COLUMN-WISE SYMMETRIC BLOCK PARTITIONED TENSOR DECOMPOSITION

Christopher Mueller-Smith and Predrag Spasojević
WINLAB, Department of Electrical and Computer Engineering, Rutgers University
chris.muellersmith@rutgers.edu, spasojev@winlab.rutgers.edu

ABSTRACT

Symmetric block partitioned tensors (SBPT) are a useful structure in signal processing applications, often generated from computing higher-order statistics on observed data. Such tensors often follow the rank-$(R_m, R_m, 1)$ SBPT structure, but in some applications the partitioning of the factor matrices is not known *a priori*. We propose a both blind and non-blind column-wise SBPT decomposition algorithms that are better scalable to high-dimensional tensors because they avoid large matrix inversions. We apply the algorithms to simulated SBPTs and demonstrate that they estimate factor matrices having high congruence with the originals across a range of collinearity values for the columns of the original factor matrices.

***Index Terms*—** block partitioned tensor, symmetric tensor, tensor decomposition

1. INTRODUCTION

Tensors are multi-way data arrays which have appeared in applications such as antenna array processing, chemometrics, and computer vision, among many others. Many tensor models have been developed over the years, with some of the most commonly known being the Canonical Decomposition/Parallel Factors (CP) model [1][2] and Tucker model [3]. More recently, the more general model of block partitioned tensors (BPT), which incorporates models such as CP as special cases, has been developed [4]. A particularly useful property of tensors, and a reason they can be found in many applications, is that their decompositions are often unique (up to column permutation and scaling) under certain conditions [5][6].

A particular case of these tensor models is tensors consisting of a collection of symmetric faces, making them symmetric in certain modes of the tensor, but not others. Such tensors often arise when their entries are generated from high order statistics. For example, one particular application we have investigated has made use of the trispectrum for the purpose of blind signal separation using a single sensor [7][8]. These tensors, consisting of sums of trispectra, can be modeled as symmetric CP tensors, however they cannot be uniquely decomposed into factor matrices (providing power spectrum and on/off activity information) under that model. To achieve a unique decomposition they must be modeled as rank-$(R_m, R_m, 1)$ symmetric block partitioned tensors (SBPT) as proposed in [4].

In the decomposition algorithm proposed for rank-$(R_m, R_m, 1)$ BPTs in [9], the matrix partitioning is known, while in situations such as the spectrum sensing application above we do not *a priori* know the partitioning as it will be a function of the mixture of the unknown signals, nor does that algorithm enforce symmetry. Additionally, it relies on an alternating least squares (ALS) approach which becomes computationally unwieldy as the tensor dimensions grow, such as when high resolution is desired in the spectrum sensing application.

Previous algorithms for the similar problem of joint block diagonalization with symmetric matrices [10] and our initial exploration into SBPT decomposition use gradient-based optimization strategies, however they become extremely slow and unreliable as the factor matrix dimensions increase. In this work we propose column-wise updating non-blind and blind SBPT tensor decomposition algorithms which iterate quickly, even at higher dimensions, as each step consists of a few matix operations, avoiding large matrix inversions. The blind version estimates matrix partitioning using clustering. After describing the algorithm we evaluate its performance through simulation on simulated data with controlled congruence between factor matrix columns. We find that, for a $20 \times 20 \times 20$ tensor of rank 16 with 4 partitions, the algorithms typically converge within 50,000 iterations and produce factor matrix estimates with low complementary cosine similarity (CCS) as compared to the true matrices for a wide range of congruences when appropriate regularization coefficients are chosen.

2. NOTATION

In this paper we use the following notation, much of it matching that in the tutorial paper [11]. Vectors are denoted as $\mathbf{x}$ and matrices are $\mathbf{X}$ with elements x_m and x_{mn}, respectively. Column i of matrix $\mathbf{X}$ is denoted as $\mathbf{x}_i$ while row i is denoted with MATLAB-like notation $\mathbf{X}_{i:}$. $\boldsymbol{\mathcal{X}} \in \mathbb{K}^{I \times J \times K}$ is a tensor with matricized unfoldings $\mathbf{X}_{(d)}$, $d = 1, 2, 3$ (see [11] for unfolding details). $\mathbf{A} \odot \mathbf{B} = \begin{bmatrix}\mathbf{a}_1 \otimes \mathbf{b}_1 \ldots \mathbf{a}_R \otimes \mathbf{b}_R\end{bmatrix}$ is the Khatri-Rao product of $\mathbf{A}$ and $\mathbf{B}$. $\circ$ is the outer product, and $*$ is the Hadamard product of matrices. The soft-threshold operator with threshold ν is $S_\nu(x) = \text{sign}(x) \cdot \max(|x| - \nu, 0)$.

3. SYMMETRIC BLOCK PARTITIONED TENSOR

Our interest is in real-valued three-way front-symmetric BPTs (SBPT) having rank-$(R_m, R_m, 1)$ partitions where tensor $\boldsymbol{\mathcal{Y}}$ has entries where $y_{ijk} = y_{jik}$. The factor matrices for this type of SBPT are $\mathbf{F} = \begin{bmatrix}\mathbf{F}_1 | \mathbf{F}_2 | \cdots | \mathbf{F}_M\end{bmatrix} \in \mathbb{R}^{I \times R}$, consisting of M partitions each of rank R_m, and $\mathbf{C} = \begin{bmatrix}\mathbf{c}_1 & \mathbf{c}_2 & \cdots & \mathbf{c}_M\end{bmatrix}$ with each column corresponding

Thanks to the National Physical Sciences Consortium for supporting this work.

to a partition in $\mathbf{F}$. We can define a matrix $\mathbf{W} \in \{0,1\}^{M \times R}$ that encodes the partitioning of $\mathbf{F}$ by placing a 1 in row m if column r of $\mathbf{F}$ is in partition m. The SBPT is constructed as

$$\boldsymbol{\mathcal{Y}} = \sum_{m=1}^{M} \mathbf{F}_m \mathbf{F}_m^T \circ \mathbf{c}_m = \sum_{m=1}^{M} \sum_{r=1}^{R} \mathbf{f}_r \mathbf{f}_r^T \circ \mathbf{C}\mathbf{w}_r = [\![\mathbf{F}, \mathbf{F}, \mathbf{CW}]\!] \tag{1}$$

where the right hand notation is the shorthand used for a CP decomposition. Conversion of the BPT to a CP model exposes the $\mathbf{W}$ partitioning matrix which will enable estimation of the partitions.

4. NON-BLIND SBPT DECOMPOSITION

We begin by devising a non-blind strategy for decomposing a SBPT (that is we know the structure of $\mathbf{W}$). Non-blind BPT algorithms have been proposed before as in [9], as have those for symmetric CP decompositions as in [2][12]. However, those have not been for for the SBPT case. In [8] we used a gradient-based approach to SBPT decomposition, but the slowness of convergence at high dimensions was prohibitive.

For the non-blind SBPT decomposition we take a cue from previous ALS algorithms where $\mathbf{F}$ is treated as two factor matrices for the purposes of solving

$$\begin{aligned}[\hat{\mathbf{F}}_A, \hat{\mathbf{F}}_B, \hat{\mathbf{C}}] = \underset{\mathbf{F}_A, \mathbf{F}_B, \mathbf{C}}{\arg\min} \, & \|\boldsymbol{\mathcal{Y}} - [\![\mathbf{F}_A, \mathbf{F}_B, \mathbf{CW}]\!]\|^2 \\ & + \lambda_F(\|\mathbf{F}_A\|_1 + \|\mathbf{F}_B\|_1) + \lambda_C \|\mathbf{C}\|_1. \end{aligned} \tag{2}$$

The addition of the L1 regularization terms is useful in improving the estimate when the factor matrices are sparse, e.g. in the spectrum sensing case where each signal in $\mathbf{F}$ occupies a small bandwidth, as well as encourage factor matrix separation.

Taking the alternating update strategy to the extreme of doing coordinate descent, we can minimize over one variable in each factor matrix at a time. For entries in the $\mathbf{F}_A$ matrix, this results in a solution of

$$\hat{f}_{Air} = S_{\lambda_F / \|\mathbf{u}_{Ar}\|^2} \left(\frac{\mathbf{e}_i^T \left(\mathbf{Y}_{(1)} - \tilde{\mathbf{F}}_A \mathbf{U}_A^T \right) \mathbf{u}_{Ar}}{\|\mathbf{u}_{Ar}\|^2} + \tilde{f}_{Air} \right) \tag{3}$$

where $\mathbf{e}_i$ is the vector of zeros with a 1 in entry i, $\tilde{\mathbf{F}}_A$ is the current estimate of $\mathbf{F}$, and $\mathbf{U}_A = \mathbf{CW} \odot \mathbf{F}_B$. If we take the Jacobian of $\boldsymbol{\mathcal{Y}} - [\![\mathbf{F}_A, \mathbf{F}_B, \mathbf{CW}]\!]$ with respect to f_{Air}, $1 \leq i \leq I$ we find that the tensor entries affected by element n of $\mathbf{f}_r$ do not intersect with the entries affected by f_{Amr}, $m \neq n$ meaning we can update the estimate of $\mathbf{F}_A$ columnwise by updating all variables in $\mathbf{f}_{Ar}$ in parallel as

$$\hat{\mathbf{f}}_{Ar} = S_{\lambda_F / \|\mathbf{u}_{Ar}\|^2} \left(\frac{\left(\mathbf{Y}_{(1)} - \tilde{\mathbf{F}}_A \mathbf{U}_A^T \right) \mathbf{u}_{Ar}}{\|\mathbf{u}_{Ar}\|^2} + \tilde{\mathbf{f}}_{Ar} \right). \tag{4}$$

Likewise, the update for $\mathbf{f}_{Br}$ is

$$\hat{\mathbf{f}}_{Br} = S_{\lambda_F / \|\mathbf{u}_{Ar}\|^2} \left(\frac{\left(\mathbf{Y}_{(2)} - \tilde{\mathbf{F}}_B \mathbf{U}_B^T \right) \mathbf{u}_{Br}}{\|\mathbf{u}_{Br}\|^2} + \tilde{\mathbf{f}}_{Br} \right) \tag{5}$$

where $\mathbf{U}_B = \mathbf{CW} \odot \mathbf{F}_A$. The update for $\mathbf{c}_m$ as

$$\hat{\mathbf{c}}_m = S_{\frac{\lambda_C}{\|\mathbf{u}_{Cm}\|^2}} \left(\frac{\left(\mathbf{Y}_{(3)} - \tilde{\mathbf{C}} \mathbf{W} \mathbf{U}_C^T \right) \mathbf{U}_C \mathbf{W}_{m:}^T}{\|\mathbf{W}_{m:} \mathbf{U}_C^T\|^2} + \tilde{\mathbf{c}}_m \right) \tag{6}$$

where $\mathbf{U}_C = \mathbf{F}_B \odot \mathbf{F}_A$.

Having treated the repeated factor matrix $\mathbf{F}$ as two different matrices for the purpose of finding a simple update equations leads to the question of how best to enforce the constraint that $\mathbf{F}_A = \mathbf{F}_B$. A common strategy in the symmetric CP case is to *not* enforce equality and hope that once the algorithm converges $\mathbf{F}_A \approx \mathbf{F}_B$ [2][11]. Empirically, we found that not including the equality constraint at all in the BPT decomposition rarely resulted in $\mathbf{F}_A \approx \mathbf{F}_B$. Instead we hard-constrain $\mathbf{F}_A = \mathbf{F}_B$ by projecting the outputs of (4) and (5) to what amounts to the average direction between the columns. We first need to align the columns by finding a scaling coefficient $\alpha_r = \arg\min_\alpha \||\alpha|\mathbf{f}_{Ar} - \frac{1}{\alpha}\mathbf{f}_{Br}\|^2$ which is given by

$$\alpha_r = \text{SGN}(\mathbf{f}_{Ar}^T \mathbf{f}_{Br}) \sqrt{\|\mathbf{f}_{Br}\| / \|\mathbf{f}_{Ar}\|} \tag{7}$$

where SGN is a modified signum function with $\text{SGN}(0) = 1$. Having found α, the solution is then updated as $\hat{\mathbf{f}}_{Ar} = \hat{\mathbf{f}}_{Br} = \frac{1}{2}|\alpha_r|\hat{\mathbf{f}}_{Ar} + \frac{1}{2\alpha_r}\hat{\mathbf{f}}_{Br}$.

The cases of $\|\mathbf{f}_{Ar}\|^2 = 0$ or $\|\mathbf{f}_{Br}\|^2 = 0$ must be handled, to avoid $\alpha = 0$ or being undefined. If either column has zero norm, then we rescale the corresponding column in the other matrix to have a norm of 1 and then continue on. One column having zero norm effectively reduces the rank of the tensor estimate, but the possibility it can reenter the active set is allowed for by normalizing the column in the other factor matrix and restricting the columns being updated to be $\mathbf{f}_{Ar} \in \mathcal{Q}_A = \{r : \|\mathbf{f}_{Br}\| \neq 0\}$ (with a similarly defined set $\mathcal{Q}_B$ for identifying the columns $\mathbf{f}_{Br}$ to update).

Since tensor decompositions have a scaling ambiguity leading to infinite solutions on a manifold, we constrain $\|\mathbf{c}_r\|^2 = 1$ and rescale the columns of $\mathbf{F}_A$ and $\mathbf{F}_B$ accordingly.

5. BLIND SBPT DECOMPOSITION

In the case where we do not *a priori* know the partitioning of the $\mathbf{F}$ matrix, we need to estimate the partitioning matrix $\mathbf{W}$. Continuing with the alternating method of the non-blind algorithm, we can introduce a step to estimate $\mathbf{W}$. We discussed that the structure of $\mathbf{W}$ is block diagonal in Sec. 3, however determining the ranks R_m is a difficult combinatorial problem, especially as R becomes large. Other work [13] has tried to solve similar diagonalization problems in the face of column permutations by introducing L1 regularization on $\mathbf{W}$ to select a small set of active variables, followed by $M \cdot R$ variable elimination tests to find the minimum solution. Our algorithm relies on finding a least squares fit for a transformed $\mathbf{W}$, and then finding partitions by constructing M clusters from the columns of the estimated matrix formed by $\mathbf{CW}$.

Input: $\mathcal{Y}$, M, R

Initialize: $\lambda_F \geq 0, \lambda_C \geq 0, \mathbf{F}_A, \mathbf{F}_B, \mathbf{C}, \mathbf{W}$

$\quad \tilde{\mathbf{F}}_A \leftarrow \mathbf{F}_A, \tilde{\mathbf{F}}_B \leftarrow \mathbf{F}_B, \tilde{\mathbf{C}} \leftarrow \mathbf{C}, \tilde{\mathbf{W}} \leftarrow \mathbf{W}$

$\quad$ **while** not converged **do**

$\quad\quad \tilde{\mathbf{f}}_{Ar} \leftarrow$ update by (4) **for** $r \in \mathcal{Q}_A$

$\quad\quad \tilde{\mathbf{f}}_{Br} \leftarrow$ update by (5) **for** $r \in \mathcal{Q}_B$

$\quad\quad \alpha_r \leftarrow$ update by (7) for $r = 1, \ldots, R$

$\quad\quad \hat{\mathbf{f}}_{Ar}, \hat{\mathbf{f}}_{Br} \leftarrow \frac{1}{2}\left(|\alpha_r|\mathbf{f}_{Ar} + \mathbf{f}_{Br}/\alpha_r\right)$ for $r \in \mathcal{Q}_A \cap \mathcal{Q}_B$

$\quad\quad \hat{\mathbf{f}}_{Ar} \leftarrow$ normalize for $r \notin \mathcal{Q}_A$

$\quad\quad \hat{\mathbf{f}}_{Br} \leftarrow$ normalize for $r \notin \mathcal{Q}_B$

$\quad\quad \tilde{\mathbf{w}}_r \leftarrow \mathrm{sign}(\alpha_r)\tilde{\mathbf{w}}_r$

$\quad\quad \tilde{\mathbf{c}}_m \leftarrow$ update by (6) **for** $m = 1, \ldots, M$

$\quad\quad \tilde{\mathbf{C}} \leftarrow \mathbf{Q}$ from QR decomposition $\mathbf{C} = \mathbf{QR}$

$\quad\quad \tilde{\mathbf{W}} \leftarrow \mathbf{RW}$ from QR decomposition $\mathbf{C} = \mathbf{QR}$

$\quad\quad \tilde{\mathbf{w}}_r \leftarrow$ update by (8) **for** $r \in \mathcal{Q}_A \cap \mathcal{Q}_B$

$\quad\quad \gamma \leftarrow \sqrt{\mathbf{1}^T(\tilde{\mathbf{W}} * \tilde{\mathbf{W}})}$

$\quad\quad \tilde{\mathbf{W}} = \mathbf{W} \cdot \mathrm{diag}(\gamma)^{-1}$

$\quad\quad \tilde{\mathbf{F}}_A \leftarrow \hat{\mathbf{F}}_A \cdot \mathrm{diag}(\sqrt{\gamma})$

$\quad\quad \tilde{\mathbf{F}}_B \leftarrow \hat{\mathbf{F}}_B \cdot \mathrm{diag}(\sqrt{\gamma})$

$\quad$ **end while**

$\quad \mathbf{f}_r \leftarrow \tilde{\mathbf{F}}_A$ or $\tilde{\mathbf{F}}_B$ for $r \in \mathcal{Q}_A \cap \mathcal{Q}_B$, $\mathbf{0}$ for $r \notin \mathcal{Q}_A \cap \mathcal{Q}_B$

$\quad [\mathbf{C}, \mathbf{d}] \leftarrow M$ cluster centers and assignments from $\bar{\mathbf{C}} = \tilde{\mathbf{C}}\tilde{\mathbf{W}}$

$\quad \mathbf{W} \leftarrow$ from cluster center assignment $\mathbf{d}$

$\quad$ **Optional:** Refine $\mathbf{F}$, $\mathbf{C}$ estimates with non-blind algorithm

Output: $\mathbf{F}, \mathbf{C}, \mathbf{W}$

Fig. 1. Algorithm for blind SBPT decomposition.

Eqs. (4)-(6) are used unchanged, but we add constraints and an equation to update the estimate for $\mathbf{W}$ in this section.

While we have defined $\mathbf{W}$ to be a matrix consisting of 0's and 1's, we can ignore this constraint and simplify computations by introducing some transformations to the problem. Namely, we constrain a new matrix $\mathbf{C}'$ to be an orthonormal matrix. If $\mathbf{C}$ and $\mathbf{W}$ are the true factor and partitioning matrices, with $\mathbf{C}' = \mathbf{Q}$ then $\mathbf{W}' = \mathbf{RW}$ where $\mathbf{C} = \mathbf{QR}$, a QR decomposition. The effect is that $\mathbf{W}'$ is not sparse and we can avoid a high-dimensional inversion of $\mathbf{C}'$ because it is unitary.

We could compute an estimate for all entries in $\mathbf{W}$ simultaneously, but the inversion of the $\mathbf{F}_B \odot \mathbf{F}_A$ matrix of size $I^2 \times R$ becomes prohibitive as the tensor dimensions grow. Instead, similar to the non-blind algorithm, we update the solution column-wise for $\hat{\mathbf{w}}'_r$ which a derivation (assuming $\lambda_C = 0$) reveals has the solution

$$\hat{\mathbf{w}}'_r = \frac{\mathbf{C}'^T\left(\mathbf{Y}_{(3)} - \mathbf{C}'\tilde{\mathbf{W}}'\mathbf{U}_C^T\right)\mathbf{u}_{Cr}}{\|\mathbf{u}_{Cr}\|^2} + \tilde{\mathbf{w}}'_r. \tag{8}$$

Again, because of the scaling indeterminancy, we constrain $\|\mathbf{w}'_r\|^2 = 1$, moving all scaling into the estimate of $\mathbf{F}$. Similar to the divide-by-zero concerns in the previous section, we only update columns $\mathbf{w}'_r$ for $r \in \mathcal{Q}_A \cap \mathcal{Q}_B$.

The decomposition terminates with estimates $\hat{\mathbf{C}}'$ and $\hat{\mathbf{W}}'$, but because of the constraint on $\mathbf{C}'$ the estimates are not the final solutions nor do they give partitioning information. Assuming we know the number of partitions M, we perform clustering on the R column vectors in the matrix $\bar{\mathbf{C}} = \hat{\mathbf{C}}'\hat{\mathbf{W}}'$

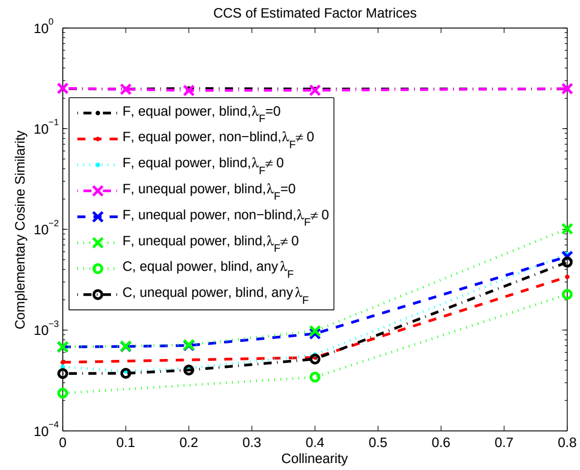

Fig. 2. The congruence of the factor matrix estimates is a function of the collinearity, with performance worsening as it increases. A non-zero value for λ_F substantially improves the estimate of $\mathbf{F}$, but has an insignificant effect on $\mathbf{C}$. Congruence worsens in the unequal power case as the estimates of the lower-power components are poor compared to the higher-power components. The estimate of $\mathbf{C}$ has better congruence than that of $\mathbf{F}$ as it has more data to be computed from.

such that M cluster centers are found with the columns each assigned to one. Clustering strategies such as k-means with $k = M$ or mean-shift clustering [14] with an appropriately chosen bandwidth parameter can be used to find the M clusters and perform assignment. We choose to use the mean-shift algorithm.

The output of the clustering algorithm is a matrix $\hat{\mathbf{C}}$ of M cluster centers, which are the final estimate for $\mathbf{C}$, and a vector $\mathbf{d}$ with R elements assigning the columns of $\bar{\mathbf{C}}$ to the centers. Using this information we construct the estimated partitioning matrix $\mathbf{W}$ by placing a 1 in entry $w_{r,d(r)}$. Once the partitioning matrix is estimated, the estimates for $\mathbf{C}$ and $\mathbf{F}$ can be refined by using the non-blind SBPT decompostion algorithm of Sec. 4. The algorithm steps are listed in Fig. 1.

6. SIMULATIONS

We evaluated the performance of this SBPT decomposition strategy through simulation. To do this, we generate tensor $\mathcal{Y}$ having factor matrices $\mathbf{F}$, $\mathbf{C}$ in a manner similar to that in [15], with modifications for the SBPT structure to create rank-$(R_m, R_m, 1)$ SBPTs with $R_m = R/M$. This is done by starting with random column-wise orthonormal matrices $\mathbf{H}_l$, $l = 1, \ldots, R_m$ of dimension $(I/R_m) \times M$ and a Cholesky factor $\mathbf{S}$ having ones on the diagonal and a desired congruence value c on the off-diagonal entries. The loading matrix is then $\mathbf{F} = \mathrm{blkdiag}(\mathbf{H}_1\mathbf{S}, \ldots, \mathbf{H}_{R_m}\mathbf{S})$ with a partition matrix $\mathbf{W} = \left[\mathbf{I}_{R_m}| \ldots |\mathbf{I}_{R_m}\right]$ consisting of M horizontally concatenated $R_m \times R_m$ identity matrices. Generating the tensor in this manner allows us to control the congruency as well as the sparsity of the factor matrices. A similar procedure is used to generate the $\mathbf{C}$ matrix. A vector $\mathbf{p}$ scales

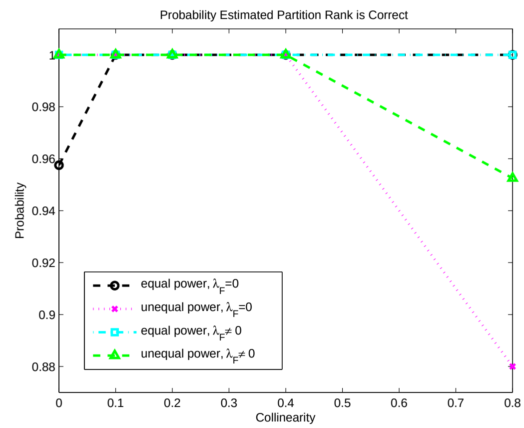

Fig. 3. When the tensor factors contain similar power levels, the success rate of estimating the correct partition rank is high across the range of collinearity values. When the factors contain unequal power the accuracy decreases at high values of collinearity and is affected by the choice of λ_F. The dip at the $c = 0$, $\lambda_F = 0$ equal power case is from several simulation runs hitting the iteration limit before the columns of $\mathbf{C}$ iterated close to their cluster centers.

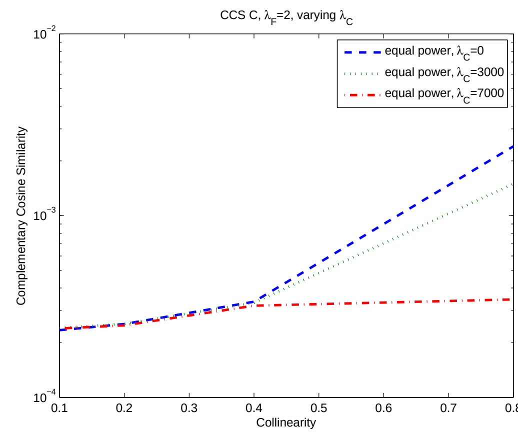

Fig. 4. Even though $\mathbf{C}$ in our simulations is not sparse, the congruence of $\hat{\mathbf{C}}$ was improved at higher collinearity values by refining the blind estimate using the non-blind algorithm with a positive, non-zero value for λ_C. The curve for the CCS of $\mathbf{C}$ from the blind algorithm coincides with the $\lambda_C = 0$ curve. A less significant improvement in the CCS for $\hat{\mathbf{F}}$ is also observed, but not plotted.

each partition to set its power so that that generated tensor is $\boldsymbol{\mathcal{Y}} = [\![\mathbf{F}, \mathbf{F}, \mathbf{C} \cdot \mathrm{diag}(\mathbf{p}) \cdot \mathbf{W}]\!]$. The input to the algorithm is $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{Y}} + \boldsymbol{\mathcal{N}}$ where $\boldsymbol{\mathcal{N}}$ is a Gaussian noise tensor scaled to such that the signal-to-noise ratio is $SNR = \|\boldsymbol{\mathcal{Y}}\|/\|\boldsymbol{\mathcal{N}}\|$.

The tensors we generated in simulation were of dimension $20 \times 20 \times 20$ with $R = 16$, $M = 4$, and $R_m = 4$, $m = 1, \ldots, 4$. The SNR in all cases was 15dB. The iterations were considered to have converged when the relative change in variables was less than 10^{-7} and the iteration limit was set to 50,000. Two power configurations were considered: in the first, all tensors had the same power so $\mathbf{p} = 125 \cdot \mathbf{1}_M$, while in the second $\mathbf{p} = \begin{bmatrix} 125 & 100 & 75 & 50 \end{bmatrix}^T$, a difference of 3.98dB between the strongest and weakest components. The collinearity between components was varied between 0 and 0.8 and the values of λ_F took values 0, 1, 2, and 3. 100 samples were generated for each data point. The accuracy of the decomposition is evaluated by the average complementary cosine similarity (CCS) measure between the true factor matrix $\mathbf{A}$ with R columns and its estimate $\hat{\mathbf{A}}$

$$CCS(\mathbf{A}, \hat{\mathbf{A}}) = \arg\min_{\boldsymbol{\pi}} \frac{1}{R} \sum_{r=1}^{R} 1 - \left| \frac{\mathbf{a}_r^T \hat{\mathbf{a}}_{\pi(r)}}{\|\mathbf{a}_r\| \|\hat{\mathbf{a}}_{\pi(r)}\|} \right| \quad (9)$$

where $\hat{\mathbf{A}}_\pi$ is a reordering of the columns of the estimate. A CCS value of 0 is a perfect match, while a value of 1 is no match.

Fig. 2 illustrates the performance of the decomposition in both power configurations considered. For this size tensor, across all collinearities, convergence was achieved in both blind and non-blind cases in about 42,000 iterations, on average. We see that an non-zero value for λ_F substantially improves the congruency of the estimate of $\mathbf{F}$. The CCS across the range of non-zero λ_F values considered varied somewhat, but the difference is not as substantial as compared to $\lambda_F = 0$.

The plots for the CCS for $\mathbf{C}$ were similar regardless of the value for λ_F, and the non-blind results for $\mathbf{C}$ were so similar to the blind that we did not plot them. The blind and non-blind algorithms have similar recovery performance at low values of collinearity, while at high collinearity the non-blind version outperforms the blind. There is a clear decrease in performance when the tensor components have unequal power, a result of the components containing more energy being better estimated, while the lower power components are less accurately estimated. The effect of the decreased accuracy can also be seen in Fig. 3 where the equal power case estimates the correct partition rank with high-probability across the range of collinearity while the unequal configuration suffers from decreased success at high values of collinearity.

The CCS of the estimates for $\mathbf{F}$ and $\mathbf{C}$ do not significantly improve with refinement of the blind algorithm output by the non-blind algorithm when $\lambda_C = 0$. Fig. 4 illustrates how a non-zero value for λ_C can improve that factor matrix's CCS in the refinement stage at high collinearity values, even for non-sparse $\mathbf{C}$ as the L1 regularization encourages the separation of factors better than a least-squares fit by .

7. CONCLUSION

We have proposed non-blind and blind column-wise alternating tensor decomposition algorithms for symmetric block-partitioned tensors. The column-wise approach, with additional constraints for the blind algorithm, reduces computations by avoiding large matrix inversions. Simulation results on synthetic SBPTs with controlled sparsity and inter-column collinearity values demonstrate that the algorithms are capable of estimating the factor matrices with low CCS values across all collinearities, and high probability of correct partition rank estimates at lower collinearity values.

8. REFERENCES

[1] R. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.

[2] 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, no. 3, pp. 283–319, 1970.

[3] L. R. Tucker, "Implications of factor analysis of three-way matrices for measurement of change," *Problems in measuring change*, vol. 15, pp. 122–137, 1963.

[4] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms-Part II: Definitions and uniqueness," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1033–1066, 2008.

[5] J. Kruskal, "Three way arrays: Rank and uniqueness of of trilinear decompositions with application to arithmetic complexity and statistics," *Linear Algebra and its Applications*, vol. 18, no. 2, pp. 95–138, 1977.

[6] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of n-way arrays," *Journal of chemometrics*, vol. 14, no. 3, pp. 229–239, 2000.

[7] C. Mueller-Smith and P. Spasojević, "Single Sensor Blind Time-Frequency Activity Estimation of a Mixture of Radio Signals via CP Tensor Decomposition," *2014 IEEE Military Communications Conference*, Oct. 2014.

[8] C. Mueller-Smith and P. Spasojevic, "Blind block partitioned tensor decomposition for radio activity estimation by a single sensor," *Conference on Information Sciences and Systems*, pp. 1–6, 2015.

[9] L. De Lathauwer and D. Nion, "Decompositions of a Higher-Order Tensor in Block Terms—Part III: Alternating Least Squares Algorithms," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1067–1083, Jan. 2008.

[10] G. Ivkovic, P. Spasojevic, and I. Seskar, "Single sensor radio scene analysis for packet based radio signals," *Information Theory and Applications Workshop 2010*, pp. 1–10, 2010.

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

[12] J. Brachat, P. Comon, B. Mourrain, and E. Tsigaridas, "Symmetric tensor decomposition," *Linear Algebra and its Applications*, vol. 433, no. 11-12, p. 1851, Dec. 2010.

[13] G. Ivkovic, "Localization of packet based radio transmitters in space, time, and frequency," Ph.D. dissertation, Rutgers University, New Brunswick, NJ, 2012.

[14] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[15] R. Bro and G. Tomasi, "A comparison of algorithms for fitting the PARAFAC model," *Computational Statistics & Data Analysis*, 2006.