Tensor space–time (TST) coding for MIMO wireless communication systems

Gérard Favier\textsuperscript{a,*}, Michele N. da Costa\textsuperscript{a,b}, André L.F. de Almeida\textsuperscript{c}, João Marcos T. Romano\textsuperscript{b}

\textsuperscript{a} I3S Laboratory, University of Nice-Sophia Antipolis, CNRS, France
\textsuperscript{b} DSPCom Laboratory, University of Campinas (UNICAMP), Brazil
\textsuperscript{c} GTEL-Wireless Telecom Research Group, Federal University of Ceará, Fortaleza, Brazil

\textbf{Abstract}

In this paper, we propose a tensor space–time (TST) coding for multiple-input multiple-output (MIMO) wireless communication systems. The originality of TST coding is that it allows spreading and multiplexing the transmitted symbols, belonging to \( K \) data streams, in both space (antennas) and time (chips and blocks) domains, owing the use of two (stream- and antenna-to-block) allocation matrices. This TST coding is defined in terms of a third-order code tensor admitting transmit antenna, data stream and chip as modes. Assuming flat Rayleigh fading propagation channels, the signals received by \( K \) receive antennas during \( P \) time blocks, composed of \( N \) symbol periods each, with \( J \) chips per symbol, form a fourth-order tensor that satisfies a new constrained tensor model, called a PARATUCK-(2,4) model. Conditions for identifiability and uniqueness of this model are established, and a performance analysis of TST coding is made, before presenting a blind receiver for joint channel estimation and symbol recovery. Finally, some simulation results are provided to evaluate the performance of this receiver.

\textcopyright 2011 Elsevier B.V. All rights reserved.

\section{1. Introduction}

Future cellular wireless communication (3G and 4G) systems must be designed to support a growing demand for high-quality multimedia services, with the best trade-offs between error performance (in terms of symbol or bit error rates, abbreviated as SER and BER, respectively), transmission rate (in symbols or bits per channel use), power efficiency, and receiver complexity for symbol recovery. A key idea for improving the error performance is to jointly exploit several diversities, which means redundancy into the information-bearing signals available at the receiver. This redundancy can be provided by the channels as it is the case with frequency-selective and time-selective channels, leading to what is called frequency (or multipath) diversity and Doppler diversity, respectively. Redundant information can also be obtained through spreading operations at the transmitter, in space, time and/or frequency domains [1–3].

Generally speaking, space diversity results from the use of multiple antennas at both transmitter and receiver ends, which leads to multiple-antenna communication systems with multiple-input multiple-output (MIMO) channels. As now well known, the deployment of multiple antennas in wireless systems allows improving the transmission rate (i.e. spectral efficiency) and reliability (i.e. error rate) over single-transmit antenna systems, while keeping the same transmission bandwidth and power [4–7].

Space spreading results from the use of several transmit antennas for transmitting the same symbol or data stream, whereas time spreading consists in repeating the same symbol multiplied by spreading codes, during several chip periods associated with each symbol. Time spreading can also be obtained by transmitting the same symbols or data streams over multiple blocks, each symbol period corresponding to a single channel use.
On the other hand, space multiplexing that consists in transmitting independent data streams in parallel on multiple transmit antennas, can be used for increasing the transmission rate [8,9,6].

Space–time (ST) coding is one of the most popular approaches relying on multi-antenna transmissions for achieving the fundamental tradeoff between error performance and transmission rate, also known as the diversity-multiplexing tradeoff [6,10]. There exists different ways for classifying ST code designs. These classifications are depending on:

- Type of MIMO channel (flat fading/frequency selective/time varying MIMO channels).
- Linearity/nonlinearity property of ST codes.
- Amount of channel state information (CSI) knowledge at the transmitter.
- Design criteria to be optimized (i.e. tradeoffs to be achieved).

Three main classes of ST coding schemes can be distinguished:

- ST trellis coding (STTC) with the pioneering work of Tarokh et al. [5].
- ST block coding (STBC) with the V-BLAST (vertical-Bell Labs layered space–time) transceiver architecture that exploits spatial multiplexing [8,9], and the orthogonal-STBC (O-STBC) design of Alamouti using two transmit antennas [11], extended to schemes with more than two transmit antennas in [12,13], and to quasi-orthogonal ST block coding (QO-STBC) [14,15], and rotated QO-STBC [16–18].
- Algebraic ST coding [19,20].

See [2,21] for an overview of matrix-based ST coding schemes.

During the last decade, several tensorial approaches have been developed for space–time MIMO wireless communication systems with matrix ST coding and blind receivers [22–29]. See [30] for a comparison of space–time spreading–multiplexing capacity of these different tensor-based approaches.

In [22], a ST coding based on the Khatri–Rao (KR) matrix product was proposed for achieving a variable rate-diversity tradeoff for any transmit–receive antenna configuration. This KRST coding combines a linear spatial pre-coding with a temporal-only spreading, and a joint channel and symbol detection is afforded thanks to the parallel temporal-only spreading, and a joint blind channel estimation. This product was proposed for achieving a variable rate-diversity tensor-based approaches.

More general ST codings including resource allocation were proposed in [25,26,30]. These ST codings lead to third-order constrained tensor models for the received signals, characterized by two or three allocation matrices that allow defining multiple-antenna transmission structures with varying degree of spatial spreading and multiplexing. In [25], two allocation matrices define the allocation of users’ data streams and spreading codes to their transmit antennas. The received signals satisfy a third-order constrained Tucker tensor model [34] whose core tensor is defined in terms of the two allocation matrices. In [26] that constitutes a generalization of [25], the transceiver is defined as a joint stream-code-antenna multiplexer that is decomposed by means of three allocation matrices. The received signals obey a third-order constrained PARAFAC model, the so-called CONFAC model. In [30], the received signals satisfy a PARATUCK-2 tensor model [35] whose interaction matrices are antenna-to-block and stream-to-block allocation matrices.

A common feature of all the tensorial approaches is to perform a joint blind symbol and channel estimation at the receiver, i.e. without a priori CSI knowledge at the receiver, which is a consequence of the essential uniqueness property of the underlying received signal tensor models.

A more detailed comparison of the above considered tensorial approaches is made in Section 3, and a unified presentation of the associated tensor models for the transmitted and received signals is given in Table 1.

In this paper, we propose a new tensorial approach that is based on a tensor space–time (TST) coding. This TST coding allows spreading and multiplexing the transmitted symbols, belonging to $R$ data streams, in both space (transmit antennas) and time (chips and blocks) domains, through the use of a third-order code tensor admitting transmit antenna, data stream and chip as modes, and two allocation matrices that allocate transmit antennas and data streams to each block. Assuming flat Rayleigh fading propagation channels, the signals received by $K$ receive antennas during $P$ blocks, composed of $N$ symbol periods each, with $J$ chips per symbol, form a fourth-order tensor that satisfies a new constrained tensor model, called a PARATUCK-(2,4) model. The proposed transmission system can be viewed as an extension of the ST transmission system of [30] that relies on a PARATUCK-2 tensor model for the received signals. This extension results from the introduction of a time-spreading code, which implies the transformation of the PARATUCK-2 model into a PARATUCK-(2,4) one. As for the PARATUCK-2 model in [30], the PARATUCK-(2,4) model allows a symbol recovery without CSI knowledge at the receiver. Indeed, a blind CSI-TST receiver using a two-step alternating least squares (ALS) algorithm can be derived for joint channel and symbol estimation.

The rest of the paper is organized as follows. After introducing a new constrained tensor model, the so-called PARATUCK-(N, $N$) model, Section 2 details the TST coding structure and presents the system tensor modeling. In Section 3, this TST-based modeling is compared with several existing tensor models. A performance analysis is carried
The scalar $\alpha$ (resp. $\beta$, $\gamma$, $\delta$) represents the $i$th row (resp. $j$th column) of $A$. The scalar $a_{i_1\ldots i_n}$ denotes the $(i_1, \ldots, i_N)$-th entry of $A$.

$D_1(A)$ is the diagonal matrix formed with the $i$th row of $A$; $I_N$ is the identity matrix of order $N$. $I_N$ is the all-one column vector of dimension $(N,1)$, and $\| \cdot \|_F$ is the Frobenius norm. The operator $\text{vec}(\cdot)$ forms a vector by stacking the columns of its matrix argument, whereas $\text{diag}(\cdot)$ forms a diagonal matrix from its vector argument.

The Kronecker and Khatri–Rao (column-wise Kronecker) products are denoted by $\otimes$ and $\odot$, respectively. We have the following properties:

$$
A \otimes C = [A_1 \otimes C_1 | \ldots | A_K \otimes C_K] = \begin{bmatrix}
CD_1(A) \\
\vdots \\
CD_K(A)
\end{bmatrix},
$$

$$
vec(BCA^T) = (A \otimes B)vec(C),
$$

$$
(A \otimes B)(E \otimes F) = (AE \otimes BF)
$$

for $A \in \mathbb{C}^{N \times N}$, $B \in \mathbb{C}^{K \times S}$ and $C \in \mathbb{C}^{S \times K}$, $E \in \mathbb{C}^{R \times K}$ and $F \in \mathbb{C}^{S \times L}$.

*In [37], $\tau$ means spectral efficiency instead of transmission rate.*

out in Section 4. Section 5 discusses the identifiability and uniqueness issues, and a blind receiver for joint channel and symbol estimation is formulated. In Section 6, some Monte Carlo simulation results are provided to evaluate the performance of this receiver under different parameter settings and to compare its performance with that of other tensor-based receivers. Finally, Section 7 draws some conclusions and perspectives for future work.

Notations: Scalars, column vectors, matrices and higher-order tensors are written as lower-case ($a, b, \ldots$), boldface lower-case ($\mathbf{a}, \mathbf{b}, \ldots$), boldface upper-case ($\mathbf{A}, \mathbf{B}, \ldots$), and blackboard ($\mathbb{A}, \mathbb{B}, \ldots$) letters, respectively. $\mathbf{A}^T$, $\mathbf{A}^H$, $\mathbf{A}^*$, and $\mathbf{A}^t$ stand for transpose, transconjugate (Hermitian transpose), complex conjugate, and Moore–Penrose pseudo-inverse of $\mathbf{A}$, respectively. The vector $\mathbf{a}_i$ (resp. $\mathbf{A}_i$) represents the $i$th row (resp. $j$th column) of $\mathbf{A}$. The scalar $a_{i_1 \ldots i_N}$ denotes the $(i_1, \ldots, i_N)$-th entry of $\mathbf{A}$. The Kronecker and Khatri–Rao (column-wise Kronecker) products are denoted by $\otimes$ and $\odot$, respectively. We have the following properties:

$$
\mathbf{A} \otimes \mathbf{C} = [\mathbf{A}_1 \otimes \mathbf{C}_1 | \ldots | \mathbf{A}_K \otimes \mathbf{C}_K] = \begin{bmatrix}
CD_1(\mathbf{A}) \\
\vdots \\
CD_K(\mathbf{A})
\end{bmatrix},
$$

$$
\text{vec}(\mathbf{B}\mathbf{C}\mathbf{A}^T) = (\mathbf{A} \otimes \mathbf{B})\text{vec}(\mathbf{C}),
$$

$$
(\mathbf{A} \otimes \mathbf{B})(\mathbf{E} \otimes \mathbf{F}) = (\mathbf{A}\mathbf{E} \otimes \mathbf{B}\mathbf{F})
$$

for $\mathbf{A} \in \mathbb{C}^{N \times N}$, $\mathbf{B} \in \mathbb{C}^{K \times S}$ and $\mathbf{C} \in \mathbb{C}^{S \times K}$, $\mathbf{E} \in \mathbb{C}^{R \times K}$ and $\mathbf{F} \in \mathbb{C}^{S \times L}$.

*In [37], $\tau$ means spectral efficiency instead of transmission rate.*

### 2. System tensor modeling

Before describing the proposed TST coding and the associated tensor modeling of transmitted and received signals, we present a new constrained tensor model that generalizes the well known PARATUCK-2 model [35,36].

#### 2.1. The PARATUCK-(N, N) model

Given an $N$th-order tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$, of dimensions $I_1 \times \ldots \times I_N$, with entries $x_{i_1 \ldots i_N} \in \mathbb{C}$ ($i_n = 1, 2, \ldots, I_n$ for $n=1,2,\ldots,N$), each index $i_n$ being associated with a mode,
and $I_n$ being the mode-n dimension, the PARATUCK-(N1, N) model of $\mathfrak{X}$, with $N > N_1$, is defined in the scalar form by the following expression:

$$X_{i_1,\ldots,i_N} = \sum_{r_1=1}^{R_1} \cdots \sum_{r_N=1}^{R_N} c_{r_1,\ldots,r_N,i_1,\ldots,i_N},$$

where $a_{i_r}$ and $\phi_{i_r}$ are the entries of the factor matrix $A^{(m)} \in \mathbb{C}^{I_r \times R_r}$ and the weighting (also called allocation) matrix $\Phi^{(m)} \in \mathbb{C}^{R_r \times I_r}$, $\forall n = 1, \ldots, N_1$, respectively.

This model can be interpreted as the transformation of the input tensor $X \in \mathbb{C}^{I_1 \times \cdots \times I_N}$ via its multiplication by the factor matrices $A^{(m)}$, $n = 1, \ldots, N_1$, along its first $N_1$ modes, combined with a mode-n input resource allocation ($n = 1, \ldots, N_1$) relatively to the mode-n ($N_1+1$) of the transformed tensor $\mathfrak{X}$.

**Special cases:**

- For $N_1=2$ and $N=3$, we obtain the standard PARATUCK-2 model introduced in [35]. Eq. (4) then becomes

$$X_{i_1, i_2, i_3} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} c_{r_1, r_2, i_1, i_2} a_{i_1, r_1} \phi_{i_2, r_2} \phi_{i_3, r_1},$$

where $a_{i_r}$ and $\phi_{i_r}$ are the entries of the factor matrix $A^{(m)} \in \mathbb{C}^{I_r \times R_r}$ and the weighting (also called allocation) matrix $\Phi^{(m)} \in \mathbb{C}^{R_r \times I_r}$, $\forall n = 1, \ldots, N_1$, respectively.

- For $N_1=2$ and $N=4$, we get the following PARATUCK-(2,4) model that is used in the sequel:

$$X_{i_1, i_2, i_3, i_4} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \sum_{r_3=1}^{R_3} \sum_{r_4=1}^{R_4} c_{r_1, r_2, r_3, r_4, i_1, i_2, i_3, i_4} a_{i_1, r_1} a_{i_2, r_2} \phi_{i_3, r_3} \phi_{i_4, r_4},$$

where $a_{i_r}$ and $\phi_{i_r}$ are the entries of the factor matrix $A^{(m)} \in \mathbb{C}^{I_r \times R_r}$ and the weighting (also called allocation) matrix $\Phi^{(m)} \in \mathbb{C}^{R_r \times I_r}$, $\forall n = 1, \ldots, N_1$, respectively.

2.2. **TST coding**

We consider a MIMO wireless communication system with $M$ transmit antennas and $K$ receive antennas, and we denote by $S_n$ the nth symbol of the nth data stream, each data stream ($r = 1, \ldots, R$) being composed of $N$ information symbols.

The transmission is assumed to be decomposed into $P$ data blocks, each block being formed of $N$ time slots. At each time slot $n$ of the $p$th block, the transceiver transmits a linear combination of the $n$th symbols of the data streams determined by the stream-to-block allocation matrix $\Psi \in \mathbb{C}^{P \times R}$ across a set of transmit antennas fixed by the antenna-to-block allocation matrix $\Phi \in \mathbb{C}^{P \times M}$. It is important to notice that, during each block $p$, a different set of data streams can be sent using a different set of transmit antennas, these two sets depending on the row vectors $\Psi_p$ and $\Phi_p$ of the two allocation matrices, respectively.

Each symbol $S_n$ is replicated several times after multiplication by a three-dimensional spreading code $w_{m,r,i}$ in such a way that the signal transmitted from the $m$th transmit antenna during the $n$th time slot of the $p$th block, and associated with the $i$th chip, is given by

$$u_{m,n,p,i} = \sum_{r=1}^{R} w_{m,r,i} S_n \phi_{p,m} \psi_{i,p},$$

with

$$g_{m,r,i,p} = w_{m,r,i} \phi_{p,m} \psi_{i,p}.$$  (8)

Eq. (7) defines the transceiver whose architecture is composed of three blocks carrying the following operations: data stream selection, TST coding, and transmit antenna allocation.

In Fig. 1, the functioning of this transceiver is illustrated for the $p$th block and the $i$th chip. The first and third black boxes ($D_p(\Psi)$ and $D_p(\Phi)$) select the data streams to be sent and the transmit antennas to be used for transmission during the $p$th block, respectively, whereas the second black box ($W_j$) spreads the selected data streams on the selected antennas to deliver the following matrix of coded signals:

$$U_{p,j} = G_{p,j} S_n T \in \mathbb{C}^{M \times N},$$

where $G_{p,j} \in \mathbb{C}^{M \times R}$ can be deduced from (8)

$$G_{p,j} = D_p(\Phi) W_j D_p(\Psi).$$

During the $j$th chip of the $n$th time slot of the $p$th block, the transceiver sends the following vector of coded signals:

$$U_{n,p,j} = G_{p,j} S_n T \in \mathbb{C}^{M \times 1}$$

with

$$S_n = [S_n, \ldots, S_n] \in \mathbb{C}^{1 \times R}.$$  (12)

Defining $\mu_p \in [1,R]$ and $\gamma_p \in [1,M]$ as the numbers of data streams and transmit antennas selected at the $p$th block, respectively:

$$\mu_p = \sum_{r=1}^{R} \psi_{r,p}, \quad \gamma_p = \sum_{m=1}^{M} \phi_{p,m}$$

the vector (11) of coded signals can be reduced in dimension as follows:

$$U_{n,p,j} = G_{p,j} S_n (\Psi) (\Phi) T \in \mathbb{C}^{M \times 1}$$

with $G_{p,j} \in \mathbb{C}^{\mu_p \times \gamma_p}$ and $S_n (\Psi) (\Phi) T \in \mathbb{C}^{M \times 1}$.

2.3. **Received signal tensor**

In the noiseless case and assuming flat Rayleigh fading propagation channels, the discrete-time baseband-equivalent model for the signal received at the $k$th receive antenna during the $j$th chip period of the $n$th symbol period of the $p$th block, is given by

$$x_{k,n,p,j} = \sum_{m=1}^{M} h_{k,m} u_{m,n,p,j},$$

where

$$x_{k,n,p,j} = \sum_{m=1}^{M} \sum_{r=1}^{R} w_{m,r,j} h_{k,m} S_n \phi_{p,m} \psi_{i,p},$$

Fig. 1. Transceiver based on TST coding with resource allocations.
The fading coefficients \( h_{m,n} \) between transmit antennas (m) and receive antennas (k) are assumed to be independent and identically distributed (i.i.d.) zero-mean complex Gaussian random variables. They are also assumed to be constant during at least P blocks.

In the noisy case, an additive complex-valued white Gaussian noise \( v_{k,n,p} \) is added to the right-hand side of (15).

Comparing (16) with (6), we can conclude that the fourth-order received signal tensor \( \mathbb{X} \in C^{K \times N \times P} \) satisfies a PARATUCK-(2, 4) model with the following correspondences:

\[(I_1, I_2, I_3, I_4, R_1, R_2) \leftrightarrow (K, N, P, J, M, R),\]

\[(C, A^{(1)}, A^{(2)}, \Phi^{(1)}, \Phi^{(2)}) \leftrightarrow (\mathbb{W}, H, S, \Phi^T, \Psi^T).\]

Let us define \( \mathbb{X}_{p,j} \in C^{K \times N} \) as the matrix slice of the received signal tensor \( \mathbb{X} \) obtained by slicing it along the plane \((p, j)\), i.e. by fixing the last two indices. Using (9) and (15) leads to the following factorization:

\[
\mathbb{X}_{p,j} = \mathbb{H} \mathbb{U}_{p,j} = \mathbb{H} \mathbb{G}_{p,j} \mathbb{S}^T
\]

with \( \mathbb{G}_{p,j} \) defined in (10).

The decomposition of the tensorial slice \( \mathbb{X}_{p,j} \in C^{K \times N \times J} \) of \( \mathbb{X} \), containing all the signals received during the \( p \)-th block and deduced from (18) and (10), is visualized in Fig. 2.

By stacking column-wise the set of matrix slices \( \{\mathbb{X}_{p,1,1}, \ldots, \mathbb{X}_{p,1,J}\} \) and \( \{\mathbb{X}_{p,1,1}^T, \ldots, \mathbb{X}_{p,1,J}^T\} \), and using (18), we can deduce, respectively, the following two matrix unfoldings of the received signal tensor \( \mathbb{X} \):

\[
\mathbb{X}_2 = \left[ \begin{array}{c} \mathbb{X}_{1,1}^T \\ \vdots \\ \mathbb{X}_{1,1}^T \\ \vdots \\ \mathbb{X}_{P,J}^T \end{array} \right] \in C^{P \times K \times N}, \quad \mathbb{X}_3 = \left[ \begin{array}{c} \mathbb{X}_{1,1}^T \\ \vdots \\ \mathbb{X}_{1,1}^T \\ \vdots \\ \mathbb{X}_{P,J}^T \end{array} \right] \in C^{N \times P \times K}
\]

\[
= (I_p \otimes H) \mathbb{G}_2 \mathbb{S}^T = (I_p \otimes S) \mathbb{G}_3 \mathbb{H}^T
\]

with

\[
\mathbb{G}_2 = \left[ \begin{array}{c} \mathbf{G}_{1,1} \\ \vdots \\ \mathbf{G}_{P,1} \\ \vdots \\ \mathbf{G}_{1,J} \\ \vdots \\ \mathbf{G}_{P,J} \end{array} \right] \in C^{J \times P \times R}, \quad \mathbb{G}_3 = \left[ \begin{array}{c} \mathbf{G}_{1,1}^T \\ \vdots \\ \mathbf{G}_{P,1}^T \\ \vdots \\ \mathbf{G}_{1,J}^T \\ \vdots \\ \mathbf{G}_{P,J}^T \end{array} \right] \in C^{R \times J \times M}.
\]

Applying property (2) to Eq. (18) and (10) gives

\[
\text{vec}(\mathbb{X}_{p,j}) = (\mathbb{S} \otimes \mathbb{H}) \text{vec}(\mathbb{G}_{p,j})
\]

and

\[
\text{vec}(\mathbb{G}_{p,j}) = (D_p(\Psi) \otimes D_p(\Phi)) \text{vec}(\mathbb{W}_{j,p}) = \text{diag}(\text{vec}(\mathbb{W}_{j,p}))(\Psi_p^T \otimes \Phi_p^T).
\]

Replacing \( \text{vec}(\mathbb{G}_{p,j}) \) by its expression (22) into (21) allows rewriting \( \text{vec}(\mathbb{X}_{p,j}) \) as

\[
\text{vec}(\mathbb{X}_{p,j}) = (\mathbb{S} \otimes \mathbb{H}) \text{diag}(\text{vec}(\mathbb{W}_{j,p}))(\Psi_p^T \otimes \Phi_p^T).
\]

Using (23), we can build a third matrix unfolding of \( \mathbb{X} \) as

\[
\mathbb{X}_1 = \{\text{ vec}(\mathbb{X}_{1,1}) \cdot \ldots \cdot \text{ vec}(\mathbb{X}_{1,1}) \cdot \ldots \cdot \text{ vec}(\mathbb{X}_{1,J}) \cdot \ldots \cdot \text{ vec}(\mathbb{X}_{P,J})\}
\]

in \( C^{NK \times J} = (\mathbb{S} \otimes \mathbb{H}) \mathbb{G}_1 \).

\[
\mathbb{G}_1 = \{\text{ vec}(\mathbb{G}_{1,1}) \cdot \ldots \cdot \text{ vec}(\mathbb{G}_{P,1}) \cdot \ldots \cdot \text{ vec}(\mathbb{G}_{1,J}) \cdot \ldots \cdot \text{ vec}(\mathbb{G}_{P,J})\}
\]

in \( C^{RM \times J} \).

Defining the following matrix unfolding of the code tensor \( \mathbb{W} \in C^{M \times R} \):

\[
\mathbb{W}_1 = \{\text{ vec}(\mathbb{W}_{1,1}) \cdot \ldots \cdot \text{ vec}(\mathbb{W}_{1,J})\} \in C^{R \times J}.
\]

\( \mathbb{G}_1 \) can be also written as

\[
\mathbb{G}_1 = [D_1(\mathbb{W}_1^T)(\Psi^T \otimes \Phi^T) \cdot \ldots \cdot D_J(\mathbb{W}_1^T)(\Psi^T \otimes \Phi^T)].
\]

Using (1) gives

\[
\mathbb{G}_1 = [\mathbb{W}_1^T (\Psi^T \otimes \Phi^T)]^T.
\]

Fig. 2. Visualization of the tensor slice \( \mathbb{X}_{p,j} \) of the PARATUCK-(2, 4) model.
In the sequel, the matrix representations $X_2$ and $X_3$ defined in (19) will be exploited for deriving a blind joint symbol and channel estimation algorithm that is based on the ALS technique, whereas $X_1$ will be used for studying the uniqueness of the proposed PARATUCK-(2,4) model of the received signal tensor.

3. Comparison with existing tensor models

Two existing tensor models can be directly deduced as particular cases of the proposed TST coding system:

- **PARATUCK-2 model [30]:** When $J=1$, (7) becomes independent of $j$:
  \[
  U_{m,n,p} = \sum_{r=1}^{R} W_{m,n,r} S_{n,r} \phi_{p,m} \psi_{p,r} = \sum_{r=1}^{R} g_{m,r,p} S_{n,r}
  \]
  (30)
  with
  \[
  g_{m,r,p} = W_{m,r} \phi_{p,m} \psi_{p,r},
  \]
  (31)
  which corresponds to the ST transmission system proposed in [30].

- **PARAFAC model [37]:** If we set $R=P=1$, and replace $S_{n,r}$ by $S_{n,m}$, and $W_{n,t,j}$ by $C_{j,m}$, with the constraints $\phi_{1,m} = \psi_{1,1} = 1$, $\forall m = 1, \ldots, M$, (7) and (16) become
  \[
  U_{m,n,j} = S_{n,m} C_{j,m}.
  \]
  (32)

  \[
  x_{k,n,j} = \sum_{m=1}^{M} h_{k,m} U_{m,n,j} = \sum_{m=1}^{M} h_{k,m} S_{n,m} C_{j,m},
  \]
  (33)
  which corresponds to the PARAFAC model of the DS-CDMA (direct-sequence code-division multiple access) system of [37], with matrix factors $(H, S, C)$. This simplest case allows space and time spreading but no multiplexing.

Now, we compare the proposed PARATUCK-(2,4) model with the PARAFAC model obtained with a Khatri–Rao space–time (KRST) coding.

- **PARAFAC model with KRST coding (PARAFAC-KRST) [22]:** In this case only one data stream, composed of $M$ symbols, is transmitted from $M$ transmit antennas, during each time block $t$ of $J$ slots, using two coding matrices $W \in \mathbb{C}^{M \times M}$ and $C \in \mathbb{C}^{J \times M}$. The first one allows to combine $M$ symbols onto each transmit antenna, for a given block $t$, which gives the pre-coded signal $v_{t,m} = \sum_{l=1}^{M} W_{m,l} s_{t,l}$, whereas the aim of the second one is to spread such a combination transmitted by each antenna over $J$ slots, which provides a third-order tensor for the transmitted signals defined as $u_{m,t,j} = v_{t,m} C_{j,m}$. Assuming that the channels are constant for $T$ blocks, the tensor of signals received by $K$ receive antennas, during $T$ blocks composed of $J$ slots each, satisfies the following third-order PARAFAC model,

\[
\begin{aligned}
& x_{k,t,j} = \sum_{m=1}^{M} h_{k,m} v_{t,m} C_{j,m} \\
& \quad \text{for } t=1, \ldots, T, \quad j=1, \ldots, J,
\end{aligned}
\]

(34)

Comparing the PARAFAC-KRST approach with the PARAFAC one for which the matrix factors are $(H, S, C)$ and the symbol matrix is directly estimated, jointly with the channel matrix, using the same two-step ALS method, we conclude that the introduction of space spreading by means of pre-coding induces the need of a supplementary decoding step.

On the other hand, the PARATUCK-2 and PARATUCK-(2,4) approaches are more general than the PARAFAC-KRST one in the sense that they provide different degrees of space and time spreading and multiplexing depending on the choice of the allocation matrices $\Phi$ and $\Psi$, as illustrated in [30]. Comparing (35) and (36) with (11) and (12), we can deduce the following correspondences between the TST and KRST codings:

\[
(R,P,N,J) \leftrightarrow (M,T,J-1)
\]

(37)

showing that, for KRST, the number of data streams is forced to be equal to the number of transmit antennas, while it can be chosen to be equal to $R \geq M$ with TST. In addition, the use of tensor coding instead of matrix-based pre- and post-coding presents the advantages of an extra time spreading on chip and not needing decoding at the receiver, the channel and symbol matrices being blindly and jointly estimated by means of the two-step ALS method, as shown in Section 5.2.

The proposed TST coding can be viewed as a combination of the spreading code of the PARAFAC approach with the two dimensional code of the PARATUCK-2 approach, under the form of a three dimensional code tensor. Assuming the codes and the allocation matrices known at the receiver, it is important to notice that TST coding allows introducing one extra time diversity with the same receiver complexity than the PARATUCK-2 approach.

An unified presentation of the three above considered tensorial approaches, the new PARATUCK-(2,4) model based solution and the tensor solutions of [33,25,26], is given in Table 1 where we give the reference numbers, the types of tensor model for the received signals, the expressions of the transmitted and received signal tensors, and the ratio $\tau$ such as the transmission rate (in bits per
channel use) is given by
\[ V = \tau \log_2(\mu), \]
where \( \mu \) is the cardinality of the information symbol constellation.

It is interesting to compare the diversities and the resource allocations of the PARATUCK-(2,4) based solution with those taken into account by the tensor approaches of [25,26,30,33]. In [33], a spatial multiplexing and spreading is combined with a temporal spreading thanks to a 3D coding. Three diversities associated with the dimensions of the received signal tensor \( X \in C^{K \times N \times N} \) are exploited: space (\( K \) receive antennas), time (\( N \) time slots), and code (code length \( J \)). In [25,26], the same diversities as in [33] are exploited using a linear combination of \( Q \) different spreading codes with an allocation of data streams and spreading codes to transmit antennas by means of two or three allocation matrices, respectively. In [30], no spreading code is used, and two time diversities (\( P \) blocks of \( N \) time slots each) and one space diversity (\( K \) receive antennas) are considered in the tensor \( X \in C^{K \times N \times N} \). While spatial and temporal allocations are provided by antenna-to-slot and stream-to-slot allocation matrices. As already mentioned, the solution allows the same resource allocation as in [30], the improvement resulting from an additional diversity associated with the spreading code of length \( J \) that corresponds to the third dimension of the code tensor \( W \in C^{M \times K \times J} \) and the fourth dimension of the received signal tensor \( X \in C^{K \times N \times N \times P \times J} \).

4. Performance analysis

In this section, we analyze the performance of TST coding and derive the maximum diversity gain over a flat fading channel. This performance analysis assumes that the channel matrix \( H \) has independent entries following a circular symmetric complex Gaussian random distribution, i.e. \( h_{k,m} \sim CN(0,1) \), or equivalently its real and imaginary components are i.i.d. and distributed as \( \mathcal{N}(0,1/2) \), which corresponds to the assumption of flat Rayleigh fading. We also assume that the receiver has perfect knowledge of \( H \) and of the TST coding parameter set \( (W,\Psi,\Phi) \).

For matrix ST coding, the transmitted ST code matrix, also called ST codeword, is defined as the matrix associated with the coding mapping:
\[ s_n \in C^{M \times T} \rightarrow C_n \in C^{M \times T}, \]

where \( M \) and \( T \) denote the space and time spreading lengths. In the case of the proposed tensor ST coding, the ST codeword is the fourth-order tensor associated with the coding mapping:
\[ S_T \in C^{K \times N \times N \times P \times J} \]

whose dimensions are the lengths of space, time, block and chip spreadsings.

As already mentioned, matrix ST coding approaches are based on codeword estimation, followed by a decoding step for estimating the transmitted symbols. So, the performance analysis is generally based on the pairwise error probability (PEP) of the maximum likelihood (ML) estimator of the codeword matrix, defined as the probability that the ML estimator estimates \( \hat{C}_n \) when \( C_n \) is actually sent. In the case of TST coding, it is possible to directly estimate the symbol matrix instead of the codeword tensor, as shown in Section 5.2, which explains why our performance analysis is based on the PEP of the ML estimator of \( S \) instead of \( U \).

The diversity gain \( d \) is defined as the negative of the asymptotic slope of the plot PEP(\( \rho \)) on a log-log scale, where \( \rho \) denotes the received signal-to-noise ratio (SNR), and PEP is hereafter the probability that the ML estimator estimates \( \hat{S} \) when \( S \) is actually transmitted.

In the sequel, we first determine the function PEP(\( \rho \)), and then we deduce the diversity gain for TST coding.

The conditional PEP between \( S \) and \( \hat{S} \) can be approximated by [5,38]
\[ P(S \rightarrow \hat{S} | H) = Q \left( \frac{1}{2 \sqrt{N_0/2}} \|X - \hat{X}\|_F \right), \]

where \( N_0/2 \) is the noise variance per (real and imaginary) dimension and \( Q(\cdot) \) is the complementary cumulative distribution function of a Gaussian variable.

Defining the difference between the matrix slices \( (p,j) \) of the codeword tensors \( U \) and \( \hat{U} \) as
\[ E^{(p,j)} = U_{p,j} - \hat{U}_{p,j} \in C^{M \times N}, \]

and using (15) and (18), we have
\[ \|X - \hat{X}\|_F^2 = \sum_{k=1}^{K} \sum_{n=1}^{N} \sum_{l=1}^{L} \sum_{p=1}^{P} \sum_{j=1}^{J} |X_{k,n,l,p,j} - \hat{X}_{k,n,l,p,j}|^2 \]

\[ = \sum_{p=1}^{P} \sum_{j=1}^{J} \|HE^{(p,j)}\|_F^2 = \sum_{p=1}^{P} \sum_{j=1}^{J} \text{tr}(HA^{(p,j)}H^H), \]

where \( A^{(p,j)} = E^{(p,j)}(E^{(p,j)}H)^H \) is Hermitian and nonnegative definite, so that its eigenvalues are nonnegative real numbers, and it can be diagonalized by an unitary matrix, such as
\[ A^{(p,j)} = V^{(p,j)} \text{diag}(\lambda^{(p,j)}_1, \ldots, \lambda^{(p,j)}_r) (V^{(p,j)}H)^H, \]

\( r^{(p,j)} \) and \( \lambda^{(p,j)}_m, m = 1, \ldots, r^{(p,j)} \), denoting the rank, and the nonzero eigenvalues of \( A^{(p,j)} \), respectively, and the columns of \( V^{(p,j)} \in C^{M \times r^{(p,j)}} \) being the orthonormal eigenvectors of \( A^{(p,j)} \) associated with its nonzero eigenvalues, for \( p = 1, \ldots, P, j = 1, \ldots, J \).

Defining the transformation
\[ B^{(p,j)} = HV^{(p,j)}, \]

the expression (42) of the conditional PEP can be rewritten as
\[ P(S \rightarrow \hat{S} | H) = Q \left( \frac{1}{2 \sqrt{N_0}} \sum_{p=1}^{P} \sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{m=1}^{r^{(p,j)}} \lambda^{(p,j)}_m |a_{k,m}^{(p,j)}|^2 \right). \]

Using the Chernoff bound, it can be upper bounded by [38]
\[ P(S \rightarrow \hat{S} | H) \leq \exp \left( -\frac{1}{4N_0} \sum_{p=1}^{P} \sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{m=1}^{r^{(p,j)}} \lambda^{(p,j)}_m |a_{k,m}^{(p,j)}|^2 \right). \]
The unconditional PEP upper bound is obtained by statistically averaging (47) over the channel coefficients \( h_{k,m} \), or equivalently over the modulus \( |b_{k,m}^{p,j}| \). Due to the linear transformation (45) and the orthonormality property of the columns of \( V^{p,j} \), the coefficients \( b_{k,m}^{p,j} \) have the same distribution \( \mathcal{C}\mathcal{N}(0,1) \) as the elements of \( H \) and therefore their modulus \( |b_{k,m}^{p,j}| \) is Rayleigh distributed (\( |b_{k,m}^{p,j}| \sim \text{Rayleigh}(0, \sqrt{1/2}) \)), which gives [2,5]

\[
P(S \rightarrow \hat{S}) \leq \prod_{p=1}^{P} \prod_{j=1}^{J} \prod_{m=1}^{M} \left( 1 + \frac{r_{m}^{p,j}}{4N_0} \right)^{-K}. 
\]

(48)

At high SNR, i.e. for small values of \( N_0 \), the above upper bound on the PEP becomes

\[
P(S \rightarrow \hat{S}) \leq \prod_{p=1}^{P} \prod_{j=1}^{J} \prod_{m=1}^{M} \left( \frac{r_{m}^{p,j}}{4N_0} \right)^{-K}. 
\]

(49)

which gives the following diversity gain:

\[
d = K \sum_{j=1}^{J} \sum_{p=1}^{P} r^{p,j}.
\]

(50)

Recalling that \( A^{p,j} = E^{p,j}(E^{p,j})^H \), we have

\[
r^{p,j} = \text{rank}(A^{p,j}) - \text{rank}(E^{p,j}).
\]

(51)

Using (9), the difference (42) of the codeword matrix slices can be rewritten as

\[
E^{p,j} = G_{p,j}(S \rightarrow \hat{S}).
\]

(52)

Applying the well-known property \( \text{rank}(AB) \leq \min(\text{rank}(A), \text{rank}(B)) \) for arbitrary matrices \( A \) and \( B \), and using the fact that \( S \) must be full column-rank, which implies \( N \geq R \), for uniqueness of the LS estimate of \( H \) from (19), we deduce that

\[
r^{p,j} \leq \min(\text{rank}(G_{p,j}), \text{rank}(S \rightarrow \hat{S}))
\]

(53)

\[
r^{p,j} \leq \min(\text{min}(M,R), \text{min}(N,R))
\]

(54)

\[
r^{p,j} \leq \text{min}(M,R), \quad \forall p,j,
\]

(55)

which leads to the maximum diversity gain given in Theorem 1.

**Theorem 1.** Assuming that \( S \) is full column-rank, with \( S \neq \hat{S} \), and that the code tensor \( \Psi \) and the allocation matrices \( \Phi \) and \( \Psi \) are chosen such as all the matrices \( E^{p,j} \) are full rank, then the TST coding characterized by the design parameter set \( \{P,J,M,R\} \) provides a maximum diversity gain equal to

\[
d_{\text{max}} = KJP \min(M,R).
\]

(56)

This expression of the maximum diversity gain is deduced from (50) in replacing \( r^{p,j} \) by its upper bound given in (55) for all the values of \( p \) and \( j \).

So, we can conclude that the TST coding provides a better diversity than standard matrix ST coding schemes that ensure a maximum diversity gain of \( KM \). Moreover, for fixed numbers \( (K \) and \( M \)) of receive and transmit antennas, the maximum diversity gain can be increased by independently increasing the design parameters \( P, J \) and \( R \) (up to \( R = M \)). However, we have to recall that an increase of \( P \) decreases the transmission rate, while an increase of \( R \) increases the transmission rate. Moreover, for a fixed \( R \), i.e. a fixed number of symbols to be estimated, an increase of \( P \) or \( J \) implies an increase of the number of received signals to be used for channel and symbol estimation, and thus an improvement of the estimation quality, while for fixed \( P \) and \( J \), an increase of \( R \) implies an increase of the number of parameters (symbols) to be estimated, which degrades the quality of estimation. From these considerations, we see that the design parameters \( \{P, J, M, R\} \) must be chosen in such a way that the best tradeoff between transmission rate and BER performance be satisfied. This tradeoff will be confirmed by means of simulation results in Section 6.

5. **Blind receiver**

In this section, the algebraic structure of the PARATUCK-(2,4) model of the received signal tensor is exploited for studying identifiability and uniqueness issues as well as for deriving a blind joint symbol and channel estimation algorithm based on the ALS technique.

5.1. **Identifiability and uniqueness issues**

5.1.1. **Structure of the third-order code tensor** \( \Psi_{JPM} \in \mathbb{C}^{M \times R \times J} \)

For the code tensor, we choose a third-order Vandermonde tensor the elements of which are given by

\[
w_{m,r,j} = e^{j2\pi n m j |M|}. 
\]

(57)

where \( j^2 = -1 \). An important reason behind this choice is that the Vandermonde structure guarantees the existence of a minimum value of the spreading length \( J \) ensuring the identifiability in the LS sense of the channel \( (H) \) and symbol \( (S) \) matrices, as shown in Theorem 3.

5.1.2. **Identifiability**

Each matrix \( S \) and \( H \) is estimated by alternately solving (19) in the LS sense with respect to one matrix conditionally to the knowledge of previously estimated value of the other matrix. Assuming that the symbol and channel matrices are full column-rank, which implies \( N \geq R \) and \( K \geq M \), uniqueness of their conditional LS estimates requires that \( G_2 \in C^{PM \times R} \) and \( G_3 \in C^{PR \times M} \) be also full column-rank. From this double condition, we deduce the following theorem.

**Theorem 2.** Assuming that \( S \) and \( H \) are full column-rank, a necessary condition for LS identifiability is given by

\[
P_{J} \geq \max \left( \left\lfloor \frac{R}{M} \right\rfloor \right).
\]

(58)

where \( \lfloor x \rfloor \) denotes the smallest integer number greater than or equal to \( x \).

**Proof.** Let us rewrite both Eq. (19) as \( X_2 = Z_2 S^T \) and \( X_3 = Z_3 H^T \), where \( Z_2 = (I_{P} \otimes H) G_2 \) and \( Z_3 = (I_{P} \otimes S) G_3 \). Uniqueness of the LS solution for \( S \) and \( H \) requires that \( Z_2 \) and \( Z_3 \) be full column-rank. Assuming that \( S \) and \( H \) are full column-rank implies that \( I_P \otimes S \) and \( I_P \otimes H \) are also
full column-rank. Consequently, \( \text{rank}(Z_2) = \text{rank}(G_2) \) and \( \text{rank}(Z_3) = \text{rank}(G_3) \), which means that \( G_2 \) and \( G_3 \) must be full column-rank to ensure the identifiability of \( S \) and \( H \), implying \( JPM \geq R \) and \( JPR \geq M \), or equivalently \( (58) \). \( \square \)

This condition \( (58) \) defines a constraint that the design parameters \((P, J, M, R)\) must satisfy. It is interesting to notice that the supplementary diversity introduced by the time-spreading mode \((j)\) of the code tensor allows us to get a more relaxed condition on the number \( P \) of data blocks than the one obtained in [30].

**Theorem 3.** Assuming that \( S \) and \( H \) are full column-rank, with the Vandermonde structure \((57)\) for the code tensor, and \((\Phi_p = 1^{M}_{M}, \Psi_p = 1^{R}_{R})\) for a given \( p \in \{1, \ldots, P\} \), Table 2 gives the minimum value of the spreading length ensuring the LS identifiability of \( S \) and \( H \), for \( M \) and \( R \) in \( \{1, \ldots, 8\} \):

**Proof.** For \( \Phi_p = 1^{M}_{M} \) and \( \Psi_p = 1^{R}_{R} \), \((10)\) gives \( G_{p,j} = W_{j} \) and then from \((20)\) we can extract the following two sub-matrices associated with the block \( p \):

\[
F_2 = \begin{bmatrix}
W_1 \\
\vdots \\
W_j
\end{bmatrix} \in C^{M \times R}, \quad F_3 = \begin{bmatrix}
W^T_1 \\
\vdots \\
W^T_j
\end{bmatrix} \in C^{R \times M}.
\]

(59)

These two matrices contain truncated Vandermonde sub-matrices. First, we have to notice the symmetric role played by \( M \) and \( R \), with the block \( W_{j} \) given by

\[
W_j = \begin{bmatrix}
w^1 w^2 \cdots w^R \\
w^{j_1} w^{j_2} \cdots w^{j_R} \\
\vdots \\
w^{Mj_1} w^{Mj_2} \cdots w^{Mj_R}
\end{bmatrix},
\]

(60)

where \( w = e^{2\pi i/Mj} \). When \( M = R \), \( W_1 \) is non-singular, which implies that \( G_2 \) and \( G_3 \) are full column-rank, and therefore the LS estimate of \( S \) and \( H \) is unique with \( J_{\text{min}} = 1 \), which corresponds to Theorem 3 in [30]. In the case where \( M > R \), the block \( W_1 \) and therefore \( G_2 \) are full column-rank, whereas \( W_1^T \) is full row-rank equal to \( R \). The number \( J_{\text{min}} \) of blocks \( W_{j} \) to be considered in \( F_3 \) to guaranty its full column rank property is given in Table 2 for \( M, R \in \{1, \ldots, 8\} \). When \( R > M \), the same reasoning can be applied to determine the minimum number \( J_{\text{min}} \) of blocks \( W_{j} \) to be considered in \( F_2 \) for guarantying its full column rank property, which explains the symmetric form of Table 2. \( \square \)

**Remark 1.**

- The constraints \( \Psi_p = 1^{R}_{R} \) and \( \Phi_p = 1^{M}_{M} \) mean that all the \( R \) data streams are sent using all the \( M \) transmit antennas, during the time block \( p \).
- It is important to notice that Theorem 3 gives a sufficient condition in terms of a minimum value \( J_{\text{min}} \) for uniqueness of the LS estimate of \( S \) and \( H \), independently of the value of \( P \geq 1 \), while Theorem 2 establishes a necessary (but not necessarily sufficient) condition for LS identifiability.
- Introducing the time-spreading mode in the Vandermonde code tensor allows us to derive a minimum value of the spreading length \( J \) that ensures the identifiability of \( S \) and \( H \) in the case \( M \neq R \), which is not possible in [30] with \( J = 1 \).

5.1.3. Uniqueness

We first recall the following lemma:

**Lemma 1** (Sidiropoulos et al. [39]). Suppose that \( A \in C^s \times F \) and \( B \in C^t \times F \) are such that neither \( A \) nor \( B \) has a zero column. Then \( A \odot B \) is full column-rank if \( \text{rank}(A) \geq \text{rank}(B) \geq F + 1 \).

**Proof.** According to Lemma 1, if \( k_A + k_B \geq F + 1 \), then \( A \odot B \) is full column rank. Note that the \( k \)-rank is more constraining than the column-rank in the sense that \( k_A \) is the number of any set of \( k \) columns of \( A \) is independent, whereas \( r(A) = k \) implies that there exists at least one set of \( k \) independent columns. Thus, we have: \( k_A \leq \text{rank}(A) \). It follows that \( \text{rank}(A) + \text{rank}(B) \geq k_A + k_B \geq F + 1 \). \( \square \)

Application of Lemma 1 to \((29)\) allows deducing that the matrix unfolding \((25)\) of \( G \), i.e. \( G_1 \), is full row-rank if \( W_i, \Phi \) and \( \Psi \) satisfy the following condition:

\[
\text{rank}(W_i) + \text{rank}(\Psi^T \odot \Phi^T) \geq RM + 1.
\]

(61)

Considering \( \hat{S} \) and \( \hat{H} \) as alternative solutions that satisfy \((24)\) and assuming \( S = SU \) and \( H = HV \), with \( U \in C^{s \times s} \) and \( V \in C^{t \times t} \) non-singular, uniqueness of the PARATUCK-(2,4) model can be proved in applying property \((3)\) to the expression \((24)\) of the matrix unfolding \( X_1 \) that can be rewritten as

\[
X_1 = (S \odot H)G_1 = (\hat{S} \odot \hat{H})(U \odot V)G_1.
\]

(62)

Therefore, it is necessary to remove any ambiguity caused by the nonsingular transformation matrices \( U \) and \( V \) to recover both matrices \( S \) and \( H \).

**Theorem 4.** If the condition \((61)\) is satisfied, i.e. \( G_1 \) is full row-rank, then \( S \) and \( H \) are unique up to a scalar factor, i.e.

\[
S = \alpha \hat{S}, \quad H = \frac{1}{\alpha} \hat{H}
\]

(63)

and the scaling ambiguity factor \( \alpha \) can be eliminated by simply transmitting a known symbol \( s_{1,1} \).

\( \square \)

\( \text{Note that } A \odot B \text{ full column-rank implies that } A \odot B \text{ is tall, i.e. } ll \geq F. \)
Proof. Note that if $G_1$ is full row-rank, i.e. if the condition (61) is satisfied, uniqueness of $S$ and $H$ implies from (62) that:

$$U \otimes V = G_1 G_1^T = I_{RM}.$$  

(64)

The only possible solution for $U \otimes V = I_{RM}$ happens when both $U$ and $V$ are identity matrices up to scalar factors that compensate each other, i.e. $U = 2I_K$ and $V = (1/2)I_M$, which leads to (63). □

Thus, as in [30] for the PARATUCK-2 model, uniqueness of the PARATUCK-(2,4) model is ensured under some condition to be satisfied by the design parameters. However, as for Theorem 2, the supplementary diversity induced by the time-spreading mode $(j)$ of the code tensor provides more flexibility in the choice of allocation matrices comparatively with the solution of [30].

Remark 2. Taking the Vandermonde form of the code tensor into account implies that each column-block in (26), is such that $	ext{rank}(\text{diag} (\text{vec}(W_1^{(j)})(\Phi^T \otimes \Phi^T)) = \text{rank}(\Phi^T \otimes \Phi^T)$, and consequently if $\Phi^T \otimes \Phi^T$ is full row-rank, which implies $P \geq RM$, then $G_1$ is also full row-rank, and the uniqueness is ensured. However, we have to notice that this condition, identical to that of Theorem 3 in [30], is more restrictive than (61). This last constraint can be satisfied owing an appropriate choice of the code tensor and allocation matrices, i.e. the set of transceiver parameters $(W_1, \Phi, \Psi)$. We have to mention that the unfolded matrix $W_1$ of the Vandermonde code tensor is full row-rank if and only if $j=RM$, which occurs only for $M=1$ or $R=1$.

5.2. Blind joint symbol and channel estimation algorithm

Assuming that the code tensor $C$ and the allocation matrices $\Phi$ and $\Psi$ are known at the receiver, the matrices $G_2$ and $G_3$ can be pre-calculated.

Application of the ALS technique to (19) gives the blind joint symbol and channel estimation algorithm summarized in Table 3, where $\tilde{X}_2 = X_2 + V_2$ and $\tilde{X}_3 = X_3 + V_3$ are noisy versions of $X_2$ and $X_3$, respectively, the noise being assumed to be additive zero-mean complex-valued white Gaussian.

It is important to emphasize that the receiver complexity mainly results from the computation of the pseudo-inverses of $(I_P \otimes \hat{H}_{(it-1)})G_2 \in \mathbb{C}^{JR \times K}$ and $(I_P \otimes S_{(it)}^H)G_3 \in \mathbb{C}^{JR \times M}$ at each ALS iteration. This complexity is directly linked to the system parameters $(J, P, N, K, M, R)$.

Table 3

<table>
<thead>
<tr>
<th>ALS algorithm for blind joint symbol and channel estimation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Initialization ($it=0$): randomly initialize $\tilde{H}_0$.</td>
</tr>
<tr>
<td>2. $it = it + 1$</td>
</tr>
<tr>
<td>3. Calculate an LS estimate of $S$ from (19): $S_{(it)}^T = (I_P \otimes \tilde{H}_{(it-1)})G_2 \tilde{X}_2$.</td>
</tr>
<tr>
<td>4. Calculate an LS estimate of $H$ from (19): $H_{(it)} = (I_P \otimes S_{(it)}^H)G_3 \tilde{X}_1$.</td>
</tr>
<tr>
<td>5. Repeat steps (2)–(4) until convergence.</td>
</tr>
<tr>
<td>6. Eliminate the scaling ambiguity using (64) with $\alpha = s_{11}/\tilde{s}<em>{11}(\infty)$, where $\tilde{s}</em>{11}(\infty)$ is the estimated value of $s_{11}$ at convergence.</td>
</tr>
<tr>
<td>7. Project the estimated symbols onto the symbol alphabet.</td>
</tr>
</tbody>
</table>

Note that this complexity is the same as the one of the approach in [30] when $J=1$, but it increases when $J > 1$.

The convergence of the algorithm is decided when the errors between the noisy received signal tensor and its values reconstructed using the channel and symbol matrices estimated at two successive iterations are such as

$$\frac{|e_{(it+1)} - e_{(it)}|}{e_{(it)}} \leq 10^{-6},$$

where $e_{(it)} = \| \tilde{X}_2 - (I_P \otimes \hat{H}_{(it)})G_2 S_{(it)}^T \|^2_F$.

6. Simulation results

The performance of the proposed TST coding and the associated ALS-based blind receiver is evaluated by means of Monte Carlo simulations, in terms of BER and normalized mean square error (NMSE) on channel estimation, defined as

$$\text{NMSE}_{db} = 10 \log_{10} \left( \frac{1}{L} \sum_{l=1}^{L} \frac{\| H - \hat{H}_{(\infty)} \|^2_F}{\| H \|^2_F} \right),$$

(65)

where $\hat{H}_{(\infty)}$ is the channel matrix estimated at convergence of the $l$th run, and $L=2000$ is the total number of Monte Carlo runs corresponding to 2000 random wireless channels, with different symbol sequences randomly drawn from a QPSK constellation, and different random noise sequences, for each simulated channel. A different random initialization $H_{(0)}$ is also used for each run. The BER is calculated by averaging the results obtained for the $R$ data streams and the $L$ Monte Carlo runs. The signal-to-noise ratio (SNR) is determined by

$$\text{SNR} = 10 \log_{10} \left( \frac{\| \tilde{X}_2 \|^2_F}{\| V_2 \|^2_F} \right).$$

(66)

The default values of the tuning parameters are chosen as follows: $R=2$, $N=10$, $J=3$, $P=4$, $K=M=2$, and

$$\Phi_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad \Psi_4 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$  

(67)

which implies a transmission rate of 1 bit per channel use.

In the sequel, we study the influence of the spreading code length ($J$), and of the numbers of blocks ($P$), data streams ($R$) and receive antennas ($K$). Then, we compare the proposed TST coding with the KRST coding [22], and a comparison is also made with the zero-forcing (ZF) receiver assuming a perfect knowledge of the channel matrix.

6.1. Influence of the spreading code length

Figs. 3–5 show the channel NMSE, the BER and the iteration number needed for convergence, versus SNR, for four values of the spreading code length ($J \in \{1, 3, 6, 10\}$), respectively. From Figs. 3 and 4, we can conclude that an increase of $J$ induces a significant performance improvement in terms of both channel estimation and symbol recovery.
Thus, the BER is canceled for a \( \text{SNR} = [12; 6; 3; 2] \) dB for 
\( J = [1; 3; 6; 10] \), respectively. Moreover, the use of 
\( J = 4 \) implies a faster convergence comparatively to the one 
obtained with \( J = 1 \) (see Fig. 5). This improvement is due 
to the fact that the extra time-spreading introduced by the 
TST coding provides more output measurements to estimate 
the same number of parameters, which makes the conver-
gence faster. It is to be recalled that the case \( J = 1 \) corre-
sponds to the blind receiver proposed in [30]. Evidently, an

increase of \( J \) provides a better performance at the cost of a 
computation complexity increase.

We observed in simulations that the BER is canceled 
for all values of SNR when \( J = 6 \). So, it is not useful to 
choose \( J = 4 \).

6.2. Influence of the number of blocks

In Fig. 6, we have plotted the BER versus SNR for two 
different values of the number of blocks (\( P \in [4, 6] \)) and of 
the spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 6 \):

\[
\Phi_6 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \Psi_6 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},
\]

where \( \Phi_4 \) and \( \Psi_4 \) are given in (67).

From this figure, we note that the BER performance 
can be significantly improved by increasing either the 
number of time blocks or the spreading code length, since 
both actions induce an increase of time diversity.

6.3. Spreading code length versus number of blocks

In order to emphasize the advantage of using the 
spreading code, we analyze the BER performance by 
varying the number of blocks (\( P \in [4, 10, 11] \)) and the 
spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 11 \):

\[
\Phi_{11} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, \quad \Psi_{11} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix},
\]

The matrices \( \Phi_P \) and \( \Psi_P \) for \( P = 4 \) and \( P = 10 \) are obtained 
by discarding the last rows of \( \Phi_{11} \) and \( \Psi_{11} \), respectively.

In Fig. 7, we can note that when the spreading length is 
not used, i.e for \( J = 1 \), we need to increase the number of

Thus, the BER is canceled for a \( \text{SNR} = [12; 6; 3; 2] \) dB for 
\( J = [1; 3; 6; 10] \), respectively. Moreover, the use of \( J = 1 \) 
implies a faster convergence comparatively to the one 
obtained with \( J = 1 \) (see Fig. 5). This improvement is due 
to the fact that the extra time-spreading introduced by the 
TST coding provides more output measurements to estimate 
the same number of parameters, which makes the conver-
gence faster. It is to be recalled that the case \( J = 1 \) corre-
sponds to the blind receiver proposed in [30]. Evidently, an

increase of \( J \) provides a better performance at the cost of a 
computation complexity increase.

We observed in simulations that the BER is canceled 
for all values of SNR when \( J = 6 \). So, it is not useful to 
choose \( J = 4 \).

6.2. Influence of the number of blocks

In Fig. 6, we have plotted the BER versus SNR for two 
different values of the number of blocks (\( P \in [4, 6] \)) and of 
the spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 6 \):

\[
\Phi_6 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \Psi_6 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},
\]

where \( \Phi_4 \) and \( \Psi_4 \) are given in (67).

From this figure, we note that the BER performance 
can be significantly improved by increasing either the 
number of time blocks or the spreading code length, since 
both actions induce an increase of time diversity.

6.3. Spreading code length versus number of blocks

In order to emphasize the advantage of using the 
spreading code, we analyze the BER performance by 
varying the number of blocks (\( P \in [4, 10, 11] \)) and the 
spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 11 \):

\[
\Phi_{11} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, \quad \Psi_{11} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix},
\]

The matrices \( \Phi_P \) and \( \Psi_P \) for \( P = 4 \) and \( P = 10 \) are obtained 
by discarding the last rows of \( \Phi_{11} \) and \( \Psi_{11} \), respectively.

In Fig. 7, we can note that when the spreading length is 
not used, i.e for \( J = 1 \), we need to increase the number of

Thus, the BER is canceled for a \( \text{SNR} = [12; 6; 3; 2] \) dB for 
\( J = [1; 3; 6; 10] \), respectively. Moreover, the use of \( J = 1 \) 
implies a faster convergence comparatively to the one 
obtained with \( J = 1 \) (see Fig. 5). This improvement is due 
to the fact that the extra time-spreading introduced by the 
TST coding provides more output measurements to estimate 
the same number of parameters, which makes the conver-
gence faster. It is to be recalled that the case \( J = 1 \) corre-
sponds to the blind receiver proposed in [30]. Evidently, an

increase of \( J \) provides a better performance at the cost of a 
computation complexity increase.

We observed in simulations that the BER is canceled 
for all values of SNR when \( J = 6 \). So, it is not useful to 
choose \( J = 4 \).

6.2. Influence of the number of blocks

In Fig. 6, we have plotted the BER versus SNR for two 
different values of the number of blocks (\( P \in [4, 6] \)) and of 
the spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 6 \):

\[
\Phi_6 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \Psi_6 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},
\]

where \( \Phi_4 \) and \( \Psi_4 \) are given in (67).

From this figure, we note that the BER performance 
can be significantly improved by increasing either the 
number of time blocks or the spreading code length, since 
both actions induce an increase of time diversity.

6.3. Spreading code length versus number of blocks

In order to emphasize the advantage of using the 
spreading code, we analyze the BER performance by 
varying the number of blocks (\( P \in [4, 10, 11] \)) and the 
spreading code length (\( J \in [1, 3] \)), with the following 
allocation matrices for \( P = 11 \):

\[
\Phi_{11} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, \quad \Psi_{11} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{bmatrix},
\]

The matrices \( \Phi_P \) and \( \Psi_P \) for \( P = 4 \) and \( P = 10 \) are obtained 
by discarding the last rows of \( \Phi_{11} \) and \( \Psi_{11} \), respectively.

In Fig. 7, we can note that when the spreading length is 
not used, i.e for \( J = 1 \), we need to increase the number of
blocks up to $P=11$ to obtain the same BER as that obtained with $J=3$ and $P=4$. According to (38), with $\tau$ given in Table 1, the transmission rate is equal to 1 and 4/11 bit per channel use for $P=4$ and $P=11$, respectively.

### 6.4. Influence of the number of data streams

Fig. 8 shows the BER versus SNR for configurations with different numbers of data streams ($R \in \{2, 5\}$), and spreading code lengths ($J \in \{1, 4\}$). We assume $P=10$ and the following allocation matrices for $R=5$:

$$
\Psi_{10} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1
\end{bmatrix},
\Phi_{10} = \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 \\
1 & 1 & 1 & 0
\end{bmatrix}.
$$

The matrix $\Psi_{10}$ in the case $R=2$ is obtained in discarding the three last columns of $\Psi_{10}$ for $R=5$.

According to (38), the transmission rate decreases from 1 to 2/5 bit per channel use when $R$ is decreased from 5 to 2. On the other hand, as expected and shown in Fig. 8, the BER performance is improved when $R$ is reduced from 5 to 2, due to the fact that fewer symbols have to be estimated using the same number of received signals. That illustrates the tradeoff between error performance and transmission rate that can be achieved with the proposed TST coding. It is interesting to note that the error performance for $J=1$ and $R=2$ is close to the one obtained with $J=4$ and $R=5$, which shows that TST coding allows to multiply the transmission rate by a factor 2.5 without modifying much the BER performance.

### 6.5. Influence of the number of receive antennas

Fig. 9 shows the BER versus SNR for two different numbers of receive antennas ($K \in \{2, 4\}$) and two spreading code lengths ($J \in \{1, 3\}$). As expected, we can see that larger is the number of receive antennas, smaller is the BER. It is interesting to notice that the proposed receiver gives a better performance with $K=2$ antennas than the receiver of [30] with $K=4$ antennas thanks to extra time diversity.

### 6.6. Comparison with blind KRST and non-blind TST-ZF receivers

In the next experiment, the blind TST-ALS based receiver is compared with the non-blind TST-ZF receiver that assumes perfect knowledge of the channel. This receiver is obtained by solving (19) with respect to the symbol matrix. That gives

$$
\tilde{S}_{ZF}^T = \left( I_{mP} \otimes H G_2 \right)^\dagger \tilde{X}_2.
$$

For this comparison, the design parameters were set to their default values. The BER curves obtained with the two algorithms (ALS and ZF) are plotted in Fig. 10 for $J \in \{1, 3\}$. From these results, we can conclude that for a BER equal to $10^{-3}$, the gap between blind TST-ALS and non-blind TST-ZF receivers is around 3 dB in terms of SNR, for both values $J=1$ and $J=3$.

The results obtained with the blind KRST-ALS based receiver of [22] using an identity pre-coding matrix ($\Psi_{\psi}=I_M$) are also plotted in Figs. 3–5 showing that...
the proposed TST-ALS based receiver outperforms the KRST-ALS based receiver in terms of channel estimation accuracy, BER and transmission rate (1 bit per channel use for TST coding, instead of 2/5 bit for KRST coding), at the cost of a slower convergence due to the greater number of parameters to be estimated (PNR symbols for TST coding and PM symbols for KRST coding).

7. Conclusion

In this paper, a new tensor space–time coding has been proposed for MIMO wireless communication systems. The associated transceiver is characterized by a third-order code tensor and two allocation matrices that allow space–time spreading–multiplexing of the transmitted symbols. As shown by the performance analysis, TST coding increases the maximum diversity gain. The introduction of one extra time diversity via the third mode of the code tensor induces a significant performance improvement in terms of BER and channel estimation comparatively to existing tensor-based solutions [22,30], as illustrated by means of simulation results. This extra time diversity leads to a more relaxed condition on the number of data blocks to be processed for ensuring LS identifiability of channel and symbol matrices that can be jointly and blindly estimated using a two-step ALS technique. Moreover, taking the Vandermonde structure of the code tensor into account allowed us to determine a (sufficient but not necessary) minimum value of the spreading length that ensures LS identifiability. There are several perspectives of this work that include extensions to channel and symbol matrices optimization, alternative receivers using an enhanced line search scheme [40] or a Levenberg–Marquardt algorithm [41], and/or an adaptive algorithm [42], and blind receiver when the code tensor is unknown at the receiver.

Acknowledgements

This work was supported by the CAPES/COFECUB Project no. 544/07. André L.F. de Almeida is partially supported by CNPq/Brazil (Grant no. 303238/2010-0).

References


Fig. 10. TST-ALS vs. TST-ZF.
[34] L. Tucker, Some mathematical notes on three-mode factor analysis, Psychometrika 31 (3) (1966) 279–311.