EXTENSION OF THE SEMI-ALGEBRAIC FRAMEWORK FOR APPROXIMATE CP DECOMPOSITIONS VIA NON-SYMMETRIC SIMULTANEOUS MATRIX DIAGONALIZATION

Kristina Naskovska, Martin Haardt

Ilmenau University of Technology
Communications Research Laboratory
D-98684 Ilmenau, Germany

Petr Tichavský[⋆]*, Gilles Chabriel*[†]*, Jean Barrère*[†]

[⋆] Institute of Information Theory and Automation
182 08 Prague 8, Czech Republic
[†] Aix Marseille Université, CNRS,
Université de Toulon, IM2NP UMR 7334,
CS 60584 - 83041 Toulon Cedex 9, France

ABSTRACT

With the increased importance of the CP decomposition (CANDECOMP / PARAFAC decomposition), efficient methods for its calculation are necessary. In this paper we present an extension of the SECSI (SEmi-algebraic framework for approximate CP decomposition via SImultaneous matrix diagonalization) that is based on new non-symmetric SMDs (Simultaneous Matrix Diagonalizations). Moreover, two different algorithms to calculate non-symmetric SMDs are presented as examples, the TEDIA (TEnsor DIAgonalization) algorithm and the IDIEM-NS (Improved DIagonalization using Equivalent Matrices-Non Symmetric) algorithm. The SECSI-TEDIA framework has an increased computational complexity but often achieves a better performance than the original SECSI framework. On the other hand, the SECSI-IDIEM framework offers a lower computational complexity while sacrificing some performance accuracy.

Index Terms— CP decomposition, semi-algebraic framework, non-symmetric simultaneous matrix diagonalization, PARAFAC

1. INTRODUCTION

Tensors provide a useful tool for the analysis of multidimensional data. A comprehensive review of tensor concepts is provided in [1]. Tensors have a very broad range of applications especially in signal processing such as compressed sensing, processing of big data, blind source separation and many more [2]. Often a tensor should be decomposed into the minimum number of rank one components. This decomposition is know as PARAFAC (PARallel FACtors) , CANDECOMP (Canonical Decomposition), or CP (CANDECOMP/PARAFAC).

The CP decomposition is often calculated via the iterative multilinear-ALS (Alternating Least Square) algorithm [1]. ALS based algorithms require a lot of iterations to calculate the CP decomposition and there is no convergence guarantee. Moreover, ALS based algorithms perform less accurate for ill-conditioned scenarios, for instance, if the columns of the factor matrices are correlated.

There are already many ALS based algorithms for calculating the CP decomposition such as the ones presented in [3] that either introduce constraints to reduce the number of iterations or are based on line search. Alternatively, semi-algebraic solutions have been proposed in the literature based on SMDs (Simultaneous Matrix Diagonalizations), that are also called JMDs (Joint Matrix Diagonalizations). Such examples include [4], [5], [6], [7], and [8]. The SECSI framework [7] calculates all possible SMDs, and then selects the best available solution in a final step via appropriate heuristics. All of these algorithms consider symmetric SMDs [9], whereas in this paper we propose a semi-algebraic framework for approximate CP decompositions via non-symmetric SMDs. Moreover, we consider two different algorithms to calculate the non-symmetric SMDs, the TEDIA algorithm [10] and an extended version of the IDIEM algorithm [11], [12] that provides a closed-form solution for the non-symmetric SMD problem. In this paper we consider the computation of a three-way tensor. It is easy to generalize this concept to higher order tensors by combining the presented SECSI framework with generalized unfoldings as discussed in [8].

In this paper the following notation is used. Scalars are denoted either as capitals or lower-case italic letters, A, a. Vectors and matrices, are denoted as bold-face capital and lower-case letters, $\boldsymbol{a}$, $\boldsymbol{A}$, respectively. Finally, tensors are represented by bold-face calligraphic letter $\boldsymbol{\mathcal{A}}$. The following superscripts, ^{T}, ^{H}, ${}^{-1}$, and ${}^+$ denote transposition, Hermitian transposition, matrix inversion and Moore-Penrose pseudo matrix inversion, respectively. The outer product, Kronecker product and Khatri-Rao product are denoted as $\circ$, $\otimes$, and $\diamond$, respectively. The operators $||.||_{\mathrm{F}}^2$ and $||.||_{\mathrm{H}}^2$ denote the Frobenius norm and the Higher order norm, respectively.

Moreover, for the tensors operations we use the following notation. An n-mode product between a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \cdots \times I_N}$ and a matrix $\boldsymbol{B} \in \mathbb{C}^{J \times I_n}$ is defined as $\boldsymbol{\mathcal{A}} \times_n \boldsymbol{B}$, for $n = 1, 2, \ldots N$ [13]. A super-diagonal or identity N-way tensor of dimensions $R \times R \ldots \times R$ is denoted as $\boldsymbol{\mathcal{I}}_{N,R}$.

2. TENSOR DECOMPOSITIONS

The CP decomposition of a low rank tensor $\boldsymbol{\mathcal{X}}_0 \in \mathbb{C}^{I \times J \times K}$ is defined as [1], [2]

$$\boldsymbol{\mathcal{X}}_0 = \sum_{r=1}^{R} \boldsymbol{f}_1^{(r)} \circ \boldsymbol{f}_2^{(r)} \circ \boldsymbol{f}_3^{(r)} \tag{1}$$

$$= \boldsymbol{\mathcal{I}}_{3,R} \times_1 \boldsymbol{F}_1 \times_2 \boldsymbol{F}_2 \times_3 \boldsymbol{F}_3. \tag{2}$$

The CP decomposition decomposes a given tensor into the sum of the minimum number of rank one tensors. According to equation (1)

The work of P. Tichavský was supported by the Czech Science Foundation through Project No. 14-13713S.

the tensor rank is equal to R. The vectors $\boldsymbol{f}_n^{(r)}$ are the corresponding columns of the factor matrices $\boldsymbol{F}_n$, for $n = 1, 2, 3$.

The CP decomposition is essentially unique under mild conditions, which means that the factor matrices $\boldsymbol{F}_n$ can be identified up to a permutation and scaling ambiguity.

In practice we can only observe a noise corrupted version of the low rank tensor $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}}$, where $\boldsymbol{\mathcal{N}}$ contains uncorrelated zero mean circularly symmetric Gaussian noise. Hence, we have to calculate a rank R approximation of $\boldsymbol{\mathcal{X}}$,

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{I}}_{3,R} \times_1 \boldsymbol{F}_1 \times_2 \boldsymbol{F}_2 \times_3 \boldsymbol{F}_3. \tag{3}$$

Another, multilinear extension of the SVD (Singular Value Decomposition) is the HOSVD (Higher Order Singular Value Decomposition) which is much easier to calculate than the CP decomposition. The HOSVD of the rank R tensor $\boldsymbol{\mathcal{X}}_0 \in \mathbb{C}^{I \times J \times K}$ is given by [13],

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{S}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3 \tag{4}$$

where $\boldsymbol{\mathcal{S}} \in \mathbb{C}^{I \times J \times K}$ is the core tensor. The matrices $\boldsymbol{U}_1 \in \mathbb{C}^{I \times I}$, $\boldsymbol{U}_2 \in \mathbb{C}^{J \times J}$ and $\boldsymbol{U}_3 \in \mathbb{C}^{K \times K}$ are unitary matrices which span the column space of the n-mode unfolding of $\boldsymbol{\mathcal{X}}_0$, for $n = 1, 2, 3$ respectively. Accordingly, the truncated HOSVD is defined as

$$\boldsymbol{\mathcal{X}}_0^{[s]} = \boldsymbol{\mathcal{S}}^{[s]} \times_1 \boldsymbol{U}_1^{[s]} \times_2 \boldsymbol{U}_2^{[s]} \times_3 \boldsymbol{U}_3^{[s]} \tag{5}$$

where $\boldsymbol{\mathcal{S}}^{[s]} \in \mathbb{C}^{R \times R \times R}$ is a truncated core tensor and the matrices $\boldsymbol{U}_1 \in \mathbb{C}^{I \times R}$, $\boldsymbol{U}_2 \in \mathbb{C}^{J \times R}$ and $\boldsymbol{U}_3 \in \mathbb{C}^{K \times R}$ have unitary columns.

3. SEMI-ALGEBRAIC FRAMEWORK FOR APPROXIMATE CP DECOMPOSITION VIA NON-SYMMETRIC SIMULTANEOUS MATRIX DIAGONALIZATION

Within this section we will point out the differences between the original SECSI framework [5], [7], and the modified framework which is a point of interest in this paper. The whole derivation will not be provided, because it follows the derivation of the original SECSI framework.

The SECSI framework starts by computing the truncated HOSVD of the noise corrupted tensor $\boldsymbol{\mathcal{X}}$ in order to calculate an approximate low rank CP decomposition. textcolormycl3Thereby we get an approximation of

$$\boldsymbol{\mathcal{X}}_0 = \left(\boldsymbol{\mathcal{S}}^{[s]} \times_3 \boldsymbol{U}_3^{[s]}\right) \times_1 \boldsymbol{U}_1^{[s]} \times_2 \boldsymbol{U}_2^{[s]} \tag{6}$$

$$= \left(\boldsymbol{\mathcal{I}}_{3,R} \times_3 \underbrace{(\boldsymbol{U}_3^{[s]} \cdot \boldsymbol{T}_3)}_{\boldsymbol{F}_3}\right) \times_1 \underbrace{(\boldsymbol{U}_1^{[s]} \cdot \boldsymbol{T}_1)}_{\boldsymbol{F}_1} \times_2 \underbrace{(\boldsymbol{U}_2^{[s]} \cdot \boldsymbol{T}_2)}_{\boldsymbol{F}_2} \tag{7}$$

where equations (6) and (7) represent the truncated HOSVD and the CP decomposition of the noiseless tensor, respectively. The invertible matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$ and $\boldsymbol{T}_3$ of dimensions $R \times R$ diagonalize the truncated core tensor $\boldsymbol{\mathcal{S}}^{[s]}$ as shown in [5].

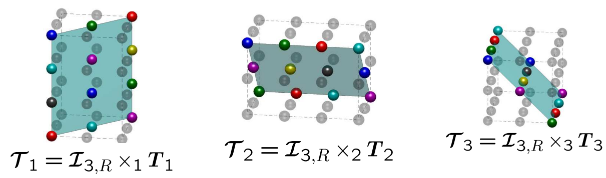
Therefore, it follows that

$$\boldsymbol{\mathcal{S}}^{[s]} = (\boldsymbol{\mathcal{I}}_{3,R} \times_3 \boldsymbol{T}_3) \times_1 \boldsymbol{T}_1 \times_2 \boldsymbol{T}_2. \tag{8}$$

In contrast to the original SECSI framework, we do not calculate 6 sets of symmetric SMDs but only 3 sets of non-symmetric SMDs, for a smaller number of matrices. To this end, we define the tensor $\boldsymbol{\mathcal{T}}_3 = (\boldsymbol{\mathcal{I}}_{3,R} \times_3 \boldsymbol{T}_3)$, as depicted in Fig. 1(c). Notice that $\boldsymbol{\mathcal{T}}_3$ contains diagonal slices along the third mode. Hence, we need to diagonalize the truncated core tensor $\boldsymbol{\mathcal{S}}^{[s]}$, or in other words we need to estimate the matrices $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$ that diagonalize the tensor $\boldsymbol{\mathcal{S}}^{[s]}$.

$$\boldsymbol{\mathcal{S}}^{[s]} \times_1 \boldsymbol{T}_1^{-1} \times_2 \boldsymbol{T}_2^{-1} = \boldsymbol{\mathcal{T}}_3 \tag{9}$$

In order to generate the set of matrices that we can use for non-symmetric SMD, the truncated core tensor has to be sliced. When we use the third mode of the tensor as presented up to now, the diagonal matrices are aligned along the 3-mode slices of the tensor. In order to select the slices from the 3-mode of the tensor we multiply along the 3-mode with a transpose of a vector $\boldsymbol{e}_k$ that is the k-th column of a $R \times R$ identity matrix. Therefore, each of the corresponding slices is defined as $\boldsymbol{S}_k^{[s]} = \boldsymbol{\mathcal{S}}^{[s]} \times_3 \boldsymbol{e}_k^T$ and $\boldsymbol{T}_{3,k} = \boldsymbol{\mathcal{T}}_3 \times_3 \boldsymbol{e}_k^T$ for the left and right hand side of equation (9).



(a) 1-mode (b) 2-mode (c) 3-mode

Fig. 1. Diagonalized core tensor for mode 1, 2 and 3.

The described slicing of the truncated core tensor results in the following set of equations,

$$\boldsymbol{T}_1^{-1} \cdot \boldsymbol{S}_k^{[s]} \cdot \boldsymbol{T}_2^{-1} = \boldsymbol{T}_{3,k}, \quad k = 1, 2, \ldots R. \tag{10}$$

Equation (10) represents a non-symmetric SMD problem. Note that we have a set of R equations instead of the K ($K \geq R$) equations of the original SECSI framework, which reduces the computational complexity of the non-symmetric SMD. Therefore, in this framework we use new algorithms for the non-symmetric SMD, which are presented in the following subsections. Thereby, an estimate of the matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ is achieved, while $\boldsymbol{T}_3$ is calculated from $\boldsymbol{\mathcal{T}}_3$, as depicted in Fig. 1(c).

Finally, from the knowledge of these three matrices, the factor matrices of the CP decomposition can be estimated, which is our final goal. From equation (7) it follows that

$$\hat{\boldsymbol{F}}_{1,\mathrm{I}} = \boldsymbol{U}_1^{[S]} \cdot \boldsymbol{T}_1 \tag{11}$$

$$\hat{\boldsymbol{F}}_{2,\mathrm{I}} = \boldsymbol{U}_2^{[S]} \cdot \boldsymbol{T}_2 \tag{12}$$

$$\hat{\boldsymbol{F}}_{3,\mathrm{I}} = \boldsymbol{U}_3^{[S]} \cdot \boldsymbol{T}_3. \tag{13}$$

The two additional tensor modes can be exploited such that 2 more sets of factor matrices are estimated, see Fig. 1. Accordingly, the core tensor should be sliced along its 1-mode and 2-mode, and then diagonalized via non-symmetric SMDs. Therefore, we get a set of estimated factor matrices $\hat{\boldsymbol{F}}_{1,\mathrm{I}}$, $\hat{\boldsymbol{F}}_{1,\mathrm{II}}$, $\hat{\boldsymbol{F}}_{1,\mathrm{III}}$, $\hat{\boldsymbol{F}}_{2,\mathrm{I}}$, $\hat{\boldsymbol{F}}_{2,\mathrm{II}}$, $\hat{\boldsymbol{F}}_{2,\mathrm{III}}$, $\hat{\boldsymbol{F}}_{3,\mathrm{I}}$, $\hat{\boldsymbol{F}}_{3,\mathrm{II}}$, and $\hat{\boldsymbol{F}}_{3,\mathrm{III}}$. From this set of estimated factor matrices different combinations can be selected, while searching for the best available solution. The different combinations lead to different heuristics, such as BM (Best Matching) and RES (Residuals) [7]. The BM solves all the SMDs and the final estimate is the one that leads to the lowest reconstruction error. The reconstruction error is calculated according to

$$\mathrm{RSE} = \frac{\left|\left|\hat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}\right|\right|_{\mathrm{H}}^2}{||\boldsymbol{\mathcal{X}}||_{\mathrm{H}}^2}. \quad (14)$$

On the other hand, RES also solves all SMDs, but as a final estimate we choose the resulting factor matrices of the non-symmetric SMD that has the smallest residual error. Note that the original SECSI framework has 6 estimates of the factor matrices due to the 6 symmetric SMDs, whereas we only have 3 non-symmetric SMDs.

3.1. TEDIA

TEDIA is an algorithm that solves the non-symmetric SMD problem [10], and we propose it as an option for solving the non-symmetric diagonalization problem within the SECSI framework. The goal of TEDIA is to find non-orthogonal matrices $\boldsymbol{A}_L \in \mathbb{C}^{R\times R}$ and $\boldsymbol{A}_R \in \mathbb{C}^{R\times R}$ that diagonalize the set of matrices $\boldsymbol{M}_k \in \mathbb{C}^{R\times R}$, resulting in a set of diagonal matrices $\boldsymbol{D}_k \in \mathbb{C}^{R\times R}$, for $k = 1, 2, \ldots K$.

$$\boldsymbol{D}_k = \boldsymbol{A}_L^{-1} \cdot \boldsymbol{M}_k \cdot \boldsymbol{A}_R^{-1}, \quad k = 1, 2, \ldots K. \quad (15)$$

Note that, $\boldsymbol{A}_L$, $\boldsymbol{A}_R$, $\boldsymbol{D}_k$ and $\boldsymbol{M}_k$, correspond to $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, $\boldsymbol{T}_{3,k}$ and $\boldsymbol{S}_{\boldsymbol{k}}^{[\boldsymbol{s}]}$ in equation (10), respectively. TEDIA does not try to minimize the off diagonal elements but rather to achieve a block-revealing condition, ideally leading to a diagonalized tensor. The algorithm is based on a search for elementary rotations that are applied to the matrices $\boldsymbol{A}_L$ and $\boldsymbol{A}_R$ and minimize the off diagonal elements of $\boldsymbol{M}_{\boldsymbol{k}}$ based on a damped Gauss-Newton method.

The TEDIA algorithm can be implemented in either a sequential or a parallel fashion and its main computational complexity comes from the different sweeps and the calculation of the Hessian matrix.

3.2. IDIEM-NS

Although the IDIEM algorithm [11] was initially proposed for symmetric approximate diagonalization, it can deal with the non-symmetrical problem as well [12]. IDIEM provides an approximate closed form solution for the minimization of the following so-called direct LS (Least Squares) cost function

$$\sum_{k=1}^{K} ||\boldsymbol{M}_k - \boldsymbol{A}_L \boldsymbol{D}_k \boldsymbol{A}_R||_{\mathrm{F}}^2 \quad (16)$$

where the matrices $\boldsymbol{A}_L$ and $\boldsymbol{A}_R$ of size $R \times R$ are the left and right-diagonalizer, respectively. These two matrices diagonalize the set of matrices $\boldsymbol{M}_k$, resulting in a set of diagonal matrices $\boldsymbol{D}_k$, for $k = 1, 2, ..., K$. Since IDIEM does not assume any explicit link between the two diagonalizers, the right-diagonalizer is simply obtained by using the rows of the R matrices $\tilde{\mathbf{R}}_m$ instead of columns in Step 4 of Table I in [11]. Where, the matrix $\tilde{\mathbf{R}}_m$ represents the m-th eigenvector of $\tilde{\boldsymbol{M}} = \sum_{k=1}^{K} \mathrm{vec}(\boldsymbol{M}_k)\mathrm{vec}(\boldsymbol{M}_k)^H$ by the means of inverse vector operation.

We propose this algorithm, which is called IDIEM-NS (IDIEM Non-Symmetric), because it is not iterative and therefore very fast and computationally efficient. Its closed form solution is a very practical choice for the non-symmetric SECSI framework.

4. SIMULATION RESULTS

In this section the non-symmetric extension of SECSI with its two implementations based on the TEDIA and the IDIEM-NS algorithm is compared to the original framework. For this reason, we have computed Monte Carlo simulation using 1000 realizations.

For simulation purposes two different, real-valued tensors, of size $4 \times 7 \times 3$ with tensor rank $R = 3$ have been designed. Each of the tensors is designed according to the CP decomposition.

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{I}}_{3,3} \times_1 \boldsymbol{F}_1 \times_2 \boldsymbol{F}_2 \times_3 \boldsymbol{F}_3 \quad (17)$$

where the factor matrices $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, and $\boldsymbol{F}_3$ have i.i.d. zero mean Gaussian distributed random entries with variance one. Moreover, we want one of the tensors to have correlated factor matrices, therefore we add correlation via

$$\boldsymbol{F}_n \leftarrow \boldsymbol{F}_n \cdot \boldsymbol{R}(\rho_n) \quad (18)$$

$$\boldsymbol{R}(\rho) = (1-\rho) \cdot \boldsymbol{I}_{R,R} + \frac{\rho}{R} \cdot \mathbf{1}_{R\times R}, \quad (19)$$

where $\boldsymbol{R}(\rho)$ is the correlation matrix with correlation factor ρ and $\mathbf{1}_{R\times R}$ denotes a matrix of ones. The second tensor has correlated factor matrices, with correlation coefficients $\rho_1 = 0.9$, $\rho_2 = 0.1$ and $\rho_3 = 0.1$ for $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, and $\boldsymbol{F}_3$, respectively.

Finally, the synthetic data is generated by adding i.i.d. zero mean Gaussian noise with variance σ_n^2. The resulting SNR (Signal to Noise Ratio) for the noisy tensor $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}}$ is SNR$= \sigma_n^{-2}$.

In the simulation results the TMSFE (Total relative Mean Square Factor Error)

$$\mathrm{TMSFE} = \mathbb{E}\left\{\sum_{n=1}^{N} \min_{\boldsymbol{P}_n \in \mathcal{M}_{\mathrm{PD}}(R)} \frac{\left|\left|\hat{\boldsymbol{F}}_n \cdot \boldsymbol{P}_n - \boldsymbol{F}_n\right|\right|_{\mathrm{F}}^2}{||\boldsymbol{F}_n||_{\mathrm{F}}^2}\right\} \quad (20)$$

is used as an accuracy measure, where $\mathcal{M}_{\mathrm{PD}}(R)$ is a set of permuted diagonal matrices that resolves the permutation ambiguity of the CP decomposition.

Since the SECSI framework has already been compared to the state of the art algorithms for various scenarios, we only compare our proposed framework to the original SECSI framework in [7], [5]. In the simulation results an accuracy and computational time comparison of the SECSI-IDIEM-NS, SECSI-TEDIA, SECSI, and SECSI Truncated is provided. SECSI-IDIEM-NS and SECSI-TEDIA denote the new proposed extension of the SECSI framework with non-symmetric SMDs based on IDIEM-NS and TEDIA, respectively. SECSI denotes the original framework, and SECSI Truncated denotes the framework when only the new truncation step is included using symmetric SMDs. Moreover, two sets of curves are presented for each algorithm, each of them representing the BM and RES heuristics to choose the final solutions. The vertical doted lines, correspond to the mean value of the estimates.

In Fig. 2 the CCDF (Complementary Cumulative Distribution Function) of the TMSFE for the tensor with uncorrelated factor matrices is presented, for SNR = 30 dB. Notice that the SECSI-TEDIA algorithm is the most accurate. The BM version for all of the algorithms is more accurate than the RES. The SECSI-IDIEM-NS BM is as accurate as the SECSI RES. With the truncation step we sacrifice accuracy but reduces the computational complexity. However, the non-symmetric SMD compensates this loss of accuracy.

In Fig. 3 the simulation results for the tensor with correlated factor matrices are visualized, for SNR = 30 dB. The SECSI-IDIEM-NS framework is the least accurate. On the other hand, the SECSI-TEDIA framework still performs accurate for ill-conditioned scenarios.

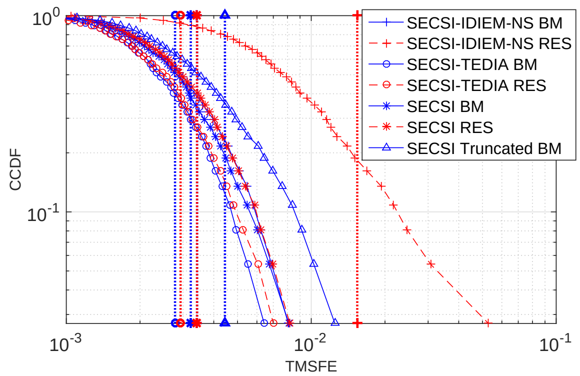


Fig. 2. Complementary cumulative distribution function of the total relative mean square factor error for real-valued tensor with dimensions $4 \times 7 \times 3$ and tensor rank $R = 3$, SNR = 30 dB.

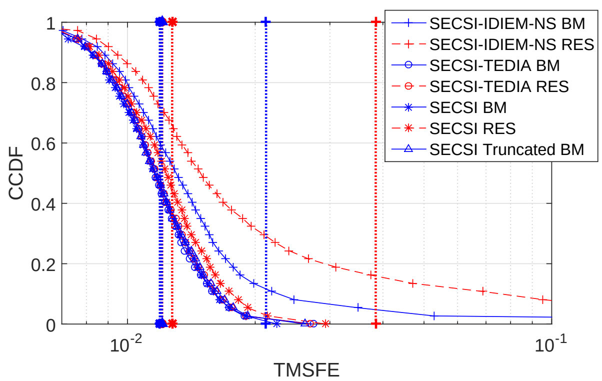


Fig. 3. Complementary cumulative distribution function of the total relative mean square factor error for real-valued tensor with dimensions $4 \times 7 \times 3$, tensor rank $R = 3$ and factor matrices with mutually correlated columns, SNR = 30 dB.

In Table 1 a summary of the average computational time in seconds is provided for the different algorithms. The SECSI-IDIEM-NS outperforms the rest of the algorithms with respect to the computational time, while the TEDIA extension requires more computational time.

5. CONCLUSIONS

In this paper we have presented an extension of the SECSI framework, by solving non-symmetric SDMs based on the TEDIA and the IDIEM-NS algorithm. The SECSI-TEDIA framework offers a high accuracy, while the SECSI-IDIEM-NS algorithm offers a very fast approximation for the CP decomposition with a reasonable accuracy. Notice that SECSI-IDIEM-NS provides a closed-form solution for CP decomposition, since the non-symmetric SMDs can be calculated in closed form [12], [10]. In contrast to the original framework we calculate 3 sets of non-symmetric SMDs instead of 6 sets of symmetric SMDs for a smaller number of matrices ($R \leq K$). The computational advantages provided by the truncations become more pronounced as the tensor size increases.

	No correlation	Correlation
SECSI-IDIEM-NS BM	0.0831 s	0.0145 s
SECSI-IDIEM-NS RES	0.0290 s	0.0042 s
SECSI-TEDIA BM	4.8274 s	0.8912 s
SECSI-TEDIA RES	5.5233 s	0.8916 s
SECSI BM	0.3480 s	0.1164 s
SECSI RES	0.0721 s	0.0442 s
SECSI Truncated BM	0.3290 s	0.1105 s

Table 1. Average computational time in [s].

6. REFERENCES

[1] T.G. Kolda and B.W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, pp. 455–500, 2009.

[2] A. Cichocki, D. Mandic, A.H. Phan, C.F. Caiafa, G. Zhou, Q. Zhao, and L. de Lathauwer, "Tensor decompositions for signal processing applications: From two-way to multiway," *IEEE Signal Processing Magazine*, vol. 32, pp. 145–163, 2015.

[3] R. Bro, N.D. Sidiropoulos, and G. B. Giannakis, "Least squares algorithm for separating trilinear mixtures.," *in Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation*, pp. 11–15, January 1999.

[4] L. de Lathauwer, "Parallel factor analysis by means of simultaneous matrix decompositions," *Proc. First IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2005)*, pp. 125–128, December 2005.

[5] F. Roemer and M. Haardt, "A closed-form solution for parallel factor (PARAFAC) analysis," *Proc. IEEE Int. Con. on Acoustics, Speech and Sig. Proc. (ICASSP 2008)*, pp. 2365–2368, April 2008.

[6] X. Luciani and L. Albera, "Semi-algebraic canonical decomposition of multi-way arrays and joint eigenvalue decomposition," *IEEE Int. Con. on Acoustics, Speech and Sig. Proc. (ICASSP 2011)*, pp. 4104–4107, May 2011.

[7] F. Roemer and M. Haardt, "A semi-algebraic framework for approximate cp decomposition via simultaneous matrix diagonalization (SECSI)," *Signal Processing*, vol. 93, pp. 2722–2738, September 2013.

[8] F. Roemer, C. Schroeter, and M. Haardt, "A semi-algebraic framework for approximate CP decompositions via joint matrix diagonalization and generalized unfoldings," *in Proc. of the 46th Asilomar Conference on Signals, Systems, and Computers, (Pacific Grove, CA)*, pp. 2023–2027, November 2012.

[9] T. Fu and X. Gao, "Simultaneous diagonalization with similarity transformation for non-defective matrices," *in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, vol. 4, pp. 1137–1140, May 2006.

[10] P. Tichavsky, A.H. Phan, and A. S Cichocki, "Two-sided diagonalization of order-three tensors," *Proceedings of the 23rd European Signal Processing Conference (EUSIPCO 2015)*, pp. 998–1002, September 2015.

[11] G. Chabriel and J. Barrere, "A direct algorithm for nonorthogonal approximate joint diagonalization," *IEEE Transactions on Signal Processing*, vol. 60, pp. 39–47, January 2012.

[12] G. Chabriel, M. Kleinsteuber, E. Moreau, H. Shen, P. Tichavsky, and A. Yeredor, "Joint matrices decompositions and blind source separation: A survey of methods, identification, and applications," *IEEE Signal Processing Magazine*, vol. 31, pp. 34–43, May 2014.

[13] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl. (SIMAX)*, vol. 21, pp. 1253–1278, 2000.