

A Survey on PARATREE and SUSVD Decomposition Techniques and Their Use in Array Signal Processing

Vineet Bhatt

Department of Mathematics, HNB Garhwal University, Campus Badshahi Thaul, Tehri Garhwal-249199, Uttarakhand, India

vineet.bhatt58@gmail.com

Sandeep Kumar*

Department of Mathematics, Govt. P.G. College, New Tehri, Tehri Garhwal, Pin: 249 001, Uttarakhand, India
drsandeepkumarmath@hotmail.com

Abstract —The present manuscript is intended to review few applications of tensor decomposition model in array signal processing. Tensor decomposition models like HOSVD, SVD and PARAFAC are useful in signal processing. In this paper we shall use higher order tensor decomposition in signal processing. Also, a novel orthogonal non-iterative tensor decomposition technique (SUSVD), which is scalable to arbitrary high dimensional tensor, has been applied in MIMO channel estimation. The SUSVD provides a tensor model with hierarchical tree structure between the factors in different dimensions. We shall use a new model known as PARATREE, which is related to PARAFAC tensor models. The PARAFAC and PARATREE both describe a tensor as a sum of rank-1 tensors, but PARATREE has several advantages over PARAFAC, when it is applied as a lower rank approximation technique. PARATREE is orthogonal, fast and reliable to compute, and the order of the decomposition can be adaptively adjusted. The low rank PARATREE approximation has been applied to measure noise suppression for tensor valued MIMO channel sounding measurements.

Index Terms —MIMO, SVD, PARATREE, SUSVD, tensor decompositions, signal processing

I. INTRODUCTION

A higher order tensor is any N-dimensional collection of data. It is generally known as tensor or a multidimensional array. Tensor decompositions and factorizations were initiated by Hitchcock in 1927[1], [2] and later developed by Cattell in 1944 [3] and by Tucker in 1966[4]. Tensor factorizations or decompositions play a fundamental role in enhancing the data and extracting latent components. In the era, tensor is used in a wide variety of applications such as in signal processing [5], data mining [6], neuroscience [7], and many more. In various signal processing applications, instrumental data contains information in more than two dimensions. Recently, researchers have contributed a large amount

of research regarding several application areas of well-established matrix operations up to their tensor equivalents. Unfortunately, these extensions from their matrix counterparts are not trivial. For example the SVD has proven to be a powerful tool for analyzing matrix or 2nd-order tensors, its generalization to higher order tensors is not straightforward. There are several approaches for doing this, and none of them is superior in all aspects. Basically there are three fundamental approaches to decompose a higher order tensor, first one is Tucker model (multi-linear SVD or HOSVD) [3], second is CANDECOMP/PARAFAC [8] and the third approach is non-negative tensor factorization. Both the CP and Tucker tensor decomposition can be considered as higher-order generalization of the matrix SVD and PCA respectively. If any tensor decomposes into sum of rank-1 tensor, this type of decomposition is often called “Canonical Decomposition” (CANDECOMP) or “Parallel Factors” model (PARAFAC) [8]. It has been applied in many signal processing applications, such as image recognition, acoustics, wireless channel estimation [9] and array signal processing [10], [11]. Recently, a Tucker-model based HOSVD [12] tensor decomposition subspace technique has also been formulated to improve multidimensional harmonic retrieval problems [13]. In this paper, we are attempting to pursue the contribution of higher order tensor decomposition in signal processing. In signal processing, data obtained from MIMO channel sounding measurements is a good example of tensor-valued data. It is well known that a PARATREE model is an enhanced version of PARAFAC model. Also the PARATREE tensor model is useful in signal processing; it is applied to suppress measurement noise in multidimensional MIMO radio channel measurements. This is Performed by identifying the PARATREE components spanning the noise subspace, and removing their contribution from the channel observation. Therefore in subsection 3 of section II, a novel PARATREE tensor model has been introduced, which is accompanied with SUSVD algorithm. As the rank-1

tensor decomposition technique is suitable for several tensor decomposition models therefore by additionally imposing this technique, the PARATREE model can be efficiently applied to approximate higher-order tensor. One example of such application involves interpreting the vector of eigenvalues of a large covariance matrix as a tensor, which is then used in a linear algebraic expression to finding the FIM. Approximating this tensor using PARATREE decomposition allows a significant reduction in computational complexity over a straight-forward matrix multiplication or any other exact solution. PARATREE also achieves a significant complexity reduction against HOSVD and PARAFAC. However, the use of PARATREE in practice is far more convenient than PARAFAC since PARATREE is capable of decomposition n-way array into a sequence of 1-rank tensor and hence SUSVD does work even when the convergence problem by PARATREE arises. Also the order of the PARATREE decomposition can be easily controlled, and the corresponding approximation error is well defined.

This manuscript is organized as follows: In Section II, the useful tensor models and tensor operations are described which form the general framework for the present study. We have also described three important tensor models viz. Tucker, PARAFAC and PARATREE. Also, the methods for computing tensor decompositions like ALS, HOSVD and SUSVD are discussed in brief. Rank approximation, deflating the full SUSVD and Deflating the full HOSVD is also described herein. Further, in Section III, we have delineated the MIMO propagation channel modelling, MIMO system channel and signal model, dense multipath component, FIM-key quantity of parameter estimation, noise suppression of multidimensional radio channel measurements and have pursued some of the relevant applications. At the end, conclusion over the proposed study has been outlined.

II. USEFUL TENSORS MODELS AND TENSOR OPERATIONS

Some tensor models and tensor operations are useful for signal processing e.g. PARAFAC model, which is commonly used for signal modelling and estimation purposes, and the orthogonal models such as HOSVD which better suit for tensor approximation, data compression, and filtering applications. In this manuscript we are going to apply tensor models to MIMO channel modelling. As a preliminary, first we describe general tensor operation and models as below:

A. General tensor operation

Here we introduce operations for N-mode tensors, the term N-mode or N-way tensors can be used to describe any N-dimensional data structure. An N-dimensional data structure is known as higher order tensor. Some basic operations for an N-dimensional tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_n \times \dots \times I_N}$ are defined as below:

1. The N-mode matrix unfolding of a higher order tensor

The N-mode matrix unfolding \mathbf{A}_n of a tensor \mathcal{A} comprises of:

- 1.) Permutation of tensor dimensions into an order $\{n, n + 1, \dots, N, 1, \dots, n - 1\}$ and
- 2.) Reshaping the permuted tensor into a matrix \mathbf{A}_n such that $\mathbf{A}_n \in \mathbb{C}^{I_n \times \prod_{i \neq n} I_i}$, i.e.,

$$\mathbf{A}_n = \text{reshape}[\text{permute}\{\mathcal{A}, (n, n + 1, \dots, N, 1, \dots, n - 1)\}, \{I_n, \prod_{i \neq n} I_i\}] \quad (1)$$

The order in which the columns of the matrix after unfolding are chosen in the latter step is not important, as long as, the order is known and remains constant throughout the calculations. A more general treatment of the unfolding (or matricization), including nesting of several modes in the matrix rows, is given in [14].

2. The n-mode multiplication

The n-mode multiplication $\mathcal{A} \times_n \mathbf{U} \in \mathbb{C}^{I_1 \times \dots \times R_n \times \dots \times I_N}$ of a tensor \mathcal{A} and a matrix $\mathbf{U} \in \mathbb{C}^{R_n \times I_n}$ is defined as

$$\mathcal{A} \times_n \mathbf{U} = \text{Permute}\{\mathcal{A}_B, (n, n + 1, \dots, N, 1, \dots, n - 1)\} \quad (2)$$

Where

$$\mathcal{A}_B = \text{Reshape}\{\mathbf{U}\mathbf{A}_n, (R_n, I_{n+1}, \dots, I_1, I_N, \dots, I_{n-1})\} \quad (3)$$

3. Rank-1 tensors

A tensor \mathcal{A} is of rank-1, if it can be expressed as an outer product of N vectors [15]. For instance the tensor \mathcal{A} in terms of outer product of N vectors can be expressed as:

$$\mathcal{A} = \mathbf{a}^{(1)} \mathbf{a}^{(2)} \mathbf{a}^{(3)} \dots \mathbf{a}^{(N)} \quad (4)$$

The element of \mathcal{A} are defined as

$$\alpha_{i_1 i_2 \dots i_N} = \prod_{n=1}^N (\mathbf{a}^{(n)})_{i_n} = a_{i_1}^1 \cdot a_{i_2}^2 \dots a_{i_N}^N \quad (5)$$

B. Tensor decomposition models

As for as the MIMO study concerns, there are two important tensor decomposition data models often appeared in multi-antenna (MIMO) communications, one is Tucker model and other is PARAFAC.

1. Tucker Model

Tucker models (HOSVD) are the models that decompose the higher order tensors [4], [16]. In this model the key idea is to form a limited set of basis vectors for each mode, and express the tensor as a linear combination of the outer product of the basis vectors of different modes. A higher order tensor \mathcal{A} can be decompose by Tucker model as

$$\mathcal{A} = \mathcal{C} \times_1 \mathbf{U}^1 \times_2 \mathbf{U}^2 \times_3 \dots \times_N \mathbf{U}^N \quad (6)$$

where $\mathcal{C} \in \mathbb{C}^{R_1 \times \dots \times R_N}$ is called the core tensor and the matrices $\mathbf{U}^n \in \mathbb{C}^{I_n \times R_n}$ contain the basis vectors. Tucker

model for third order tensor can be obtained by putting $N = 3$ in (6) and can be shown as in Fig.1.

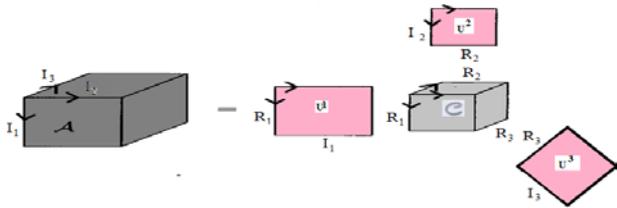


Figure 1 Tucker Model for $N = 3$

2. PARAFAC Model

The PARAFAC model or decomposition of a tensor is a sum of Rank-1 tensors. There are a number of ways to express PARAFAC decomposition [16]. Let us consider an N -mode tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ and N matrices $\mathbf{A}^n \in \mathbb{C}^{I_n \times R}$, where R is the number of factors equal to the rank of the tensor. Then the matrices $\mathbf{A}^n, n \in [1, 2, \dots, N]$, with columns $\mathbf{a}_r^n, r \in [1, 2, \dots, R]$, can be formed such that the tensor \mathcal{A} is the sum of outer products

$$\mathcal{A} = \sum_{r=1}^R \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \dots \circ \mathbf{a}_r^{(N)}, \tag{7}$$

where each outer product of the vectors \mathbf{a}_r^N is a rank-1 tensor. Equivalently, the PARAFAC model can be expressed element wise as

$$a_{i_1, i_2, \dots, i_N} = \sum_{r=1}^R a_{r, i_1}^1 \cdot a_{r, i_2}^2 \cdot \dots \cdot a_{r, i_N}^N \tag{8}$$

where i_i denotes the index in i^{th} mode. A vectorized definition is given by

$$\text{Vec}(\mathcal{A}) = (\mathbf{A}^N \circ \mathbf{A}^{N-1} \circ \dots \circ \mathbf{A}^1) \mathbf{1}_R = \sum_{r=1}^R \mathbf{a}_r^N \otimes \mathbf{a}_r^{N-1} \otimes \dots \otimes \mathbf{a}_r^1, \tag{9}$$

where $\mathbf{1}_R$ is a column vector of R ones. For third order tensor ($N = 3$), PARAFAC model is visualized by Fig.2, where the relations from (7) to (9) are given by $\mathbf{a}_r^1 = \mathbf{a}_r, \mathbf{a}_r^2 = \mathbf{b}_r$ and $\mathbf{a}_r^3 = \mathbf{c}_r$.

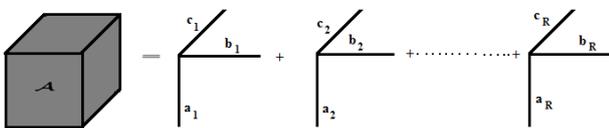


Figure 2 PARAFAC Decomposition (a sum of R rank-1 tensors)

3. PARATREE Model

In the present context a novel tensor model (PARATREE), which belongs to PARAFAC model, has been introduced. This new model has distinct hierarchical tree structures. PARATREE model can be efficiently applied to approximate higher order tensors [28]. One example of such application involves interpreting the vector of eigenvalues of large covariance matrix as a tensor, which is then used in a linear algebraic expression for finding FIM.

Approximating this tensor using PARATREE decomposition allows a significant reduction in computational complexity over a straight-forward matrix multiplication or any other exact solution. PARATREE also achieves a significant complexity reduction against HOSVD and PARAFAC. However, the use of PARATREE in practice is far more convenient than PARAFAC, since the SUSVD does not suffer from convergence problems. Also the order of the PARATREE decomposition can be easily controlled, and the corresponding approximation error is well defined [28]. In a second novel application the PARATREE model is applied to suppress measurement noise in multidimensional MIMO radio channel measurements [28]. This is performed by identifying the PARATREE components spanning the noise subspace, and removing their contribution from the channel observation. To conclude the above mentioned properties the PARATREE model, we have following benefits of this model,

1. It is reduced computational complexity in high dimensional inverse problems.
2. It is beneficial to measurement noise suppression or subspace filtering.
3. It is beneficial to compression of data or similar to low rank matrix approximation.
4. It is beneficial for fast and reliable computation and adaptive order (rank) selection.
5. It is beneficial to revealing of hidden structures and dependencies of data.

We can also think of a PARATREE model as a novel hierarchical formulation for PARAFAC-type model having not only different number of factors in different modes as in block-PARAFAC [18] or PARALIND [17], but additionally the number of factors in each mode can vary for each branch in the hierarchical tree structure. The PARATREE model for a higher order tensor can be expressed as the sum of outer products as follows

$$\mathcal{A} = \sum_{r_1=1}^{R_1} \mathbf{a}_{r_1}^1 \circ (\sum_{r_2=1}^{R_2} \mathbf{a}_{r_1, r_2}^2 \circ \dots \circ \sum_{r_{N-1}=1}^{R_{N-1}} (\mathbf{a}_{r_1, \dots, r_{N-2}, r_{N-1}}^{N-1} \circ \mathbf{a}_{r_1}^N)) \tag{10}$$

The vector $\mathbf{a}_{r_1, \dots, r_n}^n$ in (10) denotes the r_n^{th} column of the n^{th} mode matrix of basis vectors $\mathbf{A}_{r_1, \dots, r_{n-1}}^n$. The subscript r_1, r_2, \dots, r_{n-1} indicates the dependency of these matrices on the indices of the previous factors of that branch in tree. Also the number of factors R_n within each mode n can vary over different branches, i.e. R_n in (10) is actually a shorthand notation for $R_{r_1, \dots, r_{n-1}}^n$.

The visualization of PARATREE model for third way tensor in Fig: (3).

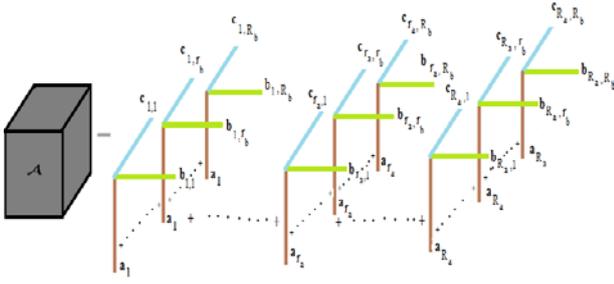


Figure 3 Three-way PARATREE decomposition, a hierarchical sum of R rank-1 tensors

Let us observe the PARAFAC model in Fig.2, the r_a^{th} basis vector \mathbf{a}_{r_a} in the first mode may be common for several factors in the remaining modes. To clarify the illustration of Fig. 3, we simply (10) for $N=3$ as follows

$$\mathcal{A} = \sum_{r_a=1}^{R_a} \mathbf{a}_{r_a} \circ \sum_{r_b=1}^{R_b} (\mathbf{b}_{r_a, r_b} \circ \mathbf{c}_{r_a, r_b}) \quad (11)$$

where the relation to (10) is obtained by setting $\{\mathbf{a}_{r_a}, \mathbf{b}_{r_a, r_b}, \mathbf{c}_{r_a, r_b}\} \equiv \{\mathbf{a}_{r_1}^1, \mathbf{b}_{r_1, r_2}^2, \mathbf{c}_{r_1, r_2}^3\}$. It is remarkable that the number of factor $R_b = R_b^{r_a}$ for the second and third mode (vector \mathbf{b}_{r_a, r_b} and \mathbf{c}_{r_a, r_b}) may depend on the factor index r_a of the first mode. In addition, the numbers of factors in the last two modes are equal. For $N=2$, the PARATREE reverts to the regular matrix SVD model. It is noteworthy that one can connect various tensor decomposition models using the well-known established connection [16]. For instance a PARAFAC model can be written as a Tucker model with a super diagonal core tensor [16]. On the other hand a Tucker model can be written as a PARAFAC model (with R equal to the number of elements in the core tensor) [16]. Hence, it is straightforward to write the PARATREE model in terms of PARAFAC or Tucker models as well. A general framework unifying the different decompositions has been recently introduced in [19].

C. Methods for computing the tensor decompositions

1. ALS Method

ALS method is the most common algorithm for fitting a PARAFAC model [21]. The basic idea is to have the number of factors R fixed and obtain an update of the n^{th} mode basis vector \mathbf{A}^n as

$$\hat{\mathbf{A}}_{\text{ALS}}^n = \mathbf{X}_n \cdot ((\mathbf{A}^N \circ \dots \circ \mathbf{A}^{n+1} \circ \mathbf{A}^{n-1} \circ \dots \circ \mathbf{A}^1)^\dagger)^t, \quad (12)$$

while keeping the basis vectors of the outer modes fixed. An iterative update of the matrices $\mathbf{A}^{(n)}$ is obtained by altering $n \in [1, 2, \dots, N]$ until a convergence is reached. The improvement of fit is monotonic. However, depending on the initial values for the matrices $\mathbf{A}^{(n)}$, a local optimum may be reached instead of the global one or the convergence may be very slow. Therefore, $\mathbf{A}^{(n)}$ are typically initialized by either using multiple random initial values, or so called rational start (based on either generalized rank annihilation or DTLD), or a semi-rational start (based on SVD/EVD) [20].

2. Higher order singular value decomposition (HOSVD)

The HOSVD is obtained by computing the matrix SVD for each 1-mode unfolding of the tensor \mathcal{A} and selecting the left singular vectors as the orthonormal basis of each mode, respectively. For the complete HOSVD, the basis matrices $\mathbf{U}^n \in \mathbb{C}^{I_n \times R_n}$ are hence given by the first $R_n = \text{rank}(\mathbf{X}_n)$ left-hand singular vectors of the SVD of \mathbf{X}_n , defined as

$$\mathbf{X}_n = \mathbf{U}^{(n)} \Sigma^{(n)} \mathbf{V}^{(n)H}. \quad (13)$$

Having computed the matrices $\mathbf{U}^{(n)}$, $n \in [1, 2, \dots, N]$ the core tensor $\mathcal{C} \in \mathbb{C}^{R_1 \times R_2 \times \dots \times R_N}$ is given in closed form as

$$\mathcal{C} = \mathcal{A} \times_1 \mathbf{U}^{(1)H} \times_2 \mathbf{U}^{(2)H} \times_3 \dots \times_N \mathbf{U}^{(N)H} \quad (14)$$

3. Sequential Unfolding Singular Value Decomposition (SUSVD)

Let us describe the SUSVD algorithm for estimating the PARATREE tensor model [28]. It is a computational method for obtaining an orthogonal PARATREE model. It is based on the idea of sequentially applying the matrix SVD [20], on an unfolding tensor formed from each of the right singular vectors of the SVD in the previous mode. SUSVD can be applied for any N-dimensional real or complex tensor and for $N = 2$, it is equal to the matrix SVD. The SUSVD decomposition for an N-way tensor $\mathcal{A}^{I_1 \times I_2 \times \dots \times I_N}$ ($I_1 \geq I_2 \geq \dots \geq I_N$) is described by the following algorithm:

Algorithm 1 [28]

{ $\{\mathbf{S}\}, \{\mathbf{U}\}$ } = SUSVD{ \mathcal{A} }

Set $\mathcal{T}_0^1 = \mathcal{A}$

Set $R^0 = 1$

For each $n = \{1, 2, \dots, N - 1\}$:

For each $r_{n-1} = \{1, 2, \dots, R_{n-1}\}$:

1. Unfold the tensor

$$\mathbf{T}_{r_1, r_2, \dots, r_{n-1}}^n = (\mathcal{T}_{r_1, r_2, \dots, r_{n-1}}^n)_{(1)'}'$$

2. Compute the SVD $\mathbf{T}_{r_1, r_2, \dots, r_{n-1}}^n = \mathbf{U}^n \Sigma^n \mathbf{V}^{nH}$,

3. For each $r_n \in \{1, 2, \dots, R^n\}$, with $R^n = \text{rank}(\mathbf{T}_{r_1, r_2, \dots, r_{n-1}}^n)$

(a) Store $\sigma_{r_1, r_2, \dots, r_{n-1}}^n = (\Sigma^n)_{r_n r_n}$ in $\{\mathbf{S}\}$ and $\mathbf{U}_{r_1, r_2, \dots, r_{n-1}}^n = (\mathbf{U}^n)_{r_n}$ in $\{\mathbf{U}\}$,

(b) Then, if $n < N - 1$,

-Reshape $(\mathbf{V}^{(n)*})_{r_n}$ in to a tensor $\mathcal{T}_{r_1, r_2, \dots, r_n}^{n+1} \in \mathbb{C}^{I_{n+1} \times \dots \times I_N}$, or else,

-Store the vector

$$\mathbf{u}_{r_1, r_2, \dots, r_{N-1}}^N = (\mathbf{V}^{(N-1)*})_{r_{N-1}}.$$

The SUSVD method and its reconstruction are visualized for a $(2 \times 2 \times 2)$ -third order tensor in fig.4.

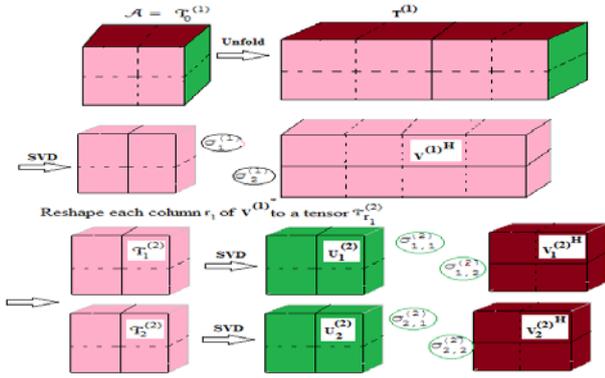


Figure 4 The SUSVD decomposition for an arbitrary (2x2x2)-tensor

The core idea of the algorithm is to apply the matrix SVD on the 1-mode matrix unfolding of the tensor (see Definition 1) to form the basis vectors of the first mode. Then each of the conjugated right-hand singular vectors $\mathbf{V}_{r_1}^{1*}$ is reshaped into tensors, and the matrix SVD is applied on the 1-mode unfolding of these tensors. This is repeated to construct the PARATREE model, until there are only the elements of the last mode contained in the right-hand singular vectors. Note that for a full SUSVD (all possible factors included) described in algorithm 1, the number of basis vectors within each mode is the same for all branches and is given by

$$R_n = \min\{M_n, \prod_{j=n+1}^{N-1} M_j\}. \quad (15)$$

Hence, the total number of orthogonal components in the decomposition is given by

$$R = \prod_{n=1}^{N-1} R_n \quad (16)$$

The $(2 \times 2 \times 2)$ -third order tensor in fig. 4 can be reconstructed with the PARATREE model as

$$\mathcal{A} = \sum_{r_1=1}^{R_1} \sigma_{r_1}^1 \cdot \mathbf{u}_{r_1}^1 \circ \sum_{r_2=1}^{R_2} \sigma_{r_1,r_2}^2 \cdot \mathbf{u}_{r_1,r_2}^2 \circ \mathbf{u}_{r_1,r_2}^3 \quad (17)$$

For avoiding the confusion in Fig.4 of the SUSVD decomposition for $(2 \times 2 \times 2)$ tensor. Different colors shows to different dimensions of the tensor. A square σ denotes a singular value, dashed blocks are elements of the tensors, and solid lines are used to separate the columns vectors. The tensor is first unfolded to a matrix $\mathbf{T}_0^{(1)}$. After applying SVD on this matrix, each of the right-hand singular vectors is reshaped and another SVD is applied on them. The procedure is repeated for each “branch” and “sub-branch”, until no additional dimensions remain in the right hand basis vectors, i.e., the matrix \mathbf{V}^{N-1} has only I_N rows. The full $(R_1 = 2, R_2 = 2)$ reconstruction is given in Fig.5. In this fig.5 a PARATREE tensor is reconstructed as a sum of outer product of weighted (by $\sigma_{r_1}^1 \sigma_{r_1,r_2}^2$) basis vectors $\mathbf{u}_{r_1}^1$, \mathbf{u}_{r_1,r_2}^2 and \mathbf{u}_{r_1,r_2}^3 . The tree structure allows common basis vectors in the previous dimensions.

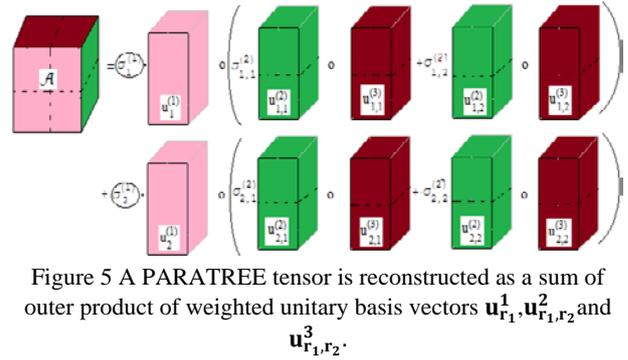


Figure 5 A PARATREE tensor is reconstructed as a sum of outer product of weighted unitary basis vectors $\mathbf{u}_{r_1}^1, \mathbf{u}_{r_1,r_2}^2$ and \mathbf{u}_{r_1,r_2}^3 .

The relation of the values in (17) to the ones in the 3D-PARATREE formulation (11) or the general form (10) is given by

$$\begin{aligned} \mathbf{a}_{r_a} &= \mathbf{a}_{r_1}^1 = \sigma_{r_1}^1 \mathbf{u}_{r_1}^1 \\ \mathbf{b}_{r_a,r_b} &= \mathbf{a}_{r_1,r_2}^2 = \sigma_{r_1,r_2}^2 \mathbf{u}_{r_1,r_2}^2 \\ \mathbf{c}_{r_a,r_b} &= \mathbf{a}_{r_1,r_2}^3 = \mathbf{u}_{r_1,r_2}^3. \end{aligned}$$

Note that the basis vectors of the SUSVD are exactly the same for the first mode as those of the HOSVD. However, the number of basis vectors of the latter modes is limited to $R_n = \text{rank}(X_n)$ for HOSVD, whereas in SUSVD the basis is formed independently for each branch. The result is that the total number R of individual rank-1 contributions, (e.g., as if the decomposition would be written in PARAFAC) from (7), is typically much less for the SUSVD than for the HOSVD. Another difference between the two decompositions is the fact that the HOSVD is unique, whereas for the SUSVD the solution depends on the order of the modes.

4. Reduced Rank Approximations

The individual rank-1 contributions of the HOSVD and SUSVD are orthogonal to each other. The practical implication of this property is that for a reduced rank approximation \mathcal{A}_A of a tensor \mathcal{A} , the squared magnitudes of individual terms directly contribute to the squared magnitude of the approximated tensor. Hence, the squared Frobenius norm of the tensor approximation is given for the SUSVD by

$$\|\mathcal{A}_{A,SU}\|_F^2 = \sum_{r_A} \|a_{r_A}^1 o a_{r_A}^2 o \dots o a_{r_A}^N\|_F^2 = \sum_{r_A} \sigma_{r_A}^1 \dots \sigma_{r_A}^{N-1}, \quad (18)$$

where $\sigma_{r_A}^n$ denotes the n^{th} mode singular value and r_A denotes an index of a rank-1 component included in the reduced rank decomposition. Equivalently, the squared Frobenius norm of the HOSVD approximation is given by

$$\|\mathcal{A}_{A,HO}\|_F^2 = \sum_{r_A} |(\mathcal{C})_{r_A}|^2 \quad (19)$$

where $(\mathcal{C})_{r_A}$ denotes an element of the HOSVD core tensor, and index r_A denotes the indexes contributing to the approximation.

5. Deflating the full SUSVD

The PARATREE model builds with SUSVD can be deflated to form reduced rank approximation of tensors. This can be done either offline after building the full SUSVD or online during the computation of the decomposition [28]. Here the discussion is limited to the offline approach. Due to orthogonality of the decomposition, the approximation error can be equivalently expressed in terms of the sum of the product of the singular values related to each factor. These can be interpreted as the magnitudes of the single rank-1 components in the PARATREE and are given by

$$\hat{\sigma}_{r_1, \dots, r_{N-1}} = \sigma_{r_1}^1 \cdot \sigma_{r_1, r_2}^2 \dots \sigma_{r_1, \dots, r_{n-1}}^{n-1} \quad (20)$$

By stacking all the R in (15)-(16), magnitude values of (20) is in descending order to a vector $\hat{\sigma} \in \mathbb{R}^{R \times 1}$, the normalized SUSVD approximation error can be expressed as

$$\epsilon_{r,SU} = \frac{\|\mathcal{A} - \mathcal{A}_A\|_F}{\|\mathcal{A}\|_F} = \sqrt{1 - \frac{\sum_{r=1}^{R_A} \hat{\sigma}_r^2}{\sum_{r=1}^R \hat{\sigma}_r^2}}$$

Or equivalently $\epsilon_{r,SU}^2 = 1 - \frac{\|\hat{\sigma}_A\|_F^2}{\|\hat{\sigma}\|_F^2} \quad (21)$

The Offline PARATREE approximation method is described by the following algorithm:

Algorithm 2[28]

- 1. Compute the products of the singular value (20) for each of the R Possible branches (15)-(16) of the full PARATREE structure.
- 2. Sort all the products in descending order to a vector $\hat{\sigma} \in \mathbb{R}^{R \times 1}$.
- 3. Pick the minimum number R_A of singular values $\hat{\sigma}_A \in \mathbb{R}^{R \times 1}$, fulfilling the criterion equation (21)

$$\|\hat{\sigma}_A\|_F^2 \geq (1 - \epsilon_r^2) \|\hat{\sigma}\|_F^2$$

- 4. Construct $\{S_A\}, \{U_A\}$ based on the selected singular values $\hat{\sigma}_A$

The input data to the approximation function are the abstract tree structures $\{S\}$ and $\{U\}$ from the SUSVD and threshold ϵ_r for the target normalized approximation error (21). The output data consist of similar structures, but with reduced number of factors to approximate the tensor. The rank of the approximation is given by

$$R_A = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \dots \sum_{r_{N-2}=1}^{R_{N-2}} R_{r_1, \dots, r_{N-2}}^{N-1} \quad (22)$$

The number of factor $R_{r_1, r_2, \dots, r_{n-1}}^n$ in each mode n of each branch r_1, r_2, \dots, r_{n-1} , depends on how the factor magnitudes are distributed among different branches. The described offline approach allows for defining the achieved relative approximation error (21) precisely at

the price of having to form the full SUSVD. This is most useful, when saving in the computational complexity of the SUSVD decomposition itself is not crucial for the application at hand. This is the case in both of the example applications in next Section. Further, improvements in computational complexity can be achieved by truncating the decomposition in an online fashion, but then controlling the normalized error of the obtained approximation would not be as straightforward.

6. Deflating the full HOSVD

The HOSVD can be deflated either by reducing the least significant basis vectors from the matrices U^n , or by considering the full decomposition, and selecting the elements from the core tensor that one wishes to include in the approximation. Here, the discussion is limited to the latter approach as that strategy for controlling the approximation error is similar to the one applied for the SUSVD deflation. Hence, the deflation of the HOSVD is obtained by setting the undesired contributions in the core tensor to zero, yielding a sparse core tensor C_A . The obtained HOSVD approximation error is given by

$$\epsilon_{r,HO}^2 = 1 - \frac{\|C_A\|_F^2}{\|C\|_F^2} \quad (23)$$

It should be noted that the low rank tensor approximations obtained by deflation are suboptimal for both SUSVD and HOSVD. However, in the proposed application example in next Section, the optimality of the approximation in LS sense may be sacrificed to the benefit of a computationally efficient method yielding an approximation with a low rank-1 number factor in the decomposition (nonzero core tensor elements for HOSVD). For HOSVD, the truncation could be performed also by reducing the rank of each mode, which would allow optimizing the solution using Tucker3-ALS [16]. However, this method is time consuming, the achievable approximation error would not be as precisely controllable, and the obtained decomposition may still contain numerous in significant rank-1 factors. Applying a similar target approximation error requirement for a general PARAFAC model would require a trial and error approach for finding a proper rank. Also the convergence of the alternating least squares algorithms used for PARAFAC is very slow for high dimensional or ill-conditioned problems [22], [23].

III. APPLICATIONS

Let us explain some important application of above tensor decomposition models in signal processing.

A. MIMO propagation channel modeling

The forging SUSVD algorithm for estimating the PARATREE model for any given tensor valued data is useful in multi-antenna communications (MIMO). The concept of SUSVD algorithm and the PARATREE tensor model emerged while solving the problem of restoring the Kronecker structure of a data mode

appearing in multi-antenna communication. In order to understand this concept we shall go through MIMO system as below.

1. MIMO System

In wireless communications a MIMO system is a wireless communication system that has more than one antenna in both transmitter and receiver [24]. The use of MIMO systems enables many different benefits that can be used to increase the performance of a data transferring network. These include spatial multiplexing gain, spatial diversity gain and array gain, often referred to as beam forming gain [28].

2. Channel and Signal Model

We can present a typical MIMO system including the signal processing subsystems,

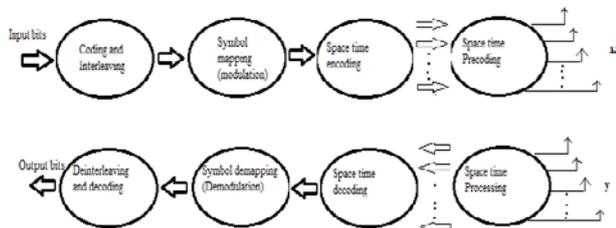


Figure 6 Process of MIMO system

and wireless channel part is extracted below:

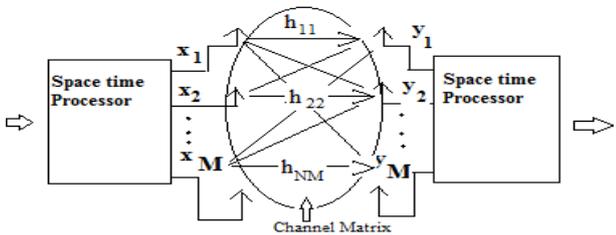


Figure 7 MIMO wireless channel part

In the above fig7 \mathbf{x} is transmitted signal vector, \mathbf{y} is received signal vector, \mathbf{n} is noise vector and \mathbf{H} is a channel matrix. The received signal vector \mathbf{y} can be expressed in terms of the channel matrix \mathbf{H} as:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{24}$$

where the symbols are

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \text{received signal vector}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \text{transmitted signal vector},$$

$$\mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_n \end{bmatrix} = \text{noise vector}$$

$$\text{and } \mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1M} \\ h_{21} & h_{22} & \dots & h_{2M} \\ \vdots & \vdots & \dots & \vdots \\ h_{N1} & h_{N2} & \dots & h_{NM} \end{bmatrix} = \text{Channel matrix}$$

3. Tensor valued data in MIMO communications (MIMO principal)

The principle of a MIMO communication system is illustrated in fig 8. The system in fig 8 has multiple (M_T) transmit (T_x) antennas and multiple (M_R) receive (R_x) antennas. The coefficients of the spatial channel between each T_x - T_R antenna pair are typically represented by a narrowband MIMO channel tensor \mathcal{H} .

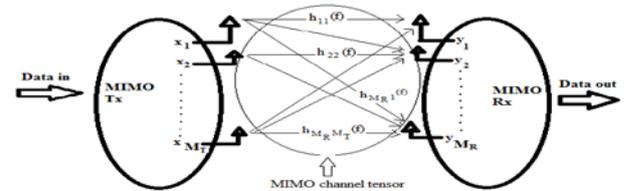


Figure 8 MIMO principal

The tensor valued measured radio channel $\mathcal{H} \in \mathbb{C}^{M_f \times M_T \times M_R}$ contains the complex coefficients describing the channel transfer function of the radio channel between the M_T transmitter antenna ports and M_R receiver antenna ports sampled at M_f frequencies. Realizations of such MIMO channels are obtained from MIMO channel sounding measurements. The measurement model for the radio wave propagation is assumed to be comprised of two model components: the dominant propagation path \mathcal{H}_s , and the diffuse scattering \mathcal{H}_D also DMC. In the following, the polarization effects are ignored to simplify the description. The propagation paths can be expressed as a constrained PARAFAC model.

$$\mathcal{H}_S(\theta) = \sum_{p=1}^P \gamma_p \{ \mathbf{b}^f(\tau_p) \circ \mathbf{b}^{(T)}(\boldsymbol{\varphi}_{T,p}, \boldsymbol{\vartheta}_{T,p}) \circ \mathbf{b}^{(R)}(\boldsymbol{\varphi}_{R,p}, \boldsymbol{\vartheta}_{R,p}) \} \tag{25}$$

where P is the number of individual propagation paths. Each rank-1 component in model (25) is parameterized with L parameters. The parameter vector $\theta \in \mathbb{R}^{L \times P \times 1}$ is defined as

$$\theta = [\boldsymbol{\tau}^T \boldsymbol{\varphi}_T^T \boldsymbol{\vartheta}_R^T \boldsymbol{\varphi}_R^T \Re\{\log(\gamma^T)\} \Im\{\log(\gamma^T)\}]^T \tag{26}$$

B. Dense multipath component

The Dense Multipath Component (DMC) is a necessary part of the MIMO channel model in order to provide means to mathematically describe the contribution of the rich diffuse scattering in the propagation channel, which cannot be modeled by the dominant propagation paths. Another way to distinguish between a propagation path and DMC is through their time varying nature. If a channel would be constant, one may be able to fully reconstruct the whole channel transfer function using a very high number of propagation paths. However, this model fails if the channel changes even slightly. This is due to the fact that most of the individually less significant channel contributions result from a superposition of signals having different spatio-temporal structure. Moreover, the

true dominant propagation paths shall prevail within a larger spatial region. Especially in a dynamic channel while either one of the terminals, or possibly a source of propagation mechanism is moving, it is evident that, regardless of the measurement system, only part of the radio channel can be modeled using the dominant propagation paths. The diffuse scattering component is defined as a tensor valued complex circular symmetric normal distributed random variable

$$\mathcal{H} \sim \mathcal{N}_c(\mathbf{0}, \mathcal{R}), \quad (27)$$

with a covariance tensor $\mathcal{R} \in \mathbb{C}^{M_f \times M_T \times M_R \times M_f \times M_T \times M_R}$.

The following Kronecker structure is assumed for the covariance tensor reshaped into a $(M \times M)$, with $M = M_f M_T M_R$ matrix

$$\mathbf{R}_D = E\{\text{vec}(\mathcal{H}_D)\text{vec}(\mathcal{H}_D)^H\} = \mathbf{R}_f \otimes \mathbf{R}_T \otimes \mathbf{R}_R \in \mathbb{C}^{M \times M}. \quad (28)$$

The matrices $\mathbf{R}_f \in \mathbb{C}^{M_f \times M_f}$, $\mathbf{R}_T \in \mathbb{C}^{M_T \times M_T}$ and $\mathbf{R}_R \in \mathbb{C}^{M_R \times M_R}$ are the covariance matrices of the frequency, the transmit array, and the receive array modes, respectively. Estimation of these covariance matrices is discussed in [24]. Furthermore, a tensor $\mathcal{H}_N(\mathbf{k})$ is defined, denoting zero mean normal distributed complex circular symmetric noise with covariance

$$\mathbf{R}_N = E\{\text{vec}(\mathcal{H}_N)\text{vec}(\mathcal{H}_N)^H\} = \sigma_N^2 \mathbf{I}. \quad (29)$$

Using (25), (27) and (29) the model for the full measured complex transfer function of the radio channel tensor at time k is defined as

$$\mathcal{H}(\mathbf{k}) = \mathcal{H}_S(\mathbf{k}) + \mathcal{H}_D(\mathbf{k}) + \mathcal{H}_N(\mathbf{k}) \sim \mathcal{N}_c(\mathcal{H}_S, \mathcal{R}), \quad (30)$$

where the covariance tensor is defined as $\mathcal{R} = \mathcal{R}_D + \mathcal{R}_N$.

The covariance tensor of (30) can be written in matrix form as

$$\mathbf{R} = E\{\text{vec}(\mathcal{H}_N)\text{vec}(\mathcal{H}_N)^H\} = \mathbf{R}_f \otimes \mathbf{R}_T \otimes \mathbf{R}_R + \sigma_N^2 \mathbf{I} \quad (31)$$

The contribution of the model components \mathcal{H}_S , \mathcal{H}_D , and \mathcal{H}_N in the PADP of a MIMO radio channel measurements.

C. The FIM – key quantity of parameter estimation

1. Computational challenges of the FIM in propagation Parameter Estimation

The model for the propagation paths (25) in vectorized form is defined as

$$h_s(\theta) = \text{vec}(\mathcal{H}_S(\theta)). \quad (32)$$

The Parameters (θ) used for identifying the model $s(\theta)$ may be estimated using, iterative Maximum Likelihood (ML) [25], or Extended Kalman Filter (EKF) [26], [27]. Both of these estimation methods rely on the evaluation of the expression

$$\mathbf{J} = 2\Re\{\mathbf{D}^H \mathbf{R}^{-1} \mathbf{D}\}, \quad (33)$$

which is commonly known as the FIM- a measure of the amount of information about θ carried in $h_s(\theta)$. This expression contains as inverse of the measurement covariance matrix \mathbf{R} (31), as well as a Jacobian matrix

$$\mathbf{D} = \frac{\partial h_s(\theta)}{\partial \theta} \in \mathbb{C}^{M \times L'}. \quad (34)$$

Due to the structure of the data model in our example application (25), the expression of the FIM (33) can be expanded as

$$\mathbf{J} = 2\Re\{(\mathbf{D}_f \diamond \mathbf{D}_T \diamond \mathbf{D}_R)^H \cdot (\mathbf{R}_f \otimes \mathbf{R}_T \otimes \mathbf{R}_R + \sigma_N^2 \mathbf{I})^{-1} \cdot (\mathbf{D}_f \diamond \mathbf{D}_T \diamond \mathbf{D}_R)^H\}. \quad (35)$$

Straight-forward computation of (34) has very high computational complexity $\mathcal{O}(\prod_i M_i^3 = M^3)$. Expression (34) also requires memory for storing the full matrix $\mathbf{R} \in \mathbb{C}^{M \times M}$ and $\mathbf{D} \in \mathbb{C}^{M \times L'}$. To facilitate feasible computation of the FIM (32), the positive definite covariance matrix \mathbf{R} (31) can be expressed in terms of its eigenvalue decomposition as

$$\mathbf{R} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^H + \sigma^2 \mathbf{I}_M = (\mathbf{U}_R \otimes \mathbf{U}_T \otimes \mathbf{U}_f) (\mathbf{\Lambda}_R \otimes \mathbf{\Lambda}_T \otimes \mathbf{\Lambda}_f + \sigma^2 \mathbf{I}_M) (\mathbf{U}_R \otimes \mathbf{U}_T \otimes \mathbf{U}_f)^H. \quad (36)$$

The FIM can thus be expressed as can thus be expressed as

$$\mathbf{J} = 2\Re\{(\mathbf{D}'_R \diamond \mathbf{D}'_T \diamond \mathbf{D}'_f)^H \mathbf{\Lambda}^{-1} (\mathbf{D}'_R \diamond \mathbf{D}'_T \diamond \mathbf{D}'_f)\} \quad (37)$$

Where

$$\mathbf{D}'_i = \mathbf{U}_i^H \mathbf{D}_i, i \in \{f, T, R\} \quad (38)$$

$$\text{and } \mathbf{\Lambda} = (\mathbf{\Lambda}_R \otimes \mathbf{\Lambda}_T \otimes \mathbf{\Lambda}_f + \sigma^2 \mathbf{I}_M). \quad (39)$$

This form has computational complexity $\mathcal{O}(L'^2 \prod_i M_i)$, and it requires storing the full matrix \mathbf{D} (33). Let us reshape the diagonal elements of $\mathbf{\Lambda}^{-1}$ in (38) into a tensor $\mathcal{L} \in \mathbb{C}^{M_f \times M_T \times M_R}$ as

$$\mathcal{L} = \text{reshape}(\text{diag}(\mathbf{\Lambda}^{-1}), \{\mathbf{M}_f, \mathbf{M}_T, \mathbf{M}_R\}). \quad (40)$$

This tensor is, in general, of full rank. One feasible solution for computing the FIM (36) is then given by

$$\mathbf{J}_0 = 2\Re\left\{\sum_{m_f=1}^{M_f} \left[\mathbf{d}'_{f,m_f} \mathbf{d}'_{f,m_f} \odot \sum_{m_T=1}^{M_T} \left[\mathbf{d}'_{T,m_T} \mathbf{d}'_{T,m_T} \sum_{m_R=1}^{M_R} \left[\mathbf{1}_{m_f, m_T, m_R} \cdot \mathbf{d}'_{R,m_R} \mathbf{d}'_{R,m_R} \right] \right] \right] \right\}, \quad (41)$$

where \mathbf{d}'_{m_i} , $i \in \{f, T, R\}$, denotes the m_i^{th} row of matrix \mathbf{D}'_i in (37). Expression (40) is exact and does not require storing the full matrix \mathbf{D} in (33), but has the same computational complexity

$$\mathcal{O}(L'^2 \prod_i M_i) \quad (42)$$

as in (37).

2. PARATREE model for solving the FIM

The PARATREE model can be applied to reduce the computational complexity of (41). The tensor \mathcal{L} in (40) is decomposed into a PARATREE model with a single matrix of basis vectors $\mathbf{L}_f^f \in \mathbb{R}^{M_f \times R_f}$ for the f-mode, and \mathbf{R}_f matrices $\mathbf{L}_{rf}^T \in \mathbb{R}^{M_T \times R_T}$ and $\mathbf{L}_{rf}^R \in \mathbb{R}^{M_R \times R_T}$ for the T-modes and R-modes. Then a PARATREE approximation \mathbf{J}_{PT} for the FIM can be expressed as

$$\mathbf{J}_{PT} = 2\Re \left\{ \sum_{(r_f=1)}^{R_f} \left[(\mathbf{D}_f^H \boldsymbol{\Lambda}_{rf}^{(f)} \mathbf{D}_f') \odot \sum_{r_T}^{R_T} [(\mathbf{D}_T^H \boldsymbol{\Lambda}_{rf,r_T}^{(T)} \mathbf{D}_T') \odot (\mathbf{D}_R^H \text{diag} \boldsymbol{\Lambda}_{rf,r_T}^{(R)} \mathbf{D}_R)] \right] \right\}, \quad (43)$$

where $\boldsymbol{\Lambda}_{rf,r_T}^R = \text{diag}((\mathbf{L}_{rf}^R)_{r_T})$ denotes a diagonal matrix formed from the r_T^{th} column of \mathbf{L}_{rf}^R etc. this solution has computational complexity

$$\mathcal{O}(L'^2 \cdot 2R_f(M_f + R_T(M_T + M_R))). \quad (44)$$

In practice $R_f \ll M_f$ and $R_T \ll M_T$, see [36], which provides a significant reduction in computational complexity compared to (42). Similar to (43), an expression for evaluating the FIM using PARAFAC is given by

$$\mathbf{J}_{PF} = 2\Re \left\{ \sum_{r_f}^{R_f} \left[(\mathbf{D}_f^H \boldsymbol{\Lambda}_r^{(f)} \mathbf{D}_f') \odot (\mathbf{D}_T^H \boldsymbol{\Lambda}_r^{(T)} \mathbf{D}_T') \odot (\mathbf{D}_R^H \boldsymbol{\Lambda}_r^{(R)} \mathbf{D}_R) \right] \right\}, \quad (45)$$

which has computational complexity in the order of

$$\mathcal{O}(L'^2 \cdot 2R(M_T + M_R + M_f)). \quad (46)$$

Furthermore, a computational strategy for evaluating the FIM using HOSVD is given by

$$\mathbf{J}_{HO} = 2\Re \left\{ \sum_{r_f}^{R_f} \left[(\mathbf{D}_f^H \boldsymbol{\Lambda}_{rf}^{(f)} \mathbf{D}_f') \odot \sum_{r_T}^{R_T} \left[(\mathbf{D}_T^H \boldsymbol{\Lambda}_{r_T}^{(T)} \mathbf{D}_T') \odot \sum_{r_R}^{R_R} [s_{r_f,r_T,r_R} \cdot (\mathbf{D}_R^H \boldsymbol{\Lambda}_{r_T}^{(R)} \mathbf{D}_R)] \right] \right] \right\}, \quad (47)$$

where s_{r_f,r_T,r_R} denotes an element of the core tensor. In (47) only the terms corresponding to a nonzero core tensor value need to be evaluated. Hence, the computational complexity of (47) is given by

$$\mathcal{O}(L'^2 \cdot 2(R_f M_f + \sum_{r_f} [R_{rf}^T M_T + M_R \cdot \sum_{r_T} R_{rf,r_T}^R])), \quad (48)$$

where numerical value is again based on the decomposition yielding. It should be mentioned that further reduction in computational complexity using HOSVD could be achieved, at the cost of very high memory consumption, if all the terms $\mathbf{D}_i^H \boldsymbol{\Lambda}_{r_i}^{(i)} \mathbf{D}_i'$ would be stored while computed for the first time. However, with the current system dimensions in Algorithm 3 (and also considering that a much higher value for L' is possible), the memory requirements for such strategy

become prohibitive. Given the proposed approaches to approximate the FIM using PARATREE (43), PARAFAC (45) and HOSVD (47), the PARATREE/SUSVD provides the best performance in terms of computational complexity.

D. D. Noise Suppression of Multidimensional Radio Channel Measurements

Another novel application to utilize the PARATREE/SUSVD is noise suppression for MIMO channel sounding measurement data [37], [38]. These data are often directly used, e.g., in link-level simulations of a wireless communication system (as opposed to drawing channel realizations based on measurement-based parametric modeling [39]). The tensor decomposition based filtering is very useful for enhancing the SNR (Signal-to-Noise Ratio) of the measured channel data to be used in the simulator. This allows a wider range of noise power (or other interfering signals) to be defined within the simulation. For convenience, the time index k in (30) is dropped and a single snapshot \mathcal{H} of a channel sounding measurement is considered. The nominal SNR of the measurement is defined as

$$S_{dB}(\mathcal{H}) = 10 \cdot \log_{10} \frac{P_H - P_N}{P_N}, \quad (49)$$

where $P_H = \|\mathcal{H}\|_F^2$ is the total power in the measurement, and $P_N = \|\mathcal{H}_N\|_F^2$ is the power of the measurement noise. These quantities are assumed to be known, which is a valid assumption in channel sounding (the noise power P_N may be assessed, e.g., by sampling while T_X is off or by estimating it from excess delay samples). The suppression of the measurement noise is achieved by the following procedure:

Algorithm 3

1. Compute the SUSVD of \mathcal{H} , as described above algorithm in section.
2. Define a threshold ϵ_r (21) for selecting the factors, i.e., the signal subspace, for the approximation.
. Here $\epsilon_r = \sqrt{\frac{P_N}{P_H}}$ is chosen, i.e., only the factors whose cumulative power exceeds the noise power are included in the decomposition.
3. Approximate \mathcal{H} by \mathcal{H}_A , with the procedure described in Table II.

This filter can be equivalently expressed using a projector matrix to the signal subspace [37] defined as

$$\mathbf{\Pi}_A = \left(\mathbf{U}_A^{(R)} \diamond \mathbf{U}_A^{(T)} \diamond \mathbf{U}_A^{(f)} \right) \left(\mathbf{U}_A^{(R)} \diamond \mathbf{U}_A^{(T)} \diamond \mathbf{U}_A^{(f)} \right)^H, \quad (50)$$

where the matrices $\mathbf{U}_A^i \in \mathbb{C}^{M_i \times R_A}$ contain all the R_A (23) factor combinations expanded in PARAFAC fashion (redundancy in columns possible). The filtered channel estimate is then given by

$$\text{vec}(\mathcal{H}_A) = \mathbf{\Pi}_A \text{vec}(\mathcal{H}) \quad (51)$$

This method effectively suppresses the measurement noise. It will also be shown that the approximation is beneficial in terms of data compression.

IV CONCLUSION

This paper reviews sequential unfolding SVD (SUSVD) a novel orthogonal, non-iterative tensor decomposition technique, which is scalable to arbitrary high dimensional tensors. The SUSVD provides a tensor model with hierarchical tree structure between the factors in different dimensions. The new model, named as PARATREE, is related to the family of PARAFAC tensor models. Links between PARATREE and other existing models are enlightened in the paper as well. The PARATREE model can be used for flexible low rank tensor approximation with precisely defined approximation error level through deflation of the orthogonal rank-1 components. The low rank PARATREE approximation can be used for reducing the computational complexity in high dimensional problems, measurement noise suppression, data compression, as well as providing insight on structures and dependencies in the data. Whereas similar strategy is obtainable for Tucker3 based HOSVD, the PARATREE clearly outperforms HOSVD (as well as PARAFAC) in terms of computational complexity while solving the Fisher Information matrix in a given application example. This is due to rich structure of the basis vectors resulting from the SUSVD, yielding more precise approximation with smaller tensor rank. Another novel application example involves noise suppression for tensor valued MIMO channel sounding measurements. The results for this application also indicate the superiority of PARATREE over HOSVD.

ACKNOWLEDGMENT

The author(s) express their hearty thanks and gratefulness to all those scientists whose masterpieces have been consulted during the preparation of the present research article. Special thanks are due to Professor Jussi Salmi [28], who's fabulous research work have been entirely consulted to expose the present manuscript.

REFERENCES

- [1] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. Phys.*, Vol. 6, pp. 164-189, 1927.
- [2] F. L. Hitchcock, "Multiple invariants and generalized rank of a p-way matrix or tensor," *J. Math. Phys.*, Vol. 7, pp. 39-79, 1927.
- [3] R. B. Cattell, "Parallel proportional profiles and other principles for determining the choice of factors by rotation," *Psychometrika*, Vol. 9, pp. 267-283, 1944.
- [4] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, Vol. 31, pp. 279-311, 1966.
- [5] P. Comon, "Tensor decompositions: State of the art and applications," in *Mathematics in Signal Processing*, V, J. G. McWhirter and I. K. Proudler, eds., Oxford University Press, pp. 1-24, 2001.
- [6] E. Acar, S. A. C. Amtepe, M. S. Krishnamoorthy, and B. Yener, "Modeling and multi-way analysis of chatroom tensors," in *ISI (2005): Proceedings of the IEEE International Conference on Intelligence and Security Informatics, Lecture Notes in Comput. Sci.* 3495, Springer, pp. 256-268, 2005.
- [7] E. Acar, C. A. Bingol, H. Bingol, R. Bro, and B. Yener, "Multi-way analysis of epilepsy tensors," *Bioinformatics*, Vol. 23, pp. 110-118, 2007.
- [8] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, Vol. 16, pp. 1-84, 1970.
- [9] A. L. F. de Almeida, G. Favier, J. C. M. Mota, and R. L. de Lacerda, "Estimation of frequency-selective block-fading MIMO channels using PARAFAC modeling and alternating least squares," in *The 40th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 1630-1634, Oct.-Nov. 2006.
- [10] N. Sidiropoulos, R. Bro, and G. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2377-2388, Aug. 2000.
- [11] A. de Almeida, G. Favier, and J. Mota, "A constrained factor decomposition with application to MIMO antenna systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2429-2442, Jun. 2008.
- [12] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253-1278, 2000.
- [13] M. Haardt, F. Roemer, and G. Del Galdo, "Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3198-3213, Jul. 2008.
- [14] T. G. Kolda, "Multilinear operators for higher-order decompositions," Sandia National Laboratories, Albuquerque, NM and Livermore, CA, Technical Report SAND2006-2081, 2006.
- [15] L. De Lathauwer, B. De Moor, and J. Vandewalle, "On the best rank-1 and rank-(r_1 ; r_2 ; ...; r_n) approximation of higher-order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324-1342, 2000.
- [16] T.G. Kolda, Brett W. Bader, "Tensor Decomposition and application," *Society of industrial and applied mathematics, SIAM review*, Vol. 51, No. 3, pp. 455-500, 2009.

[17] R. Bro, R. A. Harshman, and N. D. Sidiropoulos, "Modeling multiway data with linearly dependent loadings," Dept. of Dairy and Food Science, The Royal Veterinary and Agricultural University, Copenhagen, Denmark, Technical Report 2005-176, 2005.

[18] A. L. F. de Almeida, G. Favier, J. C. M. Mota, and R. L. de Lacerda, "Estimation of frequency-selective block-fading MIMO channels using PARAFAC modeling and alternating least squares," in The 40th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, pp. 1630–1634, Oct.–Nov. 2006.

[19] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms — Part II: Definitions and uniqueness," SIAM J. Matrix Anal. Appl., Vol. 30, no. 3, pp. 1033–1066, 2008.

[20] A. Smilde, R. Bro, and P. Geladi, "Multiway Analysis with Applications in the Chemical Sciences," Chichester, England: John Wiley and Sons, Ltd, 381 p, 2004.

[21] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an 'explanatory' multi-modal factor analysis," UCLA working papers in phonetics, vol. 16, 1970.

[22] G. Tomasi and R. Bro, "A comparison of algorithms for fitting the PARAFAC model," Computational Statistics & Data Analysis, Vol. 50, no. 7, pp.1700–1734, Apr.2006.

[23] P. K. Hopke, P. Paatero, H. Jia, R. T. Ross, and R. A. Harshman, "Three-way (PARAFAC) factor analysis: examination and comparison of alternative computational methods as applied to ill-conditioned data," Chemometrics and Intelligent Laboratory Systems, vol. 43, pp. 25–42, Sept. 1998.

[24] A. Richter, J. Salmi, and V. Koivunen, "ML estimation of covariance matrix for tensor valued signals in noise," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2008), Las Vegas, USA, pp. 2349–2352, Mar. 31–Apr. 4 2008.

[25] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," SIAM J. Matrix Anal. Appl., vol. 21, no. 4, pp. 1253-1278, 2000.

[26] V. Degli-Esposti, D. Guiducci, A. deMarsi, P. Azzi, and F. Fuschini, "An advanced eld prediction model including diffuse scattering," IEEE Transactions on Antennas and Propagation, Vol. 52, no. 7, pp. 1717-1728, Jul. 2004.

[27] G. Del Galdo, "Geometry-based channel modeling for multi-user MIMO systems and applications," Ph.D. dissertation, Ilmenau University of Technology, Ilmenau, Germany, ISBN: 978-3-938843-27-7, 2007.

[28] Jussi Salmi, "Contributions to measurement-based dynamic MIMO channel modeling and propagation parameter estimation," Helsinki University of Technology, Faculty of Electronics, Communications and Automation, Department of Signal Processing and Acoustics, ISBN 978-952-248-019-4 (PDF), ISSN 1797-4267, 2009.



Vineet Bhatt is a Research Scholar at H.N.B. Garhwal University, Campus Badshahi Thaul, Tehri Garhwal, Uttarakhand, India.



Sandeep Kumaris working as a visiting assistant professor of Maths at the department of mathematics Govt. P.G. College New Tehri, T.G., Uttarakhand, India.

Nomenclatures

List of Symbols		List of Abbreviations	
A, B, ...	Matrix	FIM	Fisher Information Matrix
$\mathcal{A}, \mathcal{C}, \dots$	Higher Order Tensor	MIMO	Multiple-Input Multiple-Output
\circ	Outer Product	ALS	Alternative Least Squares
\diamond	Khatri-Rao Product (Column-wise product)	SVD	Singular Value Decomposition
\otimes	Kronecker Product	EVD	Eigen Value Decomposition
\odot	Schur-Hadamard (element wise) Product	HOSVD	Higher Order SVD
\times_n	n-mode multiplication	CANDECOMP	Canonical Decomposition
\mathbb{C}	Set of Complex Numbers	PARAFAC	Parallel Factorization
a, b, ...	Vectors	PARATREE	Tree Structured Tensor Model
a, b, ...	Scalars	Rx	Radio Receiver
\mathbf{A}^*	Complex Conjugate of \mathbf{A}	Tx	Radio Transmitter
$\Re\{\mathbf{A}\}$	Real part of \mathbf{A}	PADP	Power-Angular-Delay-Profile
Diag(\mathbf{A})	Vector of the diagonal elements of \mathbf{A}	DCM	Dense Multipath Component
Vec(\mathbf{A})	Stacks all the elements of \mathbf{A} into a vector	M	Total number of samples in an observation
\mathbf{A}^n	Matrix of n^{th} mode basis vectors	\mathbf{R}	Number of factors in a tensor decomposition
H	MIMO channel matrix	t	Time
\mathcal{H}	MIMO channel tensor	Λ	Diagonal matrix of eigenvalues
P	Number of propagation paths.	σ	Singular value
τ	Time delay of arrival	γ	Vector of all complex path weights
$\Phi_{T/R}$	Azimuth angle of departure/arrival at Tx/Rx	\mathbf{x}	Vector of transmitted signal
$\vartheta_{T/R}$	Elevation angle of departure/arrival at Tx/Rx	\mathbf{y}	Vector of received/measured signal
$\mathbf{b}(\boldsymbol{\varphi}, \boldsymbol{\vartheta})$	Steering vector	\mathcal{T}	Auxiliary tensor in SUSVD