COUPLED RANK-(\(L_m, L_n\)) BLOCK TERM DECOMPOSITION BY COUPLED BLOCK SIMULTANEOUS GENERALIZED SCHUR DECOMPOSITION

Xiao-Feng Gong\(^{*+} \), Qiu-Hua Lin\(^*\), Otto Debals\(^{+\dagger\‡} \), Nico Vervliet\(^{\dagger\‡} \), Lieven De Lathauwer\(^{+\dagger\‡} \)

\(^*\)School of Information and Communication Engineering, Dalian University of Technology, China, \(^\dagger\)Group Science, Engineering and Technology, KU Leuven, KULAK, Belgium, \(^\‡\)Department of Electrical Engineering (ESAT), KU Leuven, Belgium, \(^\dagger\)iMinds Medical IT, Leuven, Belgium. E-mails: \{xfgong, qhlin\}@dlut.edu.cn, \{otto.debal, nico.vervliet\}@esat.kuleuven.be, lieven.delathauwer@kuleuven-kulak.be

ABSTRACT

Coupled decompositions of multiple tensors are fundamental tools for multi-set data fusion. In this paper, we introduce a coupled version of the rank-(\(L_m, L_n\)) block term decomposition (BTD), applicable to joint independent subspace analysis. We propose two algorithms for its computation based on a coupled block simultaneous generalized Schur decomposition scheme. Numerical results are given to show the performance of the proposed algorithms.

Index Terms — Tensor, block term decomposition, coupled tensor decomposition, multi-set data fusion

1. INTRODUCTION

In the past decade, the block term decomposition (BTD) has attracted increased attention in various signal processing applications. In comparison to the well-known canonical polyadic decomposition (CPD) that writes a tensor as the sum of a minimal number of rank-1 terms, BTD decomposes a tensor into a set of terms of low multilinear rank, which is more flexible and better adapted to applications where the signals are multidimensional. Uniqueness and algorithms for various types of BTD were studied in [1-4] and applications in signal processing were reported in [5-10].

Recently, coupled tensor decompositions have been studied in the context of multi-set data fusion, which assumes some links or shared factor matrices among target tensors. They are mainly extensions of CPD to multi-set cases, including the works in [11-20] and the references therein. In addition, the coupled rank-(\(L_r, L_m, L_n\)) 1) BTD was studied in [15, 16]. Structured data fusion (SDF) was presented in [21] as a flexible framework for coupled and/or structured decompositions. The above works have shown that the coupled decomposition is able to both improve the accuracy and relax the identifiability condition compared with the decomposition of a single tensor.

In this paper, we combine the concepts of coupling and BTD, and introduce a particular coupled rank-(\(L_m, L_n\)) BTD. We will show that the particular multi-set data fusion problem of joint independent subspace analysis (J-ISA) can be turned into a coupled rank-(\(L_m, L_n\)) BTD. We present two algorithms for the coupled block version of so-called simultaneous generalized Schur decomposition scheme (SGSD, [22-24]). SGSD involves only unitary factors. It has found use in the analytical constant modulus algorithm [22], in the computation of CPD [23], in numerically stable representations of ill-posed CPD problems [24], etc. We here limit ourselves to the overdetermined case. The more challenging under-determined coupled rank-(\(L_m, L_n\)) BTD will be addressed in the future.

Notation and definitions: vectors, matrices and tensors are denoted by lowercase boldface, uppercase boldface and uppercase calligraphic letters, respectively. The \(r\)th column vector and the \((i, j)\)th entry of \(A\) are denoted by \(a_r\) and \(a_{ij}\), respectively. Symbols ‘\(\otimes\)’, ‘\(\odot\)’, ‘\(\times\)’ and ‘\(\circ\)’ denote the Kronecker product, block-wise Khatri-Rao product, mode-\(n\) product, and outer product, respectively, defined as:

\[
A \otimes B = \begin{bmatrix} a_{1,1} B & a_{1,2} B & \cdots \\ a_{2,1} B & a_{2,2} B & \cdots \\ \vdots & \vdots & \ddots \\ a_{m,1} B & a_{m,2} B & \cdots \end{bmatrix},
\]

\[
A \odot B = [A \otimes B_1, \ldots, A \otimes B_k],
\]

\[
(T \times_a G)_{a_1,\ldots,a_r,b_1,\ldots,b_s} \triangleq \sum_{t_{1-\cdots-r}} g_{t_{1-\cdots-r}} (a \odot b \odot c)_{t_{1-\cdots-r},b_{1-\cdots-s}} \triangleq a_{b_1} c_{b_2}.
\]
In the above definitions, we assume \( A \triangleq [A_1, \ldots, A_r] \) and \( B \triangleq [B_1, \ldots, B_r] \) for the block-wise Khatri-Rao product. For the mode-\( n \) product we assume that the \( n \)th dimension of \( \mathbf{T} \) is equal to the number of columns of \( G \). We denote the identity matrix and the all-zero matrix as \( I_{m \times m} \in \mathbb{C}^{m \times m} \) and \( \theta_{m \times n} \in \mathbb{C}^{m \times n} \), respectively. Transpose, conjugate, conjugated transpose, Moore-Penrose pseudo inverse, and Frobenius norm are denoted as \((\cdot)^T \), \((\cdot)^* \), \((\cdot)^H \), \((\cdot)^{\dagger} \), \((\cdot)^{1\vec{\cdot}} \), respectively. Mathematical expectation is denoted as \( \mathbb{E} \{ \cdot \} \). MATLAB notations will be used to denote submatrices of a tensor. For instance, we use \( T_{(i,k)} \) to denote the frontal slice of a tensor by fixing the third index to \( k \).

For a given matrix \( \mathbf{T} \in \mathbb{C}^{r \times d} \), \( \text{vec}(\mathbf{T}) \triangleq [t_{11}^T, \ldots, t_{r1}^T]^T \in \mathbb{C}^{rd} \) denotes column-wise vectorization of \( \mathbf{T} \) and \( \text{unvec}(\cdot) \) performs the inverse. For a third-order tensor \( \mathbf{T} \in \mathbb{C}^{r \times j \times k} \), notations \( T_{(i,j,k)} \in \mathbb{C}^{r \times j \times k} \), \( T_{(i,j,k)} \in \mathbb{C}^{r \times j \times k} \) denote three types of matricization, defined by:

\[
(T_{(i,j,k)})_{(l-1)j+l,k} = (T_{(i,j,k)})_{(l-1)j+l,k} = T_{(i,j,k)}
\]

The mode-\( n \) vectors of \( \mathbf{T} \) are obtained by fixing all but the \( n \)th index of \( \mathbf{T} \). The mode-\( n \) rank of \( \mathbf{T} \) is defined as the dimension of the subspace spanned by the mode-\( n \) vectors. It is easy to understand that the mode-1, mode-2, and mode-3 rank of \( \mathbf{T} \) is equal to the rank of \( T_{(i,j,k)} \), \( T_{(i,j,k)} \), and \( T_{(i,j,k)} \), respectively. Third-order tensors with mode-1, mode-2, mode-3 rank equal to \( L_1 \), \( L_2 \), \( L_3 \), respectively, are said to have multilinear rank \((L_1, L_2, L_3)\). Third-order tensors with mode-1 and mode-2 rank equal to \( L_1 \) and \( L_2 \), without rank constraint in the third mode, are said to have rank-four rank \((L_1, L_2, \cdot)\). Rank-(1,1,1) terms simply correspond to rank-1 terms.

### 2. Problem Formulation

In this paper, we consider the following coupled rank-\((L_m, L_n, \cdot)\) BTD of a set of tensors \( \mathbf{T}^{(m,n)} \in \mathbb{C}^{L_m \times L_n \times \mathbb{K}} \):

\[
\mathbf{T}^{(m,n)} = \sum_{s=1}^{L_m} C^{(m,n)}(s) A^{(m)}(s) \times A^{(n)}(s), \quad m, n = 1, \ldots, M,
\]

where \( C^{(m,n)}(s) \in \mathbb{C}^{L_m \times L_n \times \mathbb{K}} \) has mode-1 rank equal to \( L_m \) and mode-2 rank equal to \( L_n \), and \( A^{(m)}(s) \in \mathbb{C}^{L_m \times \mathbb{K}} \) has full column rank. Eq. (1) suggests that each tensor \( \mathbf{T}^{(m,n)} \) admits by itself a rank-(\( L_m, L_n, \cdot)\) BTD [1] (see Fig. 1 for an illustration). In addition, each tensor \( \mathbf{T}^{(m,n)} \) is coupled with \( \mathbf{T}^{(m,n)} \) in the first mode by factor matrix \( A^{(m)}(s) \) and at the same time coupled with \( A^{(m)}(s) \) in the second mode by \( A^{(n)}(s) \), \( m \neq m', n \neq n' \). This double coupling structure is illustrated in Fig. 2. The matrix representation of (1) is given by:

\[
\mathbf{T}^{(m,n)}_{(i,j,k)} = (A^{(m)}(s) \otimes A^{(n)}(s)) C^{(m,n)}(s),
\]

where \( C^{(m,n)}(s) \triangleq [(C^{(m,n)}(s))_{1,j,k}, \ldots, (C^{(m,n)}(s))_{L_m,j,k}]^T \in \mathbb{C}^{L_m \times L_n \times \mathbb{K}} \). In addition, the frontal slices of \( \mathbf{T}^{(m,n)} \) take the following form:

\[
\mathbf{T}^{(m,n)}_{(i,k)} = A^{(m)}(s) \Sigma^{(m,n)}_{(i,k)} A^{(n)}(s)^H,
\]

where \( \Sigma^{(m,n)}_{(i,k)} \in \mathbb{C}^{L_m \times L_n} \) is a block-diagonal matrix with blocks of size \( L_m \times L_n \) containing the \( k \)th frontal slice of \( C^{(m,n)}(s) \) as the \( r \)th block on its main diagonal.

Note that in (1) we can arbitrarily permute the terms if it is done consistently for all tensors involved. We can also post-multiply \( A^{(m)}(s) \) and \( A^{(n)}(s) \) by non-singular matrices \( F^{(m)}(s) \in \mathbb{C}^{L_m \times L_m} \) and \( F^{(n)}(s) \in \mathbb{C}^{L_n \times L_n} \) provided that \( A^{(m)}(s) \) is replaced by \( A^{(m)}(s) \times A^{(n)}(s) \) for all values of \( m \) and \( n \). The goal of coupled rank-(\( L_m, L_n, \cdot)\) BTD is then to solve (1) up to these trivial indeterminacies.

![Fig. 1](image1.png)

Fig. 1. The rank-(\( L_m, L_n, \cdot)\) BTD writes each target tensor \( \mathbf{T}^{(m,n)} \) as the sum of multiple block terms of low multilinear rank.

![Fig. 2](image2.png)

Fig. 2. The double coupling structure. The target tensors are placed at different nodes of a grid according to their indices. Each tensor is coupled with tensors along two modes by two factor matrices.

Next we explain how the coupled rank-(\( L_m, L_n, \cdot\)) BTD (1) is related to the J-ISA problem [25]. The multi-set data model for J-ISA is formulated as:

\[
x^{(m)}(s) = A^{(m)}(s) s^{(m)}(t),
\]

where \( x^{(m)}(s) \in \mathbb{C}^{L_m} \) denotes the observed mixture, and \( s^{(m)}(t) \in \mathbb{C}^{L_m} \) is the latent source vector at time instant \( t \), and \( \mathbf{A}^{(m)}(s) \in \mathbb{C}^{L_m \times \mathbb{K}} \) denotes the mixing matrix of the \( m \)th dataset. We partition the source vector \( s^{(m)}(t) \) into \( R \) sub-vectors \( [s^{(m)}(t), \ldots, s^{(m)}(t)]^T \neq s^{(m)}(t) \), with each sub-vector \( s^{(m)}(t) \) representing a group of \( L_m \) sources. In addition we partition the mixing matrix \( [A^{(1)n}, \ldots, A^{(r)n}] \neq A^{(m)n} \), where \( A^{(m)^n} \) is the sub-matrix of \( A^{(m)n} \) associated with \( s^{(m)}(t) \).

We make the following assumptions: 1) each sub-vector \( s^{(m)}(t) \) contains \( L_m \) dependent sources; 2) \( s^{(m)}(t) \) and \( s^{(n)}(r) \) are independent for \( r \neq r \) and dependent for \( r = r \) regardless of the values of \( m \) and \( m \); 3) the sources are temporally non-stationary with zero mean and unit variance.

We calculate the cross-covariance tensor as follows:

\[
\mathbf{C}^{(m,n)}_{(i,k)} = \text{E} [x^{(m)}(s) [x^{(n)}(t)]^{\dagger}],
\]

where \( \mathbf{C}^{(m,n)}_{(i,k)} \in \mathbb{C}^{L_m \times L_n \times \mathbb{K}} \), \( 1 \leq m, n \leq M \), \( K \) denotes the number of time-frames for which such a cross-covariance is computed. The matrix \( \Sigma_{(i,k)} \neq \text{E} [s^{(m)}(s) [s^{(n)}(t)]^{\dagger}] \) is block-diagonal with blocks of size \( L_m \times L_n \) under the above assumptions. Comparing (5) and (3), we can see that the J-ISA data model has been converted to a coupled rank-(\( L_m, L_n, \cdot)\) BTD formulation by using second-order statistics.
3. ALGORITHMS

We limit ourselves to the overdetermined case where $A^{(n)}$ and $C^{(m,n)}$ in (2) are assumed to have full column rank for all the values of $m,n$, and propose two algorithms for the computation of coupled rank-$(L_m, L_n, \cdot )$ BTD, which extend the SGS scheme to the coupled block case.

3.1. Coupled Block SGSD

The goal of coupled rank-$(L_m, L_n, \cdot )$ BTD is to find $A^{(n)}, A^{(n)}$ and $\Sigma^{(n)}$ that minimize the following function:

$$\eta(\Omega) = \sum_{m=1}^{M} \sum_{k=1}^{K} \left\| T_{(m),k}^{(n)} - A^{(n)} \Sigma^{(n),k} A^{(n)\top} \right\|^2,$$

where $\Omega \triangleq \{A^{(n)}, \Sigma^{(n),k}, m, n = 1, \ldots , M, k = 1, \ldots , K\}$ denotes the set containing all the arguments. We let $A^{(n)} \in \mathbb{C}^{m\times n}$ be a block QR decomposition of $A^{(n)}$ and $A^{(n)} \in \mathbb{C}^{m\times n}$ be a block ZL decomposition of $A^{(n)}$, where $Q^{(n)} \in \mathbb{C}^{m \times m}, Z^{(n)} \in \mathbb{C}^{m \times m}$ are unitary matrices, and $R^{(n)} \in \mathbb{C}^{m \times n}$ is a block upper-triangular matrix. Here we introduce a block variant of the algorithm in [22], which alternates between updates of $Q^{(n)}$ and $Z^{(n)}$ to optimize the cost function in (9). In each iteration, we update $Q^{(n)}$ as $Q^{(n)} \leftarrow T_{(m),k}^{(n)} C^{(m,n)}$ with $Z^{(n)}$ fixed (vice-versa for the update of $Z^{(n)}$). Here $Q^{(n)} \in \mathbb{C}^{m \times n}$ is a unitary matrix constructed as the product of $R$ unitary matrices:

$$Q^{(n)} = I_{(1,2)} R^{(n)}_1 \cdots R^{(n)}_{L_m}.$$

where $I_{(1,2)}$ denotes the Frobenius norm of the strictly block lower-triangular part. Note that (9) involves only unitary factor matrices, which are optimally conditioned. This has been achieved by relaxing the block-diagonal structure of $\Sigma^{(n),k}$ in (6) to the block-upper-triangular structure of $\Gamma^{(n),k}$ in (8).

When the coupled block SGSD is solved (algorithms will be presented later in subsections 3.2 and 3.3), we obtain the block upper-triangular matrices $\Gamma^{(n),k}$. We recall that $I^{(n),k} \triangleq R^{(n)} \Sigma^{(n),k} \Gamma^{(n),k}$ are given in (7), and we have:

$$I^{(n),k} = \Gamma^{(n),k}_1 \cdots \Gamma^{(n),k}_L,$$

where $\Gamma^{(n),k} \in \mathbb{C}^{m \times n}$, $\mathbb{C}^{m \times n}$, $\mathbb{C}^{m \times n}$, and $\mathbb{C}^{m \times n}$ are the rth block on the main diagonal of $\Gamma^{(n),k}$, $\Gamma^{(n),k}$, and $\Gamma^{(n),k}$, respectively. Note that $\Gamma^{(n),k}$ and $\Gamma^{(n),k}$ are invertible, and (10) implies that $\Gamma^{(n),k}$ is an estimate of $\Gamma^{(n),k}$. According to (10) we have the following:

$$C^{(m,n)} = \begin{bmatrix} \hat{R}_l^{(n)} \otimes \hat{E}_l^{(n)} & \ldots & \hat{R}_l^{(n)} \otimes \hat{E}_l^{(n)} \end{bmatrix} C^{(m,n)} \left( \begin{bmatrix} \hat{R}_l^{(n)} \otimes \hat{E}_l^{(n)} \end{bmatrix}\right)^{-1} \hat{R}_l^{(n)} \otimes \hat{E}_l^{(n)}$$

By (2) and (11), and recall that $C^{(m,n)}$ is assumed to have full column rank, we have the following:

$$V^{(m,n)} = I_{(1,2)} \Gamma^{(m,n)} C^{(m,n)} \left( I_{(1,2)} \Gamma^{(m,n)} C^{(m,n)} \right)^{-1}.$$

where $A^{(m)} \triangleq A^{(m)}R^{(m)}$ and $B^{(m)} \triangleq A^{(m)}F^{(m)}$. This can be taken as estimates of $A^{(m)}$ and $A^{(n)}$ up to trivial indeterminacies. Now the problem is how to calculate $A^{(m)}$ and $B^{(m)}$ from $V^{(m,n)}$. We partition $V^{(m,n)}$ as $[V_{(r)}^{(m,n)}, \ldots , V_{(r)}^{(m,n)}]$ where $V_{(r)}^{(m,n)} = A^{(m)}R^{(m)}B^{(m)} \in \mathbb{C}^{m \times m}$, and reshape $V_{(r)}^{(m,n)}$ into a matrix $V_{(r)}^{(m,n)} \in \mathbb{C}^{m \times m}$ as follows:

$$V_{(r)}^{(m,n)} = \left[ \begin{array}{ccc} \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \end{array} \right] \begin{bmatrix} V_{(r)}^{(m,n)} & V_{(r)}^{(m,n)} & \ldots & V_{(r)}^{(m,n)} \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \end{array}$$

By definition we have $V^{(m,n)} = \text{vec}(A^{(m)})^{' \text{vec}(B^{(m)})}$. Moreover, we stack the matrices $V^{(m,n)}$ for all the values of $m$ and $n$ into a larger rank-1 matrix $M_r$ as follows:

$$M_r = \begin{bmatrix} V^{(1,1)} & V^{(1,2)} & \cdots & V^{(1,M)} \\ V^{(2,1)} & V^{(2,2)} & \cdots & V^{(2,M)} \\ \vdots & \vdots & \ddots & \vdots \\ V^{(M,1)} & V^{(M,2)} & \cdots & V^{(M,M)} \\ \end{bmatrix} = \begin{bmatrix} \text{vec}(A^{(1)} & \text{vec}(A^{(2)} & \cdots & \text{vec}(A^{(M)} & \text{vec}(B^{(1)} & \cdots & \text{vec}(B^{(M)} & \text{vec}(B^{(M)}) \end{bmatrix}.$$
to minimize the strictly below block-diagonal norms of the second $L_m$ columns of $\mathcal{T}(n,m)$ without affecting the first $L_n$ columns. This can be done by looking at a reduced problem where the same reasoning as in the calculation of $H(n)^{\top}$ is followed for $\mathcal{T}(n,m)$ with the first $L_n$ columns removed. After $H(n)^{\top}$ is computed, the matrices $H_1(n)^{\top}, \ldots, H_{L_m}(n)^{\top}$ follow sequentially. The update of $Z(n)$ is similar to $Q(n)$.

3.3. Jacobi Iteration

Here we alternate between updates of $Q(n)$ and $Z(n)$, each computed as the product of a sequence of elementary Givens matrices. For the update of $Q(n)$, in each step the matrices $Q(n)$ and $\Gamma(n)$ are updated by a Givens matrix $G_{uv}(n)$ as $Q(n) \leftarrow G_{uv}(n) Q(n)$. $\Gamma(n) \leftarrow G_{uv}(n) \Gamma(n) G_{uv}(n)^{\top}$, $1 \leq u < v \leq L$, where $G_{uv}(n)$ is equal to the identity matrix except $(G_{uv}(n))_{u,v} = 1, (G_{uv}(n))_{u,v} = s_{uy}(n), (G_{uv}(n))_{u,v} = s_{vy}(n)$, $(G_{uv}(n))_{u,v} = c_{uy}(n)$, with $c_{uy}(n) = \cos(\theta_{uy}(n))$ and $s_{uy}(n) = \sin(\theta_{uy}(n))$.

Denoting $\bar{G}_{uv}(n) = \sum_u (G(n)_{uv} G(n)^{\top}_{uv})$, the cost function (9) can be written as $\bar{E} = \sum_u \xi_u$. Then for fixed $m$, an iteration step $(u,v)$ consists of finding $c_{uy}(n)$ and $s_{uy}(n)$ that minimize $\bar{E}$. Note that $G_{uv}(n)$ only affects the $uv$ and $v$th rows of $\Gamma(n)$, and that the minimization of $\bar{E}$ amounts to minimizing the strictly lower-block-triangular parts of the $uv$th and $v$th columns of $\Gamma(n)$ for all values of $n$ and $k$.

With some technical derivations we obtain the following:

\[ \bar{E} = \sum_{u,v} (M_{uv}(n) M_{uv}(n)^{\top} w_{uv}), \]

where $M_{uv}(n) = \sum_M M_{uv}(n)^{\top} (M_{uv}(n)^{\top})^{\top}$, and $w_{uv}(n) = \{c_{uy}(n), \cos(s_{uy}(n))\}$. Each $M_{uv}(n)$ and $w_{uv}(n)$ are defined as:

\[ M_{uv}(n)^{\top} = \frac{1}{2} (\Gamma(n)^{\top})_{u,v}^{\top} (\Gamma(n)^{\top})_{v,u}^{\top}, \quad M_{uv}(n)^{\top} = \frac{1}{2} (\Gamma(n)^{\top})_{u,v} (\Gamma(n)^{\top})_{v,u}. \]

Then $w_{uv}(n)$ is taken equal to the least significant eigenvector of $M_{uv}(n)$. In each iteration for the update of $Q(n)$, we find the optimal rotation angles $c_{uy}(n), s_{uy}(n)$ from an EVD. The update of $Z(n)$ can be addressed similarly, and is not further discussed.

4. NUMERICAL RESULTS

The target tensors are constructed as:

\[ \mathbf{P}(n,m) = \mathbf{T}(n,m) \| R + \sigma_n \mathbf{N}(n,m)\| \| \mathbf{N}(n,m)\|, \]

where $\mathbf{T}(n,m)$ is generated by (1) with the entries of $A(n,m) \in \mathbb{C}^{L \times L}$, $C(n,m) \in \mathbb{C}^{L \times L}$ and $\mathbf{N}(n,m) \in \mathbb{C}^{L \times L}$ randomly drawn from complex normal distributions, $m, n = 1, \ldots, M$. We set $M = 2$, $K = 200$, $R = 2$, $L = 6$. The signal-to-noise ratio (SNR) is defined with the signal level $\sigma$ and noise level $\sigma_n$, as $\text{SNR} = 20 \log_{10}(\sigma / \sigma_n)$.

The proposed coupled rank-$(L_m, L_n)$ BTD algorithms based on the extended QZ and Jacobi iteration are denoted as CLLD-EQZ and CLLD-Jacobi, respectively. For comparison, we implemented coupled rank-$(L_m, L_n)$ BTD with structured data fusion (CLLD-SDF) [21]. We also include in the comparison the computation of rank-$(L_m, L_n)$ BTD by alternating least squares (LLD-ALS, [2]) for each tensor separately. For CLLD-EQZ and CLLD-Jacobi we initialize with identity matrices. For CLLD-SDF, we initialize with the results from CLLD-EQZ. For LLD-ALS, we initialize with randomly generated factor matrices. With the obtained estimates we can reconstruct a set of tensors by (1). The average relative fitting error $\bar{E}$ used to evaluate the performance is defined as:

\[ \bar{E} = \frac{\sum_{m,n} \left\| \mathbf{P}(n,m) - \mathbf{P}(n,m) \right\|}{\left\| \mathbf{P}(n,m) \right\|}, \]

where $\mathbf{P}(n,m)$ is the reconstructed tensor. For CLLD-EQZ, CLLD-Jacobi and LLD-ALS, we terminate the iteration when $|\bar{E}_{m,n} - \bar{E}_{m,n+1}| / |\bar{E}_{m,n}| \leq 10^{-3}$, where $\bar{E}_{m,n}$ and $\bar{E}_{m,n+1}$ denote the relative fitting errors in the current and previous iteration, respectively. For CLLD-SDF, we set the tolerance parameters ToFun and ToIX in the ‘SDF_NLS’ function of Tensorlab [26] to 0.001 and 0.03, respectively. For each SNR value, we perform 200 independent runs of all the algorithms. The results of mean $\bar{E}$ and CPU time versus SNR are drawn in Fig. 3. We can see that the proposed algorithms provide more accurate estimates and higher computational efficiency than LLD-ALS for moderate to high SNR. CLLD-SDF gives the most accurate results, at a higher computational cost. This illustrates the interest of taking the coupling into account. It also shows that CLLD-EQZ and CLLD-Jacobi (i) provide good estimates for sufficiently high SNR and (ii) may be used as a low-cost initialization of the more expensive CLLD-SDF.

5. CONCLUSION

We have proposed two algorithms for the computation of a new coupled rank-$(L_m, L_n)$ BTD problem. The proposed algorithms are based on a coupled block version of the SGSD scheme, and can be used for the particular multi-set data fusion problem of J-ISA. Numerical results have shown that the proposed algorithms have fast computation and good accuracy, which makes them useful tools as such. When high accuracy is desired, they may be used to initialize algorithms that minimize the block-diagonal criterion (6) instead of the block-triangular criterion (8).
6. REFERENCES


