IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 6, 2019, accepted March 25, 2019, date of publication March 29, 2019, date of current version April 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2908207

Tensor-Based Channel Estimation for Massive MIMO-OFDM Systems

DANIEL COSTA ARAÚJO[1], ANDRÉ L. F. DE ALMEIDA[1], (Senior Member, IEEE), JOÃO P. C. L. DA COSTA[2], (Senior Member, IEEE), AND RAFAEL T. DE SOUSA, Jr.,[2]

[1]Wireless Telecom Research Group (GTEL), Department of Teleinformatics Engineering (DETI), Federal University of Ceará (UFC), Fortaleza 60020-181, Brazil
[2]Department of Electrical Engineering, University of Brasília, Brasília 70910-900, Brazil

Corresponding author: André L. F. de Almeida (andre@gtel.ufc.br)

This work was partially supported by the Brazilian National Councils CNPq (Grant: 23038.007604/2014-69) and CAPES (Grants: 88887.144009/2017-00; 23038.007604/2014-69), and in part by the FAP-DF (Grants: 0193.001366/2016; 0193.001365/2016), and by LATITUDE/UnB (Grant: 23106.099441/2016-43).

ABSTRACT Channel estimation is a crucial problem for massive multiple input multiple output (MIMO) systems to achieve the expected benefits in terms of spectrum and energy efficiencies. However, a considerable number of pilots are usually distributed over a large number of time-frequency resources using orthogonal frequency division multiplexing (OFDM) to effectively estimate a large number of channel coefficients in space and frequency domains, sacrificing spectral efficiency. In this paper, by assuming MIMO-OFDM transmission, we start by proposing a tensor-based minimum mean square error (MMSE) channel estimator that exploits the multidimensional nature of the frequency-selective massive MIMO channel in the frequency-domain, being a low-complexity alternative to the well-known vector-MMSE channel estimation. Then, by incorporating a 3D sparse representation into the tensor-based channel model, a tensor compressive sensing (tensor-CS) model is formulated by assuming that the channel is compressively sampled in space (radio-frequency chains), time (symbol periods), and frequency (pilot subcarriers). This tensor-CS model is used as the basis for the formulation of a tensor-orthogonal matching-pursuit (T-OMP) estimator that solves a greedy problem per dimension of the measured tensor data. The proposed channel estimator has two variants which may either resort to a joint search per tensor dimension or to a sequential search that progressively reduces the search space across the tensor dimensions. The complexities of the different tensor-based algorithms are studied and compared to those of the traditional vector-MMSE and vector-CS estimators. Our results also corroborate the performance-complexity tradeoffs between T-MMSE and T-OMP estimators, both being competing alternatives to their vector-based MMSE and OMP counterparts.

INDEX TERMS Channel estimation, massive MIMO, compressive sensing, tensor analysis, Tucker3 decomposition.

I. INTRODUCTION

Massive multiple-input multiple-output (MIMO) is a key technology for future 5G wireless communication systems. Employing hundreds of antennas at the base station (BS) can potentially provide spectral efficiency gains compared to multi-antenna solutions used in current Long Term Evolution (LTE) systems [1]. The promised gains can only be achieved if the BS has an accurate knowledge of the channel state information (CSI). Otherwise, high beamforming gains cannot be enjoyed by the user equipment (UE) [1], [2]. A major bottleneck arising in the problem of channel estimation for massive MIMO is the training overhead. The higher is the number of antennas, the higher is the number (and length) of pilot sequences [3]. This becomes a huge problem in systems whose duplexing mode employs downlink pilots, since the minimum length of the pilot sequences necessary to provide a least squares channel estimate is determined by the number of transmit antennas [4], [5]. In addition, since most of the current wireless communication standards are based on frequency division duplexing (FDD) and rely on downlink channel estimation, the training overhead problem places a challenge for their evolution towards the massive MIMO scenario. Gao *et al.* [5] propose a closed-loop

The associate editor coordinating the review of this manuscript and approving it for publication was Nizar Zorba.

channel estimation scheme that adapts the training sequence at every channel block, by optimizing the pilots over time. The training overhead is reduced by resorting to compressed channel representations. In [5], channel compression is achieved thanks to the use of the eigenvectors of the spatial channel covariance matrix. Indeed, sparse channel models for wireless communications have been studied in a number of works and different channel estimation methods based on compressive sensing (CS) have been presented [6], [7], [9].

Adopting structured multipath channel models is particularly important in poor scattering propagation scenarios. Examples of structured MIMO channels available in the literature are the Kronecker model [10], the Weichselberger's model [11] or the virtual channel representation [12]. In the context of millimeter wave MIMO systems, exploiting structured (low-rank) channel models is possible via CS-based estimators, as shown in [9], [13], and [14]. Bajwa *et al.* [9] link sparse reconstruction tools to the channel estimation problem and discuss the design of pilot assisted schemes. The work in [14] capitalizes on structured channel models to estimate the angles of arrival and departure, which allows reconstructing the instantaneous CSI with good accuracy. In [15], a sparse channel estimation technique is proposed to recover the 2D angular information of the dominant channel paths by exploiting the knowledge of the visibility region of a planar array as well as the adaptation of the measurement matrix (beam directions) during the channel acquisition process. CS-based channel estimation has also been addressed in a number of recent works in the context of mmWave MIMO systems using hybrid analog-digital architectures [16]–[18]. In [16], a hierarchical multi-resolution codebook is used to design the training beams to estimate the channel angular parameters and adaptive CS-based algorithm is proposed.

Since modern wireless communication systems may operate on frequency-selective channels, solutions that takes into account channel frequency-selectivity have been proposed recently in [17]–[19], where the authors derive CS-based algorithms that cope with wideband mmWave systems. The solutions derived in [17] and [18] do not explicitly exploit the multidimensional structure of the sparse massive MIMO channel, by operating instead on the "vectorized" measurements, such that vector-based sparse reconstruction algorithms can be applied. Nasser and Elsabrouty [19] propose a channel estimation algorithm that separately retrieves the delay and angle parameters of the channel. However, the solution proposed therein only exploits the joint sparsity at the transmitter side, due to the assumption of single-antenna users. In this work, we are interested in a more general scenario, where i) both ends of the wireless link (base station and user terminal) are equipped with antenna arrays. This is likely to happen in millimeter wave systems, where a large number of antenna elements should be used to compensate for small antenna apertures [20]; ii) the channel is frequency-selective, which is the likely the case when operating in wideband channels. Under these two main assumptions, the channel model naturally admits a multidimensional algebraic structure which is also sparse in the joint space-delay domain. Our interest is to preserve the mutidimensional structure of the channel a well as to exploit its sparsity by resorting to a tensor-based CS approach.

A. RELATED WORK

A useful tool to deal with multidimensional data is tensor analysis in multilinear algebra [21]–[23]. In the context of MIMO channel modeling and estimation, tensor analysis has been used in a number of works as a mathematical formalism to describe the algebraic structure of the channel [11], [24]–[26]. In [24], the Kronecker model was extended to the wideband case using tensor modeling. Following that work, in [25], a tensor-based wideband channel model built up from the Weichselberger's model [11] was proposed and validated experimentally. Zhang *et al.* [26] introduced a CSI tensor expression and a full correlation model for wideband MIMO channels by means of a general coupling-based framework built from the relationship between spatial and frequency channel dimensions. Beyond the importance of channel modeling on its own, tensor models such as the Parallel Factors (PARAFAC) model [27]–[29] and the Tucker model [30] have interesting structural properties that can be exploited to obtain sparse channel representations. Among these models, the Tucker one is of particular interest in this work because its structure provides us a complete multilinear sparse channel representation that can be fully exploited by a tensor-compressive sensing (CS) channel estimator, as detailed in Section VI.

The joint use of tensor modeling and CS tools for channel estimation in massive MIMO systems has been addressed in few recent works [31]–[33]. In [31], the problem of uplink channel estimation for multiuser mmWave MIMO systems is considered. Assuming a layered pilot transmission scheme, a PARAFAC-based estimator is used to separate effective channels of the different users, and a CS-based algorithm is applied to estimate each user angular parameters. A generalization of this method to frequency-selective channels was recently given in [32]. The work [33] establishes a link between the frequency-selective mmWave MIMO channel estimation problem to the theory of multi-way compressive sensing of sparse tensors via PARAFAC analysis [34]. By leveraging on this link, a two-stage algorithm for the joint estimation of the compressed channel bases (spatial transmit, spatial receive, and delay) is proposed. Although working under different assumptions, all these methods have a common reasoning; they solve the channel estimation problem in two stages. The first stage consists of a PARAFAC-based estimator via an alternating least squares (ALS) algorithm that separate multiuser signals [31], [32] or multipath components [33], while the second stage consists of sparse recovery via CS to estimate the channel parameters, such as angles and delays. Note that the approach of [31], [32] deals with the uplink multiuser scenario, where tensor-based estimation has the goal of separating multiple user transmissions, while sparsity is solved using conventional (vector-based) CS algorithms. In the present work, in addition to assuming a different system setup, we take a different route to solve the problem. By assuming actual pilot placement schemes of MIMO-OFDM, we formulate a tensor-based MMSE channel estimator that exploits the multidimensional nature of the massive MIMO-OFDM channel. Despite its conceptual simplicity, this is the first time a tensor-MMSE approach is proposed for MIMO-OFDM channel estimation, to the best of the authors knowledge. As shown later in Section VIII, the tensor-MMSE channel estimator performs close to the conventional (vector-)MMSE one, while being much less computationally complex. Regarding the exploitation of channel sparsity, in contrast to previous works that rely on vector-CS recovery, to the best of our knowledge, the present work is the first resorting to a tensor-CS approach to fully exploit the multidimensional sparse structure of the channel via a joint iterative tensor-based estimation and CS recovery,

B. CONTRIBUTIONS

The contributions of this paper are summarized as follows.

- We develop a tensor-based MMSE channel estimator that exploits the multilinear structure of the second order statistics in space, time, and frequency dimensions. The method consists of filtering each dimension of the received signal tensor by using the modal minimum mean square error (MMSE) filters that operate multilinearly on each dimension of the channel tensor. The design of such filters is based on the knowledge of the individual space, time, and frequency-domain correlation matrices, being a low-complexity alternative to the well-known vector-MMSE channel estimator, which solves a single (large-scale) 1D optimization problem based on the full correlation matrix.
- We propose a tensor-CS channel estimation approach that resorts to a sparse Tucker3 tensor model to represent the frequency-selective massive MIMO channel in the frequency-domain. More specifically, we formulate a compressible channel representation in space (radio-frequency chains), time (symbol periods) and frequency (pilot subcarriers). This tensor-CS model is used as the basis for the formulation of a T-OMP estimator that solves a greedy problem per dimension of the measured tensor data. The proposed channel estimator has two variants which may either resort to a joint search per tensor dimension or to a sequential search that progressively reduces the search space across the tensor dimensions.
- We discuss the computational complexities of T-MMSE and T-OMP algorithms and compare them with their vector-MMSE and vector-CS counterparts, showing that the proposed estimators are competing solutions for channel estimation in massive MIMO-OFDM systems, due to their good performance/complexity tradeoffs.

The remainder of this paper is organized as follows. We first present some fundamental concepts of tensor algebra in Section II. In Section III, we use a tensor framework to model the frequency-selective massive MIMO channel. Essentially, this model highlights that the channel and its sparse representation enjoy a multilinear structure. In Section IV, the basis system model and assumptions are presented. In Section V, we formulate the tensor-MMSE channel estimation approach, while in Section VI the Tensor-CS approach is developed and the corresponding algorithms are detailed. The simulation results are presented in VIII, and the conclusions are drawn in Section IX.

Notations: Scalars are denoted by lower-case letters ($a, b, \ldots$), vectors by bold lower-case letters ($\boldsymbol{a}, \boldsymbol{b}, \ldots$), matrices by bold upper-case letters ($\boldsymbol{A}, \boldsymbol{B}, \ldots$), tensors are defined by calligraphic upper-case letters ($\boldsymbol{\mathcal{A}}, \mathcal{B}, \ldots$). $\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{A}^{\dagger}, \boldsymbol{A}^{*}, \boldsymbol{A}^{\mathrm{H}}$ stand for transpose, Moore-Penrose pseudo-inverse, conjugate and Hermitian of $\boldsymbol{A}$, respectively. The operators $\otimes$, $\diamond$ and $\circ$ define the Kronecker, Khatri-Rao and the outer product, respectively. For a matrix $\mathbf{A} \in \mathbb{C}^{I \times R}$, the vec$(\cdot)$ operator vectorizes a matrix by stacking its columns, i.e., $\text{vec}(\mathbf{A}) = \mathbf{a} \in \mathbb{C}^{IR \times 1}$.

II. TENSOR PREREQUISITES

Tensors are multidimensional structures (i.e. arrays with more than two dimensions) that can be viewed as natural extensions of vectors and matrices to higher orders.

Definition 1: A tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is a N-th order array whose typical element $\boldsymbol{\mathcal{A}}(i_1, i_2, \ldots, i_N)$ is accessed via N indices, $i_n \in \{I_1, I_2, \ldots I_N\}$.

To define a multiplication between a tensor and a matrix, it is necessary to specify which mode of the tensor is affected by that operation.

Definition 2: The n-mode product of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{J_1 \times J_2 \ldots \times J_N}$ and a matrix $\boldsymbol{A} \in \mathbb{C}^{I_n \times J_n}$ is symbolized by $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}} \times_n \boldsymbol{A}$, where $\boldsymbol{\mathcal{Y}} \in \mathbb{C}^{J_1 \times J_2 \ldots \times J_{n-1} \times I_n \times J_{n+1} \ldots \times J_N}$, each element being defined as [21]:

$$\boldsymbol{\mathcal{G}}(j_1, \ldots, i_n, \ldots, j_N) \doteq \sum_{j_n=1}^{J_n} \boldsymbol{\mathcal{X}}(j_1, \ldots, j_n, \ldots, j_N)\boldsymbol{A}(i_n, j_n). \tag{1}$$

This product can be applied successively along several modes, e.g., $(\boldsymbol{\mathcal{X}} \times_n \boldsymbol{A}) \times_m \boldsymbol{B} = (\boldsymbol{\mathcal{X}} \times_m \boldsymbol{B}) \times_n \boldsymbol{A}$ $(m \neq n)$. In the case $m = n$, we have $(\boldsymbol{\mathcal{X}} \times_n \boldsymbol{A}) \times_n \boldsymbol{B} = \boldsymbol{\mathcal{X}} \times_n \boldsymbol{BA}$.

Definition 3: The Tucker3 decomposition for a three-order tensor $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{I \times J \times K}$ is defined, in scalar form, as:

$$\boldsymbol{\mathcal{G}}(i, j, k) = \sum_{p=1}^{P}\sum_{q=1}^{Q}\sum_{r=1}^{R} \boldsymbol{\mathcal{X}}(p, q, r)\boldsymbol{A}(i, p)\boldsymbol{B}(j, q)\boldsymbol{C}(k, r), \tag{2}$$

where $\boldsymbol{\mathcal{G}}(i, j, k)$ and $\boldsymbol{\mathcal{X}}(p, q, r)$ are elements of the third-order tensor $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{I \times J \times K}$ and $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{P \times Q \times R}$, respectively, $\boldsymbol{A}(i, p)$, $\boldsymbol{B}(j, q)$, and $\boldsymbol{C}(k, r)$ are typical entries of the factor matrices $\boldsymbol{A} \in \mathbb{C}^{I \times P}$, $\boldsymbol{B} \in \mathbb{C}^{J \times Q}$, and $\boldsymbol{C} \in \mathbb{C}^{K \times R}$, respectively. Using the n-mode product notation, the Tucker3 decomposition can be expressed as

$$\mathcal{G} = \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}. \tag{3}$$

The matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ are associated with 1-mode, 2-mode, and 3-mode products across the respective dimensions of the input tensor $\boldsymbol{\mathcal{X}}$, yielding the output tensor $\mathcal{G}$. Such an operation can also be interpreted as a three-dimensional filtering (i.e. a multilinear compression/decompression) mapping $\boldsymbol{\mathcal{X}}$ into $\mathcal{G}$, and is represented in Fig. 1.

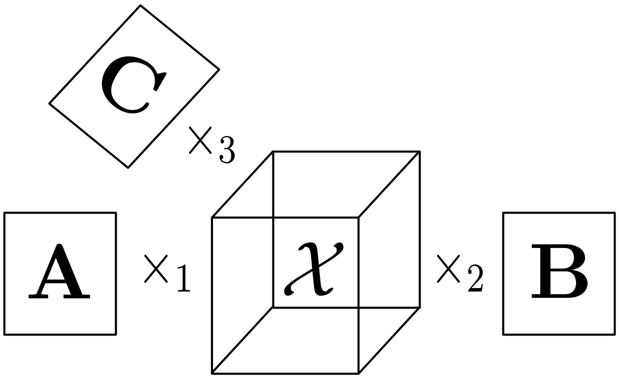

FIGURE 1. ***n*-mode product representation for a third-order tensor.**

Another useful way of representing the tensor transformation (3) consists of rewriting it as three different (but equivalent) unfolded forms, which are given by [21]

$$\boldsymbol{G}_{(1)} = \boldsymbol{A}\boldsymbol{X}_{(1)}\left[\boldsymbol{C} \otimes \boldsymbol{B}\right]^T \in \mathbb{C}^{I\times JK}, \tag{4}$$

$$\boldsymbol{G}_{(2)} = \boldsymbol{B}\boldsymbol{X}_{(2)}\left[\boldsymbol{C} \otimes \boldsymbol{A}\right]^T \in \mathbb{C}^{J\times IK}, \tag{5}$$

$$\boldsymbol{G}_{(3)} = \boldsymbol{C}\boldsymbol{X}_{(3)}\left[\boldsymbol{B} \otimes \boldsymbol{A}\right]^T \in \mathbb{C}^{K\times IJ}. \tag{6}$$

Another way of representing the above equations resorts to an equivalent vectorized form

$$\boldsymbol{g} = (\boldsymbol{C} \otimes \boldsymbol{B} \otimes \boldsymbol{A})\,\boldsymbol{x} \in \mathbb{C}^{IJK}, \tag{7}$$

where $\boldsymbol{x} \doteq \text{vec}\{\boldsymbol{X}_{(1)}\}$ and $\boldsymbol{g} \doteq \text{vec}\{\boldsymbol{G}_{(1)}\}$. Two additional vectorized forms can be obtained in a similar way from the unfoldings $\boldsymbol{G}_{(2)}$ and $\boldsymbol{G}_{(3)}$, respectively.

In the following, the tensor formalism is used to model a frequency-selective MIMO channel by means of a sparse multidimensional representation. More specifically, the factor matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ in (3) play the role of fixed channel bases in space, time and frequency domains, respectively, while the core tensor $\boldsymbol{\mathcal{X}}$ reveals the sparse structure of the channel in the joint angle-delay domain. This structure is exploited in Section VI to formulate the proposed tensor-CS channel estimators.

III. TENSOR-BASED CHANNEL FORMULATION

Consider a single user MIMO system, where the receiver and the transmitter are equipped with I and J antennas, respectively. The channel is modelled as a summation of K paths, where each of them are defined by the 6-tuple $(\phi_{T,k}, \theta_{T,k}, \phi_{R,k}, \theta_{R,k}, \tau_k, \beta_k)$. The azimuth and elevation of the transmit and receive sides are $\phi_{T,k}$, $\theta_{T,k}$, $\phi_{R,k}$ and $\theta_{R,k}$, respectively. The delay associated with the kth path is defined by τ_k, and β_k is a random complex variable that accounts for the phase rotation with zero mean, unity variance, and Normal distribution. We are interested in the case where the system bandwidth W is much larger than the inverse of the maximum delay spread, i.e., $W >> \frac{1}{\tau_{max}}$. We can represent the frequency response of a MIMO channel at the fth frequency bin as the frontal (3-mode) slice of a channel tensor $\mathcal{G} \in \mathbb{C}^{I\times J\times F}$. More specifically, we have

$$\mathcal{G}(:,:,f) = \sum_{k=1}^{K} \beta_k \boldsymbol{v}(\phi_{R,k}, \theta_{R,k})\boldsymbol{v}^H(\phi_{T,k}, \theta_{T,k})\mathrm{e}^{-\jmath 2\pi f \tau_k}, \tag{8}$$

where $v(\phi_{R,k}, \theta_{R,k})$ and $v(\phi_{T,k}, \theta_{T,k})$ are the receive and transmit steering vectors, respectively. The ith element of $v(\phi_{R,k}, \theta_{R,k})$ is given by

$$\boldsymbol{v}_i(\phi_{R,k}, \theta_{R,k}) = \mathrm{e}^{\jmath \frac{2\pi d}{\lambda}(\sin\theta_{R,k}\cos\theta_{R,k}+\cos\theta_{R,k})}, \tag{9}$$

where λ is the wavelength and d is the antenna spacing. The third-order channel tensor collecting the channel response of the F frequency bins can be expressed as

$$\mathcal{G} = \mathcal{H}^{\text{v}} \times_1 \boldsymbol{A}_{\text{R}} \times_2 \boldsymbol{A}_{\text{T}} \times_3 \boldsymbol{A}_{\text{F}}\ \mathbb{C}^{I\times J\times F}, \tag{10}$$

where $\mathcal{H}^{\text{v}} \in \mathbb{C}^{M\times N\times P}$ is the sparse core tensor, $\boldsymbol{A}_{\text{R}} \in \mathbb{C}^{I\times M}$, $\boldsymbol{A}_{\text{T}} \in \mathbb{C}^{J\times N}$, and $\boldsymbol{A}_{\text{F}} \in \mathbb{C}^{F\times P}$ are the (sparsifying) bases that map the channel tensor $\mathcal{G}$ into a sparse virtual one $\mathcal{H}^{\text{v}}$. Note that Eq. (10) follows a Tucker3 decomposition as defined in (3), where the core tensor, representing the virtual channel tensor, has a sparse 3D structure. The variables M, N, and P denote the number of discrete angles of arrival, angles of departure, and delays, respectively. Although the choice of these bases directly affect the sparse structure of $\mathcal{H}^{\text{v}}$, the optimization of $\boldsymbol{A}_{\text{R}}$, $\boldsymbol{A}_{\text{T}}$ and $\boldsymbol{A}_{\text{F}}$, also known as "dictionary learning" in the compressive sensing literature, is beyond the scope of this paper. We instead focus on the problem of acquiring the non-zero entries of the core tensor $\mathcal{H}^{\text{v}}$ by assuming predefined bases, i.e. codebooks, which are fixed at both the transmitter and the receiver. Hence, the (n, m, p)th element of the 3D virtual channel tensor can be expressed as

$$\mathcal{H}^{\text{v}}(n,m,p) = \mathcal{G} \times_1 \boldsymbol{A}_{\text{R}}^H(:,n) \times_2 \boldsymbol{A}_{\text{T}}^H(:,m) \times_3 \boldsymbol{A}_{\text{F}}^H(:,p)\,. \tag{11}$$

According to [8], [9], for uniform linear arrays (ULAs) and discrete Fourier transform (DFT) bases, the expression (11) coincides with the virtual channel representation given by

$$\begin{aligned}\mathcal{H}^{\text{v}}(n,m,p) \approx & \sum_{k\,\in\, S_{R,n}\cap S_{T,m}\cap S_{\tau,p}} \beta_k \\ & \cdot f_N(n/I - \theta_{R,k}) f_M^*(m/J - \theta_{T,k}) \\ & \cdot \text{sinc}(p - W\tau_k),\end{aligned} \tag{12}$$

where β_k is a zero mean complex Gaussian gain and unit variance, $\theta_{R,l}$ and $\theta_{T,l}$ denote the angle of arrival (AoA) and angle of departure (AoD), respectively, $\tau_k \in [0, \tau_{max}]$ is the delay related to the kth path, $f_\alpha(\gamma) = \frac{1}{\alpha}\sum_{i=0}^{\alpha-1} e^{-j2\pi i\gamma}$ is the Dirichlet kernel, and $\text{sinc}(x) = e^{-j\pi x}\sin(\pi x)/(\pi^2 x)$ is the sinc kernel. The sets over which the summation in (12) is carried out are given by [9]:

$$\begin{aligned} S_{R,n} &= \{k \quad \theta_{R,k} \in (n/I - \Delta\theta_R/2, n/I + \Delta\theta_R/2]\} \\ S_{T,m} &= \{k \quad \theta_{T,k} \in (m/J - \Delta\theta_T/2, m/J + \Delta\theta_T/2]\} \\ S_{\tau,p} &= \{k \quad \tau_k \in (p/W - \Delta\tau/2, p/W + \Delta\tau/2]\}. \end{aligned} \tag{13}$$

The sparse Tucker3 model (10) offers a tensorial representation for the massive MIMO channel in the frequency-domain. The factor matrices $\boldsymbol{A}_R$, $\boldsymbol{A}_T$, and $\boldsymbol{A}_\mathrm{F}$ are completely known a *priori* while $\boldsymbol{\mathcal{H}}^\mathrm{v}$ is the unknown sparse tensor to be recovered. Our goal is to exploit the multilinear structure of the massive MIMO channel tensor $\boldsymbol{\mathcal{G}}$, as well as its sparse structure captured by the virtual channel tensor $\boldsymbol{\mathcal{H}}^\mathrm{v}$ to derive our tensor-based channel estimators.

IV. SYSTEM MODEL

Actual wireless communication system have pilot sequences embedded in their frames that serve as reference symbols for channel estimation. Assuming an OFDM scheme, the pilot symbols occupy some subcarriers while the remaining ones are used for data transmission. Following the tensor formalism introduced in Section III, we define $\boldsymbol{\mathcal{G}}_\mathrm{p} \in \in \mathbb{C}^{I \times J \times F_\mathrm{p}}$, and $\boldsymbol{\mathcal{G}}_\mathrm{d} \in \mathbb{C}^{I \times J \times F_\mathrm{d}}$ as the pilot and data channel tensors, respectively, where F_p denotes the number of pilot subcarriers and F_d is the number of data subcarriers, with $F_\mathrm{p} + F_\mathrm{d} = F$. Let us define $\boldsymbol{F} \in \mathbb{C}^{F_p \times F}$ as a subcarrier selection matrix, where $\mathbb{F}^{F_p \times F}$ is the set of all possible selection matrices of dimensions $F_p \times F$ obtained by selecting a subset of F_p rows of the identity matrix $\boldsymbol{I}_F$. The matrix $\boldsymbol{F}$ defines the pilot placement scheme, i.e., the set of subcarriers used for pilot transmission.[1] We can define a relationship between the pilot channel, i.e., the one containing the pilot subcarriers, and the full channel as

$$\boldsymbol{\mathcal{G}}_\mathrm{p} = \boldsymbol{\mathcal{G}} \times_3 \boldsymbol{F}. \tag{14}$$

Likewise, by defining a selection matrix for the data subcarriers, we can obtain the data channel $\boldsymbol{\mathcal{G}}_\mathrm{d}$.

In this work, we assume a hybrid beamforming (HB) architecture at both transmitter and the receiver [16], [36], where L_t and L_r denote the number of radio frequency (RF) chains used at the transmitter and receiver, respectively. Each RF chain transmits a different pilot sequence (i.e. L_t pilot sequences transmitted at the same time-frequency resource). Moreover, each pilot sequence is repeated across the F_p pilot subcarriers. The pilot symbol vector at the tth OFDM symbol is given by

$$\boldsymbol{X}_0(t,:) = [x_{t,0} \ x_{t,2} \ \ldots \ x_{t,L_\mathrm{t}-1}] \in \mathbb{C}^{1 \times L_\mathrm{t}}, \tag{15}$$

where $t = 1, \ldots, T_\mathrm{p}$. Therefore, the collection of T_p OFDM symbols defines $\boldsymbol{X}_0 \in \mathbb{C}^{T_\mathrm{p} \times L_\mathrm{t}}$. In this work, we assume $\boldsymbol{X}_0$ is orthonormal, which implies $T_\mathrm{p} = L_\mathrm{t}$. Moreover, each pilot sequence is associated with a specific beam, defined by the corresponding column of the beamforming matrix $\boldsymbol{W} \in \mathbb{C}^{L_\mathrm{t} \times J}$. The effective transmitted signal is therefore given by

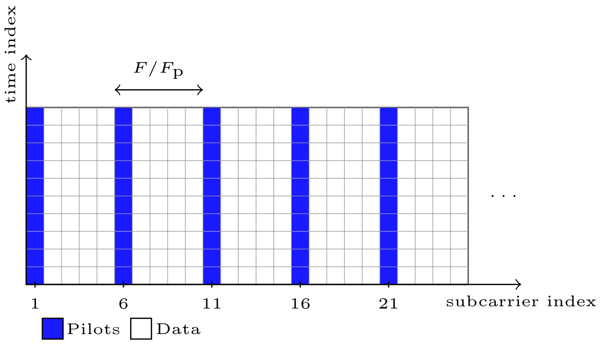

FIGURE 2. **An example of a pilot placement in the time-frequency grid. The pilot sequences are regularly spaced and their separation is larger than the coherence bandwidth.**

$$\boldsymbol{X} = \boldsymbol{X}_0 \boldsymbol{W} \in \mathbb{C}^{T_\mathrm{p} \times J}, \tag{16}$$

with $\|\boldsymbol{X}(t,:)\|^2 = P_\mathrm{T}/F_\mathrm{p}$, and P_T being the total transmit signal power. Since HB architecture is assumed, the beamforming matrix $\boldsymbol{W}$ represents a network of phase shifters. Hence, its entries are modeled as complex exponentials. Note that the columns of this matrix are beamforming vectors used to probe the channel in different directions. In this work, random phases are assumed for $\boldsymbol{W}$, which physically means that random beams are drawn to probe the channel in the range $[0, 2\pi)$ (i.e., there is no direction of preference).

The input-output relation in the frequency domain is given as

$$\boldsymbol{\mathcal{Y}}_0 = \boldsymbol{\mathcal{G}}_\mathrm{p} \times_2 \boldsymbol{X} + \boldsymbol{\mathcal{Z}} \in \mathbb{C}^{I \times T_\mathrm{p} \times F_\mathrm{p}}, \tag{17}$$

where $\boldsymbol{\mathcal{Z}}$ denotes the noise contribution whose entries are zero mean circularly symmetric (ZMCS) complex Gaussian random variables with variance $N_0/2$. Substituting (14) into (17) we obtain

$$\boldsymbol{\mathcal{Y}}_0 = \boldsymbol{\mathcal{G}} \times_2 \boldsymbol{X} \times_3 \boldsymbol{F} + \boldsymbol{\mathcal{Z}}, \tag{18}$$

which now expresses the received signal tensor as a function of the full channel tensor.

First, note that in the time-domain we have $T_\mathrm{p} << J$, since our interest is to work with short pilot sequences to reduce the training overhead. Second, in the frequency-domain we have $F_\mathrm{p} << F$ as a consequence of the pilot placement scheme. This implies that $\boldsymbol{X} \in \mathbb{C}^{T_\mathrm{p} \times J}$ and $\boldsymbol{F} \in \mathbb{C}^{F_\mathrm{p} \times F}$ are full row-rank matrices, i.e., the received signal tensor $\boldsymbol{\mathcal{Y}}_0$ in (18) is a compressed version of the channel tensor $\boldsymbol{\mathcal{G}}$, where compression is applied in the second and third modes, i.e. over the time and frequency domains, respectively. Recall that a HB architecture is assumed, and define $\boldsymbol{Q} \in \mathbb{C}^{L_\mathrm{r} \times I}$ as the receive combining/beamforming matrix, which loads the signals from the I receive antennas into the L_r RF chains,

[1] In wireless communication standards, such as LTE, pilots are regularly spaced in the frequency domain. Even though other configurations might be possible without changing the framework, we adopt the regular spacing scheme.

with $L_\mathrm{r} \ll I$. After receiver combining, we further compress the measurements (18) in the space domain, yielding

$$\mathcal{Y} = \mathcal{Y}_0 \times_1 \boldsymbol{Q}, \tag{19}$$

which, in combination with (18), gives

$$\mathcal{Y} = \mathcal{G} \times_1 \boldsymbol{Q} \times_2 \boldsymbol{X} \times_3 \boldsymbol{F} + \tilde{\mathcal{Z}} \;\in\; \mathbb{C}^{L_\mathrm{r}\times T_\mathrm{p}\times F_\mathrm{p}}, \tag{20}$$

where $\tilde{\mathcal{Z}} = \mathcal{Z} \times_1 \boldsymbol{Q}$ denotes the filtered noise term after the receive combining.

In the next section, we exploit the tensor model (20) to design tensor-MMSE filters that exploit the multidimensional angle-delay representation for the massive MIMO-OFDM channel.

V. TENSOR-MMSE CHANNEL ESTIMATION

For the sake of completeness, let us first consider the traditional vector-MMSE design, which is commonly applied to channel estimation problems formulated in vector and/or matrix forms [37]. Starting from (20) and "vectorizing" $\mathcal{Y}$, we obtain:

$$\boldsymbol{y} = (\boldsymbol{F} \otimes \boldsymbol{X} \otimes \boldsymbol{Q})\,\boldsymbol{g} + \tilde{\boldsymbol{z}} \;\in\; \mathbb{C}^{L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}}. \tag{21}$$

The problem to be solved is given by

$$\min_{\boldsymbol{U}} \mathrm{E}\left\{\|\boldsymbol{g} - \boldsymbol{U}\boldsymbol{y}\|_2^2\right\}, \tag{22}$$

where $\boldsymbol{U} \in \mathbb{C}^{IJF\times L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}}$ is the MMSE matrix filter, whose closed-form solution is known to be [37]

$$\boldsymbol{U} = \boldsymbol{R}\boldsymbol{\Phi}^H\left(\boldsymbol{\Phi}\boldsymbol{R}\boldsymbol{\Phi}^H + \boldsymbol{R}_{\boldsymbol{z}}\right)^{-1}, \tag{23}$$

where $\boldsymbol{\Phi} = (\boldsymbol{F} \otimes \boldsymbol{X} \otimes \boldsymbol{Q}) \in \mathbb{C}^{L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}\times IJF}$, $\boldsymbol{R} = \mathrm{E}\left\{\boldsymbol{g}\boldsymbol{g}^H\right\} \in \mathbb{C}^{IJF\times IJF}$, and $\boldsymbol{R}_{\boldsymbol{z}} = \mathrm{E}\left\{\tilde{\boldsymbol{z}}\tilde{\boldsymbol{z}}^H\right\} \in \mathbb{C}^{L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}\times IJF}$. In the massive MIMO case, finding the MMSE solution for $\boldsymbol{U}$ involves computing products and inversion of large matrices, which may result in a significant computational burden at the receiver.

To overcome this limitation, we propose an alternative solution that exploits the multidimensional structure of the received pilot signal, i.e., by operating on the unfolded forms of the received tensor $\mathcal{Y}$ given in (20). By analogy with (3), we conclude that $\mathcal{Y}$ follows a Tucker3 model, and the following correspondences can be deduced

$$[\mathcal{X}, \mathbf{A}, \mathbf{B}, \mathbf{C},\,] \longleftrightarrow [\mathcal{G}, \boldsymbol{Q}, \boldsymbol{X}, \boldsymbol{F}]. \tag{24}$$

From (4)-(6), we can write the three unfolded matrix forms that represent the measured tensor data $\mathcal{Y}$ as follows:

$$\boldsymbol{Y}_{(1)} = \boldsymbol{Q}\boldsymbol{G}_{(1)}\left[\boldsymbol{F} \otimes \boldsymbol{X}\right]^T \;\in\; \mathbb{C}^{L_\mathrm{r}\times T_\mathrm{p}F_\mathrm{p}}, \tag{25}$$

$$\boldsymbol{Y}_{(2)} = \boldsymbol{X}\boldsymbol{G}_{(2)}\left[\boldsymbol{F} \otimes \boldsymbol{Q}\right]^T \;\in\; \mathbb{C}^{T_\mathrm{p}\times L_\mathrm{r}F_\mathrm{p}}, \tag{26}$$

$$\boldsymbol{Y}_{(3)} = \boldsymbol{F}\boldsymbol{G}_{(3)}\left[\boldsymbol{X} \otimes \boldsymbol{Q}\right]^T \;\in\; \mathbb{C}^{F_\mathrm{p}\times L_\mathrm{r}T_\mathrm{p}}, \tag{27}$$

where $\boldsymbol{G}_{(1)} \in \mathbb{C}^{I\times JF}$, $\boldsymbol{G}_{(2)} \in \mathbb{C}^{J\times IF}$, and $\boldsymbol{G}_{(3)} \in \mathbb{C}^{F\times IJ}$ are the 1-mode, 2-mode, and 3-mode unfoldings of the channel tensor $\mathcal{G} \in \mathbb{C}^{I\times J\times F}$.

The new problem to be minimized is formulated as follows

$$\min_{\boldsymbol{U}_\mathrm{R},\boldsymbol{U}_\mathrm{T},\boldsymbol{U}_\mathrm{F}} \mathrm{E}\left\{\|\mathcal{G} - \mathcal{Y} \times_1 \boldsymbol{U}_\mathrm{R} \times_2 \boldsymbol{U}_\mathrm{T} \times_3 \boldsymbol{U}_\mathrm{F}\|_F^2\right\}, \tag{28}$$

where $\boldsymbol{U}_\mathrm{R} \in \mathbb{C}^{I\times L_\mathrm{r}}$, $\boldsymbol{U}_\mathrm{T} \in \mathbb{C}^{J\times T_\mathrm{p}}$, and $\boldsymbol{U}_\mathrm{F} \in \mathbb{C}^{F\times F_\mathrm{p}}$ are the 1-mode, 2-mode, and 3-mode MMSE matrix filters that operate over space, time, and frequency dimensions of the received tensor $\mathcal{Y}$, respectively. More specifically, instead of applying a single (and very large) space-time-frequency MMSE matrix filter over the vector signal $\boldsymbol{y} \in \mathbb{C}^{L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}}$ as a linear filter, the proposed tensorial minimum mean square error (T-MMSE) filtering operates multilinearly over the tensor signal $\mathcal{Y} \in \mathbb{C}^{L_\mathrm{r}T_\mathrm{p}F_\mathrm{p}\times IJF}$ by means of three "per-dimension" matrix filters that operate over space, time and frequency independently. Note that the T-MMSE filtering formulated in (28) follows a Tucker3 decomposition defined in (3), and the following correspondences can be deduced

$$[\mathcal{X}, \mathbf{A}, \mathbf{B}, \mathbf{C},\,] \longleftrightarrow [\mathcal{Y}, \boldsymbol{U}_\mathrm{R}, \boldsymbol{U}_\mathrm{T}, \boldsymbol{U}_\mathrm{F}]. \tag{29}$$

Hence, by analogy with (4), (5) and (6), the T-MMSE filtering problem (28) can be recast as three linear MMSE subproblems, as follows

$$\min_{\boldsymbol{U}_\mathrm{R}} \mathrm{E}\left\{\|\boldsymbol{G}_{(1)} - \boldsymbol{U}_\mathrm{R}\boldsymbol{Y}_{(1)}\left[\boldsymbol{U}_\mathrm{F} \otimes \boldsymbol{U}_\mathrm{T}\right]^T\|_F^2\right\}, \tag{30}$$

$$\min_{\boldsymbol{U}_\mathrm{T}} \mathrm{E}\left\{\|\boldsymbol{G}_{(2)} - \boldsymbol{U}_\mathrm{T}\boldsymbol{Y}_{(2)}\left[\boldsymbol{U}_\mathrm{F} \otimes \boldsymbol{U}_\mathrm{R}\right]^T\|_F^2\right\}, \tag{31}$$

$$\min_{\boldsymbol{U}_\mathrm{F}} \mathrm{E}\left\{\|\boldsymbol{G}_{(3)} - \boldsymbol{U}_\mathrm{F}\boldsymbol{Y}_{(3)}\left[\boldsymbol{U}_\mathrm{T} \otimes \boldsymbol{U}_\mathrm{R}\right]^T\|_F^2\right\}. \tag{32}$$

The solution of each sub-problem yields the n-mode MMSE filters as

$$\boldsymbol{U}_\mathrm{R} = \boldsymbol{R}_\mathrm{R}\boldsymbol{Q}^H\left(\boldsymbol{Q}\boldsymbol{R}_\mathrm{R}\boldsymbol{Q}^H + \sigma^2\boldsymbol{I}_{L_\mathrm{r}}\right)^{-1}, \tag{33}$$

$$\boldsymbol{U}_\mathrm{T} = \boldsymbol{R}_\mathrm{T}\boldsymbol{X}^H\left(\boldsymbol{X}\boldsymbol{R}_\mathrm{T}\boldsymbol{X}^H + \sigma^2\boldsymbol{I}_{T_\mathrm{p}}\right)^{-1}, \tag{34}$$

$$\boldsymbol{U}_\mathrm{F} = \boldsymbol{R}_\mathrm{F}\boldsymbol{F}^H\left(\boldsymbol{F}\boldsymbol{R}_\mathrm{F}\boldsymbol{F}^H + \sigma^2\boldsymbol{I}_{F_\mathrm{p}}\right)^{-1}, \tag{35}$$

where $\boldsymbol{R}_\mathrm{R} = \mathrm{E}\left\{\boldsymbol{G}_{(1)}\boldsymbol{G}_{(1)}^H\right\} \in \mathbb{C}^{I\times I}$, $\boldsymbol{R}_\mathrm{T} = \mathrm{E}\left\{\boldsymbol{G}_{(2)}\boldsymbol{G}_{(2)}^H\right\} \in \mathbb{C}^{J\times J}$, and $\boldsymbol{R}_\mathrm{F} = \mathrm{E}\left\{\boldsymbol{G}_{(3)}\boldsymbol{G}_{(3)}^H\right\} \in \mathbb{C}^{F\times F}$ are the space, time, and frequency channel covariance matrices, respectively. Therefore, the T-MMSE filter exploits the inherent separability of the space-time-frequency covariance matrix by designing filters that operate over each channel dimension. This is possible thanks to the multilinear structure of the channel, which follows a Tucker3 model as shown in (10).

Note that a trade-off between complexity and accuracy exists. While the vector-MMSE filter is designed from the full (large) covariance matrix, the T-MMSE solution makes use of three (much smaller) marginal covariance matrices, which may result in a considerable complexity reduction, as discussed in Section VII. Nevertheless, according to our simulation results, the T-MMSE approach provides accurate channel estimates with a small performance loss compared to the vector-MMSE one. Such a trade-off can be interesting in massive MIMO-OFDM systems operating with large numbers of antennas and subcarriers.

Despite the good trade-off between performance and complexity provided by the proposed T-MMSE estimator, our approach requires the knowledge (or estimation) of second-order channel statistics. Indeed, the complexity reduction achieved by exploiting the multilinearity of the channel tensor, the acquisition of space, time, and frequency covariance matrices may require large data blocks. In scenarios where a reliable estimation of the channel statistics may not be possible, the channel parameters can be estimated by capitalizing on a compressed sensing of the massive MIMO channel, which admits a sparse representation in angular and delay domains, especially in mmWave systems [15]–[18]. In the following, we formulate a compressed (sparse) tensor representation for the massive MIMO-OFDM channel tensor in terms of space (radio-frequency chains), time (symbol periods) and frequency (pilot subcarriers) dimensions. The resulting model is then used as the basis for the formulation of tensor-CS channel estimators that jointly exploit the multidimensional and sparse structures of the compressed channel tensor.

VI. CHANNEL ESTIMATION VIA TENSOR-CS

In this section, we propose tensor-CS channel estimators based on a sparse Tucker3 modeling for the massive MIMO-OFDM channel. The tensor-CS approach avoids the computation of channel second-order statistics by operating directly on the received pilot signals, which are recast as a tensor of observations according to (20). This equation shows that the tensor of observations $\boldsymbol{\mathcal{Y}}$ is a tensor-compressed version of the MIMO-OFDM channel tensor $\boldsymbol{\mathcal{G}}$ obtained from multilinear projections onto the set of sensing matrices $\{\boldsymbol{Q}, \boldsymbol{X}, \boldsymbol{F}\}$. Capitalizing on this tensor structure, the proposed tensor-CS algorithms operate by solving sparse reconstruction problem for each dimension of the compressed channel tensor via orthogonal matching pursuits.

A. SPARSE FORMULATION PRELIMINARIES

Let us start with the following vector-CS data model

$$\boldsymbol{y} = \bar{\boldsymbol{\Phi}}\boldsymbol{g} + \boldsymbol{e} = \bar{\boldsymbol{\Phi}}\bar{\boldsymbol{\Psi}}\boldsymbol{d} + \boldsymbol{e}, \tag{36}$$

where $\boldsymbol{g} \in \mathbb{C}^{V}$ is the signal of interest, $\bar{\boldsymbol{\Psi}} \in \mathbb{C}^{V\times V'}$ is the sparsifying basis, and $\bar{\boldsymbol{\Phi}} \in \mathbb{C}^{\bar{V}\times V}$ is the measurement matrix. Now, assume that $\boldsymbol{g} = \bar{\boldsymbol{\Psi}}\boldsymbol{d}$ has a multilinear structure according to (7), i.e., $\boldsymbol{g} = (\boldsymbol{\Psi}_3 \otimes \boldsymbol{\Psi}_2 \otimes \boldsymbol{\Psi}_1)\,\boldsymbol{d}$, where $\boldsymbol{\Psi}_1 \in \mathbb{C}^{I\times I'}$, $\boldsymbol{\Psi}_2 \in \mathbb{C}^{J\times J'}$, and $\boldsymbol{\Psi}_3 \in \mathbb{C}^{F\times F'}$ are the sparsifying bases. Therefore, we can rewrite $\boldsymbol{g} \in \mathbb{C}^{IJF}$, where $IJF = V$ and $I'J'F' = V'$, in a Tucker3 tensor format by analogy with (3) and (7), which gives

$$\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{D}} \times_1 \boldsymbol{\Psi}_1 \times_2 \boldsymbol{\Psi}_2 \times_3 \boldsymbol{\Psi}_3 \in \mathbb{C}^{I\times J\times F}, \tag{37}$$

where $\boldsymbol{\mathcal{D}} = \mathcal{T}\{\bar{\boldsymbol{d}}\} \in \mathbb{C}^{I'\times J'\times F'}$, and the operator $\mathcal{T}\{\cdot\}$ maps the elements of the vector $\boldsymbol{d}$ into a tensor $\boldsymbol{\mathcal{D}}$, as follows

$$\boldsymbol{d}_{i+(j-1)I+(f-1)IJ} \xrightarrow[\mathcal{T}\{\cdot\}]{} \boldsymbol{\mathcal{D}}(i, j, f), \tag{38}$$

where $i = \{1, \ldots, I'\}$, $j = \{1, \ldots, J'\}$, $f = \{1, \ldots, F'\}$ and $\boldsymbol{\mathcal{D}}$ is sparse tensor.

Now, let us introduce three measurement (sensing) matrices $\boldsymbol{\Phi}_1 \in \mathbb{C}^{\bar{I}\times I}$, $\boldsymbol{\Phi}_2 \in \mathbb{C}^{\bar{J}\times J}$, and $\boldsymbol{\Phi}_3 \in \mathbb{C}^{\bar{F}\times F}$, which sample the tensor $\boldsymbol{\mathcal{G}}$ multilinearly along its 1-mode, 2-mode and 3-mode, respectively. We can recast the vector-CS data model (36) as

$$\begin{aligned}\boldsymbol{\mathcal{Y}} &= \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{\Phi}_1 \times_2 \boldsymbol{\Phi}_2 \times_3 \boldsymbol{\Phi}_3 + \boldsymbol{\mathcal{E}}\\ &= \boldsymbol{\mathcal{D}} \times_1 (\boldsymbol{\Phi}_1\boldsymbol{\Psi}_1) \times_2 (\boldsymbol{\Phi}_2\boldsymbol{\Psi}_2) \times_3 (\boldsymbol{\Phi}_3\boldsymbol{\Psi}_3) + \boldsymbol{\mathcal{E}},\end{aligned} \tag{39}$$

where $\boldsymbol{\mathcal{Y}} \doteq \mathcal{T}\{\boldsymbol{y}\} \in \mathbb{C}^{\bar{I}\times\bar{J}\times\bar{F}}$ and $\boldsymbol{\mathcal{E}} \doteq \mathcal{T}\{\boldsymbol{e}\} \in \mathbb{C}^{\bar{I}\times\bar{J}\times\bar{F}}$ are the compressed data and noise tensors, respectively, which follows the mapping given by (38). Hence, the tensor-CS model expresses the tensor of observations $\boldsymbol{\mathcal{Y}}$ in terms of multilinear projections of the sparse tensor $\boldsymbol{\mathcal{D}}$ onto the compressed basis matrices $\{\boldsymbol{\Phi}_i\boldsymbol{\Psi}_i\}$, $i = 1, 2, 3$, associated with the 1-mode, 2-mode, and 3-mode of the observed tensor data.

Remark 1: Vectorizing the tensor-CS model (39) yields

$$\begin{aligned}\boldsymbol{y} &\doteq \mathrm{vec}(\boldsymbol{\mathcal{Y}})\\ &= \big[(\boldsymbol{\Phi}_3\boldsymbol{\Psi}_3) \otimes (\boldsymbol{\Phi}_2\boldsymbol{\Psi}_2) \otimes (\boldsymbol{\Phi}_1\boldsymbol{\Psi}_1)\big]\boldsymbol{d} + \boldsymbol{e}\\ &= \boldsymbol{\Upsilon}\boldsymbol{d} + \boldsymbol{e},\end{aligned} \tag{40}$$

which coincides with the Kronecker-CS model introduced in [35], for which analytical bounds are given to ensure sparse recovery in terms of mutual coherence properties of the Kronecker product. Instead of operating over the resulting vector $\boldsymbol{y} \in \mathbb{C}^{\bar{I}\bar{J}\bar{F}}$ of compressed observations using vector sparse reconstruction algorithms, we are interested in preserving the tensor structure of the compressed data, which implies operating over each mode of the tensor $\boldsymbol{\mathcal{Y}} \in \mathbb{C}^{\bar{I}\times\bar{J}\times\bar{F}}$. As shown in Section VII, this entails a significant complexity reduction of sparse recovery algorithms, such as those based on the orthogonal matching pursuit (OMP) approach.

B. TENSOR-CS DATA MODEL

Recall our measurement tensor model given by (20), which expresses the received signal after the spatial combining. Our goal is to estimate the (nonzero) coefficients of the virtual channel tensor $\boldsymbol{\mathcal{H}}^{\mathrm{v}} \in \mathbb{C}^{M\times N\times P}$, where $M \geq I$, $N \geq J$, and $P \geq F$ denote, respectively, the number of discrete receive angles, transmit angles, and delays that make up the codebook set $\{\boldsymbol{A}_{\mathrm{R}}, \boldsymbol{A}_{\mathrm{T}}, \boldsymbol{A}_{\mathrm{F}}\}$, according to (10). Replacing (10) into (20) gives

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{H}}^{\mathrm{v}} \times_1 (\boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}}) \times_2 (\boldsymbol{X}\boldsymbol{A}_{\mathrm{T}}) \times_3 (\boldsymbol{F}\boldsymbol{A}_{\mathrm{F}}) + \tilde{\boldsymbol{\mathcal{Z}}}. \tag{41}$$

By analogy with (3), we conclude that the observations tensor $\boldsymbol{\mathcal{Y}}$ follows a Tucker3 model, and we have the following correspondences:

$$\big[\boldsymbol{\mathcal{X}}, \mathbf{A}, \mathbf{B}, \mathbf{C},\big] \longleftrightarrow \big[\boldsymbol{\mathcal{H}}^{\mathrm{v}}, (\boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}}), (\boldsymbol{X}\boldsymbol{A}_{\mathrm{T}}), (\boldsymbol{F}\boldsymbol{A}_{\mathrm{F}})\big]. \tag{42}$$

This Tucker3 model has a sparse core represented by the virtual channel tensor $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$, the structure of which reveals the joint sparse angle-delay profile. Note that (41) is a multilinear

CS model. Comparing (39) and (41), we deduce the following correspondences

$$\left[\boldsymbol{\mathcal{D}}, (\boldsymbol{\Phi}_1\boldsymbol{\Psi}_1), (\boldsymbol{\Phi}_2\boldsymbol{\Psi}_2), (\boldsymbol{\Phi}_3\boldsymbol{\Psi}_3)\right] \longleftrightarrow \left[\boldsymbol{\mathcal{H}}^{\mathrm{v}}, (\boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}}), (\boldsymbol{X}\boldsymbol{A}_{\mathrm{T}}), (\boldsymbol{F}\boldsymbol{A}_{\mathrm{F}})\right]. \quad (43)$$

Hence, the received pilot tensor $\boldsymbol{\mathcal{Y}}$ can be seen as the result of a multilinear compression of the virtual channel tensor, where the transmit spatial dimension is compressed by the pilot sequence matrix $\boldsymbol{X}$, the receive spatial dimension is compressed by the combining matrix $\boldsymbol{Q}$, and the frequency dimension is compressed by subcarrier selection/mapping matrix $\boldsymbol{F}$. The composite factor matrices $(\boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}})$, $(\boldsymbol{X}\boldsymbol{A}_{\mathrm{T}})$, and $(\boldsymbol{F}\boldsymbol{A}_{\mathrm{F}})$ correspond to the sparsifying bases for the channel tensor $\boldsymbol{\mathcal{G}}$. Note that the design of the measurement matrices of our tensor-CS model is dictated by the choice of the combiner $\boldsymbol{Q}$, the pilot sequence matrix $\boldsymbol{X}$, and the pilot placement strategy defined in $\boldsymbol{F}$. Moreover, since $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$ is sparse, we can set $L_{\mathrm{r}} < I$, $T_{\mathrm{p}} < J$, $F_{\mathrm{p}} < F$. In practice, especially in mmWave MIMO systems, the number of RF chains is much smaller than the number of receive antennas, which means that $L_{\mathrm{r}} < I$ is naturally satisfied. In addition, since the pilot subcarriers is always a subset of the total number of subcarriers (see Fig. 2), we always have $F_{\mathrm{p}} < F$. The remaining inequality, $T_{\mathrm{p}} < J$, is the most interesting one since it implies operating with short pilot sequences of length T_{p} compared to the number J of transmit antennas, usually large in massive MIMO systems.

Remark 2: The usual CS approach to estimate the channel parameters, considered in previous works [17], [18], [32], consists in vectorizing (41), resulting in the Kronecker-CS model (40), i.e.,

$$\boldsymbol{y} \doteq \mathrm{vec}(\boldsymbol{\mathcal{Y}}) = \left[(\boldsymbol{F}\boldsymbol{A}_{\mathrm{F}}) \otimes (\boldsymbol{X}\boldsymbol{A}_{\mathrm{T}}) \otimes (\boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}})\right]\boldsymbol{h}^{\mathrm{v}} + \tilde{z}, \quad (47)$$

where $\boldsymbol{h}^{\mathrm{v}} \doteq \mathrm{vec}(\boldsymbol{\mathcal{H}}^{\mathrm{v}}) \in \mathbb{C}^{NMP}$ is the virtual channel vector to be estimated. In this case, the estimation of $\boldsymbol{h}^{\mathrm{v}}$ can be carried out by means of linear programming [39] or by using second order cone programs [38]. Traditional sparse recovery algorithms such as the orthogonal matching pursuit (OMP) [38]) present a higher computational complexity due to the large size of the virtual channel vector, which is lower-bounded by the product dimension $IJF \leq NMP$. Specifically, the complexity grows exponentially with the channel dimensions and even for moderate number of antennas and subcarriers, the complexity of a vector-CS approach to estimate the channel can be prohibitive. When operating on vectorized observations, however, the 3D sparse structure of the angle-delay channel tensor is destroyed, since all the channel dimensions are mixed up. The tensor-CS algorithms presented in the next section preserve tensor structure of the problem by operating directly on $\boldsymbol{\mathcal{Y}} \in \mathbb{C}^{L_{\mathrm{r}} \times T_{\mathrm{p}} \times F_{\mathrm{p}}}$ to solve a CS problem for each channel dimension.

C. TENSOR-OMP WITH JOINT SEARCH

Although the conventional orthogonal matching-pursuit (OMP) algorithm and its variants have been widely studied in the literature, they have been designed to recover a sparse vector (1D signal). For recovering a higher-order tensor (e.g. a 3D signal), applying OMP to the "vectorized" tensor can potentially yield problems in terms of memory usage and processing. The high number of operations involved and their complexity, including inversion of big matrices, call for more efficient ways to circumvent these shortcomings. Our interest is to avoid channel vectorization by explicitly operating on the virtual channel tensor. The proposed tensor-OMP with joint search T-OMP with joint search (T-OMP-JS) operates on the tensor $\boldsymbol{\mathcal{Y}}$ to recover the full channel tensor $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$ by solving a sparse recovery problem per channel dimension in an iterative way.

Let us rewrite (41) as

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{H}}^{\mathrm{v}} \times_1 \boldsymbol{\Upsilon}_1 \times_2 \boldsymbol{\Upsilon}_2 \times_3 \boldsymbol{\Upsilon}_3, + \tilde{\boldsymbol{\mathcal{Z}}}, \quad (48)$$

where $\boldsymbol{\Upsilon}_1 = \boldsymbol{Q}\boldsymbol{A}_{\mathrm{R}} \in \mathbb{C}^{L_{\mathrm{r}} \times N}$, $\boldsymbol{\Upsilon}_2 = \boldsymbol{X}\boldsymbol{A}_{\mathrm{T}} \in \mathbb{C}^{T_{\mathrm{p}} \times M}$, and $\boldsymbol{\Upsilon}_3 = \boldsymbol{F}\boldsymbol{A}_{\mathrm{F}} \in \mathbb{C}^{F_{\mathrm{p}} \times P}$.

The T-OMP-JS algorithm consists of a greedy search per dimension of $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$. The cost function to be solved at the kth iteration is given by (44) as shown at the bottom of the next page, where $\boldsymbol{\mathcal{R}}^{(k-1)} \in \mathbb{C}^{L_{\mathrm{r}} \times T_{\mathrm{p}} \times F_{\mathrm{p}}}$ is the tensor of residuals at the $(k-1)$th iteration, where $k = 1, \ldots, K$, and K denotes the maximum number of iterations to convergence. In order to exploit each channel dimension, the algorithm should determine the index triplet $[\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}]$ that maximizes (44) at each iteration. This procedure involves a search over all possible combinations of indices $i_1^{(k)}$, $i_2^{(k)}$, and $i_3^{(k)}$ per iteration, where $\diamondsuit$ denotes the Khatri-Rao (columnwise Kronecker) product. The algorithm starts by setting $k = 1$ with $\boldsymbol{\mathcal{R}}^{(k-1)} = \boldsymbol{\mathcal{R}}^{(0)} = \boldsymbol{\mathcal{Y}}$, and the estimated virtual channel tensor set to the zero tensor, i.e. $\boldsymbol{\mathcal{H}}^{\mathrm{v}} = O_{N \times M \times P}$. The index sets $\mathbb{I}_1^{(k)}$, $\mathbb{I}_2^{(k)}$, and $\mathbb{I}_3^{(k)}$ store the optimum indices found at the $(k-1)$th iteration, for the first (space), second (time) and third (frequency) channel dimensions, respectively. At the kth iteration, an estimate of the nonzero entries of the channel tensor is found from (45), as shown at the bottom of the next page, where $\boldsymbol{h}^{(k)} \in \mathbb{C}^{k \times 1}$ is a parameter vector that collects the k nonzero entries of the channel found at the kth iteration. The parameter vector $\boldsymbol{h}^{(k)}$ is then used to update the entries of the virtual channel tensor as $\hat{\boldsymbol{\mathcal{H}}}^{\mathrm{v}}(n, m, p) \xleftarrow[\boldsymbol{\mathcal{T}}\{\cdot\}]{} \boldsymbol{h}^{(k)}_{n+(m-1)N+(p-1)NM}$. Then, a new tensor of residuals is formed by subtracting, at each iteration, a rank-one tensor channel component as shown in the second term of (46), as shown at the bottom of the next page, where $\circ$ denotes the outer product. After that, the updated tensor of residuals $\boldsymbol{\mathcal{R}}^{(k)}$ is used as input to find a new index triplet in the subsequent iteration. This procedure is repeated until the energy of the tensor of residuals is small enough according to a predefined threshold, i.e. $||\boldsymbol{\mathcal{R}}^{(k)}||_F^2 < \epsilon$, where ϵ denotes the threshold. In this work, we declare the convergence when ϵ is on the order of the noise variance. This assumption is valid for the two proposed algorithms. Table 1 provides a summary of the T-OMP-JS algorithm.

TABLE 1. T-OMP with joint search (T-OMP-JS).

Require:	Tensor of observations $\boldsymbol{\mathcal{Y}}$ and $\{\boldsymbol{\Upsilon}_i\}$, $i = 1, 2, 3$ $\mathbb{I}_1 = \{\}$, $\mathbb{I}_2 = \{\}$, and $\mathbb{I}_3 = \{\}$, $\hat{\boldsymbol{\mathcal{H}}}^{\text{v}} = \boldsymbol{\mathcal{O}}_{N\times M\times P}$
Step 1	$k = 1$; Set the residual tensor $\boldsymbol{\mathcal{R}}^{(0)} = \boldsymbol{\mathcal{Y}}$
Step 2	Solve Eq. (44) to obtain $[\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}]$
Step 3	$\mathbb{I}_1^{(k)} = \{\mathbb{I}_1^{(k-1)}, \bar{i}_1^{(k)}\}$, $\mathbb{I}_2^{(k)} = \{\mathbb{I}_2^{(k-1)}, \bar{i}_2^{(k)}\}$, $\mathbb{I}_3^{(k)} = \{\mathbb{I}_3^{(k-1)}, \bar{i}_3^{(k)}\}$
Step 4	Obtain the k strongest channel entries using Eq. (45)
Step 5	Update the estimated virtual channel tensor $\hat{\boldsymbol{\mathcal{H}}}^{\text{v}}(n, m, p) \xleftarrow[\boldsymbol{\mathcal{T}}\{\cdot\}]{} \boldsymbol{h}^{(k)}_{n+(m-1)N+(p-1)NM}$
Step 6	Update the tensor of residuals $\boldsymbol{\mathcal{R}}^{(k)}$ using Eq. (46)
Step 7	$k \leftarrow k + 1$; Repeat Steps 2 to 6 until $\|\boldsymbol{\mathcal{R}}^{(k)}\|_F^2 < \epsilon$

D. TENSOR-OMP WITH SEQUENTIAL SEARCH

The T-OMP with sequential search (T-OMP-SS) is an algorithm derived from the T-OMP-JS to reduce the computation complexity of an exhaustive search. Let us remember that T-OMP-JS solves the problem in (44) by simultaneously testing all the combinations of index triplets $[i_1^{(k)}, i_2^{(k)}, i_3^{(k)}]$ at the kth iteration. The computational complexity associated with this procedure may be high when operating with large dictionaries. This is problem when considering user terminals with reduced computational resources or limited power consumption. For instance, wideband systems and/or scenarios where a high angular resolution is required, the size of the codebook set $\{\boldsymbol{A}_{\text{R}}, \boldsymbol{A}_{\text{T}}, \boldsymbol{A}_{\text{F}}\}$ can be very large, leading to a high number of index triplets to be evaluated. In order to turn the problem solvable in this scenario, the T-OMP-SS cuts out triplets that are likely to be out of the solution set, speeding up the search procedure due to a reduced number of operations. Indeed, the solution of (44) implemented in step 2 of T-OMP-JS may exhibit a high computation complexity in cases where (at least) one dimension of the virtual channel tensor $\boldsymbol{\mathcal{H}}^{v}$ is very large.

The idea behind T-OMP-SS is to solve (44) in a more efficient fashion. The algorithm exploits the n-mode product operator over the three dimensions by sequentially solving one-dimensional (1D) search problems, starting with a search on the space domain, then on the time domain, and finally on the frequency domain. The first iteration of the algorithm ($k = 1$) sets the residual tensor $\boldsymbol{\mathcal{R}}^{(k-1)} = \boldsymbol{\mathcal{R}}^{(0)} = \boldsymbol{\mathcal{Y}}$. We also define the index sets $\mathbb{I}_1^{(k)}$, $\mathbb{I}_2^{(k)}$, and $\mathbb{I}_3^{(k)}$ to store the optimum indexes found at the $(k-1)$th iteration, for the first (space), second (time) and third (frequency) channel dimensions, respectively. To obtain such indexes, the algorithm splits the three-mode products shown in (44) into three sequentially related maximization problems that are solved in three steps. The T-OMP-SS algorithm starts by scanning across the space dimension, while averaging the tensor of residuals over the time and frequency dimensions, as shown in (49) at the bottom of the next page. Once $\bar{i}_1^{(k)}$ is determined, the selected matrix-slice of residuals given by $\boldsymbol{\mathcal{R}}^{(k-1)} \times_1 \boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)}) \in \mathbb{C}^{T_{\text{P}}\times F_{\text{P}}}$ is used as an input to the second maximization problem the goal of which is to find $\bar{i}_2^{(k)}$. We stress that multiplying the tensor of residuals $\boldsymbol{\mathcal{R}}^{(k-1)}$ by the selected column of the space-domain basis $\boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)})$ returns a matrix of residuals, i.e., all the triplets that are not associated with the index $\bar{i}_1^{(k)}$ are discarded. The second step solves problem (50), as shown at the bottom of the next page by making use of the matrix of residuals $\boldsymbol{\mathcal{R}}^{(k-1)} \times_1 \boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)}) \in \mathbb{C}^{T_{\text{P}}\times F_{\text{P}}}$ to scan across the time dimension, while averaging over the frequency dimension. The solutions $\bar{i}_1^{(k)}$ and $\bar{i}_2^{(k)}$ are used to generate a vector of residuals $\boldsymbol{\mathcal{R}}^{(k-1)} \times_1 \boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)}) \times_2 \boldsymbol{\Upsilon}_2^H(:, \bar{i}_2^{(k)}) \in \mathbb{C}^{F_{\text{P}}}$ that is used as an input to the third step. Finally, in the third step, a frequency-domain search is performed using (51), as shown at the bottom of the next page, to find $\bar{i}_3^{(k)}$. The tensor of residuals is updated according to steps (45) and (46), as for the T-OMP-JS algorithm, and the same procedure is repeated until convergence.

Note that the T-OMP-SS algorithm follows the same logic as a greedy algorithm, but replace a 3D search by three sequentially related 1D searches to determine the triplet $[\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}]$ as shown in (49)-(51). Table 2 summarizes the T-OMP-SS algorithm.

TABLE 2. T-OMP with sequential search (T-OMP-SS).

Require:	Tensor of observations $\boldsymbol{\mathcal{Y}}$ and $\{\boldsymbol{\Upsilon}_i\}$, $i = 1, 2, 3$ $\mathbb{I}_1 = \{\}$, $\mathbb{I}_2 = \{\}$, and $\mathbb{I}_3 = \{\}$, $\hat{\boldsymbol{\mathcal{H}}}^{\text{v}} = \boldsymbol{\mathcal{O}}_{N\times M\times P}$
Step 1	$k = 1$; Set the residual tensor $\boldsymbol{\mathcal{R}}^{(0)} = \boldsymbol{\mathcal{Y}}$
Step 2	Solve Eq. (49), (50), and (51) to obtain $[\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}]$
Step 3	$\mathbb{I}_1^{(k)} = \{\mathbb{I}_1^{(k-1)}, \bar{i}_1^{(k)}\}$, $\mathbb{I}_2^{(k)} = \{\mathbb{I}_2^{(k-1)}, \bar{i}_2^{(k)}\}$, $\mathbb{I}_3^{(k)} = \{\mathbb{I}_3^{(k-1)}, \bar{i}_3^{(k)}\}$
Step 4	Obtain the k strongest channel entries using Eq. (45)
Step 5	Update the estimated virtual channel tensor $\hat{\boldsymbol{\mathcal{H}}}^{\text{v}}(n, m, p) \xleftarrow[\boldsymbol{\mathcal{T}}\{\cdot\}]{} \boldsymbol{h}^{(k)}_{n+(m-1)N+(p-1)NM}$
Step 6	Update the tensor of residuals $\boldsymbol{\mathcal{R}}^{(k)}$ using Eq. (46)
Step 7	$k \leftarrow k + 1$; Repeat Steps 2 to 6 until $\|\boldsymbol{\mathcal{R}}^{(k)}\|_F^2 < \epsilon$

The convergence of TOMP-SS and TOMP-JS is affected by the degree of sparsity of the channel tensor representation. More specifically, two aspects directly affect the convergence speed, which are the number of channel paths and the choice

$$\left[\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}\right] = \arg\max_{i_1, i_2, i_3} \left|\boldsymbol{\mathcal{R}}^{(k-1)} \times_1 \boldsymbol{\Upsilon}_1^H(:, i_1) \times_2 \boldsymbol{\Upsilon}_2^H(:, i_2) \times_3 \boldsymbol{\Upsilon}_3^H(:, i_3)\right|^2 \tag{44}$$

$$\boldsymbol{h}^{(k)} = \left(\boldsymbol{\Upsilon}_3(:, \mathbb{I}_3^{(k)}) \diamond \boldsymbol{\Upsilon}_2(:, \mathbb{I}_2^{(k)}) \diamond \boldsymbol{\Upsilon}_1(:, \mathbb{I}_1^{(k)})\right)^{\dagger} \text{vec}(\boldsymbol{\mathcal{Y}}) \tag{45}$$

$$\boldsymbol{\mathcal{R}}^{(k)} = \boldsymbol{\mathcal{R}}^{(k-1)} - \hat{\boldsymbol{\mathcal{H}}}^{\text{v}}(\bar{i}_1^{(k)}, \bar{i}_2^{(k)}, \bar{i}_3^{(k)}) \circ \boldsymbol{\Upsilon}_1(:, \bar{i}_1^{(k)}) \circ \boldsymbol{\Upsilon}_2(:, \bar{i}_2^{(k)}) \circ \boldsymbol{\Upsilon}_3(:, \bar{i}_3^{(k)}) \tag{46}$$

of the representation bases. The higher is the number of channel paths, the higher is the number of nonzero elements of its sparse representation and more iterations will be necessary for the convergence, for a fixed stopping criterion. In addition, the mismatch between the bases and the actual steering vectors generates a power leakage in the virtual (core) tensor, which also causes an increase on the required number of iterations for the convergence of both algorithms.

VII. COMPUTATIONAL COMPLEXITY

In the downlink, channel estimation is implemented at UEs that generally has limited computational resources compared to the BS. An analysis of the computational complexity required by the different solutions proposed in the previous section is therefore an important aspect to be considered. The MMSE method follows the conventional vector formulation, which represents the signal model by stacking all the measurements in a single vector containing space-, time- and frequency- domain samples. This approach implies solving for a linear system modeled as a Kronecker product of space, time, and filtering components, as shown in (21). The dominant cost of the MMSE estimator is associated with that of inverting the square matrix in the right-hand side of (23) which has size $L_{\mathrm{r}}T_{\mathrm{p}}F_{\mathrm{p}}$. This operation results in a complexity of $O\left((L_{\mathrm{r}}T_{\mathrm{p}}F_{\mathrm{p}})^3\right)$. The T-MMSE estimator reduces the number of operations by exploiting a tensor representation of the MIMO-OFDM channel. Instead of considering a single linear system, the tensor solution breaks the problem into three smaller ones with smaller transformation matrices. The resulting complexity is that of solving the n-mode MMSE solutions for the three smaller sub-problems given in (33)-(35), resulting in a complexity of $O\left(L_{\mathrm{r}}^3 + F_{\mathrm{p}}^3 + T_{\mathrm{p}}^3\right)$.

Both T-OMP-JS and T-OMP-SS are greedy algorithms that implement the greedy search in different manners. The first uses a joint search over the full tensor space, i.e. the algorithm search over all the dimensions simultaneously by trying all combinations of triplets $\{i_1, i_2, i_3\}$. The required number of operations implies a complexity $O(T_{\mathrm{p}}F_{\mathrm{p}}MNP)$, where M is the number of scanned directions at the receive side, N is the number of scanned directions at the transmit side, and P is the number of scanned delay bins, respectively. The product $L_{\mathrm{r}}T_{\mathrm{p}}F_{\mathrm{p}}$ correspond to the number of elements of the compressed tensor $\boldsymbol{\mathcal{Y}}$, which are much smaller than the product MNP corresponding to the number of elements of the sparse virtual channel tensor $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$. Therefore if M, N, and/or P is large, the more granular and costly will be the greedy search. On the other hand, T-OMP-SS breaks the maximization problem in (44) down to three smaller ones, given by (49), (50), and (51). The overall complexity is obtained by summing over the complexities of the individual search procedures for each channel dimension, which results in $O\left(L_{\mathrm{r}}T_{\mathrm{p}}F_{\mathrm{p}}N + T_{\mathrm{p}}F_{\mathrm{p}}M + F_{\mathrm{p}}P\right)$. In addition to the greedy search, both T-OMP-SS and T-OMP-JS calculate the pseudoinverse of the Khatri-Rao product matrix $\left(\boldsymbol{\Upsilon}_3(:, \mathbb{I}_3^{(k)}) \diamond \boldsymbol{\Upsilon}_2(:, \mathbb{I}_2^{(k)}) \diamond \boldsymbol{\Upsilon}_1(:, \mathbb{I}_1^{(k)})\right)$ as shown in (45), which is needed to estimate $\boldsymbol{\mathcal{H}}^{\mathrm{v}}$ at each iteration. This step adds a complexity of $O\left(k^3\right)$ to both algorithms, as shown in Table 3. Note that the number of columns of the Khatri-Rao product matrix in (45) is incremented as a function of the iteration index k. For sparser channels with only few paths, less iterations are needed for the convergence and the impact of the term $O\left(k^3\right)$ will be limited.

TABLE 3. Computational complexity of the different estimators.

Solution	Complexity
MMSE	$O\left((L_{\mathrm{r}}F_{\mathrm{p}}T_{\mathrm{p}})^3\right)$
T-MMSE	$O\left(L_{\mathrm{r}}^3 + F_{\mathrm{p}}^3 + T_{\mathrm{p}}^3\right)$
T-OMP-SS	$O\left(L_{\mathrm{r}}T_{\mathrm{p}}F_{\mathrm{p}}N + T_{\mathrm{p}}F_{\mathrm{p}}M + F_{\mathrm{p}}P\right) + O\left(k^3\right)$
T-OMP-JS	$O\left(T_{\mathrm{p}}F_{\mathrm{p}}L_{\mathrm{r}}MNP\right) + O\left(k^3\right)$

VIII. NUMERICAL RESULTS

In this section, we provide results from computer simulations to evaluate the performance of the proposed tensor-based estimators. In all simulations, we assume the BS and UE are equipped with uniform rectangular arrays, where the BS is equipped with $J = 64$ antennas while the UE has $I = 4$ antennas. Unless stated otherwise, our simulations assume $L_{\mathrm{t}} = 32$ and $L_{\mathrm{r}} = 2$ RF chains at the BS and UE, respectively. We also assume $T_{\mathrm{p}} = L_{\mathrm{t}} = 32$ pilot symbol periods and $F_{\mathrm{p}} = 256$ regularly-spaced pilot subcarriers. Table 4 describes the simulation setup used in all the simulations.

The performance is evaluated in terms of the normalized mean square error (NMSE) defined as

$$\mathrm{NMSE} = \frac{1}{N_{\mathrm{runs}}} \sum_{n=1}^{N_{\mathrm{runs}}} \frac{\|\boldsymbol{\mathcal{G}}_n - \hat{\boldsymbol{\mathcal{G}}}_n\|_{\mathrm{F}}^2}{\|\boldsymbol{\mathcal{G}}_n\|_{\mathrm{F}}^2}, \tag{52}$$

where N_{runs} is the number of Monte Carlo runs, while

$$\bar{i}_1^{(k)} = \arg\max_{i_1} \frac{1}{T_{\mathrm{p}}F_{\mathrm{p}}} \sum_{m=1}^{T_{\mathrm{p}}} \sum_{n=1}^{F_{\mathrm{p}}} \left| \boldsymbol{\mathcal{R}}^{(k-1)}(:, m, n) \times_1 \boldsymbol{\Upsilon}_1^H(:, i_1) \right|^2. \tag{49}$$

$$\bar{i}_2^{(k)} = \arg\max_{i_2} \frac{1}{F_{\mathrm{p}}} \sum_{m=1}^{F_{\mathrm{p}}} \left| \boldsymbol{\mathcal{R}}^{(k-1)}(:, :, m) \times_1 \boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)}) \times_2 \boldsymbol{\Upsilon}_2^H(:, i_2) \right|^2. \tag{50}$$

$$\bar{i}_3^{(k)} = \arg\max_{i_3} \left| \boldsymbol{\mathcal{R}}^{(k-1)} \times_1 \boldsymbol{\Upsilon}_1^H(:, \bar{i}_1^{(k)}) \times_2 \boldsymbol{\Upsilon}_2^H(:, \bar{i}_2^{(k)}) \times_3 \boldsymbol{\Upsilon}_3^H(:, i_3) \right|^2. \tag{51}$$

TABLE 4. **Simulation parameters.**

Parameter	Value
Azimuth and elevation distribution at Tx and Rx sides	uniform
Delay distribution	exponential
Azimuth spread $\sigma_{\phi,T}$ at Tx side	20 deg
Azimuth mean $\bar{\phi}_T$ at Tx side	90 deg
Elevation spread $\sigma_{\theta,T}$ at Tx side	5 deg
Elevation mean $\bar{\theta}_T$ at Tx side	100 deg
Azimuth spread $\sigma_{\phi,R}$ at Rx side	20 deg
Azimuth mean $\bar{\phi}_R$ at Rx side	90 deg
Elevation spread $\sigma_{\theta,R}$ at Rx side	5 deg
Elevation mean $\bar{\theta}_R$ at Rx side	5 deg
Delay mean $\bar{\tau}$	1μs
Distribution of β_k in (12)	Normal
Variance of β_k in (12)	1
Mean of β_k in (12)	0
Number of paths K	$\{8, 16\}$
Planar array configuration at the Tx side	8x8
Planar array configuration at the Rx side	2x2
Number of subcarriers (F)	1024
Codebook $\boldsymbol{A}_{\mathrm{R}}$	2D-DFT
Codebook $\boldsymbol{A}_{\mathrm{T}}$	2D-DFT
Codebook $\boldsymbol{A}_{\mathrm{F}}$	DFT
Number of RF chains in the BS	$\{2, 4, 6, \ldots, 64\}$
Number of RF chains in the UE	2
Number of subcarriers F	1024
Number of Monte Carlo runs	1000

$\mathcal{G}_n$ and $\hat{\mathcal{G}}_n$ denote the true and estimated channel tensors at the nth run. For all simulations, we assume $N_{\text{runs}} = 1000$. Each run corresponds to a different realization of the channel and noise tensors $\mathcal{G}$ and $\mathcal{Z}$, respectively. The entries (phases) of the beamforming matrix $\mathbf{W}$ are also randomly drawn at each run. We also show the impact of the channel estimation accuracy on the spectral efficiency by means of the Shannon's formula

$$C = \left(1 - \frac{T_{\mathrm{p}}F_{\mathrm{p}}}{TF}\right)\log_2\left(1 + \mathrm{SNR}\right). \tag{53}$$

Recall that T_{p} and F_{p} denote the number of symbol periods and subcarriers used for pilots, as shown in Fig. 2 (c.f. Section IV). For all simulations, the pilot subcarriers are regularly spaced and every RF chain transmits a different pilot sequence, which is spread across all the pilot subcarriers.

Recall that the channel used in our experiments follows (8). At each Monte Carlo run, to generate the kth azimuth and elevation angles for the transmit and receive sides, we assume $\theta_{T,k} \sim U(\bar{\theta}_T, \sigma_{\theta,T})$, $\theta_{R,k} \sim U(\bar{\theta}_R, \sigma_{\theta,R})$, $\phi_{T,k} \sim U(\bar{\phi}_T, \sigma_{\phi,T})$, and $\phi_{R,k} \sim U(\bar{\phi}_R, \sigma_{\phi,R})$, where $U(a, b)$ stands for uniform distribution with mean a and standard deviation b, while $\bar{\theta}_T$ and $\bar{\theta}_R$ as the mean azimuth angles at the transmit and receive sides, respectively. The parameters $\sigma_{\theta,T}$ and $\sigma_{\theta,R}$ are the transmit and receive azimuth spreads, respectively. Likewise, $\bar{\phi}_T$ and $\bar{\phi}_R$ are the respective mean elevation angles, and $\sigma_{\phi,T}$ and $\sigma_{\phi,R}$ are the elevation spreads. The power delay profile of the channel follows an exponential distribution with mean and variance equal to $\bar{\tau}$ is assumed for the path delays. The complex path gains $\{\beta_k\}$ are drawn from a normal distribution with zero mean and unitary variance.

Our first experiment evaluates the convergence of the T-OMP-JS and T-OMP-SS algorithms. Figure 3 depicts the energy of the tensor of residuals $||\mathcal{R}^{(k)}||_F^2$, calculated at Step 7 of the proposed algorithm (see Tables 1 and 2), as a function of the iteration index k. We consider two scenarios with $K = 8$ and $K = 16$ paths with the SNR fixed to 20dB. We can observe that both algorithms converge to low reconstruction errors (smaller than 0.1) around 30 iterations for $K = 8$, and 60 iterations for $K = 16$. As expected, convergence is faster for sparser channels with few dominant paths. We can also note that T-OMP-JS and T-OMP-SS have the same convergence behaviour, the difference between them being on the lower computational complexity of the second one.

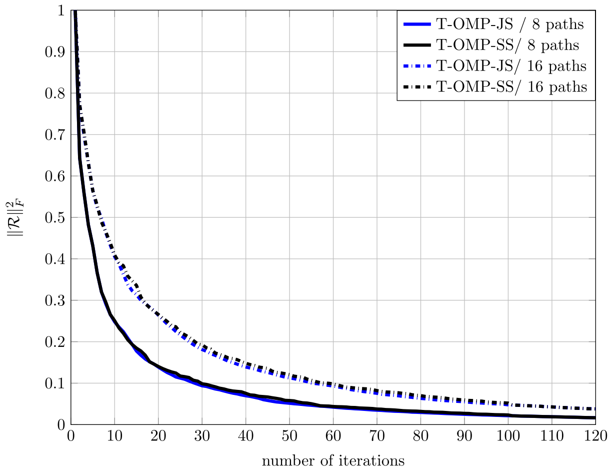

FIGURE 3. **Convergence curves for T-OMP-JS and T-OMP-SS: $||\mathcal{R}^{(k)}||_F^2$ vs. k.**

In the second set of simulations, we evaluate the NMSE performances of T-OMP-SS, T-OMP-JS, vec-OMP, and T-MMSE. For the T-MMSE estimator, we assume the knowledge of the covariance matrices to compute the n-mode MMSE filters in (33)-(35). In contrast, the sparse channel estimators (T-OMP-SS, T-OMP-JS, and vec-OMP) rely on the knowledge of the sparsifying bases (codebooks). Figure 4 shows that the sparsity-based approaches exhibit a saturation in their NMSE curves. This behaviour comes from the

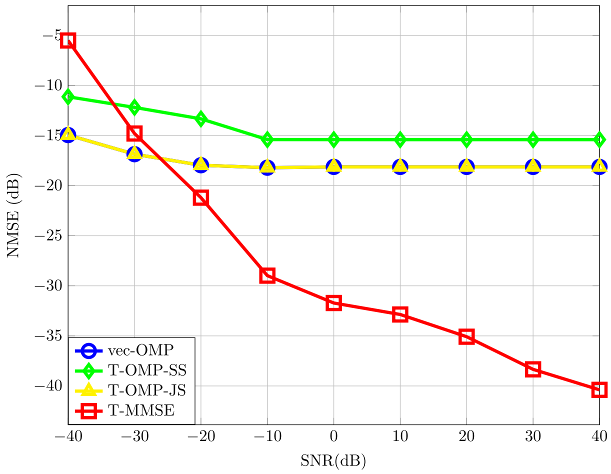

FIGURE 4. **NMSE performances of different estimators:** vec-**OMP, T-MMSE, T-OMP-JS, and T-OMP-SS. $K = 16$.**

mismatch between the codebooks and the actual steering vectors defined in Eq. (8), which generates a power leakage in the core tensor $\mathcal{H}^{\mathrm{v}}$. Indeed, such a leakage increases the number of non-zero entries of $\mathcal{H}^{\mathrm{v}}$ and, consequently, its sparsity is decreased. To overcome this off-grid problem, we would need to learn/refine the codebooks while exploiting the tensor structure of the problem. Despite being outside the scope of this work, this issue is important and will be addressed in a future contribution. Note also that vec-OMP and T-OMP-JS have similar performances, although the later is much less complex, as discussed in Section VII. By its turn, the T-OMP-SS estimator exhibits a small degradation compared to vec-OMP and T-OMP-JS, offering a trade-off between computational complexity and accuracy. As expected, the best performance is achieved with the T-MMSE estimator with perfect knowledge of the channel covariance structure.

In a second experiment, we study the influence of the number of paths in the performance of different tensor-based estimators. To this end, we keep the same parameters as assumed for the previous experiment (Fig. 4) but now reduce the number of channel paths down to $K = 8$. The results are depicted in Fig. 5. The curves show that T-OMP-JS and T-OMP-SS have similar performances. The T-MMSE solution outperforms the other two when the SNR is very low. Otherwise, the three methods exhibit nearly the same performance. Indeed, in this scenario, T-OMP-JS and T-OMP-SS are more attractive due to their good performance-complexity tradeoff. This experiment also indicates that, for a small number of paths, the gains obtained with the T-MMSE are marginal, in contrast to the gains observed in Figure 4 with $K = 16$, which are significant. Nevertheless, the proposed T-MMSE estimator is still far less complex than the traditional vector-MMSE estimator when a large number of antennas are assumed at the BS and UE, which is the situation assumed in this paper. Otherwise stated, for a moderate number of paths T-MMSE is a good choice, since it offers a good accuracy while being more computationally efficient compared with the vector-MMSE solution.

In the next experiment, we evaluate the influence of the number of RF chains at the transmitter on the performance of the different tensor-based estimators proposed in this work. As shown in Figure 6, T-MMSE, T-OMP-JS and T-OMP-SS reduce the estimation error with more RF chains at the transmitter. All the three methods have similar performances with an advantage of T-MMSE over T-OMP-JS and T-OMP-SS within the range from 10 to 45 number of RF chains. This means T-MMSE affords a simpler transmitter that employs less RF chains than that with sparsity-based algorithms.

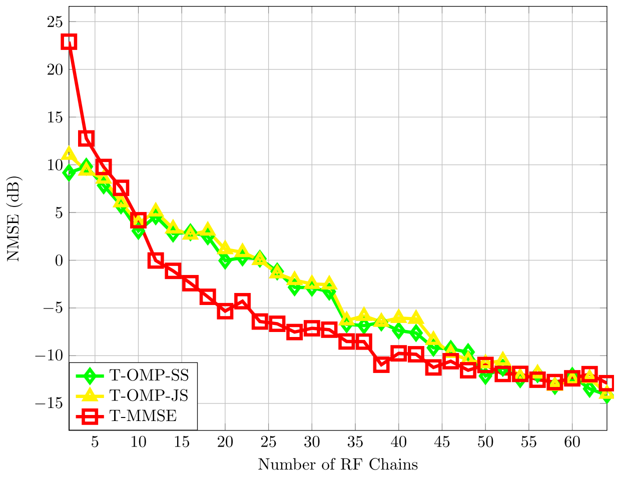

FIGURE 6. NMSE performance as a function of the number L_t of RF chains at the transmitter. $K = 8$ and SNR = 20 dB.

We have also studied the impact of the proposed estimators in terms of spectral efficiency, as depicted in Figure 7 for $K = 8$ and SNR $= 20$ dB. In this experiment, we assume an eigenbeamforming solution, where the transmit and receive beamformers for each subcarrier are designed from the left and right singular vectors of the associated MIMO channel, respectively. The spectral efficiency (in bps/Hz) is plotted

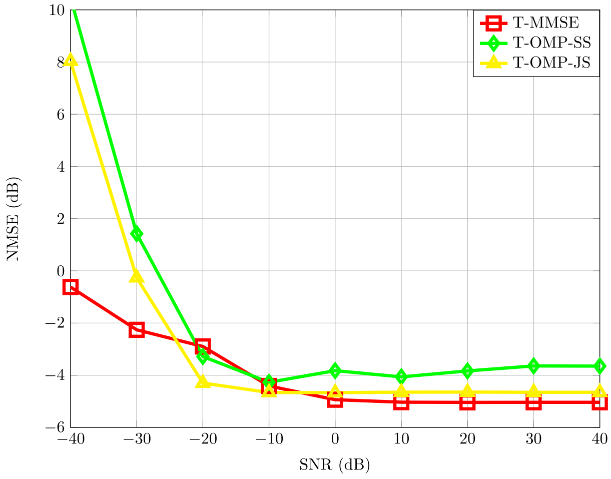

FIGURE 5. Performance of the proposed tensor-based estimators for $K = 8$.

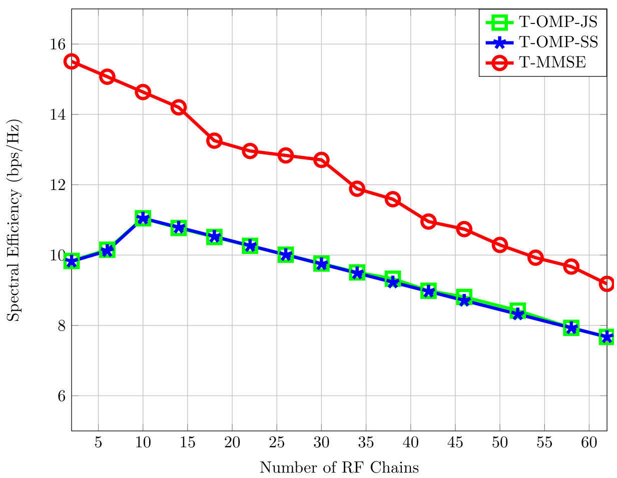

FIGURE 7. Spectral efficiency as a function of the number L_t of RF chains at the transmitter.

as a function of the number L_t of RF chains. It can be seen that T-OMP-JS and T-OMP-SS exhibit similar performances, while T-MMSE offers the best results. The superiority of T-MMSE comes at the price of an accurate knowledge of the channel second-order statistics, which are not necessary for T-OMP-SS and T-OMP-JS. As expected, using more RF chains does not always imply a higher spectral efficiency, since the number of pilot sequences used for channel estimation scales with L_t (recall that every RF chain transmits a different pilot sequence).

Figure 8 shows the behaviour of T-OMP-SS and T-OMP-JS for low and high angular spreads. To this end, we consider $\sigma_{\phi,T} = 5$ and $\sigma_{\phi,T} = 20$, respectively. With the low angular spread, both T-OMP-JS and T-OMP-SS improve their performance as the SNR increases. However, the curves saturate due to the off-grid problem. Essentially, the beams predefined by the codebook do not match to the actual channel paths, with yields a power leakage. This problem reduces the channel sparsity and limits the performance of T-OMP-JS and T-OMP-SS. Note that for the higher the angular spread value, the error floor limits the performance of both algorithms, as we can conclude by comparing blue against black curves in Fig. 8.

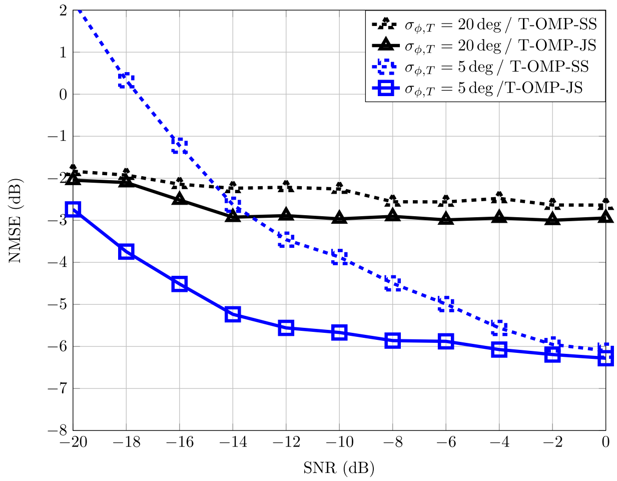

FIGURE 8. Impact of the angular spread at the transmitter side.

Another important aspect is the difference between T-OMP-JS and T-OMP-SS. In the low angular spread case, T-OMP-JS achieves better results than T-OMP-SS, but both solutions have similar performances in the high angular spread situation. In the later case, we can conclude that performing a joint search with T-OMP-JS produces marginal gains over the sequential search with T-OMP-SS. In summary, using T-OMP-JS is beneficial for low angular spreads and very low SNR. On the other hand, for high angular spreads the gap between both solutions diminishes and both of them become less attractive since the channel is no more sparse due to the large number of resolvable paths. The same can be said about the traditional vector-OMP, which also exhibits poor performance for high angular spreads. In this case, vector-MMSE and T-MMSE are preferable, the later being a better choice for massive MIMO systems due to its low complexity, as previously discussed.

IX. CONCLUSION

We have proposed new algorithms to estimate the CSI in massive MIMO-OFDM systems that jointly exploit the sparsity and multidimensional nature of the channel. The proposed tensor-based estimators enable the UEs to estimate the MIMO channel by assuming that (i) the BS and UE have a limited number of RF chains, (ii) the MIMO channel is frequency-selective, and (iii) the system has limited pilot resources for channel acquisition, allowing to reduce the training overhead. While T-MMSE is an attractive solution for a higher number of paths, T-OMP-JS and T-OMP-SS are preferable in poor scattering scenarios with a smaller number of paths due to their good complexity-performance tradeoff. The performance of the proposed T-OMP-JS and T-OMP-SS can be further enhanced to deal with off-grid problems, where the channel paths do not match to the basis set $(\boldsymbol{A}_{\mathrm{F}}, \boldsymbol{A}_{\mathrm{T}}, \boldsymbol{A}_{\mathrm{R}})$. This situation happens, for instance, when there is uncertainty in the structure of the array response vectors at the transmitter and/or receiver due to calibration errors. In this case, tensor-based dictionary learning algorithms [40], [41] could be used. In addition, the proposed tensor-CS model can be generalized to include Doppler shifts and carrier-frequency offsets.

REFERENCES

[1] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[2] M. Soltanalian, M. M. Naghsh, N. Shariati, P. Stoica, and B. Hassibi, "Training signal design for correlated massive MIMO channel estimation," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 1135–1143, Feb. 2017.

[3] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[4] J. Choi, D. J. Love, and P. Bidigare, "Downlink training techniques for FDD massive MIMO systems: Open-loop and closed-loop training with memory," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 802–814, Oct. 2014.

[5] D. J. Love, J. Choi, and P. Bidigare, "A closed-loop training approach for massive MIMO beamforming systems," in *Proc. 47th Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2013, pp. 1–5.

[6] Z. Gao, L. Dai, W. Dai, B. Shim, and Z. Wang, "Structured compressive sensing-based spatio-temporal joint channel estimation for FDD massive MIMO," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 601–617, Feb. 2016.

[7] X. Rao and V. K. N. Lau, "Compressive sensing with prior support quality information and application to massive MIMO channel estimation with temporal correlation," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4914–4924, Sep. 2015.

[8] A. M. Sayeed and T. Sivanadyan, "Wireless communication and sensing in multipath environments using multi-antenna transceivers," *Handbook on Array Processing and Sensor Networks*, K. J. R. Liu and S. Haykin, Eds. Piscataway, NJ, USA: IEEE Press, 2010.

[9] W. U. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed channel sensing: A new approach to estimating sparse multipath channels," *Proc. IEEE*, vol. 98, no. 6, pp. 1058–1076, Jun. 2010.

[10] J.-P. Kermoal, L. Schumacher, K. I. Pedersen, P. E. Mogensen, and F. Frederiksen, "A stochastic MIMO radio channel model with experimental validation," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 6, pp. 1211–1226, Aug. 2002.

[11] W. Weichselberger, M. Herdin, H. Ozcelik, and E. Bonek, "A stochastic MIMO channel model with joint correlation of both link ends," *IEEE Trans. Wireless Commun.*, vol. 5, no. 1, pp. 90–100, Jan. 2006.

[12] A. M. Sayeed, "Deconstructing multiantenna fading channels," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2563–2579, Oct. 2002.

[13] J. He, T. Kim, H. Ghauch, K. Liu, and G. Wang, "Millimeter wave MIMO channel tracking systems," in *Proc. Globecom Workshops (GC Wkshps)*, Dec. 2014, pp. 416–421.

[14] P. Schniter and A. Sayeed, "Channel estimation and precoder design for millimeter-wave communications: The sparse way," in *Proc. 48th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2014, pp. 273–277.

[15] D. C. Araujo, A. L. F. de Almeida, and J. C. M. Mota, "Compressive sensing based channel estimation for massive MIMO systems with planar arrays," *Proc. 6th IEEE Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process., (CAMSAP)*, Cancun, Mexico, Dec. 2015, pp. 413–416.

[16] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, Jr., "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[17] J, P. González–Coma, J. Rodríguez-Fernández, N. González-Prelcic, L. Castedo, and R. W. Heath, "Channel estimation and hybrid precoding for frequency selective multiuser mmWave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 2, pp. 353–367, May 2018.

[18] J. Rodríguez-Fernández, N. González-Prelcic, K. Venugopal, and R. W. Heath, "Frequency-domain compressive channel estimation for frequency-selective hybrid millimeter wave MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 2946–2960, May 2018.

[19] A. Nasser and M. Elsabrouty, "Frequency-selective massive MIMO channel estimation and feedback in angle-time domain," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2016, pp. 1018–1023.

[20] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, May 2013.

[21] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.

[22] A. Cichocki *et al.*, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 145–163, Mar. 2015.

[23] A. K. Smilde, R. Bro, and P. Geladi, *Multi-way Analysis with Applications in the Chemical Sciences*. Hoboken, NJ, USA: Wiley, 2004.

[24] K. Yu, M. Bengtsson, B. Ottersten, D. McNamara, P. Karlsson, and M. Beach, "Modeling of wide-band MIMO radio channels based on NLoS indoor measurements," *IEEE Trans. Veh. Technol.*, vol. 53, no. 3, pp. 655–665, May 2004.

[25] N. Costa and S. Haykin, "A novel wideband MIMO channel model and the wideband MIMO software defined radio," in *Proc. Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Sep. 2006, pp. 1–4.

[26] Y. Zhang *et al.*, "A general coupling-based model framework for wideband MIMO channels," *IEEE Trans. Antennas Propag.*, vol. 60, no. 2, pp. 574–586, Feb. 2012.

[27] F. L. Hitchcock, "Multiple invariants and generalized rank of a P-way matrix or tensor," *J. Math. Phys.*, vol. 7, nos. 1–4, pp. 39–79, Apr. 1928.

[28] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an explanatory multimodal factor analysis," *UCLA Work. Papers Phonetics*, vol. 16, pp. 1–84, Jun. 1970.

[29] J. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of 'Eckart-Young' decomposition," *Psychometrika*, vol. 3, pp. 283–319, 1970.

[30] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

[31] Z. Zhou, J. Fang, L. Yang, H. Li, Z. Chen, and S. Li, "Channel estimation for millimeter-wave multiuser MIMO systems via PARAFAC decomposition," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7501–7516, Nov. 2016.

[32] Z. Zhou, J. Fang, L. Yang, H. Li, Z. Chen, and R. S. Blum, "Low-rank tensor decomposition-aided channel estimation for millimeter wave MIMO-OFDM systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 7, pp. 1524–1538, Jul. 2017.

[33] D. C. Araújo and A. L. F. de Almeida, "Tensor-based compressed estimation of frequency-selective mmwave MIMO channels," in *Proc. IEEE 7th Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2017, pp. 1–5.

[34] N. D. Sidiropoulos and A. Kyrillidis, "Multi-way compressed sensing for sparse low-rank tensors," *IEEE Signal Process. Lett.*, vol. 19, no. 11, pp. 757–760, Nov. 2012.

[35] M. F. Duarte and R. G. Baraniuk, "Kronecker compressive sensing," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 494–504, Feb. 2012.

[36] D. C. Araujo, E. Karipidis, A. L. F. de Almeida, and J. C. M. Mota, "Improving spectral efficiency in large-array FDD systems with hybrid beamforming," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop (SAM)*, Jul. 2016, pp. 1–5.

[37] S. M. Kay, *Fundamentals Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[38] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[39] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1999.

[40] F. Roemer, G. Del Galdo, and M. Haardt, "Tensor-based algorithms for learning multidimensional separable dictionaries," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 3963–3967.

[41] X. Ding, W. Chen, and I. J. Wassell, "Joint sensing matrix and sparsifying dictionary optimization for tensor compressive sensing," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3632–3646, Jul. 2017.

DANIEL COSTA ARAÚJO received the B.S. degree from Fortaleza University (UNIFOR), in 2010, and the M.S. and Ph.D. degrees in teleinformatics engineering from the Federal University of Ceará (UFC), Brazil, in 2012 and 2016, respectively, where he currently holds a postdoctoral position with the Department of Teleinformatics Engineering, UFC. Since 2012, he has been a Researcher with the Wireless Communication Group (GTEL), Brazil, working in research projects within a technical cooperation with Ericsson Research, Sweden. In 2013, he was a Guest Researcher with Ericsson, and from 2015 to 2016, for a year-long period in Stockholm, Sweden. His research interests include array signal processing, tensor analysis, wireless communication systems, modeling and simulation of cellular communication, and channel estimation.

ANDRÉ L. F. DE ALMEIDA (M'08–SM'13) received the double Ph.D. degree in sciences and teleinformatics engineering from the University of Nice, Sophia Antipolis, France, and the Federal University of Ceará, Fortaleza, Brazil, in 2007. During fall 2002, he was a Visiting Researcher with Ericsson Research Labs, Stockholm, Sweden. From 2007 to 2008, he held a one-year teaching position with the University of Nice. In 2008, he received CAPES/COFECUB Research Fellowship with the I3S Laboratory, CNRS, France. He received multiples times visiting professor positions at the University of Nice, in 2012, 2013, 2015, and 2017. He is currently an Associate Professor with the Department of Teleinformatics Engineering, Federal University of Ceará. He authored or coauthored more than 60 journal articles published and accepted, 100 conference papers and five book chapters. His current research interests include tensor methods and multilinear algebra with applications to communications and signal processing. He is a member of the Sensor Array and Multichannel (SAM) Technical Committee of the IEEE Signal Processing Society (SPS) and the *EURASIP Signal Processing for Multi-Sensor Systems* Special Area Team. He is a Research Fellow of the CNPq (the Brazilian National Council for Scientific and Technological Development) and an Elected Affiliate Member of the Brazilian Academy of Sciences, since 2018. He was the General Co-Chair of the IEEE CAMSAP 2017 Workshop, and served as the Technical Co-Chair of the Symposium on Tensor Methods for Signal Processing and Machine Learning at GlobalSIP 2018. He also serves as the Technical Co-Chair of IEEE SAM 2020 Workshop, Hangzhou, China. He served as an Associate Editor for the IEEE Transactions on Signal Processing, from 2012 to 2016. He currently serves as an Associate Editor for the IEEE Signal Processing Letters.

JOÃO P. C. L. DA COSTA (SM'16) received the Diploma degree in electronic engineering from the Military Institute of Engineering (IME), Rio de Janeiro, Brazil, in 2003, the M.Sc. degree in telecommunications from University of Brasília (UnB), Brazil, in 2006, and the Ph.D. degree in electrical and information engineering from TU Ilmenau, Germany, in 2010. Since 2010, he has been coordinating with the Laboratory of Array Signal Processing, University of Brasília. From 2014 to 2017, he coordinated a special visiting researcher (PVE) project related to satellite navigation with the German Aerospace Center (DLR) supported by CAPES and CNPq, and since 2014, he has been a Project Coordinator in a distance learning project with the National School of Public Administration (Enap).

RAFAEL T. DE SOUSA, Jr. , was born in Campina Grande, Brazil, in 1961. He graduated in electrical engineering from the Federal University of Paraíba (UFPB), Campina Grande, Brazil, 1984, and the Ph.D. degree in telecommunications from the University of Rennes 1, Rennes, France, 1988. He was a Software and Network Engineer in the private sector, from 1989 to 1996. Since 1996, he has been a Network Engineering Professor with the Electrical Engineering Department, University of Brasília, Brazil. From 2006 to 2007, his work was supported by the Brazilian R&D Agency CNPq. He took a sabbatical year with the Group for the Security of Information Systems and Networks, Ecole Superiéure d'Electricité, Rennes. He is currently a member of the Post-Graduate Program on Electrical Engineering and supervises the Decision Technologies Laboratory (LATITUDE), University of Brasília. His research interests include distributed systems and network management and security.