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

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 known 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, \( A, \alpha \). Vectors and matrices, are denoted as bold-face capital letters, \( A, \mathbf{A} \), respectively. Finally, tensors are represented by bold-face calligraphic letter \( \mathbf{A} \). The following superscripts, \( ^t \), \( ^{-1} \), and \( ^* \) denote transpose, Hermitian transpose, matrix inversion and Moore-Penrose pseudo matrix inversion, respectively. The outer product, Kronecker product and Khatri-Rao product are denoted by \( \mathbf{A} \times \mathbf{B} \), \( \mathbf{A} \otimes \mathbf{B} \), and \( \mathbf{A} \odot \mathbf{B} \) respectively.

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

\[
\textbf{X}_0 = \sum_{r=1}^{R} f_1^{(r)} \circ f_2^{(r)} \circ f_3^{(r)}.
\]

The CP decomposition decomposes a given tensor into the sum of the minimum number of rank one tensors. According to equation (1)
the tensor rank is equal to \( R \). The vectors \( f_{n}^{s} \) are the corresponding columns of the factor matrices \( F_n \), for \( n = 1, 2, 3 \).

The CP decomposition is essentially unique under mild conditions, which means that the factor matrices \( F_n \) 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 \( X = X_0 + \mathcal{N} \), where \( \mathcal{N} \) contains uncorrelated zero mean circularly symmetric Gaussian noise. Hence, we have to calculate a rank \( R \) approximation of \( X \).

\[
X \approx \mathcal{I}_{3,R} \times_1 F_1 \times_2 F_2 \times_3 F_3. \tag{3}
\]

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 \( X_0 \in \mathbb{C}^{I \times J \times K} \) is given by [13],

\[
X_0 = S \times_1 U_1 \times_2 U_2 \times_3 U_3 \tag{4}
\]

where \( S \in \mathbb{C}^{I \times J \times K} \) is the core tensor. The matrices \( U_1 \in \mathbb{C}^{I \times 1} \), \( U_2 \in \mathbb{C}^{J \times J} \) and \( U_3 \in \mathbb{C}^{K \times K} \) are unitary matrices which span the column space of the \( n \)-mode unfolding of \( X_0 \), for \( n = 1, 2, 3 \) respectively. Accordingly, the truncated HOSVD is defined as

\[
X_0^{[s]} = S^{[s]} \times_1 U_1^{[s]} \times_2 U_2^{[s]} \times_3 U_3^{[s]} \tag{5}
\]

where \( S^{[s]} \in \mathbb{C}^{I \times J \times K} \) is a truncated core tensor and the matrices \( U_1 \in \mathbb{C}^{I \times s} \), \( U_2 \in \mathbb{C}^{J \times s} \) and \( U_3 \in \mathbb{C}^{K \times s} \) 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 \( X \) in order to calculate an approximate low rank CP decomposition. textcolor{magnal3}Therefore we get an approximation of

\[
X_0 = \left( S^{[s]} \times_3 U_3^{[s]} \right) \times_1 U_1^{[s]} \times_2 U_2^{[s]} \tag{6}
\]

\[
= \left( \mathcal{I}_{3,R} \times_3 (U_3^{[s]} \cdot T_{3}) \right) \times_1 (U_1^{[s]} \cdot T_{1}) \times_2 (U_2^{[s]} \cdot T_{2}) \tag{7}
\]

where equations (6) and (7) represent the truncated HOSVD and the CP decomposition of the noiseless tensor, respectively. The invertible matrices \( T_1 \), \( T_2 \) and \( T_3 \) of dimensions \( R \times R \) diagonalize the truncated core tensor \( S^{[s]} \) as shown in [5].

Therefore, it follows that

\[
S^{[s]} = \left( \mathcal{I}_{3,R} \times_3 T_{3} \right) \times_1 T_{1} \times_2 T_{2}. \tag{8}
\]

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 \( T_3 = (\mathcal{I}_{3,R} \times_3 T_{3}) \), as depicted in Fig. 1(c). Notice that \( T_3 \) contains diagonal slices along the third mode. Hence, we need to diagonalize the truncated core tensor \( S^{[s]} \), or in other words we need to estimate the matrices \( T_1 \) and \( T_2 \) that diagonalize the tensor \( S^{[s]} \).

\[
S^{[s]} \times_1 T_{1}^{-1} \times_2 T_{2}^{-1} = T_3 \tag{9}
\]

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 \( e_3^T \) that is the \( k \)-th column of a \( R \times R \) identity matrix. Therefore, each of the corresponding slices is defined as \( S_k^{[s]} = S^{[s]} \times_3 e_k^T \) and \( T_{3,k} = T_3 \times_3 e_k^T \) for the left and right hand side of equation (9).

![Fig. 1. Diagonalized core tensor for mode 1, 2 and 3.](image)

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

\[
T_{1}^{-1} \cdot S_k^{[s]} \cdot T_2^{-1} = T_{3,k}, \quad k = 1, 2, \ldots, R. \tag{10}
\]

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 \( T_1 \), \( T_2 \), and \( T_3 \) is achieved, while \( T_3 \) is calculated from \( T_{3,k} \), 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

\[
\hat{F}_{1,1} = U_1^{[s]} \cdot T_{1} \tag{11}
\]

\[
\hat{F}_{2,1} = U_2^{[s]} \cdot T_{2} \tag{12}
\]

\[
\hat{F}_{3,1} = U_3^{[s]} \cdot T_{3}. \tag{13}
\]

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 \( \hat{F}_{1,1}, \hat{F}_{1,2}, \hat{F}_{1,3}, \hat{F}_{2,2}, \hat{F}_{2,3}, \hat{F}_{3,1}, \hat{F}_{3,2}, \hat{F}_{3,3} \). 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...
The reconstruction error is symmetric line an accuracy and computational time comparative. The BM version for all of the algorithms are non-symmetric. The non-symmetric SMD problem of the damped Gauss-Newton method.

\[ \rho \]

of the off diagonal elements diagonalized tensor. The algorithm is for SNR = 30 dB resolves the factors, than.

\[ \mathbf{A} \]

of the RES. The SECSI-IDIEM-NS BM is leading to the lowest reconstruction error.

\[ \mathbf{X} \]

ues of the factor matrices due to the 6 estimates of the factor matrices.

\[ \mathbf{F} \]

in equation (10), respectively. In Fig. 2 the CCDF (Complementary Cumulative Distribution Function) is shown for the two algorithms.

\[ \mathbf{X}_0 \]

in the simulation results the TMSFE (Total relative Mean Square Error) is calculated according to.

\[ \mathbf{X}_0 = \mathbf{I} \times \mathbf{X} \]

To verify the performance of the proposed algorithm, we choose the original tensor in (9) and the new tensor in (10) for the simulation.

\[ \mathbf{F}_n = \mathbf{F} \]

with variance \( \mathbf{X}_0 \). The resulting \( \mathbf{F}_n \) is a set of permuted diagonal matrices. 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 (9). In the case of a symmetric SMD, the tensor is designed according to the CP decomposition.

\[ \mathbf{X} \]

is the correlation matrix with correlation factor \( \sigma \). The second tensor has factor matrices with correlation coefficient \( \rho = 0.3 \), \( \rho = 0.1 \) and \( \rho = 0.1 \), measured in dB.

\[ \mathbf{F}_n = \mathbf{F} \]

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.

\[ \mathbf{P}_n \]

The TEDIA algorithm can be implemented in either a parallel fashion and its main computational complexity comes from the off diagonal elements but rather to achieve a block-revealing structure. The algorithm is based on a simplified Grassmannian method.

\[ \mathbf{X}_0 \]

for each algorithm, each of them representing the BM and \( \rho \) is selected on the means of the estimates.

\[ D = \mathbf{A}_r \mathbf{M}_r \mathbf{A}_t \]

with \( \mathbf{A}_r \), \( \mathbf{M}_r \) and \( \mathbf{A}_t \) are the left and right matrices that solve the symmetric SMD problem as well [12]. A symmetric SMD problem for the non-symmetric SMD is solved by a final estimate of the non-symmetric SMD, because it is not iterative and therefore very fast.

\[ \mathbf{R} \]

in equation (10), respectively. In Fig. 2 the CCDF (Complementary Cumulative Distribution Function) is shown for the two algorithms.

\[ \mathbf{X}_0 \]

in the simulation results the TMSFE (Total relative Mean Square Error) is calculated according to.

\[ \mathbf{X}_0 = \mathbf{I} \times \mathbf{X} \]

is the correlation matrix with correlation factor \( \sigma \). The second tensor has factor matrices with correlation coefficient \( \rho = 0.3 \), \( \rho = 0.1 \) and \( \rho = 0.1 \), measured in dB.

\[ \mathbf{F}_n = \mathbf{F} \]

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.

\[ \mathbf{P}_n \]

with variance \( \mathbf{X}_0 \). The resulting \( \mathbf{F}_n \) is a set of permuted diagonal matrices. 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 (9). In the case of a symmetric SMD, the tensor is designed according to the CP decomposition.
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.

Table 1. Average computational time in [s].

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>No correlation</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECSI-IDIEM-NS BM</td>
<td>0.0831 s</td>
<td>0.0145 s</td>
</tr>
<tr>
<td>SECSI-IDIEM-NS RES</td>
<td>0.0290 s</td>
<td>0.0042 s</td>
</tr>
<tr>
<td>SECSI-TEDIA BM</td>
<td>4.8274 s</td>
<td>0.8912 s</td>
</tr>
<tr>
<td>SECSI-TEDIA RES</td>
<td>5.5233 s</td>
<td>0.8916 s</td>
</tr>
<tr>
<td>SECSI BM</td>
<td>0.3480 s</td>
<td>0.1164 s</td>
</tr>
<tr>
<td>SECSI RES</td>
<td>0.0721 s</td>
<td>0.0442 s</td>
</tr>
<tr>
<td>SECSI Truncated BM</td>
<td>0.3290 s</td>
<td>0.1105 s</td>
</tr>
</tbody>
</table>
6. REFERENCES


