^{\mathrm{T}} - \mathbf{P}^k\mathbf{A}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + \lambda\sum_{r=1}^{R}\sqrt{\left(\sum\nolimits_{l=1}^{L}\sqrt{\|\mathbf{a}_{rl}\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2}\right)^2 + \|\mathbf{c}_r^k\|_2^2 + \eta^2}, \tag{10}$$

where the $JK \times LR$ matrix $\mathbf{P}^k$ is defined as $\mathbf{P}^k \triangleq \mathbf{B}^k \odot \mathbf{C}^k$. To allow this sub-problem to have closed-form solution for $\mathbf{A}$, we define a local tight upper bound function of (10) as a rough second-order Taylor approximation of $f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}^k,\mathbf{C}^k)$ around $\mathbf{A}^k$. Namely:

$$g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k,\mathbf{B}^k,\mathbf{C}^k) = f_{\mathbf{A}}(\mathbf{A}^k|\mathbf{B}^k,\mathbf{C}^k) + \mathrm{tr}\{(\mathbf{A}-\mathbf{A}^k)\nabla_{\mathbf{A}} f_{\mathbf{A}}(\mathbf{A}^k|\mathbf{B}^k,\mathbf{C}^k)\} + \frac{1}{2}\mathrm{vec}(\mathbf{A}-\mathbf{A}^k)^{\mathrm{T}}\bar{\mathbf{H}}_{\mathbf{A}^k}\mathrm{vec}(\mathbf{A}-\mathbf{A}^k), \tag{11}$$

where the $ILR \times ILR$ approximate Hessian matrix $\bar{\mathbf{H}}_{\mathbf{A}^k}$ of $f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}^k,\mathbf{C}^k)$ at $\mathbf{A}^k$ is given (in analogy with [49]) by

$$\bar{\mathbf{H}}_{\mathbf{A}^k} = \mathbf{I}_I \otimes (\mathbf{P}^{k\mathrm{T}}\mathbf{P}^k + \lambda\mathbf{D}^k), \tag{12}$$

with $\mathbf{D}^k \triangleq (\mathbf{D}_1^k \otimes \mathbf{I}_L)\mathbf{D}_2^k$. $\mathbf{D}_1^k$ is an $R \times R$ diagonal matrix, whose rth diagonal entry is

$$\mathbf{D}_1^k(r,r) = \left[\left(\sum_{l=1}^{L} \sqrt{\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2} \right)^2 + \|\mathbf{c}_r^k\|_2^2 + \eta^2 \right]^{-1/2} \tag{13}$$

and $\mathbf{D}_2^k$ is an $RL \times RL$ diagonal matrix, whose $((r-1)L + l)$th diagonal entry is

$$\mathbf{D}_2^k((r-1)L + l, (r-1)L + l) = (\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2)^{-1/2}. \tag{14}$$

We can see from (12) that the approximate Hessian $\bar{\mathbf{H}}_{\mathbf{A}}$ is a positive definite block diagonal matrix with I identical $RL \times RL$ blocks on its main diagonal. Its relation with the exact Hessian $\mathbf{H}_{\mathbf{A}}$, which is a full yet structured $IRL \times IRL$ matrix, is clarified in Appendix A. Therein, it is also proved that the matrix $\bar{\mathbf{H}}_{\mathbf{A}} - \mathbf{H}_{\mathbf{A}}$ is positive semi-definite and thus the conditions of BSUM are satisfied by the majorization function in (11). In addition, as shown in Appendix B, minimizing $g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$ w.r.t. $\mathbf{A}$ results in the following analytical expression for the estimate of $\mathbf{A}$ at iteration $k+1$:

$$\mathbf{A}^{k+1} = \mathbf{Y}_{(1)}\mathbf{P}^k(\mathbf{P}^{k\mathrm{T}}\mathbf{P}^k + \lambda\mathbf{D}^k)^{-1}. \tag{15}$$

Similarly, $\mathbf{B}^{k+1}$ can be found from the minimization of $g_{\mathbf{B}}(\mathbf{B}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$, which has an analogous form with (11) with $\bar{\mathbf{H}}_{\mathbf{B}^k} = \mathbf{I}_J \otimes (\mathbf{Q}^{k\mathrm{T}}\mathbf{Q}^k + \lambda\mathbf{D}^k)$ and is a tight upper bound around $\mathbf{B}^k$ of

$$f_{\mathbf{B}}(\mathbf{B}|\mathbf{A}^k, \mathbf{C}^k) = \frac{1}{2}\left\|\mathbf{Y}_{(2)}^{\mathrm{T}} - \mathbf{Q}^k\mathbf{B}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + \lambda \sum_{r=1}^{R} \sqrt{\left(\sum\nolimits_{l=1}^{L} \sqrt{\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}\|_2^2 + \eta^2} \right)^2 + \|\mathbf{c}_r^k\|_2^2 + \eta^2}, \tag{16}$$

with (cf. (5)) the $IK \times LR$ matrix $\mathbf{Q}^k$ being defined as $\mathbf{Q}^k \triangleq \mathbf{C}^k \odot \mathbf{A}^k$. The unique solution of $\min_{\mathbf{B}} g_{\mathbf{B}}(\mathbf{B}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$ is given by

$$\mathbf{B}^{k+1} = \mathbf{Y}_{(2)}\mathbf{Q}^k(\mathbf{Q}^{k\mathrm{T}}\mathbf{Q}^k + \lambda\mathbf{D}^k)^{-1}. \tag{17}$$

Finally, the objective function w.r.t. $\mathbf{C}$ may be expressed as

$$f_{\mathbf{C}}(\mathbf{C}|\mathbf{A}^k, \mathbf{B}^k) = \frac{1}{2}\left\|\mathbf{Y}_{(3)}^{\mathrm{T}} - \mathbf{S}^k\mathbf{C}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + \lambda \sum_{r=1}^{R} \sqrt{\left(\sum\nolimits_{l=1}^{L} \sqrt{\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2} \right)^2 + \|\mathbf{c}_r\|_2^2 + \eta^2}, \tag{18}$$

where (cf. (6)) the $IJ \times R$ matrix $\mathbf{S}^k$ is given by

$$\mathbf{S}^k \triangleq \left[(\mathbf{A}_1^k \odot_{\mathrm{c}} \mathbf{B}_1^k)\mathbf{1}_L \cdots (\mathbf{A}_R^k \odot_{\mathrm{c}} \mathbf{B}_R^k)\mathbf{1}_L \right].$$

The factor $\mathbf{C}^{k+1}$ is found from $\min_{\mathbf{C}} g_{\mathbf{C}}(\mathbf{C}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$ as

$$\mathbf{C}^{k+1} = \mathbf{Y}_{(3)}\mathbf{S}^k(\mathbf{S}^{k\mathrm{T}}\mathbf{S}^k + \lambda\mathbf{D}_1^k)^{-1}, \tag{19}$$

Algorithm 1: BTD-HIRLS Algorithm.

Input: $\mathcal{Y}, \lambda, R_{\mathrm{ini}}, L_{\mathrm{ini}}$
Output: BTD model of $\mathcal{Y}$
Initialize: $k = 0, \mathbf{A}^0, \mathbf{B}^0, \mathbf{C}^0$
repeat
  Compute $\mathbf{D}_1^k, \mathbf{D}_2^k$ from (13) and (14)
  $\mathbf{D}^k \leftarrow (\mathbf{D}_1^k \otimes \mathbf{I}_L)\mathbf{D}_2^k$
  $\mathbf{P}^k \leftarrow \mathbf{B}^k \odot \mathbf{C}^k$
  $\mathbf{A}^{k+1} \leftarrow \mathbf{Y}_{(1)}\mathbf{P}^k(\mathbf{P}^{k\mathrm{T}}\mathbf{P}^k + \lambda\mathbf{D}^k)^{-1}$
  $\mathbf{Q}^k \leftarrow \mathbf{C}^k \odot \mathbf{A}^k$
  $\mathbf{B}^{k+1} \leftarrow \mathbf{Y}_{(2)}\mathbf{Q}^k(\mathbf{Q}^{k\mathrm{T}}\mathbf{Q}^k + \lambda\mathbf{D}^k)^{-1}$
  $\mathbf{S}^k \leftarrow [(\mathbf{A}_1^k \odot_{\mathrm{c}} \mathbf{B}_1^k)\mathbf{1}_L \cdots (\mathbf{A}_R^k \odot_{\mathrm{c}} \mathbf{B}_R^k)\mathbf{1}_L]$
  $\mathbf{C}^{k+1} \leftarrow \mathbf{Y}_{(3)}\mathbf{S}^k(\mathbf{S}^{k\mathrm{T}}\mathbf{S}^k + \lambda\mathbf{D}_1^k)^{-1}$
  $k \leftarrow k+1$
until *convergence*

where the locally upper bound function $g_{\mathbf{C}}(\mathbf{C}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$ has an analogous form with $g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k, \mathbf{B}^k, \mathbf{C}^k)$ in (11) with

$$\bar{\mathbf{H}}_{\mathbf{C}^k} = \mathbf{I}_K \otimes (\mathbf{S}^{k\mathrm{T}}\mathbf{S}^k + \lambda\mathbf{D}_1^k).$$

Summarizing the above, the steps of the proposed algorithm, which alternatingly solves for $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, in that order, are tabulated as Algorithm 1. As explained in Appendix D, the proposed algorithm is a sort of *hierarchical* iterative reweighted least squares (HIRLS) scheme, fully adjusted to promote block and column sparsity in the BTD model. This may be also seen from the expressions of $\mathbf{A}^{k+1}$, $\mathbf{B}^{k+1}$, and $\mathbf{C}^{k+1}$ given above and the form of the diagonal weighting matrices $\mathbf{D}_1^k$ and $\mathbf{D}_2^k$. Indeed, if R and L are overestimated, reweighting via $\mathbf{D}_1$ imposes *jointly* block sparsity on $\mathbf{A}$ and $\mathbf{B}$ and column sparsity on $\mathbf{C}$, hence helping in estimating R. In addition, reweighting via $\mathbf{D}_2$ promotes column sparsity *jointly* on the corresponding blocks of $\mathbf{A}$ and $\mathbf{B}$, thus estimating L_r's. This mechanism, combined with an appropriate selection of λ, can reveal the actual value of R and the true block-term ranks L_r's, as it is also empirically demonstrated in the next section.

It should be noted that the majorization functions $g_{\mathbf{A}}$, $g_{\mathbf{B}}$, and $g_{\mathbf{C}}$ used previously for the derivation of the proposed algorithm are *quadratic upper bound* functions that satisfy Assumption A [54, Table 3] required for BSUM. In addition, minimization of these functions w.r.t. the BTD factors at each iteration of the algorithm leads in all cases to unique solutions. Hence, according to Theorem 1 of [54], every limit point of the BTD-HIRLS algorithm is a stationary point of the initial objective function (9).

A notable feature of the proposed algorithm is that the closed-form expressions for the BTD factors comprise matrix operations only and relatively small-size ($RL \times RL$ and $R \times R$) matrix inversions. This should be attributed to the block diagonal form of the approximate Hessians employed in the three sub-problems. A brief analysis of the computational complexity of BTD-HIRLS is conducted in Appendix C, where it is shown that for I, J and K sufficiently larger than R and L, the number

of multiplications required per iteration is $\mathcal{O}(IJKRL)$. In contrast, in [46], the BTD factors are not computed in closed form but via a group-sparsity promoting iterative procedure in each iteration of the algorithm, which results in a much higher computational complexity per iteration compared to BTD-HIRLS, as it is empirically demonstrated in the next section. Further reduction in the computational complexity of the proposed algorithm is possible by eliminating negligible columns (column pruning) in the course of the iterations (as in [49]).

IV. Simulation Results

In this section, we report indicative simulation results with both synthetic and real data for evaluating the performance of the proposed algorithm. For comparison purposes, the classical BTD-ALS algorithm of [23], which makes no use of any low-rank regularization, and the BTD-AGL algorithm of [46], which minimizes the objective function defined in (2) enhanced by a proximal term, are also tested.[6] It should be noted that a block-pruning mechanism is implemented in both BTD-HIRLS and BTD-AGL, namely, $\mathbf{A}_r, \mathbf{B}_r$ blocks that correspond to columns of $\mathbf{C}$ with negligible energy are removed as the algorithms progress. This can be applied in both of the aforementioned algorithms due to their group-sparsity imposing characteristics.

A. Synthetic Data

In all cases, we generate BTD tensors $\boldsymbol{\mathcal{X}}$ contaminated by additive noise, i.e., $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} + \sigma\boldsymbol{\mathcal{N}}$, where $\boldsymbol{\mathcal{N}}$ contains zero-mean, independent and identically distributed (i.i.d) Gaussian entries of unit variance and σ is set so that we get a given signal-to-noise ratio (SNR), with SNR in dB defined as $\mathrm{SNR} = 10\log_{10}\|\boldsymbol{\mathcal{X}}\|_{\mathrm{F}}^2/(\sigma^2\|\boldsymbol{\mathcal{N}}\|_{\mathrm{F}}^2)$. The entries of the matrices $\mathbf{A}_r$ and $\mathbf{B}_r$ and the vectors $\mathbf{c}_r$ have been also sampled from i.i.d. zero-mean Gaussian distributions of unit variance. The tensor approximation is measured with the normalized mean squared error (NMSE) over the blocks, defined as $\mathrm{NMSE}(\hat{\mathbf{A}},\hat{\mathbf{B}},\hat{\mathbf{C}}) = \frac{1}{R}\sum_{r=1}^{R}\frac{\|\mathbf{A}_r\mathbf{B}_r^{\mathrm{T}}\circ\mathbf{c}_r - \hat{\mathbf{A}}_r\hat{\mathbf{B}}_r^{\mathrm{T}}\circ\hat{\mathbf{c}}_r\|_{\mathrm{F}}^2}{\|\mathbf{A}_r\mathbf{B}_r^{\mathrm{T}}\circ\mathbf{c}_r\|_{\mathrm{F}}^2}$, where $(\mathbf{A},\mathbf{B},\mathbf{C})$ and $(\hat{\mathbf{A}},\hat{\mathbf{B}},\hat{\mathbf{C}})$ denote the true and the estimated tensor factors, respectively. To calculate this metric, a linear assignment problem is solved to resolve the permutation ambiguity.[7] When R is overestimated (as in BTD-HIRLS and BTD-AGL), the NMSE over blocks is calculated on the basis of those of the R block terms that are "closer" to the true ones. For BTD-HIRLS and BTD-ALS, the stopping criterion is based on the relative difference between two consecutive values of the reconstruction error. For BTD-AGL, the relative difference of the values of the objective function (2) at two consecutive iterations is employed (as in [46]). The algorithms stop either when the relative difference becomes less than 10^{-6} or a maximum of 200 iterations is reached. The regularization parameter λ was empirically observed to depend on the dimensions of the tensor and the model ranks as well as on the noise strength. Hence, for the selection of the value of λ in BTD-HIRLS, we employ the heuristic rule $\lambda = L_{\mathrm{ini}}R_{\mathrm{ini}}(I+J+K)\hat{\sigma}$ with $\hat{\sigma}$ being a guess of the standard deviation of the noise. The γ-sweeping procedure employed in [46] is also adopted here for BTD-AGL, namely, for each single case (corresponding to a given realization and an initialization) BTD-AGL is applied five times with five different (and increasing) γ's (as in [46]) and the estimates from each run are used to initialize the next one.

TABLE I
NMSE and Run-Time Comparison of BTD-HIRLS, BTD-AGL and BTD-ALS for Different SNR Values

	SNR (dB)				aver. time (sec)
	5	10	15	20	
BTD-HIRLS	0.0792	0.0252	0.0082	0.0027	0.68
BTD-AGL	0.1367	0.0333	0.0076	0.0028	10.05
BTD-ALS	0.1517	0.0485	0.0170	0.0109	0.61

1) Performance in the Presence of Noise: First, we test the algorithms at different SNR values. We set $I = 18$, $J = 18$ and $K = 10$. The true R is set to 3 and the L_r's are selected as $L_1 = 8$, $L_2 = 6$ and $L_3 = 4$. The noisy tensors are generated as described above. Since the ranks of the model are in general unknown, we initialized BTD-HIRLS and BTD-AGL with overestimates of the true ones, namely $R_{\mathrm{ini}} = 10$ and $L_{\mathrm{ini}} = 10$ for all factor blocks. For BTD-ALS it was assumed that the true R is known, while all L_r's were overestimated to 10. All algorithms were randomly initialized (all entries of factors $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ were sampled from the standard Gaussian i.i.d. distribution) 10 times and their best run, in terms of the NMSE, was kept. In Table I, we report the median NMSEs of the results obtained over 100 independent realizations of the experiment. The average run-times (with Matlab 2019b in a Dell, MS Windows Pro, 2.20 GHz 10-core Intel CPU, 38 GB RAM) are also reported. Even with the knowledge of the true value of R, BTD-ALS is seen to be outperformed in terms of NMSE by the two rank-revealing methods, at all SNRs. In terms of accuracy, the proposed method seems to perform comparably with BTD-AGL, offering some gain only at sufficiently low SNR values. Moreover, BTD-AGL is considerably more computationally costly. Note that this should not be attributed to the γ-sweeping procedure only. Even a single run of BTD-AGL takes more time because, in contrast to BTD-HIRLS (and BTD-ALS), BTD-AGL does not rely on closed-form solutions for the updates of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ but it instead involves separate iterative procedures for the solution of each of the sub-problems. For example, in the above experiment, each iteration of BTD-HIRLS and BTD-AGL takes 3.4 msec and 10.05 msec, respectively.

Furthermore, BTD-HIRLS exhibits a higher rate of convergence than BTD-AGL, as demonstrated in Fig. 2, where the evolution of the NMSE for 10 realizations of the experiment at SNR $= 10$ dB is plotted versus the number of iterations. To facilitate the comparison of the 200 iterations of BTD-HIRLS with the BTD-AGL sweeping runs, its curves have been extended all the way to 1000 iterations based on the NMSE value at iteration 200. It should be clear from Fig. 2 that the γ-sweeping procedure is indeed useful for BTD-AGL as in most cases it does not converge before the 2nd-3rd round of it, that is, 400 iterations. In contrast, the proposed method converges fast, requiring no

[6] For BTD-AGL, we have used the Matlab code that Dr. J. H. de Morais Goulart kindly shared with us.

[7] The Matlab 2019b `matchpairs` function was employed for this purpose.

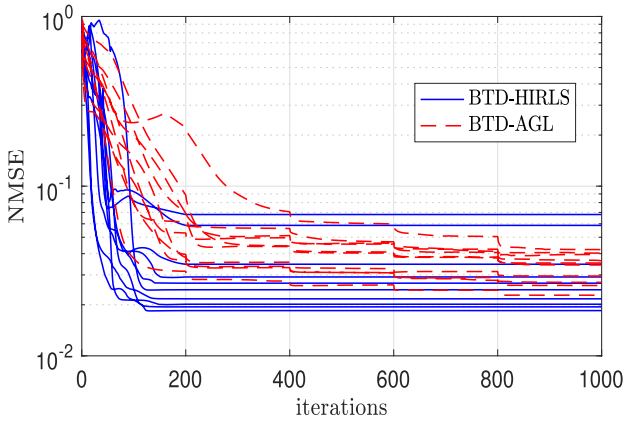


Fig. 2. NMSE of BTD-HIRLS and BTD-AGL vs. iterations for SNR=10 dB.

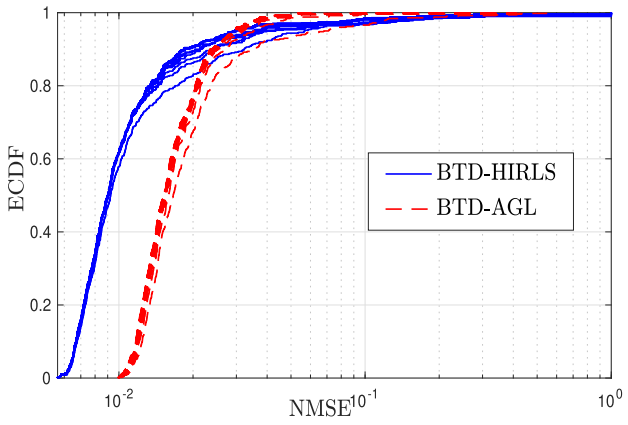


Fig. 3. Empirical cumulative distribution function (ECDF) of NMSEs obtained with BTD-HIRLS and BTD-AGL, for 500 different realizations. The ith curve from bottom to top corresponds to the result of selecting the best out of $i = 1, 2, \ldots, 12$ different initializations.

more than 100 iterations in virtually all realizations. Fig. 3 helps assessing the sensitivity to initialization of the BTD-HIRLS and BTD-AGL methods, by plotting the empirical cumulative distribution function (ECDF) of the NMSE obtained at SNR=15 and using the same experimental setting described previously for generating the tensors. For each of the two methods, the ith curve from bottom to top corresponds to selecting the best out of i initializations, for $i = 1, 2, \ldots, 12$. Clearly, BTD-HIRLS shares the insensitivity in terms of initialization demonstrated for BTD-AGL in [46], suggesting that only a small number of initializations suffices for achieving an accurate BTD model.

2) Success Rates for the Recovery of R and L_r's: In this part, our aim is to demonstrate the ability of BTD-HIRLS to reveal the true model structure. We set SNR=15 dB and we estimate the success rates in the estimation of R as well as of the L_r's for 100 different realizations of the experiment. Again, we generate tensors of dimensions $18 \times 18 \times 10$ and the true number of blocks, R, is set to 3. We compare BTD-HIRLS with BTD-AGL and for both algorithms we over-estimate R and L_r's as $R_{\text{ini}} = 10$ and $L_{\text{ini}} = 10$ for all block terms. We examine two different scenarios:

a) Scenario I: The true block ranks are $L_1 = 8$, $L_2 = 6$ and $L_3 = 4$. In this case, $\sum_{r=1}^{R} L_r = \min(I, J)$, that is, the (most well-known) sufficient uniqueness condition mentioned in the Introduction [1, Theorem 4.1] is met. As it can be seen in Fig. 4(a), BTD-HIRLS achieves success rates higher than 90% for all L_r's, outperforming BTD-AGL in the task of revealing the true L_r's. This can be explained by the properties of the regularizer of BTD-HIRLS which is carefully designed so as to better capture the structure of the decomposition model. The latter is also verified in Fig. 4(c), where we can see that BTD-HIRLS is more efficient than BTD-AGL when it comes to the success in estimating the number of block terms, R.

b) Scenario II: In this more challenging setting, we set $L_1 = 9$, $L_2 = 7$, and $L_3 = 5$. Thus, we now have $\sum_{r=1}^{R} L_r > \min(I, J)$ and hence the above sufficient uniqueness condition is violated. However, it can be observed in Fig. 4(b) that BTD-HIRLS reveals all L_r's with high relative frequencies, more or less outperforming BTD-AGL. Moreover, the success rate of accurately estimating R remains high as in the case of the simpler scenario (cf. Fig. 4(d)).

B. Experimenting With Real Data: Hyper-Spectral Image De-Noising

Hyper-spectral images are known to exhibit high coherence both in the spectral and the spatial domain [49]. As a result, low-rank matrix and tensor factorization methods have been widely employed to address related problems such as restoration, super resolution, de-noising, etc. [13], [14], [49]. Such an image can be viewed as a 3-way tensor, with its first two modes corresponding to the spatial coordinates and the third mode to the spectral bands. It is shown in [12] that the common linear mixing model assumption for HSI dictates a BTD model with its parameters having a clear physical interpretation. Here, we consider the recovery of a hyper-spectral image from its noise-corrupted version with additive Gaussian noise of SNR=5 dB, with the aid of the BTD-HIRLS and BTD-AGL algorithms. The idea is to recover the image as a low-rank BTD approximation of the noisy one, capitalizing on the low-rank structure of the HSI, which allows the removal of the high-rank noise [49]. In both cases, we set $R_{\text{ini}} = 50$ and $L_{\text{ini}} = 10$. Note that, for HSI data, and using the HSI spectral unmixing jargon, R is related to the number of the end-members (i.e., spectral signatures of the materials that exist in the depicted scene) while the L_r's reflect the ranks of the corresponding R abundance maps (i.e., the images of the percentages of a given material in the given image). In this example, we consider the Washington DC Mall AVIRIS image captured at $m = 191$ contiguous spectral bands in the 0.4 to 2.4 μm region of the visible and infrared spectrum [49]. The size of the image is 150×150 pixels. Thus, a $150 \times 150 \times 191$ tensor is formed.

We compare the performance of the proposed algorithm in this task with that of BTD-AGL, both visually and in terms of the *structural similarity index measure (SSIM)*. A popular metric of the degradation of an image as perceived change in structural information, SSIM is defined for two image windows x, y as $\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$, where μ_x, μ_y and

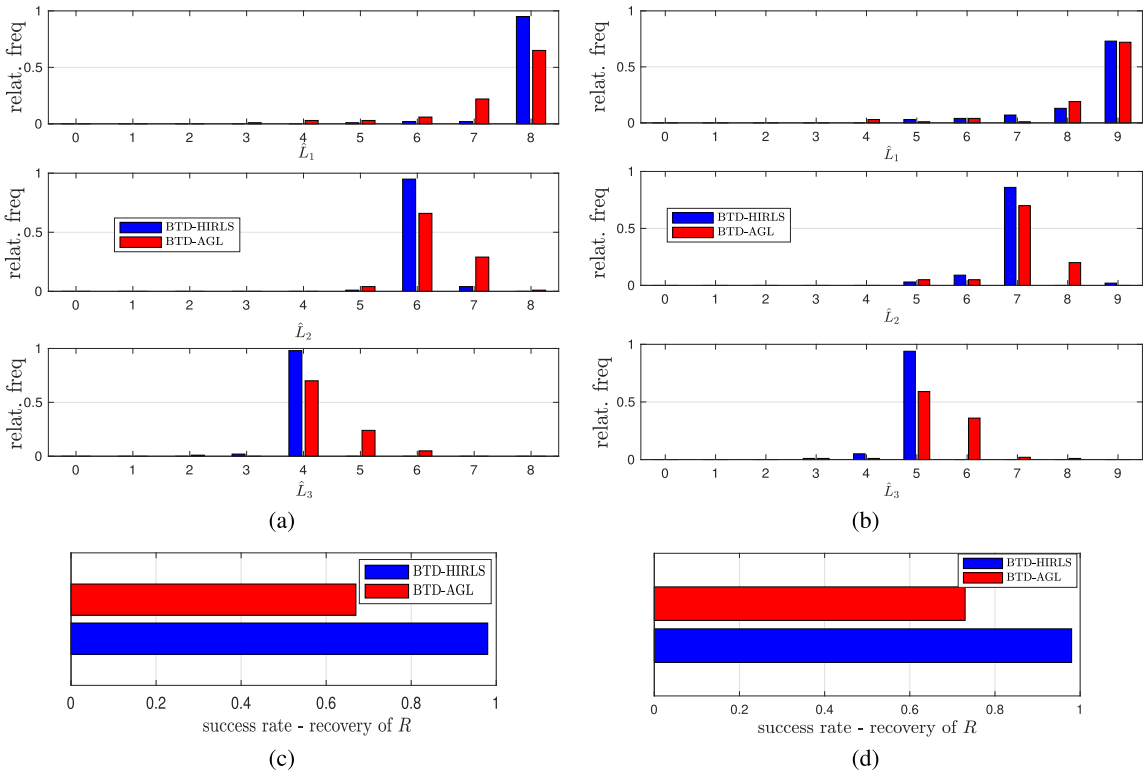


Fig. 4. Relative frequencies of the estimated values of L_r's and success rate (%) of estimating R via BTD-HIRLS and BTD-AGL. SNR $= 15$ dB. (a) Scenario I: $\sum_{r=1}^{R} L_r \leq \min(I, J)$; true L_r's are $L_1 = 8, L_2 = 6, L_3 = 4$. (b) Scenario II: $\sum_{r=1}^{R} L_r > \min(I, J)$; true L_r's are $L_1 = 9, L_2 = 7, L_3 = 5$. (c) Success rate of estimating R for Scenario I and (d) same for Scenario II.

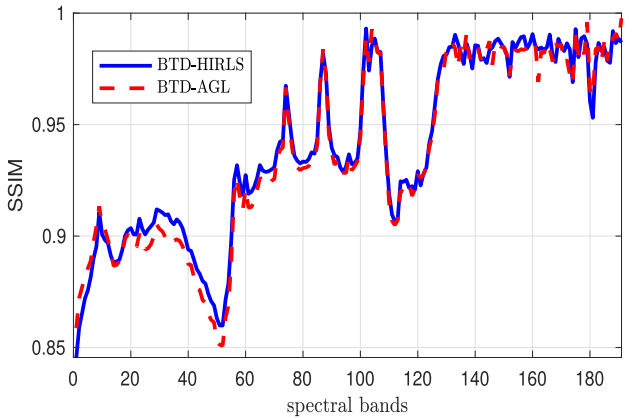


Fig. 5. SSIM of the hyper-spectral images recovered via BTD-HIRLS and BTD-AGL.

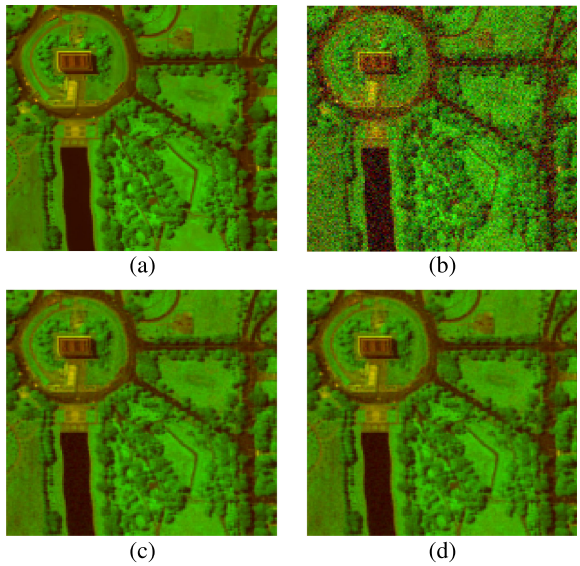


Fig. 6. False RGB color images of Washington DC Mall AVIRIS hyper-spectral image. (a) Original (b) Noisy (c) De-noised with BTD-AGL (d) De-noised with BTD-HIRLS.

σ_x^2, σ_y^2 are their averages and variances respectively, and σ_{xy} denotes their covariance. c_1, c_2 are small constants that are used for averting zero values in the numerator and the denominator. In our case[8], a separate value of SSIM is obtained for each spectral band (frontal slice of the HSI tensor) to measure the similarity of the reconstructed-denoised image with its "clean" version, considered as the ground truth. Fig. 5 plots the values of SSIM, while the results can be visually inspected in Fig. 6, where RGB false color images reconstructed from bands (24,64,135) are shown.[9] The two algorithms are seen to perform equally well in this experiment, though at a significantly shorter run-time

[8]We have employed the Matlab 2019b function `ssimval`.

[9]The choice was random, aiming only at showing examples from various regions of the spectrum.

for BTD-HIRLS. Both estimate R as 8, which agrees with the true number of the end-members in the scene depicted [49].

V. Conclusion

The challenging problem of efficiently and effectively estimating the model structure and parameters of a BTD has recently received special attention due to the increasing application range of this tensor model. This paper briefly reviews the related literature and reports our recent results on this topic, which are based on an appropriate extension to the BTD model of our earlier rank-revealing work on low-rank matrix and tensor approximation. The idea is to impose column sparsity *jointly* on the factors and in a *hierarchical* manner that matches the structure of the model, and successively estimate the ranks as the numbers of factor columns of non-negligible magnitude, with the aid of alternating hierarchical IRLS. The proposed method enjoys fast convergence and low computational complexity, also allowing the negligible columns to be pruned in the course of the procedure. Simulation results that demonstrate the effectiveness of our method in accurately estimating both the ranks and the factors in both synthetic and real-world scenarios are reported.

Future work will include the development of constrained variants of the method and (semi-)automatic ways of tuning its regularization parameter.

Acknowledgment

The authors would like to thank the reviewers for their constructive criticisms and comments that helped improve the quality of the manuscript. Special thanks go to Dr. J. H. de Morais Goulart for an insightful discussion on this subject and for kindly sharing with us the Matlab code implementing the BTD-AGL algorithm of [46].

Appendix A
Proof of Positive Semi-Definiteness

Let the $ILR \times 1$ vector $\mathbf{a} \triangleq \text{vec}(\mathbf{A})$ be $[a_{11}, \ldots, a_{1d}, \ldots, a_{I1}, \ldots, a_{Id}]^{\mathrm{T}}$ where $d \triangleq LR$. After some tedious algebra, it can be shown that the Hessian of $f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}, \mathbf{C})$ w.r.t. $\mathbf{a}$ is written as

$$\mathbf{H}_{\mathbf{A}} = \mathbf{I}_I \otimes (\mathbf{P}^{\mathrm{T}}\mathbf{P} + \lambda \mathbf{D}) - \lambda \mathbf{U} = \bar{\mathbf{H}}_{\mathbf{A}} - \lambda \mathbf{U}$$

i.e., $\bar{\mathbf{H}}_{\mathbf{A}} - \mathbf{H}_{\mathbf{A}} = \lambda \mathbf{U}$, where $\mathbf{U}$ is a $Id \times Id$ matrix which consists of I^2 $d \times d$ diagonal blocks denoted as $\mathbf{U}_{ij}$ with $i, j = 1, 2, \ldots, I$ and shown in (20) shown at the bottom of this page. Note that for the sake of the simplicity of the expressions, η^2 has been omitted. From (20) it follows that $\mathbf{U}$ can be written in the form $\mathbf{U} = \tilde{\mathbf{U}}^{\mathrm{T}}\tilde{\mathbf{U}}$, hence it is positive semi-definite. In fact, $\tilde{\mathbf{U}}$ is a $d \times Id$ matrix that comprises I $d \times d$ diagonal blocks $\tilde{\mathbf{U}}_i$ which can be written as in (21) shown at the bottom of the next page. It thus follows that, since $\lambda > 0$, $\bar{\mathbf{H}}_{\mathbf{A}} - \mathbf{H}_{\mathbf{A}}$ is also positive semi-definite, which completes the proof. Analogous results for the $\mathbf{B}$ and $\mathbf{C}$ sub-problems can be similarly arrived at.

Appendix B
Proof of Equation (15)

The gradient of $f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}^k, \mathbf{C}^k)$ (cf. (10)) w.r.t. $\mathbf{A}$ can be written as

$$\nabla_{\mathbf{A}} f_{\mathbf{A}}(\mathbf{A}|\mathbf{B}^k, \mathbf{C}^k) = -\mathbf{Y}_{(1)}\mathbf{P}^k + \mathbf{A}^k \mathbf{P}^{k\mathrm{T}}\mathbf{P}^k + \lambda \mathbf{A}\mathbf{D}_{\mathbf{A}}, \tag{22}$$

where $\mathbf{D}_{\mathbf{A}}$ is an $RL \times RL$ diagonal matrix having a form similar to $\mathbf{D}^k$ with $\mathbf{a}_{rl}$ instead of $\mathbf{a}_{rl}^k$ in (13) and (14). Hence (22) computed at $\mathbf{A}^k$ takes the form

$$\nabla_{\mathbf{A}} f_{\mathbf{A}}(\mathbf{A}^k|\mathbf{B}^k, \mathbf{C}^k) = -\mathbf{Y}_{(1)}\mathbf{P}^k + \mathbf{A}^k(\mathbf{P}^{k\mathrm{T}}\mathbf{P}^k + \lambda \mathbf{D}^k). \tag{23}$$

$$\begin{aligned}
\mathbf{U}_{ij} = \text{diag}\Bigg(& \frac{a_{i1}a_{j1}\left[\left(\|\mathbf{a}_{11}\|_2^2 + \|\mathbf{b}_{11}\|_2^2\right)^{1/2} + \left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2 + \|\mathbf{b}_{1l}\|_2^2}\right)^2 + \|\mathbf{c}_1\|_2^2\right]}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2 + \|\mathbf{b}_{1l}\|_2^2}\right)^2 + \|\mathbf{c}_1\|_2^2\right]^{3/2}\left(\|\mathbf{a}_{11}\|_2^2 + \|\mathbf{b}_{11}\|_2^2\right)^{3/2}}, \ldots \\
& \frac{a_{iL}a_{jL}\left[\left(\|\mathbf{a}_{1L}\|_2^2 + \|\mathbf{b}_{1L}\|_2^2\right)^{1/2} + \left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2 + \|\mathbf{b}_{1l}\|_2^2}\right)^2 + \|\mathbf{c}_1\|_2^2\right]}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2 + \|\mathbf{b}_{1l}\|_2^2}\right)^2 + \|\mathbf{c}_1\|_2^2\right]^{3/2}\left(\|\mathbf{a}_{1L}\|_2^2 + \|\mathbf{b}_{1L}\|_2^2\right)^{3/2}}, \\
& \frac{a_{i(L+1)}a_{j(L+1)}\left[\left(\|\mathbf{a}_{21}\|_2^2 + \|\mathbf{b}_{21}\|_2^2\right)^{1/2} + \left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{2l}\|_2^2 + \|\mathbf{b}_{2l}\|_2^2}\right)^2 + \|\mathbf{c}_2\|_2^2\right]}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{2l}\|_2^2 + \|\mathbf{b}_{2l}\|_2^2}\right)^2 + \|\mathbf{c}_2\|_2^2\right]^{3/2}\left(\|\mathbf{a}_{21}\|_2^2 + \|\mathbf{b}_{21}\|_2^2\right)^{3/2}}, \ldots \\
& \frac{a_{id}a_{jd}\left[\left(\|\mathbf{a}_{RL}\|_2^2 + \|\mathbf{b}_{RL}\|_2^2\right)^{1/2} + \left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{Rl}\|_2^2 + \|\mathbf{b}_{Rl}\|_2^2}\right)^2 + \|\mathbf{c}_R\|_2^2\right]}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{Rl}\|_2^2 + \|\mathbf{b}_{Rl}\|_2^2}\right)^2 + \|\mathbf{c}_R\|_2^2\right]^{3/2}\left(\|\mathbf{a}_{RL}\|_2^2 + \|\mathbf{b}_{RL}\|_2^2\right)^{3/2}} \Bigg)
\end{aligned} \tag{20}$$

To minimize the quadratic function $g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k,\mathbf{B}^k,\mathbf{C}^k)$, we find the unique matrix $\mathbf{A}$ for which $\nabla_{\mathbf{A}} g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k,\mathbf{B}^k,\mathbf{C}^k)=\mathbf{0}$. From (11) we have

$$\nabla_{\mathbf{A}} g_{\mathbf{A}}(\mathbf{A}|\mathbf{A}^k,\mathbf{B}^k,\mathbf{C}^k) = \nabla_{\mathbf{A}} f_{\mathbf{A}}(\mathbf{A}^k|\mathbf{B}^k,\mathbf{C}^k) + (\mathbf{A}-\mathbf{A}^k)(\mathbf{P}^{k\mathrm{T}}\mathbf{P}^k+\lambda\mathbf{D}^k). \quad (24)$$

Substituting (23) in (24), equating to zero and solving for $\mathbf{A}$ leads to (15). Eqs. (17) and (19) can be obtained in an analogous way.

Appendix C
Computational Complexity of BTD-HIRLS

Using the mixed product rule from [55, Eq. (25)], $\mathbf{S}^{\mathrm{T}}\mathbf{S}$ in (19) can be equivalently written as

$$\mathbf{S}^{\mathrm{T}}\mathbf{S} = (\mathbf{I}_R \otimes \mathbf{1}_L^{\mathrm{T}})(\mathbf{A}^{\mathrm{T}}\mathbf{A} * \mathbf{B}^{\mathrm{T}}\mathbf{B})(\mathbf{I}_R \otimes \mathbf{1}_L), \quad (25)$$

which allows it to be computed in a cheaper manner. Analogous expressions can be arrived at for the other two Grammian matrices involved in the algorithm iterations. Indeed, it is not difficult to generalize the mixed product rule to partition-wise Khatri-Rao products with column-wise partition for the one of the factors, as it is the case for $\mathbf{P}$ and $\mathbf{Q}$. One can then readily verify that the corresponding Grammian matrices, $\mathbf{P}^{\mathrm{T}}\mathbf{P}$ and $\mathbf{Q}^{\mathrm{T}}\mathbf{Q}$, required in (15) and (17), respectively, can be equivalently written as:

$$\mathbf{P}^{\mathrm{T}}\mathbf{P} = \mathbf{B}^{\mathrm{T}}\mathbf{B} * (\mathbf{C}^{\mathrm{T}}\mathbf{C} \otimes \mathbf{1}_{L\times L}) \quad (26)$$

$$\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{A}^{\mathrm{T}}\mathbf{A} * (\mathbf{C}^{\mathrm{T}}\mathbf{C} \otimes \mathbf{1}_{L\times L}). \quad (27)$$

Neglecting existing symmetries the computational cost of a BTD-HIRLS iteration can be estimated as follows. The computation of the matrices $\mathbf{P},\mathbf{Q}$, and $\mathbf{S}$ requires $(IJ+JK+KI)(LR)$ multiplications in total. When computing their Grammians as in (25), (26), and (27), $(2(I+J)+3)(LR)^2+2KR^2$ multiplications are needed. For the $\mathbf{D}$ matrix we need around $((I+J)L+K)R+LR$ multiplications (not counting the square roots). The matrix inversions require $\mathcal{O}(2(LR)^3+R^3)$ multiplications in total. Finally, for the matrix multiplications in (15), (17) and (19), $2IJKLR+(I+J)K(LR)^2+IJKR+IJR^2$ multiplications are required. Therefore, in the most common practical case of big low-rank tensors, that is when $I,J,K \gg R,L$, the per-iteration computational complexity of BTD-HIRLS amounts to $\mathcal{O}(IJKLR)$.

Appendix D
Hierarchical IRLS Nature of BTD-HIRLS

If $\mathbf{x}=[\,x_1\ x_2\ \cdots\ x_n\,]^{\mathrm{T}}$ is a sparse vector, the IRLS algorithm for estimating $\mathbf{x}$ subject to an ℓ_2 proximity criterion is derived by solving the following minimization problem at iteration $k+1$ [50]:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{b}-\mathbf{R}\mathbf{x}\|_2^2 + \frac{\lambda}{2}\sum_{i=1}^{n}\frac{x_i^2}{\sqrt{(x_i^k)^2+\eta^2}}. \quad (28)$$

This problem admits the closed-form solution $\mathbf{x}^{k+1}=(\mathbf{R}^{\mathrm{T}}\mathbf{R}+\lambda\mathbf{W}^k)^{-1}\mathbf{R}^{\mathrm{T}}\mathbf{b}$, where $\mathbf{W}^k$ is a diagonal weighting matrix whose ith diagonal entry is given by $\mathbf{W}^k(i,i)=((x_i^k)^2+\eta^2)^{-1/2}$.

In the same vein, it can be shown that the closed-form expression for the BTD factor $\mathbf{A}^{k+1}$ given in (15) can be also obtained by solving the following minimization problem

$$\mathbf{A}^{k+1} = \arg\min_{\mathbf{A}} \frac{1}{2}\left\|\mathbf{Y}_{(1)}^{\mathrm{T}}-\mathbf{P}^k\mathbf{A}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + \frac{\lambda}{2}\sum_{r=1}^{R}\frac{\left(\frac{1}{2}\sum_{l=1}^{L}\frac{\|\mathbf{a}_{rl}\|_2^2+\|\mathbf{b}_{rl}^k\|_2^2+\eta^2}{\sqrt{\|\mathbf{a}_{rl}^k\|_2^2+\|\mathbf{b}_{rl}^k\|_2^2+\eta^2}}\right)^2+\|\mathbf{c}_r^k\|_2^2+\eta^2}{\sqrt{\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{rl}^k\|_2^2+\|\mathbf{b}_{rl}^k\|_2^2+\eta^2}\right)^2+\|\mathbf{c}_r^k\|_2^2+\eta^2}}. \quad (29)$$

$$\tilde{\mathbf{U}}_i = \mathrm{diag}\left(\frac{a_{i1}\left[\left(\|\mathbf{a}_{11}\|_2^2+\|\mathbf{b}_{11}\|_2^2\right)^{1/2}+\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2+\|\mathbf{b}_{1l}\|_2^2}\right)^2+\|\mathbf{c}_1\|_2^2\right]^{1/2}}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2+\|\mathbf{b}_{1l}\|_2^2}\right)^2+\|\mathbf{c}_1\|_2^2\right]^{3/4}\left(\|\mathbf{a}_{11}\|_2^2+\|\mathbf{b}_{11}\|_2^2\right)^{3/4}},\ldots\right.$$
$$\frac{a_{iL}\left[\left(\|\mathbf{a}_{1L}\|_2^2+\|\mathbf{b}_{1L}\|_2^2\right)^{1/2}+\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2+\|\mathbf{b}_{1l}\|_2^2}\right)^2+\|\mathbf{c}_1\|_2^2\right]^{1/2}}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{1l}\|_2^2+\|\mathbf{b}_{1l}\|_2^2}\right)^2+\|\mathbf{c}_1\|_2^2\right]^{3/4}\left(\|\mathbf{a}_{1L}\|_2^2+\|\mathbf{b}_{1L}\|_2^2\right)^{3/4}},$$
$$\frac{a_{i(L+1)}\left[\left(\|\mathbf{a}_{21}\|_2^2+\|\mathbf{b}_{21}\|_2^2\right)^{1/2}+\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{2l}\|_2^2+\|\mathbf{b}_{2l}\|_2^2}\right)^2+\|\mathbf{c}_2\|_2^2\right]^{1/2}}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{2l}\|_2^2+\|\mathbf{b}_{2l}\|_2^2}\right)^2+\|\mathbf{c}_2\|_2^2\right]^{3/4}\left(\|\mathbf{a}_{21}\|_2^2+\|\mathbf{b}_{21}\|_2^2\right)^{3/4}},\ldots$$
$$\left.\frac{a_{id}\left[\left(\|\mathbf{a}_{RL}\|_2^2+\|\mathbf{b}_{RL}\|_2^2\right)^{1/2}+\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{Rl}\|_2^2+\|\mathbf{b}_{Rl}\|_2^2}\right)^2+\|\mathbf{c}_R\|_2^2\right]^{1/2}}{\left[\left(\sum_{l=1}^{L}\sqrt{\|\mathbf{a}_{Rl}\|_2^2+\|\mathbf{b}_{Rl}\|_2^2}\right)^2+\|\mathbf{c}_R\|_2^2\right]^{3/4}\left(\|\mathbf{a}_{RL}\|_2^2+\|\mathbf{b}_{RL}\|_2^2\right)^{3/4}}\right) \quad (21)$$

A similar minimization problem can be defined for computing the factor $\mathbf{B}^{k+1}$ as in (17), while we can also get (19) from

$$\mathbf{C}^{k+1} = \arg\min_{\mathbf{C}} \frac{1}{2} \left\| \mathbf{Y}_{(3)}^{\mathrm{T}} - \mathbf{S}^k \mathbf{C}^{\mathrm{T}} \right\|_{\mathrm{F}}^2 + \frac{\lambda}{2} \sum_{r=1}^{R} \frac{\left(\sum_{l=1}^{L} \sqrt{\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2} \right)^2 + \|\mathbf{c}_r\|_2^2 + \eta^2}{\sqrt{\left(\sum_{l=1}^{L} \sqrt{\|\mathbf{a}_{rl}^k\|_2^2 + \|\mathbf{b}_{rl}^k\|_2^2 + \eta^2} \right)^2 + \|\mathbf{c}_r^k\|_2^2 + \eta^2}} \tag{30}$$

By carefully inspecting the objective functions in (29) and (30) and comparing with the conventional IRLS objective function in (28), we easily recognize a hierarchical IRLS structure consisting of two separate reweighting least squares steps. Each step gives rise to a separate reweighting matrix. Namely, the first one ($\mathbf{D}_1$) is composed of the inverses of the outer summation terms of the regularizer in (9) and jointly weighs the blocks of $\mathbf{A}, \mathbf{B}$, i.e. the $\mathbf{A}_r$'s and $\mathbf{B}_r$'s, and the respective columns of $\mathbf{C}$. The second reweighting matrix ($\mathbf{D}_2$) contains the inverses of the terms of the inner summation in (9) and jointly balances the corresponding columns of the $\mathbf{A}_r$'s and $\mathbf{B}_r$'s. It thus follows that the proposed two-level $\ell_{1,2}$ regularization naturally leads to a IRLS scheme with a corresponding hierarchy.

REFERENCES

[1] 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.

[2] N. D. Sidiropoulos *et al.*, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[3] 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.

[4] M. Sørensen, F. Van Eeghem, and L. De Lathauwer, "Blind multichannel deconvolution and convolutive extensions of canonical polyadic and block term decompositions," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4132–4145, Aug. 2017.

[5] B. Hunyadi *et al.*, "Block term decomposition for modelling epileptic seizures," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 139, 2014.

[6] C. Chatzichristos *et al.*, "Blind fMRI source unmixing via higher-order tensor decompositions," *J. Neurosci. Methods*, vol. 315, pp. 17–47, Mar. 2019.

[7] Y. R. Aldana *et al.*, "Nonconvulsive epileptic seizure detection in scalp EEG using multiway data analysis," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 2, pp. 660–671, Mar. 2019.

[8] C. Stamile *et al.*, "Tensor based blind source separation in longitudinal magnetic resonance imaging analysis," in *Proc. Eng. Med. Biol. Conf.*, Berlin, Germany, Jul. 2019, pp. 3879–3883.

[9] V. Zarzoso, "Parameter estimation in block term decomposition for non-invasive atrial fibrillation analysis," in *Proc. Comput. Adv. Multi-Sensor Adaptive Process.*, Curaçao, Dutch Antilles, Dec. 2017.

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

[11] I. Mousavian, M. B. Shamsollahi, and E. Fatemizadeh, "Noninvasive fetal ECG extraction using doubly constrained block-term decomposition," *Math. Biosci. Eng.*, vol. 17, no. 1, pp. 144–159, Sep. 2019.

[12] Y. Qian *et al.*, "Matrix-vector nonnegative tensor factorization for blind unmixing of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 3, pp. 1776–1792, Mar. 2017.

[13] F. Xiong, J. Zhou, and Y. Qian, "Hyperspectral restoration via L_0 gradient regularized low-rank tensor factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 10 410–10 425, Dec. 2019.

[14] G. Zhang *et al.*, "Hyperspectral super-resolution: A coupled nonnegative block-term tensor decomposition approach," in *Proc. Comput. Adv. Multi-Sensor Adaptive Process.*, Guadeloupe, West Indies, Dec. 2019, pp. 470–474.

[15] B. L. Bianca and P. S. Gheorghe, "Unsupervised clustering for hyperspectral images," *Symmetry*, vol. 12, no. 2, 2020.

[16] E. Gujral, R. Pasricha, and E. E. Papalexakis, "Beyond rank-1: Discovering rich community structure in multi-aspect graphs," in *Proc. WWW-2020*, Taipei, Taiwan, Apr. 2020.

[17] G. Zhang *et al.*, "Spectrum cartography via coupled block-term tensor decomposition," *IEEE Trans. Signal Process.*, vol. 68, pp. 3660–3675, 2020.

[18] J. Spiegelberg, J. Rusz, and K. Pelckmans, "Tensor decompositions for the analysis of atomic resolution electron energy loss spectra," *Ultramicroscopy*, vol. 175, pp. 36–45, 2017.

[19] X. Ma *et al.*, "A tensorized transformer for language modeling," in *Proc. NeurIPS-2019*, Vancouver, Canada, Dec. 2019.

[20] J. Ye *et al.*, "Block-term tensor neural networks," *Neural Netw.*, Jun. 2020. [Online]. Available: https://doi.org/10.1016/j.neunet.2020.05.034

[21] 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.

[22] L. De Lathauwer, "Block component analysis: A new concept for blind source separation," in *Proc. Latent Variable Anal. Signal Separation*, Tel Aviv, Israel, Mar. 2012.

[23] 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.

[24] J. H. de M. Goulart and P. Comon, "On the minimal ranks of matrix pencils and the existence of a best approximate block-term tensor decomposition," *Linear Algebra Appl.*, vol. 561, pp. 161–186, 2019.

[25] I. Domanov and L. De Lathauwer, "From computation to comparison of tensor decompositions," Dec. 2019, *arXiv:1912.04694v1*.

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

[27] N. Vervliet, I. Domanov, and L. De Lathauwer, "Algebraic and optimization-based algorithms for decomposing tensors into block terms," in *Proc. 21st Householder Symp. Numer. Linear Algebra*, 2020.

[28] 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.

[29] N. Vervliet *et al.*, *Tensorlab User Guide - Release 3.0*, Mar. 2016. [Online]. Available: http://www.tensorlab.net/userguide3.pdf

[30] N. Li, "Variants of ALS on tensor decompositions and applications," Ph.D. dissertation, Dept. Math., Clarkson Univ., 2013.

[31] J. H. de M. Goulart and P. Comon, "Non-iterative low-multilinear-rank tensor approximation with application to decomposition in rank-$(1, L, L)$ terms," hal-01516167, Apr. 2017. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01516167

[32] G. Olikier, P.-A. Absil, and L. De Lathauwer, "A variable projection method for block term decomposition of higher-order tensors," in *Proc. Latent Variable Anal. Signal Separation*, Guildford, U.K., Jul. 2018.

[33] P. Tichavsky, A.-H. Phan, and A. Cichocki, "Non-orthogonal tensor diagonalization," *Signal Process.*, vol. 138, pp. 313–320, Sep. 2017.

[34] X. Fu and K. Huang, "Block-term tensor decomposition via constrained matrix factorization," in *Proc. Mach. Learn. Signal Process.*, Pittsburg, PA, Oct. 2019.

[35] X. Han *et al.*, "Block term decomposition with rank estimation using group sparsity," in *Proc. Comput. Adv. Multi-Sensor Adaptive Process.*, Curaçao, Dutch Antilles, Dec. 2017.

[36] E. E. Papalexakis, "Automatic unsupervised tensor mining with quality assessment," in *Proc. SDM-2016*, Miami, FA, May 2016.

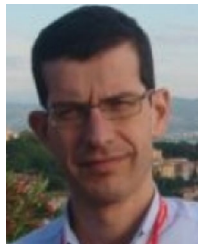
[37] T. Yokota, N. Lee, and A. Cichocki, "Robust multilinear tensor rank estimation using higher order singular value decomposition and information criteria," *IEEE Trans. Signal Process.*, vol. 65, no. 5, pp. 1196–1206, Mar. 2017.

[38] Q. Xie *et al.*, "Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1888–1902, Aug. 2018.

[39] X. Han *et al.*, "Robust multilinear decomposition of low rank tensors," in *Proc. Latent Variable Anal. Signal Separation*, Guilford, U.K., Jul. 2018.

[40] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.
[41] A. J. Brockmeier *et al.*, "Greedy algorithm for model selection of tensor decompositions," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Vancouver, Canada, May 2013, pp. 6113–6117.
[42] A. H. Phan *et al.*, "From basis components to complex structural patterns," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Vancouver, Canada, May 2013, pp. 3228–3232.
[43] X. Han, "Robust low-rank tensor approximations using group sparsity," Ph.D. dissertation, ECOLE DOCTORALE N° 601: *Mathématiques et Sciences et Technologies de l' Information et de la Communication*, L' Université de Rennes 1, Rennes, France, Jan. 2019.
[44] B. Yang, G. Wang, and N. D. Sidiropoulos, "Tensor completion via group-sparse regularization," in *Proc. Asilomar Conf. Signals, Syst. Comput.,* Asilomar Conf. Grounds, Pacific Grove, CA, Nov. 2016, pp. 1750–1754.
[45] I. C. Tsaknakis *et al.*, "A computationally efficient tensor completion algorithm," *IEEE Signal Process. Lett.*, vol. 25, no. 8, pp. 1266–1270, Aug. 2018.
[46] J. H. de M. Goulart *et al.*, "Alternating group lasso for block-term tensor decomposition and application to ECG source separation," *IEEE Trans. Signal Process.*, vol. 68, pp. 2682–2696, Apr. 2020.
[47] P. V. Giampouras, A. A. Rontogiannis, and K. D. Koutroumbas, "Online low-rank subspace learning from incomplete data using rank revealing ℓ_2/ℓ_1 regularization," in *Proc. Stat. Signal Process. Workshop*, Palma de Mallorca, Spain, Jun. 2016.
[48] P. V. Giampouras, A. A. Rontogiannis, and K. D. Koutroumbas, "ℓ_1/ℓ_2 regularized non-convex low-rank matrix factorization," in *Proc. Signal Process. Adaptive Sparse Struct. Representations*, Lisbon, Portugal, Jul. 2017, pp. 619–642.
[49] P. V. Giampouras, A. A. Rontogiannis, and K. D. Koutroumbas, "Alternating iteratively reweighted least squares minimization for low-rank matrix factorization," *IEEE Trans. Signal Process.*, vol. 67, no. 2, pp. 490–503, Jan. 2019.
[50] I. Daubechies *et al.*, "Iteratively reweighted least squares minimization for sparse recovery," *Commun. Pure Appl. Math.*, vol. 63, pp. 1–38, 2010.
[51] A. A. Rontogiannis, E. Kofidis, and P. V. Giampouras, "Block-term tensor decomposition: Model selection and computation," in *Proc. Eur. Signal Process. Conf.*, Amsterdam, The Netherlands, Aug. 2020.
[52] Y. Hu *et al.*, "Group sparse optimization via $\ell_{p,q}$ regularization," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 960–1011, 2017.
[53] P. V. Giampouras, A. A. Rontogiannis, and K. D. Koutroumbas, "A projected Newton-type algorithm for non-negative matrix factorization with model order selection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Brighton, U.K., May 2019, pp. 3497–3501.
[54] M. Hong *et al.*, "A unified algorithmic framework for block-structured optimization involving big data," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 57–77, Jan. 2016.
[55] S. Liu and G. Trenkler, "Hadamard, Khatri-Rao, Kronecker and other matrix products," *Int. J. Info. Syst. Sci.*, vol. 4, no. 1, pp. 160–177, 2008.

Athanasios A. Rontogiannis (Member, IEEE) received the five year Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1991, the M.A.Sc. degree in electrical and computer engineering from the University of Victoria, Victoria, BC, Canada, in 1993, and the Ph.D. degree in signal processing and communications from the National and Kapodistrian University of Athens, Athens, Greece, in 1997. From 1998 to 2003, he was a Lecturer with the University of Ioannina, Ioannina, Greece. In 2003, he joined the National Observatory of Athens, Athens, Greece, where he is currently a Research Director with the Institute for Astronomy, Astrophysics, Space Applications and Remote Sensing. He has coauthored more than 100 articles in refereed journals and conference proceedings. His research interests include the general area of statistical signal and image processing with emphasis on adaptive signal processing, hyper-spectral image processing, compressive sensing, sparse and low-rank signal representations, and fast signal processing algorithms. He was a Program Committee Member in more than 25 conferences, in one of them as the Co-Chair and in three of them as an Area Chair. He was on the Editorial Boards of the EURASIP *Journal on Advances in Signal Processing*, Springer (2008–2017), and the EURASIP *Signal Processing* Journal, Elsevier (since 2011). Since 2017, he has been an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He is a Member of the EURASIP and the Technical Chamber of Greece. Since January 2018, he has also been the Chair of the IEEE Signal Processing Society Greece Chapter.



Eleftherios Kofidis (Member, IEEE) received the Diploma and Ph.D. degrees from the Department of Computer Engineering and Informatics, University of Patras, Patras, Greece, in 1990 and 1996, respectively. From 1996 to 1998, he was with the Hellenic Army, Greece. From 1998 to 2000, he was a Postdoctoral Fellow with the Institut National des Télécommunications (INT), Évry, France (now Télécom SudParis). From 2001 to 2004, he was a Research Associate with the University of Athens, Athens, Greece, and an Adjunct Professor with the University of Peloponnese, Tripoli, Greece, and the University of Piraeus, Piraeus, Greece. In 2004, he joined the Department of Statistics and Insurance Science, University of Piraeus, where he is currently an Associate Professor. He is also affiliated with the Computer Technology Institute and Press "Diophantus", Patras, Greece. His research interests include signal processing and machine learning, with applications including communications and medical imaging. He was a Technical Program Co-Chair in two international conferences, CIP-2008 and DSP-2009, and a Technical Program Committee Member and a Reviewer in a number of conferences and journals. He has also served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the EURASIP *Journal on Advances in Signal Processing* (JASP), and the IET *Signal Processing journal*. He coorganized three special sessions on filter bank-based multicarrier systems, which include ISWCS-2012, EW-2014, and SPAWC-2015, and was the Lead Guest Editor for a JASP Special Issue on this subject.

Paris V. Giampouras was born in Athens, Greece, in 1986. He received the Diploma degree in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, in 2011, the M.Sc. degree in information technologies in medicine and biology from the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Athens, Greece, in 2014, and the Ph.D. degree from the National and Kapodistrian University of Athens in 2018. Since 2019, he has been a Marie Skłodowska-Curie Postdoctoral Fellow with the Mathematical Institute for Data Science, Johns Hopkins University, Baltimore, MD, USA. His main research interests include signal processing and machine learning focusing on nonconvex optimization and representation learning using sparse and low-rank priors with application to collaborative filtering and hyper-spectral image processing.