Semi-algebraic canonical decomposition of multi-way arrays and joint eigenvalue decomposition

Xavier Luciani, Laurent Albera

SEMI-ALGEBRAIC CANONICAL DECOMPOSITION OF MULTI-WAY ARRAYS AND JOINT EIGENVALUE DECOMPOSITION

Xavier Luciani[1,2] *and Laurent Albera*[1,2]

[1]Inserm, UMR 642, France and [2]Université de Rennes 1, LTSI, Rennes, F-35000, France

ABSTRACT

A semi-algebraic algorithm based on Joint EigenValue Decomposition (JEVD) is proposed to compute the CP decomposition of multi-way arrays. The iterative part of the method is thus limited to the JEVD computation. In addition it involves less restrictive hypothesis than other recent semi-algebraic approaches. We also propose an original JEVD technique based on the $\boldsymbol{LU}$ factorization. Numerical examples highlight the main advantages of the proposed methods to solve both the JEVD and CP problems.

Index Terms— Tensor decomposition, CP, PARAFAC, joint eigenvalue decomposition, non defective matrices.

1. INTRODUCTION

Tensor or multi-way array decompositions are used in numerous application areas such as Psycometrics [1], Biomedical Engineering [2] or Chemometrics [3]. Thanks to its uniqueness property [4, 5], the CP decomposition (for CANDECOMP/PARAFAC) [1, 6] is probably the most popular nowadays.

Many iterative algorithms have been proposed to compute the CP decomposition. One of the most famous resorts to an iterative Alternating Least Squares (ALS) procedure [6]. However these approaches suffer from classical convergence problems (local minima, slow convergence or high computational cost per iteration). Recently, an Enhanced Line Search (ELS) [7, 8] procedure has allowed to confine this disadvantage but it still exist some simple cases for which any iterative algorithm fails [9]. An other approach is to rephrase the CP decomposition as a joint diagonalization problem [5, 10, 11]. Notably, the "Closed Form Solution" (CFS) presented in [10] and [11] resorts to the Joint EigenValue Decomposition (JEVD) of a set of non-defective matrices. These methods can be called semi-algebraic since they algebraically rewrite the CP problem into a more classical matrix problem, which is then iteratively solved by means of a Jacobi-like procedure. However such methods generally involve some strongest hypothesis to work. For instance, CFS requires that the rank of the considered tensor does not exceed two of its dimensions.

We propose here a new formulation of the CP decomposition as a JEVD problem, leading to a novel semi-algebraic solution, named SALT (Semi-ALgebraic Tensor decomposition) which does not impose this limitation. At this occasion we first propose an original Jacobi-like JEVD algorithm, called JET (Joint Eigenvalue decomposition algorithm based on Triangular matrices).

2. JOINT EIGENVALUE DECOMPOSITION

In the following, the subset of $\mathbb{N}$ included in $[x;y]$ is denoted by $[x;y]_{\mathbb{N}}$.

The JEVD problem consists in finding an eigenvector matrix $\boldsymbol{A}$ from a set of non-defective matrices $\boldsymbol{M}^{(k)}$ verifying:

$$\forall k \in [1;K]_{\mathbb{N}},\ \boldsymbol{M}^{(k)} = \boldsymbol{A}\boldsymbol{D}^{(k)}\boldsymbol{A}^{-1}, \tag{1}$$

where the K diagonal matrices $\boldsymbol{D}^{(k)}$ are unknown. It can be shown that the JEVD is unique up to a permutation and a scaling of the columns of $\boldsymbol{A}$ within conditions on matrices $\boldsymbol{D}^{(k)}$ [12].

Although it is encountered in other contexts such as 2-D DOA estimation [13], few authors have addressed the JEVD problem. Two main kinds of Jacobi-like algorithms have been developed based on either the $\boldsymbol{QR}$ factorization [14] or the polar decomposition [15, 16, 17] of $\boldsymbol{A}$.

We propose here a third Jacobi-like approach, based on the $\boldsymbol{LU}$ factorization of the eigenvector matrix and we show that the iterative optimization is then reduced to the search for only one triangular matrix.

Definition 1 *A unit matrix is a matrix whose all the diagonal elements are equal to* 1.

Definition 2 *An elementary triangular matrix $\boldsymbol{L}^{(i,j)}(a)$ is a unit triangular matrix whose non-diagonal components are zero except the (i,j)-th one, which is equal to a.*

A generalization of the $\boldsymbol{LU}$ factorization easily shows that any non-singular square matrix $\boldsymbol{A}$ can be factorized as $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{\Pi}$ where $\boldsymbol{L}$ is a unit lower triangular matrix, $\boldsymbol{V}$ is a unit upper triangular matrix, $\boldsymbol{\Lambda}$ is a diagonal matrix and $\boldsymbol{\Pi}$ is a permutation matrix. Thereby, due to the indeterminacies of the JEVD problem, the matrix $\boldsymbol{A}$ solving (1) can be chosen of the form $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{V}$ without loss of generality. The JEVD problem is then reduced to find a unit lower triangular matrix $\boldsymbol{L}$ and a unit upper triangular matrix $\boldsymbol{V}$ verifying:

$$\forall k \in [1;K]_{\mathbb{N}},\ \boldsymbol{L}^{-1}\boldsymbol{M}^{(k)}\boldsymbol{L} = \boldsymbol{V}\boldsymbol{D}^{(k)}\boldsymbol{V}^{-1}, \tag{2}$$

where the K matrices $\boldsymbol{R}^{(k)} = \boldsymbol{V}\boldsymbol{D}^{(k)}\boldsymbol{V}^{-1}$ are upper triangular. As a consequence $\boldsymbol{L}$ performs the joint triangularization of matrices $\boldsymbol{M}^{(k)}$. Let us propose a Jacobi-like procedure to identify it, based on the following lemma:

Lemma 1 *Any unit lower triangular matrix $\boldsymbol{L}$ of size $(N \times N)$ can be factorized as a product of $M = N(N-1)/2$ elementary lower triangular matrices.*

The proof is skipped due to the lack of space. Now by taking into account that elementary lower triangular matrices commute, (2) and lemma 1 yield:

$$\exists \{x_m\}_{m\in[1;M]_{\mathbb{N}}} \text{ such that, } \forall k \in [1;K]_{\mathbb{N}},$$
$$\boldsymbol{R}^{(k)} = \prod_{m=1}^{M} \left(\boldsymbol{L}^{(m)}(x_m)\right)^{-1} \boldsymbol{M}^{(k)} \prod_{m=1}^{M} \boldsymbol{L}^{(m)}(x_m), \quad (3)$$

where each index m corresponds to a distinct couple (i,j) $(1 \leq j < i \leq N)$. As a consequence, ideally, we have to found only M parameters x_m to triangularize the K matrices $\boldsymbol{M}^{(k)}$. Instead of simultaneously identifying these M parameters, a Jacobi-like procedure will repeat several sequence of M sequential optimizations until convergence, each optimization with respect to only one parameter. A sequence of M optimizations is generally called a sweep. Thereby, we then look for a matrix $\boldsymbol{L}$ of the form $\boldsymbol{L} = \prod_{n_s=1}^{N_s} \prod_{m=1}^{M} \boldsymbol{L}^{(m,n_s)}(x_m^{n_s})$, where N_s is the number of sweeps.

$\forall (k,m,n_s) \in [1;K]_{\mathbb{N}} \times [2;M]_{\mathbb{N}} \times [1;N_s]_{\mathbb{N}}$, we define:

$$\boldsymbol{M}^{(k,0,1)} = \boldsymbol{M}^{(k)} \quad (4)$$
$$\boldsymbol{M}^{(k,1,n_s)} = \left(\boldsymbol{L}^{(1)}(y_1^{n_s})\right)^{-1} \boldsymbol{M}^{(k,M,n_s-1)} \boldsymbol{L}^{(1)}(y_1^{n_s}) \quad (5)$$
$$\boldsymbol{M}^{(k,m,n_s)} = \left(\boldsymbol{L}^{(m)}(y_m^{n_s})\right)^{-1} \boldsymbol{M}^{(k,m-1,n_s)} \boldsymbol{L}^{(m)}(y_m^{n_s}) \quad (6)$$

A natural way to compute the optimal (m,n_s)-th parameter $x_m^{n_s}$ is given by:

$$\forall (m,n_s) \in [1;M]_{\mathbb{N}} \times [1;N_s]_{\mathbb{N}},\ x_m^{n_s} = \text{Argmin}_{y_m^{n_s}} \left(\zeta^{m,n_s}(y_m^{n_s})\right),$$

with:

$$\zeta^{m,n_s}(y_m^{n_s}) = \sum_{k=1}^{K} \sum_{q=1}^{N-1} \sum_{p=q+1}^{N} \left(M_{p,q}^{(k,m,n_s)}\right)^2.$$

Components of $\boldsymbol{M}^{(k,m,n_s)}$ are deduced from those of $\boldsymbol{M}^{(k,m-1,n_s)}$ within only a few computations. This is an advantage of using elementary triangular matrices. Indeed, (4)-(6) yield:

$\forall (k,m,n_s) \in [1;K]_{\mathbb{N}} \times [1;M]_{\mathbb{N}} \times [1;N_s]_{\mathbb{N}}$,

$$\begin{aligned}
M_{p,q}^{(k,m,n_s)} &= M_{p,q}^{(k,m-1,n_s)} \text{ if } p \neq i \text{ and } q \neq j,\\
M_{p,q}^{(k,m,n_s)} &= -y_m^{n_s} M_{j,q}^{(k,m-1,n_s)} + M_{p,q}^{(k,m-1,n_s)}\\
&\quad \text{if } p = i \text{ and } q \neq j,\\
M_{p,q}^{(k,m,n_s)} &= y_m^{n_s} M_{p,i}^{(k,m-1,n_s)} + M_{p,q}^{(k,m-1,n_s)}\\
&\quad \text{if } p \neq i \text{ and } q = j,\\
M_{i,j}^{(k,m,n_s)} &= -\left(y_m^{n_s}\right)^2 M_{j,i}^{(k,m-1,n_s)} + M_{i,j}^{(k,m-1,n_s)}\\
&\quad + y_m^{n_s}\left(M_{i,i}^{(k,m-1,n_s)} - M_{j,j}^{(k,m-1,n_s)}\right).
\end{aligned}$$

Consequently ζ^{m,n_s} can be expressed as a fourth degree polynomial in variable $y_m^{n_s}$ and thus easily minimized by computing the roots of its derivative. Finally, $\boldsymbol{L}$ is estimated by sequentially minimizing the $N_s M$ criteria ζ^{m,n_s} and we deduce the estimate of each upper triangular matrix $\boldsymbol{R}^{(k)}$ from (2).

We now show how the unit upper triangular matrix $\boldsymbol{V}$ can be algebraically computed from the set of matrices $\boldsymbol{R}^{(k)} = \boldsymbol{V}\boldsymbol{D}^{(k)}\boldsymbol{V}^{-1}$. Such a computation is achieved component by component. The relationship between $\boldsymbol{R}^{(k)}$, $\boldsymbol{V}$ and $\boldsymbol{D}^{(k)}$ yields:

$$\forall (i,j) \in [1;N]_{\mathbb{N}}^2,\ \left(\boldsymbol{R}^{(k)}\boldsymbol{V}\right)_{i,j} = \left(\boldsymbol{V}\boldsymbol{D}^{(k)}\right)_{i,j}.$$

So we have $\forall k \in [1;K]_{\mathbb{N}},\ \forall (i,j) \in [1;N]_{\mathbb{N}}^2$ with $i < j$:

$$\left(D_{j,j}^{(k)} - R_{i,i}^{(k)}\right) V_{i,j} = \sum_{p=i+1}^{j} R_{i,p}^{(k)} V_{p,j}. \quad (7)$$

Since $\boldsymbol{D}^{(k)}$ is actually the diagonal matrix of eigenvalues of $\boldsymbol{R}^{(k)}$ and since $\boldsymbol{R}^{(k)}$ is a triangular matrix, the diagonal components of $\boldsymbol{D}^{(k)}$ are known and equal to the diagonal components of $\boldsymbol{R}^{(k)}$. Then the left-hand side of (7) becomes $\left(R_{j,j}^{(k)} - R_{i,i}^{(k)}\right) V_{i,j}$. Now, let:

$$a_k^{(i,j)} = R_{j,j}^{(k)} - R_{i,i}^{(k)} \quad \text{and} \quad b_k^{(i,j)} = \sum_{p=i+1}^{j} R_{i,p}^{(k)} V_{p,j}$$

be the k-th components of vectors $\boldsymbol{a}^{(i,j)}$ and $\boldsymbol{b}^{(i,j)}$, respectively. Then (7) can be rewritten as follows:

$$\forall (i,j) \in [1;N]_{\mathbb{N}}^2,\ i < j,\ V_{i,j}\,\boldsymbol{a}^{(i,j)} = \boldsymbol{b}^{(i,j)}.$$

Thereby, the identification of $V_{i,j}$ in the least square sense is given by:

$$\forall (i,j) \in [1;N]_{\mathbb{N}}^2,\ i < j,\ V_{i,j} = \frac{\boldsymbol{a}^{(i,j)\intercal}\,\boldsymbol{b}^{(i,j)}}{\|\boldsymbol{a}^{(i,j)}\|^2}. \quad (8)$$

The use of (8) requires to scan the values of i from $j-1$ to 1 for a given value of j. Indeed, $\boldsymbol{b}^{(j-1,j)}$ only depends on $V_{j,j}$ which is equal to 1. Consequently, from (8), we can compute $V_{j-1,j}$, then we deduce $\boldsymbol{b}^{(j-2,j)}$ and so on. Columns of $\boldsymbol{V}$ are obtained by repeating this process for all j in $[1;N]_{\mathbb{N}}$. We finally compute $\boldsymbol{A}$ from $\boldsymbol{L}$ and $\boldsymbol{V}$.

3. A SEMI-ALGEBRAIC CP DECOMPOSITION

The CP decomposition states that for any Q-th order tensor (or Q-way array) $\boldsymbol{\mathcal{T}} = (\mathcal{T}_{i_1,\cdot,i_Q})$ of size $(I_1 \times \cdots \times I_Q)$, it exists a minimal integer R such that $\boldsymbol{\mathcal{T}}$ can be exactly decomposed as:

$$\mathcal{T}_{i_1,\cdots,i_Q} = \sum_{r=1}^{R} X_{i_1,r}^{(1)} \cdots X_{i_Q,r}^{(Q)}, \quad (9)$$

where $\boldsymbol{X}^{(q)}$ defines the q-th "factor" matrix of size $(I_q \times R)$. R is called the tensor rank. The problem is thus to find the Q factor matrices from $\boldsymbol{\mathcal{T}}$.

Tensor dimensions can be merged in order to store all tensor entries in a single "unfolding" matrix. Obviously, there are many possible unfolding matrices. This choice has an impact on the identifiability conditions and on the performances of the CP method. We define $\pi_a^b = I_a I_{a+1} \cdots I_b$. Let $\boldsymbol{T}(P)$ be the $(\pi_1^P \times \pi_{P+1}^Q)$ unfolding matrix of $\boldsymbol{\mathcal{T}}$ given by:

$$\forall (m,n) \in [1;\pi_1^P]_{\mathbb{N}} \times [1;\pi_{P+1}^Q]_{\mathbb{N}},\ T(P)_{m,n} = \mathcal{T}_{i_1,\cdots,i_Q}, \tag{10}$$

with:

$$m = i_1 + \sum_{q=2}^{P} (i_q - 1)\pi_1^{q-1}\ ;\ n = i_{P+1} + \sum_{q=P+2}^{Q} (i_q - 1)\pi_{P+1}^{q-1}.$$

Any unfolding matrix of $\boldsymbol{\mathcal{T}}$ can be merely obtained by permuting the tensor dimensions and varying the P value. Then by using the Khatri-Rao product denoted by $\odot$ and after some straightforward computations, (9) can be rewritten as:

$$\boldsymbol{T}(P) = \boldsymbol{Y}_{\boldsymbol{X}}^{(P,1)} \boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)\intercal},$$

with:

$$\boldsymbol{Y}_{\boldsymbol{X}}^{(b,a)} = \boldsymbol{X}^{(b)} \odot \boldsymbol{X}^{(b-1)} \odot \boldsymbol{X}^{(b-2)} \odot \cdots \odot \boldsymbol{X}^{(a)}, (b > a).$$

As the SALT method is considered, $\boldsymbol{T}(P)$ has to be of rank R (hypothesis $\mathcal{H}_1$). Let $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\intercal$ be the singular value decomposition of $\boldsymbol{T}(P)$, truncated at order R. Thus it exists a non singular square matrix $\boldsymbol{W}$ of size $(R \times R)$ such that:

$$\boldsymbol{Y}_{\boldsymbol{X}}^{(P,1)} = \boldsymbol{U}\boldsymbol{W} \text{ and } \boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)\intercal} = \boldsymbol{W}^{-1}\boldsymbol{S}\boldsymbol{V}^\intercal. \tag{11}$$

Recalling that $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)} = \boldsymbol{X}^{(Q)} \odot \boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)}$ and using the definition of the Khatri-Rao product, $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)\intercal}$ can be seen as an horizontal block matrix:

$$\boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)\intercal} = \left[\boldsymbol{\phi}^{(1)}\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)\intercal}, \cdots, \boldsymbol{\phi}^{(I_Q)}\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)\intercal}\right], \tag{12}$$

where $\boldsymbol{\phi}^{(1)}, \cdots, \boldsymbol{\phi}^{(I_Q)}$ are the I_Q diagonal matrices built from the I_Q rows of matrix $\boldsymbol{X}^{(Q)}$. As a consequence, (11) and (12) yield:

$$\boldsymbol{S}\boldsymbol{V}^\intercal = \left[\boldsymbol{\Gamma}^{(1)\intercal}, \cdots, \boldsymbol{\Gamma}^{(I_Q)\intercal}\right],$$

where $\boldsymbol{\Gamma}^{(i)} = \boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)}\boldsymbol{\phi}^{(i)}\boldsymbol{W}^\intercal$ for any $i \in [1;I_Q]_{\mathbb{N}}$. All matrices $\boldsymbol{\Gamma}^{(i)}$ and matrix $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)}$ are of size $(\pi_{P+1}^{Q-1} \times R)$. Assuming that these are full column rank (hypothesis $\mathcal{H}_2$), then they all admit a Moore-Penrose matrix inverse denoted by $\sharp$. Thereby, we can define for any couple (i_1,i_2) belonging to $[1;I_Q]_{\mathbb{N}}^2$:

$$\begin{aligned}
\boldsymbol{\Theta}^{(i_1,i_2)} &= \boldsymbol{\Gamma}^{(i_1)\sharp}\boldsymbol{\Gamma}^{(i_2)}, \\
&= \boldsymbol{W}^{-\intercal}\boldsymbol{\phi}^{(i_1)\sharp}\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)\sharp}\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)}\boldsymbol{\phi}^{(i_2)}\boldsymbol{W}^\intercal, \\
&= \boldsymbol{W}^{-\intercal}\boldsymbol{\Lambda}^{i_1,i_2}\boldsymbol{W}^\intercal,
\end{aligned}$$

where $\boldsymbol{\Lambda}^{(i_1,i_2)} = \boldsymbol{\phi}^{(i_1)\sharp}\boldsymbol{\phi}^{(i_2)}$ are diagonal matrices. As a result, $\boldsymbol{W}^{-\intercal}$ performs the JEVD of the set of matrices $\boldsymbol{\Theta}$ which are full rank. Assuming that $\boldsymbol{X}^{(Q)}$ has at least two rows whose entries are non-zero (hypothesis $\mathcal{H}_3$), this subset is not empty and $\boldsymbol{W}^{-\intercal}$ can thus be estimated by the JET algorithm. Then one can immediately deduce $\boldsymbol{Y}_{\boldsymbol{X}}^{(P,1)}$ and $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)}$ from (11).

At this stage, column r of $\boldsymbol{Y}_{\boldsymbol{X}}^{(P,1)}$ can be reshaped into a P-th order, rank-1 tensor $\boldsymbol{\mathcal{Y}}_{\boldsymbol{X}_r}^{(1,P)}$ whose factor vectors are the r-th columns of the P matrices $\boldsymbol{X}^{(1)}, \cdots, \boldsymbol{X}^{(P)}$, respectively. Thereby a simple rank-1 HOSVD [18] of $\boldsymbol{\mathcal{Y}}_{\boldsymbol{X}_r}^{(P,1)}$ provides their estimation. In the same way, the column r of $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q,P+1)}$ can be reshaped in a $(Q-P)$-th order, rank-1 tensor $\boldsymbol{\mathcal{Y}}_{\boldsymbol{X}_r}^{(Q,P+1)}$ whose factor vectors are the r-th columns of matrices $\boldsymbol{X}^{(P+1)} \cdots \boldsymbol{X}^{(Q)}$, which can be estimated from the rank-1 HOSVD of $\boldsymbol{\mathcal{Y}}_{\boldsymbol{X}_r}^{(Q,P+1)}$. Finally, we have just to repeat both operations for all the r values to solve the problem.

We must choose a permutation of the tensor dimensions and a P value that ensure $\mathcal{H}_1$, $\mathcal{H}_2$ and $\mathcal{H}_3$. This set of conditions is necessary and sufficient to compute the CP decomposition using the SALT algorithm. It is worth mentioning that these conditions become weak for high order arrays. Notably, at orders higher than 3, the rank of the considered tensor is not required to exceed two of its dimensions contrary to the CFS algorithm. Note that $\mathcal{H}_1$ and $\mathcal{H}_2$ imply $R \leq \min(\pi_1^P, \pi_{P+1}^{Q-1})$. Even if several candidates often fulfill the conditions, we recommend to choose a value of P and a permutation of the tensor dimensions that give matrices $\boldsymbol{T}(P)$ and $\boldsymbol{Y}_{\boldsymbol{X}}^{(Q-1,P+1)}$ with the highest maximal rank. In practice, this usually leads to maximize $\min(\pi_1^P, \pi_{P+1}^{Q-1})$.

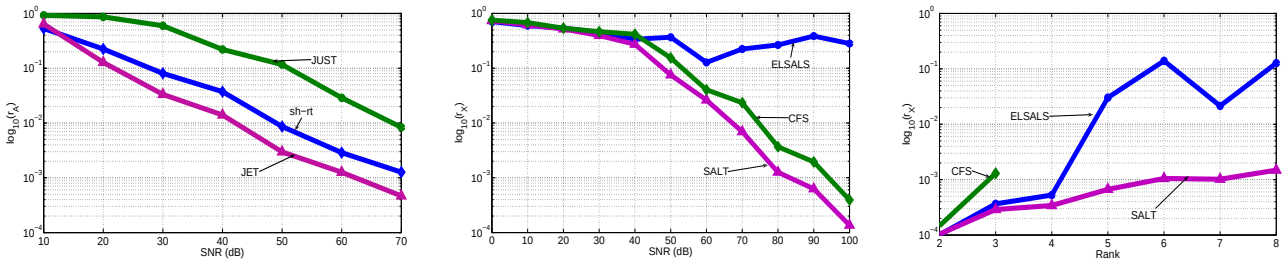
4. NUMERICAL RESULTS

4.1. Performances comparison of the JET algorithm

The JET algorithm is compared to the sh-rt [15] and JUST [16] methods by means of Monte-Carlo (MC) simulations. Entries of the eigenvector $\boldsymbol{A}$ and diagonal matrices $\boldsymbol{D}^{(k)}$ are randomly drawn according to a standard normal distribution. A Gaussian white noise is added to the matrix set to be jointly diagonalized. Algorithms are evaluated according to a normalized root mean squared error on the estimated eigenvector matrix, denoted by r_A. We vary the SNR from 10 dB to 70 dB whereas K and N are fixed to 10 and 5, respectively. The median value of r_A obtained from the 100 MC runs is plotted on figure 1(a). It appears that at 10 dB, JET and sh-rt provide very closed results. Conversely, beyond 10 dB, the JET algorithm consistently outperforms both techniques based on the polar decomposition.

4.2. Performance comparison of the SALT algorithm

We have compared SALT with the CFS and ALS with ELS (ELSALS) algorithms. Implemented versions of SALT and CFS resort to the JET algorithm to solve the JEVD problem. The ELS procedure is run every 3 ALS iterations. Each algorithm gives for each factor matrix a normalized root mean squared estimation error whose median values are computed from 100 MC experiments and denoted by $r_X^{(q)}$. Our estimation criterion r_X is then: $r_X = \frac{1}{Q}\sum_{q=1}^{Q} r_X^{(q)}$.



(a) *The JEVD problem.* (b) *CP decomposition with correlated factors.* (c) *CP decomposition of* 8*-th order tensors.*

Fig. 1. JEVD and CP decomposition algorithm comparison. Evolution of the estimation errors.

The SALT algorithm should be particularly interesting in two cases: when some columns in the factor matrices are almost collinear and/or when the tensor order is high. In the first case, iterative algorithms have difficulties to avoid local minima. This is highlighted by our first simulation, namely the CP of a third order tensor of size $(4 \times 4 \times 4)$ and rank 3. Two columns of the random factor matrices are correlated. A white Gaussian noise is added and we vary the SNR from 100 to 10 dB. r_X values are plotted on figure 1(b). We also notice that SALT performs slightly better than CFS. In the second case one can take benefit of the tensor dimensions to easily ensure the necessary conditions and choose the more suitable unfolding matrix $\boldsymbol{T}(P)$. This is pointed out by our second simulation for which we consider 8-th order tensors whose all dimensions are equal to 3. The SNR is set to 50 dB, factors are uncorrelated, the SALT parameter P is set to 4 and we vary the tensor rank from 2 to 8. Results are plotted on figure 1(c). In this case, CFS cannot go beyond rank 3 because of its necessary condition while ELSALS provides very poor results beyond rank 4. Conversely, SALT gives satisfying results whatever the considered rank.

5. CONCLUSION

Our contribution is twofold. Indeed we have proposed a new semi-algebraic approach for the CP decomposition along with an original JEVD algorithm. Combined together, these methods define a reliable CP decomposition algorithm called SALT. Simulation results show i) the efficiency of our JEVD algorithm and ii) that SALT can overcome standard CP decomposition algorithms in several situations, notably in the case of high order tensors or when two or more factors are correlated.

6. REFERENCES

[1] J. D. Carroll and J. J. Chang, "Analysis of Individual Differences in Multidimensional Scaling via N-Way Generalization of Eckart-Young Decomposition," *Psychometrika*, 35 (3), 283-319 (1970).

[2] H. Becker, P. Comon, L. Albera, M. Haardt and I. Merlet, "Multiway Space-Time-Wave-Vector Analysis for Source Localization and Extraction," in *EUSIPCO 2010*, Aalborg.

[3] R. Bro, "PARAFAC, Tutorial and Applications," *Chemom. Intel. Lab. Syst.*, 38, 149-171 (1997).

[4] J. B. Kruskal, "Three-Way Arrays: Rank and Uniqueness of Trilinear Decompositions," *Linear Algebra and Applications*, 18, 95-138 (1977).

[5] L. De Lathauwer, "A Link between Canonical Decomposition in Multilinear Algebra and Simultaneous Matrix Diagonalization," *SIAM Journal on Matrix Analysis*, 28 (3), 642-666 (2006).

[6] R. Harshman, "Foundations of the Parafac procedure: Models and conditions for an explanatory multimodal factor analysis," *UCLA Working Papers in Phonetics*, 16, 1-84 (1970).

[7] M. Rajih, P. Comon and R. Harshman, "Enhanced Line Search : A Novel Method to Accelerate PARAFAC," *SIAM Journal on Matrix Analysis and Applications*, 30 (3), 1148-1171 (2008).

[8] A.Karfoul, L. Albera and L. De Lathauwer "Iterative methods for the canonical decomposition of multi-way arrays: application to blind underdetermined mixture identification," *to appear in Elsevier Signal Processing*.

[9] P. Comon, X. Luciani and A.L.F. De Almeida, "Tensor Decompositions, Alternating Least Squares and other Tales," *Journal of Chemometrics*, 23 (9), 393-405 (2009).

[10] F. Roemer and M. Haardt, A closed-form solution for Parallel Factor (PARAFAC) Analysis, In *IEEE ICASSP 2008*, 2365-2368.

[11] F. Roemer and M. Haardt, A closed-form solution for multilinear PARAFAC decompositions, In *IEEE SAM 2008*, 487-491.

[12] L. De Lathauwer, B. De Moor and J. Vandewalle, "Computation of the Canonical Decomposition by Means of a Simultaneous Generalized Schur Decomposition," *SIAM Journal on Matrix Analysis and Applications*, 26, 295-327 (2001).

[13] A. J. van der Veen, P. B. Ober and E. F. Deprettere, "Azimuth and elevation computation in high resolution DOA estimation," *IEEE Trans. Signal Proc.* 40, 1828-1832 (1992).

[14] M. Haardt and J. A. Nossek, "Simultaneous Schur decomposition of several nonsymmetric matrices to achieve automatic pairing in multidimensional harmonic retrieveal problems," *IEEE Trans. Signal Proc.* 46, 161-169 (1998).

[15] T. Fu and X. Gao, "Simultaneous Diagonalization with Similarity Transformation for Non-defective Matrices," *In IEEE ICASSP 2006*, 1137-1140.

[16] R. Iferroudjene, K. Abed Meraim and A. Belouchrani, " A New Jacobi-like Method for Joint Diagonalization of Arbitrary non-defective Matrices," *Applied Mathematics and Computation* 211, 363-373 (2009)

[17] X. Luciani and L. Albera, Joint Eigenvalue Decomposition using Polar Matrix Factorization, In *LVA/ICA 2010*, 555-562.

[18] L. De Lathauwer, B. De Moor and J. Vandewalle, "A multilinear singular value decomposition," *SIAM Journal on Matrix Analysis and Applications* 21 (4), 1253-1278 (2000).