The Tensor Multi-linear Channel and its Shannon Capacity

DIVYANSHU PANDEY1 and HARRY LEIB2

1Department of Electrical and Computer Engineering, McGill University, Montreal, Quebec, Canada, H3A 0E9. (email: divyanshu.pandey@mail.mcgill.ca)
2Department of Electrical and Computer Engineering, McGill University, Montreal, Quebec, Canada, H3A 0E9. (email: harry.leib@mcgill.ca)

Corresponding author: Divyanshu Pandey (e-mail: divyanshu.pandey@mail.mcgill.ca).

This work was supported by grant RGPIN-2016-03647 entitled “Tensor modulation for space-time-frequency communication systems”, awarded by the Natural Sciences and Engineering Research Council of Canada (NSERC).

ABSTRACT Tensors are multi-way arrays which can be used to model systems spanning many domains. This work proposes to use tensors for characterizing, analyzing, and designing multi-domain communication systems. Most modern day communication systems make use of coding and modulation across different domains such as space, frequency, time. Hence a unified mathematical framework characterizing such a multiple-domain system in an intuitive manner is well needed. In this paper, we present such a unified framework that characterizes a communication system with N input domains, M output domains and an M + N domains multi-linear tensor channel. The proposed framework is generic where the physical interpretation of the domains is system specific. We illustrate a few examples from multi-antenna multi-carrier and multi-user systems that fit the proposed framework. Assuming a fixed tensor channel, we provide an information theoretic analysis by deriving its Shannon capacity and power allocation under a variety of power constraints. In this paper we show how the tensor framework’s suitability to mathematically describe a family of power constraints can be used to design and analyze various multiple domain communication systems. The tensor based approach extends water filling from a matrix setting to tensors, encapsulating the effects of multiple domains thereby allowing joint multi-domain pre-coding. We show that the capacity pre-log for a tensor channel increases exponentially in the number of domains, indicating the potential of tensor based multi-domain communication systems to provide the large information transmission rates envisaged for 5G and beyond systems. We also show the application of the tensor framework in characterizing the capacity and rate regions of multi-user MIMO channels. Both Multiple Access and Interference channel are considered where the tensor based approach leads to a coordinated users transmission scheme. Such a scheme ensures higher achievable sum rates as compared to independent user transmissions.

INDEX TERMS Tensors, MIMO channels, Multi-user MIMO, Shannon capacity, Tensor SVD and EVD.

I. INTRODUCTION

A tensor is a multi-domain array that can be seen as an Nth order generalization of a vector or a matrix, where a vector is a tensor of order one and a matrix is a tensor of order two [1]. Tensors were introduced in the early nineteenth century with applications in Physics where such mathematical structures can be seen as a mapping from a linear space to another whose coordinates transform multilinearly under a change of bases [2]. Later, tensors found applications in psychometrics in the sixties with the work of Tucker [3] as an extension of two-way data analysis to higher-order datasets, and in chemometrics in the eighties [4], [5]. In the last few decades, tensors as an extension of matrices have found extensive applications in various engineering disciplines including computer vision [6], [7], data mining [8], [9], machine learning [10], neuroscience [11], signal processing [12]–[14] and multi-linear system theory [15], [16]. Tensors provide a unified and intuitive framework to represent processes with dependencies on more than two indices. Through a tensor based approach, we can develop models which capture interactions between various parameters enhancing the understanding of their mutual effects. A detailed summary of tensor algebra results can be found in many publications such as [1], [9], [10], [17]–[20].

In the field of communication systems, so far tensors have been used primarily from a signal processing point of view. Most modern communication systems use multi-domain modulation and coding schemes for effectively ex-
exploiting resources in transmission and reception. A common example in wireless communications is that of using the space domain, i.e., multiple input multiple output (MIMO) systems based on multiple antennas, to achieve a significant increase in channel capacity [21]. Systems including MIMO in conjunction with multi-carrier techniques such as Orthogonal Frequency Division Multiplexing (OFDM) [22], [23], Generalized Frequency Division Multiplexing (GFDM) [24], [25], Filter Bank Multi-carrier (FBMC) [26], etc., have been widely researched over past few years. Techniques have been developed to improve link reliability through space-time, space-frequency, and space-time-frequency coding methods [27] that exploit diversity in all spatial, temporal and frequency domains. Hence the received and transmitted signals have an inherent multi-domain structure which can be represented by tensors. Such a multi-domain approach using tensors has been considered in [28], [29], where the focus has been around developing blind detection techniques at the receiver. A tensor based space-time-frequency coding structure is proposed in [30] for a MIMO OFDM-Code Division Multiple Access (CDMA) system where signals are represented using order-5 tensors. Applications of tensors are also being considered for modelling various 5G and 6G communication technologies such as Intelligent Reflecting Surfaces [31], massive MIMO [32], [33], millimeter wave [34], [35], MIMO relay systems [36]. Recently, a tensor based framework for a general multi-domain communication system using the tensor contracted product has been developed in [37]. In addition, [37] also considers tensor based joint-domain equalization at the receiver to combat inter-domain interferences. Hence the tensor approach of handling a multi-domain communication system can lead to design of new and improved transmission and reception schemes. This paper considers information theoretic aspects of discrete version of the tensor-modelled multi-domain communication systems of [37].

The domains in a communication system can represent space, time, frequency, users, propagation delay, spreading sequence etc. This necessitates a generic unified mathematical framework with which we can model any multi-domain communication system, and hence tensors naturally come into play. Having such a framework would not only provide a mathematical basis for existing schemes, but would also act as a stepping stone for developing new and improved systems spanning multiple domains such as G.hn networks [38], 5G and beyond [39], as well as 6G [40]. The main contribution of this paper is the generic tensor based system model where an order $M+N$ channel tensor is used to connect an order $N$ input tensor with an order $M$ output tensor using the Einstein product. We find the Shannon capacity of such a deterministic tensor channel with various input constraints, revealing its exponential increase with the number of domains. A tensor framework endowed with the Einstein product has been used in [15], [16] to extend basic linear system concepts to the multi-linear realm. Our work introduces the multi-linear tensor channel and extends basic Information Theoretic concepts to such models.

The initial motivation to use tensors for our purpose stems from their unique suitability to retain the distinction between multiple domains in the system model, thereby allowing a convenient representation of a variety of power constraints across domains. Even though tensor entities occur naturally in multi-domain communication systems, it should be noted that in principle a tensor can be represented using a matrix or a vector. For instance, the slices of a third order tensor can be stacked together to form a bigger matrix. Such matrix representations are sometimes used in order to leverage the well established linear algebra concepts for analysis. However, representing a naturally occurring higher order tensor using a lower order array such as a matrix or vector collapses the distinct multiple indices which are used to identify the domains. Thus in order to restore the identifiability of domains, it becomes imperative to use tensors [41]. In addition, the tensor based approach leads to joint domain pre-coding and power allocation operations which perform much better than matrix approaches using per domain processing. Also, the proposed framework in this paper can be used to find the capacity of multi-user MIMO channels. The tensor model allows to keep users and antennas as distinct domains in a multi-user system. Hence the problem of finding the channel capacity under per user power constraints for any number of users can be approached using the tensor framework.

Capacity of the MIMO matrix channel was analyzed in [21] by converting the MIMO channel into parallel non-interfering scalar channels through a singular value decomposition of the channel matrix. Since the late 1990’s and early 2000’s a significant research effort has been invested in extending the work in [21] under various assumptions. A detailed summary can be found in [42]. Capacity behaviour of OFDM based MIMO systems has been explored in [43] by concatenating the transmit vectors over different antennas and different sub-carriers into a single vector and using a block-diagonal channel matrix. There has not been an attempt so far to quantify the capacity for a channel spanning more than two domains in terms of the number of domains itself and also to find the multi-domain power allocation required to achieve capacity, which calls for a tensor based approach. One impediment en-route such application of tensors in information theory is the inability to completely diagonalize a tensor and thereby convert a higher order channel into scalar channels. Complete diagonalization of any tensor with tensor decomposition techniques such as Parallel Factorization (PARAFAC) or Tucker decomposition is not achievable in general [1]. However, recent work in [18], [44] proposes a tensor singular value decomposition (SVD) and eigenvalue decomposition (EVD) approach as special cases of Tucker decomposition for solving multi-linear systems of equations. A generalization of such a tensor SVD and EVD is presented in this paper which is used to compute the channel capacity.

The capacity of a MIMO GFDM channel modelled as a sixth order tensor, under sum power constraint is considered in [45]. Further, [46] presents the notion of tensor partial
response signalling as a means to generate multi-domain signals with desired spectral and cross-spectral properties. The trade-off between domains of a communication system as revealed through the tensor approach is also studied in [46]. In [47], the capacity of tensor channels under a family of power constraints, which includes per antenna or sum power constraints as its specific cases, is considered. Going beyond [47], in this paper we consider the channel characterization as a tensor and its Shannon capacity under different power constraints in detail with new results and applications. In particular, we consider the application of the tensor framework for MIMO multiple access and interference channels. We use our proposed solution to characterize rate regions and quantify achievable sum rates in a multi-user system under user cooperation with per user power constraints. A comparison of these results with other methods in literature which assume independent user transmissions is also included. We also present examples of several multi-domain systems such as multi-user MIMO OFDM where the channel can be characterized as higher order tensor. For any such multi-domain system, we present an algorithmic approach of approximating the optimal input covariance under a variety of input power constraints along with a discussion on its computational complexity. For MIMO GFDM systems, we use our proposed solution to prescribe a precoding scheme under per-antenna power constraints and analyze its BER performance. We also present results on multiplexing gain achieved by the tensor channel. Through several numerical examples, we present the capacity behaviour of tensor channels for various channel sizes, with different domain power constraints, and channel with correlated entries.

This paper is organized as follows: A brief review of tensor algebra is presented in section II. Section III introduces the discrete system model for a multi-domain communication system using tensors. The notion of a channel as a higher order tensor is proposed with an example. Section IV presents the capacity characterization of a channel as a higher order tensor. Section V considers the capacity of tensor channels under a family of power constraints, an algorithm for finding the input covariance along with its complexity, and multiplexing gain provided by tensor channels. Section VI presents numerical and simulation results showing an application of our work to MIMO GFDM systems. Section VII considers multi-user MIMO Multiple Access Channels (MAC) and Interference Channels (IC) using the tensor framework. The paper is concluded in Section VIII.

II. ELEMENTS OF TENSOR ALGEBRA

In this section we present essential tensor algebra results needed for this paper. A detailed treatment of tensor algebra can be found in [1], [18], [37] and references within.

A. NOTATIONS

Throughout this paper, deterministic vectors are represented using lower-case underline fonts, e.g. \( \mathbf{x} \), matrices using upper-case fonts, e.g. \( X \) and tensors using upper-case calligraphic fonts, e.g. \( \mathbf{X} \). Their corresponding random quantities are denoted by bold fonts, e.g. \( \mathbf{x} \), \( X \) and \( \mathbf{X} \) for random vectors, matrices and tensors respectively. The individual entries of a tensor are denoted by indices in subscript, e.g. the \((i,j,k)\)th element of a third order tensor \( \mathbf{X} \) is denoted by \( X_{i,j,k} \). A colon in subscript is used to indicate all elements of a mode, e.g. \( X_{:,j,k} \) represents all the elements of first mode corresponding to the \( j \)th second and the \( k \)th third mode. The \( n \)th element in a sequence is denoted by a superscript in parentheses, e.g. \( A^{(n)} \) denotes the \( n \)th tensor in a sequence of tensors. Expectation is denoted by \( \mathbb{E} [\cdot] \), entropy by \( \mathcal{H} (\cdot) \) and mutual information by \( I (\cdot; \cdot) \). The set of complex numbers is denoted by \( \mathbb{C} \). An all-zero tensor is represented by \( 0_{\cdot} \).

B. BASIC TENSOR OPERATIONS AND DEFINITIONS

Tensors are multi-way arrays with components indexed by \( N \) indices also known as modes. The number of modes, \( N \), is called the order of the tensor. The set of all tensors of size \( I_1 \times \ldots \times I_K \) over \( \mathbb{C} \) forms a linear space, denoted as \( \mathbb{T}_{I_1,...,I_K} (\mathbb{C}) \).

Definition 1. Matricization Transformation: Let us denote the linear space of \( P \times Q \) matrices over \( \mathbb{C} \) as \( \mathbb{M}_{P} (\mathbb{C}) \). For an order \( K = N + M \) tensor \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \), the matricization transformation \( f_1,...,I_N|J_1,...,J_M : \mathbb{T}_{I_1,...,I_N|J_1,...,J_M} (\mathbb{C}) \rightarrow \mathbb{M}_{I_1,...,I_N,\prod_{k=1}^{K-1} I_k,\prod_{k=1}^{K-1} J_k} (\mathbb{C}) \) with \( f_1,...,I_N|J_1,...,J_M (A) = \Lambda \) is defined component-wise as [18]:

\[
A_{i_1,i_2,...,i_N,j_1,j_2,...,j_M} = \sum_{k=1}^{N} \sum_{l=1}^{M} A_{i_1+\sum_{k=2}^{N} (i_k-1)\prod_{l=1}^{k-1} I_l,j_1+\sum_{k=2}^{M} (j_k-1)\prod_{l=1}^{k-1} J_l}
\]

This transformation is referred to as matricization, or matrix unfolding by partitioning the indices into two disjoint subsets [17]. The vectorization operation as defined in [48] can be seen as a specific case of (1) by using \( J_1 = \ldots = J_M = 1 \). The bar in subscript of \( f_1,...,I_N|J_1,...,J_M \) represents the partitioning after \( N \) modes of an \( N + M \) order tensor where first \( N \) modes correspond to the rows of the representing matrix, and the last \( M \) modes correspond to the columns of the representing matrix. This mapping is bijective [49], and it preserves addition and scalar multiplication operations. Hence the linear spaces \( \mathbb{T}_{I_1,...,I_N|J_1,...,J_M} (\mathbb{C}) \) and \( \mathbb{M}_{I_1,...,I_N,\prod_{k=1}^{K-1} I_k,\prod_{k=1}^{K-1} J_k} (\mathbb{C}) \) are isomorphic and the transformation \( f_1,...,I_N|J_1,...,J_M \) is an isomorphism between the linear spaces.

Definition 2. Tensor Contracted product [17]: Consider two tensors \( X \in \mathbb{C}^{I_1 \times \ldots \times I_M \times J_1 \times \ldots \times J_N} \) and \( Y \in \mathbb{C}^{I_1 \times \ldots \times I_M \times K_1 \times \ldots \times K_P} \). A contraction between the two tensors along their \( M \) common modes denoted by \( \langle X, Y \rangle_{\{1,\ldots,M; 1,\ldots,M\}} \) leads to a resulting tensor \( Z \in \mathbb{C}^{I_1 \times \ldots \times I_N \times K_1 \times \ldots \times K_P} \), given by:

\[
Z_{j_1,...,j_N,k_1,...,k_P} = \sum_{i_1,...,i_M} X_{i_1,...,i_M,j_1,...,j_N} Y_{i_1,...,i_M,k_1,...,k_P}
\]

In general for the contracted product, the modes to be
Definition 3. Einstein product [18]: For any positive integer \( N \), the Einstein product is defined using the operation \( *_N \) by

\[
(A *_N B)_{i_1,...,i_N} = \sum_{k_1,...,k_N} A_{i_1,...,i_N;k_1,...,k_N} B_{k_1,...,k_N} \quad (3)
\]

where \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N \times K_1 \times \ldots \times K_N} \) and \( B \in \mathbb{C}^{K_1 \times \ldots \times K_N \times J_1 \times \ldots \times J_M} \).

Einstein product is a tensor contracted product where contraction is over the last \( N \) consecutive modes of \( A \) and first \( N \) consecutive modes of \( B \). The outer and inner products can be seen as special cases of Einstein product. For tensors \( X, Y \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) and \( Z \in \mathbb{C}^{J_1 \times \ldots \times J_M} \), we have:

**Inner Product**: \( \langle X, Y \rangle = X *_N Y = Y *_N X \)

\[
= \sum_{i_1=1}^{I_1} \ldots \sum_{i_N=1}^{I_N} X_{i_1,...,i_N} Y_{i_1,...,i_N} \quad (4)
\]

**Outer Product**: \( \langle X \odot Z \rangle_{i_1,...,i_N} = \sum_{j_1,...,j_M} X_{i_1,...,i_N} Z_{j_1,...,j_M} = X *_N Z \quad (5)\)

where \( \langle X, Y \rangle \) is a scalar and \( (X \odot Z) \in C^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \).

Definition 4. mode-n product [50]: The mode-n product of a tensor \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \) with a matrix \( U \in \mathbb{C}^{J_n \times I_k} \) is denoted by \( X \times_n U \) and is defined as

\[
(X \times_n U)_{i_1,...,i_n-1,j,i_n+1,...,i_N} = \sum_{j=1}^{I_n} X_{i_1,...,i_n-1,j,i_n+1,...,i_N} U_{j,i_n} \quad (6)
\]

where \( (X \times_n U) \in \mathbb{C}^{I_1 \times \ldots \times I_n-1 \times J_1 \times \ldots \times J_M} \).

Definition 5. Square tensors [49]: A tensor \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \) is called a square tensor if \( N = M \) and \( I_k = J_k \) for \( k = 1, \ldots, N \).

For square tensors \( A, B \) of size \( I \times J \times I \) and \( J \), it was shown in [18] that

\[
f_{I_1,...,I_N|K_1,...,K_P}(A *_M B) = f_{I_1,...,I_N|J_1,...,J_M}(A) \cdot f_{J_1,...,J_M|K_1,...,K_P}(B) \quad (7)
\]

where \( f_{I_1,...,I_N|K_1,...,K_P} \) denotes the following holds:

The tensor SVD: For \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \), the tensor Singular Value Decomposition (SVD) of \( A \) has the form

\[
A = \sum_{\ell=1}^{\min(N,M)} \sigma_{\ell} U_{\ell,1} \otimes \ldots \otimes U_{\ell,N} V_{\ell,1} \otimes \ldots \otimes V_{\ell,M} \quad (8)
\]

where \( \sigma_{\ell} \) are the singular values of \( A \), and \( U_{\ell} \) and \( V_{\ell} \) are unitary matrices. This decomposition is the multi-linear generalization of the singular value decomposition for matrices.
applications particularly in extracting low rank structures in higher dimensional data [54]. The relation between Tucker decomposition and the tensor SVD presented here is explained in [18].

**Tensor EVD:** Let \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), \( \lambda \in \mathbb{C} \), where \( X \) and \( \lambda \) satisfy \( A \ast_N X = \lambda X \), then we call \( X \) and \( \lambda \) as eigentensor and eigenvalue of \( A \) respectively. The tensor Eigenvalue Decomposition (EVD) of a tensor \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) is given as \( A = \bigcup \ast_N D \ast_N U^H \) where \( U \in \mathbb{C}^{I_1 \times \ldots \times I_N \times I_1 \times \ldots \times I_N} \) is a unitary tensor and \( D \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) is a square pseudo-diagonal tensor with its non-zero values being the eigenvalues of \( A \) and \( U \) containing the eigentensors of \( A \). As a generalization of matrix eigenvalues, several other definitions exist in literature for tensor eigenvalues [55], [56]. But most of these definitions apply to super-symmetric tensors which are defined as a class of tensors that are invariant under any permutation of their indices [56]. Such an approach has applications in Physics and Mechanics [56]. But there is no single generalization to the tensor case that preserves all the properties of matrix eigenvalues [57]. For our purposes, we have presented a particular generalization from [44] which can be used for any square tensor, irrespective of symmetry in its elements.

The eigenvalues of a Hermitian tensor are real. Subsequently, a Hermitian tensor \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) is defined as positive semi-definite, denoted by \( X \succeq 0 \), if all its eigenvalues are non-negative, and as positive definite, denoted by \( X \succ 0 \), if all its eigenvalues are strictly positive. Determinant of \( X \), denoted by \( \det(X) \), is defined as the product of its eigenvalues. Trace of a tensor \( A \), denoted as \( \text{tr}(A) \), is defined as the sum of its pseudo-diagonal entries. Also, the following properties can be established using Lemma 1:

(a) **Associativity:** For tensors \( A \in \mathbb{C}^{I_1 \times \ldots \times I_P \times J_1 \times \ldots \times J_N} \), \( B \in \mathbb{C}^{J_1 \times \ldots \times J_P \times K_1 \times \ldots \times K_M} \), and \( C \in \mathbb{C}^{K_1 \times \ldots \times K_M \times T_1 \times \ldots \times T_Q} \), we have

\[
( A \ast_N B ) \ast_M C = A \ast_N ( B \ast_M C )
\]

\[
( A \ast_N B ) \circ C = A \ast_N ( B \circ C )
\]

(b) **Commutativity:** Einstein product is not commutative in general. However for the specific case where the contraction is over all the \( N \) modes of one of the tensors, say for tensors \( A \in \mathbb{C}^{I_1 \times \ldots \times I_P \times J_1 \times \ldots \times J_N} \) and \( B \in \mathbb{C}^{J_1 \times \ldots \times J_N} \), we have

\[
A \ast_N B = B \ast_N A^T
\]

(c) For invertible tensors \( A \) and \( B \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), we have

\[
( A \ast_N B )^{-1} = B^{-1} \ast_N A^{-1}
\]

\[
\det( A \ast_N B ) = \det( B \ast_N A ) = \det( A ) \cdot \det( B )
\]

(d) For \( A, B \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) of same size and order \( N \),

\[
A \ast_N B = B \ast_N A = \text{tr}(A \circ B) = \text{tr}(B \circ A)
\]

(e) For tensors \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \) and \( B \in \mathbb{C}^{J_1 \times \ldots \times J_M \times I_1 \times \ldots \times I_N} \), we have:

\[
\det(J_N \ast A \ast N B ) = \det(J_M \ast B \ast N A)
\]

To prove (13), we can use Lemma 1 and Sylveste’s matrix determinant identity [58].

**Tensor Gradient:** Consider a real-valued scalar function \( g(X) \) of a complex matrix \( X \). The corresponding complex gradient matrix is defined as \( \nabla_X g \triangleq \partial g / \partial X \), where \( \nabla_X g_{i,j} = \frac{\partial g}{\partial X_{i,j}} \triangleq \frac{1}{2} \left( \frac{\partial g}{\partial \text{Re}[X_{i,j}]} + j \frac{\partial g}{\partial \text{Im}[X_{i,j}]} \right) \) [59]. We similarly define the complex gradient of a scalar function \( f(X) \) of a tensor \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) as \( \nabla_X f \triangleq \partial f / \partial X \), where the gradient is a tensor of the same size as \( X \) whose individual components are the derivatives with respect to the components of \( X \), i.e. \( \nabla_X f_{i_1,\ldots,i_N} = \partial f / \partial X^*_{i_1,\ldots,i_N} \). Using this definition, we can extend several results on matrix complex gradients from [59], [60], [61], Appendix A.7 to a tensor setting. For instance, given \( A \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), and Hermitian positive semi-definite tensors \( B, C \in \mathbb{C}^{I_1 \times \ldots \times I_N} \), using results from [59], [60] and Lemma 1, we can write:

\[
\nabla_c \log[\det(J_M \ast A \ast N C \ast N A^H)] = A^H \ast_M (A \ast_N C \ast N A^H + J_M)^{-1} \ast_M A
\]

\[
\nabla_c \text{tr}(B \ast N C) = B
\]

### III. DISCRETE TENSOR SYSTEM MODEL

#### A. THE TENSOR CHANNEL

We define the input and output in a multi-domain communication system as tensor symbols of order \( N \) and \( M \) respectively. Let \( X \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) represent the input (transmitted) tensor symbol with \( I_1, I_2, \ldots, I_N \) as the dimensions of its \( N \) domains where each component \( X_{i_1,\ldots,i_N} \) is a discrete complex symbol. Similarly, we represent the output (received) tensor symbol by \( Y \in \mathbb{C}^{J_1 \times \ldots \times J_M} \) with \( J_1, J_2, \ldots, J_M \) as the dimensions of its \( M \) domains. We define a multi-linear channel between the transmit and the receive side as a tensor of order \( M + N \) represented by \( H \in \mathbb{C}^{I_1 \times \ldots \times I_N \times J_1 \times \ldots \times J_M} \).

With additive noise, the system model can be specified by using the Einstein product of the channel tensor with input tensor where we contract along all the modes of the input tensor:

\[
Y = H \ast_N X + N
\]

where

\[
Y_{j_1,\ldots,j_M} = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} H_{j_1,\ldots,j_M,i_1,\ldots,i_N} X_{i_1,\ldots,i_N} + N_{j_1,\ldots,j_M}
\]

with \( N \) representing the received noise tensor of same size as \( Y \). It is straightforward to see that the narrowband discrete time MIMO matrix channel model is a special case of this tensor model where input and output are order-1 tensors.
(vectors), the channel is an order-2 tensor (matrix) and the Einstein product reduces to regular matrix multiplication, \( y = Hx + n \).

In case of a matrix representation of a discrete MIMO channel which characterizes only the space domain, each component of channel matrix \( H_{i,j} \) represents the complex gain (i.e. amounting for both phase change and amplitude gain) of different paths between transmit and receive antennas. In (17), each component of the tensor channel is a complex gain that couples the elements between the order \( N \) input and order \( M \) output tensors. The proposed system model is generic and the number of modes along with the physical interpretation of the individual modes is systemspecific. Such modes can represent space, time, frequency, propagation delay, users, spreading sequence, etc., depending on the system.

For a MIMO communication system with \( N_R \) receive and \( N_T \) transmit antennas, the continuous time-input-output relation in a linear time varying channel is given as [62]:

\[
y(t) = \int H(t, \tau)x(t-\tau)d\tau + n(t) \tag{18}
\]

where \( x(t) \) is the \( N_T \times 1 \) continuous time input vector, \( n(t) \) and \( y(t) \) are the \( N_R \times 1 \) noise and received vectors respectively. The \( N_R \times N_T \) matrix \( H(t, \tau) \) has components \( H_{n,r}(t, \tau) \) that represent the channel impulse response between transmit antenna \( n_t \) and receive antenna \( n_r \) at time instant \( t \) and delay \( \tau \). If the channel is assumed time-invariant with a maximum delay \( \tau_{max} \), the discretization of (18) at a sampling frequency \( f_s \) gives the input/output relation at an instant \( k \) as [62]:

\[
y[k] = \sum_{d=1}^{D} H[d]x[k-(d-1)] + n[k], \quad k = 0, 1, \ldots, (N-1) \tag{19}
\]

where \( x[k] \) is the \( N_T \times 1 \) channel input at time index \( k \), \( n[k] \) and \( y[k] \) are the \( N_R \times 1 \) noise and received vectors respectively. The \( N_R \times N_T \) matrix \( H[d] \) has components \( H_{n,r,d}(t, \tau) \) that represents the length \( D \) channel impulse response between transmit antenna \( n_t \) and receive antenna \( n_r \) at delay \( d \), where \( D = \lceil f_s \tau_{max} \rceil \). Delay can be considered as another domain in the system model. So for a time-invariant channel, at any instant \( k \) the relation in (19) can be expressed using tensor model as [63]:

\[
y[k] = \mathcal{H} \ast_2 x[k] + n[k] \tag{20}
\]

where all the individual vectors \( y[k] \) from (21) for \( k = 0, 1, \ldots, (N-1) \) form the columns of the matrix \( Y \) of size \( N_R \times N \). Similarly vectors \( n[k] \) from (21) form the columns of the matrix \( N \) of size \( N_R \times N \) and vectors \( x[d'] \) where \( d' = k - (d-1) \) form the columns of the matrix \( X \) of size \( N_T \times D' \). All the individual matrices \( H[k,d] \) from (21) are now sub-tensors of the order-four channel tensor \( \mathcal{H} \) of size \( N_R \times N \times N_T \times D' \) where \( \mathcal{H}_{i,k,:,:} = H[k,d] \). We can see how the tensor system model in (20) simply evolved in (22) to account for time variation of the channel as well.

### B. Tensor Model Applied to MU-MIMO OFDM

The model presented in (16) can be used for a wide variety of systems. For instance, the system model used in [44] for image restoration is a specific case of the system model of (16). To stress the relevance of the proposed tensor model, particularly in multi-domain communication systems, we now present an example of a multi-user MIMO OFDM system which can be represented using the tensor framework.

OFDM is one of the most popular multi-carrier schemes and has been used extensively with MIMO in 4G standards and Wi-Fi [64]. A conventional model for a MIMO OFDM

---

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3160187, IEEE Access
where each vector $\tilde{y}[p]$, $\tilde{x}[p]$ and $\tilde{n}[p]$ are the frequency domain received, transmitted and noise symbol vectors at sub-carrier $p$, while $N_{sc}$ denotes the number of sub-carriers. The model represented by (23) can be obtained from (21) by taking the $N_{sc}$-point discrete Fourier Transform (DFT) of $\{y[k]\}$ where $DFT\{y[k]\} = \tilde{y}[p]$, $DFT\{n[k]\} = \tilde{n}[p]$ and $DFT\{x[k]\} = \tilde{x}[p]$. The frequency domain $N_R \times N_T$ channel matrix between transmit sub-carrier $q$ and receive sub-carrier $p$ is $H[p,q]$, whose individual elements $H_{n_r,n_t}[p,q]$ can be obtained from the discrete time varying channel between the $n_r$th receive antenna and $n_t$th transmit antenna, $H_{n_r,n_t}[k,d]$ (based on DFT of (21)) as:

$$\tilde{H}_{n_r,n_t}[p,q] = \frac{1}{N_{sc}} \sum_{k=0}^{N_{sc}-1} \sum_{d=0}^{D-1} H_{n_r,n_t}[k,d] e^{j2\pi (k (q-p) + n_r d) / N_{sc}} e^{-j2\pi q d / N_{sc}}$$

(24)

where $1 \leq n_r \leq N_R, 1 \leq n_t \leq N_T$ and $0 \leq p, q \leq N_{sc}-1$. Using tensors, we can represent the frequency domain input/output relation in MIMO OFDM of (23) as:

$$\tilde{Y} = \mathcal{F} \ast \tilde{X} + \tilde{N}$$

(25)

where each vector $\tilde{y}[p]$ and $\tilde{n}[p]$ from (23) for $p = 0, \ldots, N_{sc}-1$ form the columns of matrices $\tilde{Y}$ and $\tilde{N}$ of size $N_R \times N_{sc}$, and vectors $\tilde{x}[p]$ form the columns of matrix $\tilde{X}$ of size $N_T \times N_{sc}$. The input and output are connected by an order 4 tensor channel $\mathcal{H} \in \mathbb{C}^{N_R \times N_{sc} \times N_T \times N_{sc}}$ where each element $H_{n_r,n_t}[p,q]$ from (24) is now an element in the channel tensor as $\mathcal{H}_{n_r,n_t,p,q}$. We can expand the MIMO OFDM system model to include users as an additional domain in the model which will lead to a sixth order tensor channel. In the case of multi-user MIMO OFDM, the frequency domain channel matrix is often represented as an $N_R \times N_T$ matrix corresponding to a specific user and a specific sub-carrier [65], [66]. To account for inter-carrier interference (ICI) as well, the channel matrix could be represented as $\tilde{H}[u,p,q] \in \mathbb{C}^{N_R \times N_T}$ corresponding to the $u$th user for transmit sub-carrier $q$ and receive sub-carrier $p$. Consider a multi-user MIMO OFDM downlink system where a base station is catering to $U$ users having $N_R$ receive antennas each. The system model is a generalization of (23) and it is given by

$$\tilde{y}[u,p] = H[u,p] \tilde{x}[p] + \sum_{q=0,q\neq p}^{N_{sc}-1} H[u,p,q] \tilde{x}[q] + \tilde{n}[u,p]$$

(26)

for $p,q = 0, \ldots, N_{sc}-1$ and $u = 1, \ldots, U$. The entities $\tilde{y}[u,p] \in \mathbb{C}^{N_R \times 1}$ and $\tilde{n}[u,p] \in \mathbb{C}^{N_R \times 1}$ represent the received signal vector and noise vector on sub-carrier $p$ for the $u$th user, respectively. Also, $\tilde{x}[q] \in \mathbb{C}^{N_R \times 1}$ denotes the transmit vector from the base station at sub-carrier $q$, which is given by (27):

$$\tilde{x}[q] = \sum_{u'=1}^{U} M[u',q] \tilde{d}[u',q]$$

(27)

where $M[u',q] \in \mathbb{C}^{N_R \times N_T}$ denotes the pre-coding matrix used to transmit data vector $\tilde{d}[u',q] \in \mathbb{C}^{N_T \times 1}$ to user $u'$ on sub-carrier $q$. Hence the system model of (26) becomes

$$\tilde{y}[u,p] = \sum_{q=0}^{N_{sc}-1} H[u,p,q] \left( \sum_{u'=1}^{U} M[u',q] \tilde{d}[u',q] \right) + \tilde{n}[u,p]$$

(28)

Let $H[u,p,q] \cdot M[u',q] = G[u,u',p,q] \in \mathbb{C}^{N_R \times N_T}$ denote the equivalent channel between transmit data vector $\tilde{d}[u',q]$ and receive vector $\tilde{y}[u,p]$, then the input/output relation can be written as:

$$\tilde{y}[u,p] = \sum_{q=0}^{N_{sc}-1} \sum_{u'=1}^{U} G[u,u',p,q] \tilde{d}[u',q] + \tilde{n}[u,p]$$

(29)

In this case, the output and noise tensors can be rearranged into order three tensors $\tilde{Y}, \tilde{N} \in \mathbb{C}^{U \times N_R \times N_{sc}}$. The components $\tilde{y}_{u,n_r}[u,p]$ and $\tilde{n}_{u,n_r}[u,p]$ are mapped to elements of third order tensors, denoted by $\tilde{Y}_{u,n_r,p}$ and $\tilde{N}_{u,n_r,p}$ respectively. Similarly, the input can be rearranged as an order three tensor $\tilde{D} \in \mathbb{C}^{U \times N_R \times N_{sc}}$ where $\tilde{d}_{u,n_r}[u',q]$ is mapped to $\tilde{D}_{u,n_r,q}$. Subsequently, the channel can be represented as an order 6 tensor $\tilde{G} \in \mathbb{C}^{U \times N_R \times N_{sc} \times U \times N_T \times N_{sc}}$ where each element of matrix $\tilde{G}[u,n_r,p,u',q]$ from (28) is mapped to an element $\tilde{G}_{u,n_r,p,u',q}$ of the sixth order tensor channel. The system model then becomes:

$$\tilde{y} = \tilde{G} \ast \tilde{D} + \tilde{N}$$

(30)

The tensor model represented by (30) can be considered as an evolution of the common matrix MIMO model in the space domain only, to a tensor MIMO model that in addition to space it encapsulates also the frequency and user domains. Similarly, other systems such as MIMO DSL [68], MIMO CDMA [69], MIMO FBMC [26] and MIMO GFDM [24] can be represented using the tensor based system model.

Representing (29) using (30), or (23) using (25) gives us a unified framework which encapsulates the multitude of signalling domains into well structured tensor entities. The matrix representations from (29) and (23) often leads to a per user or per sub carrier processing of the signals, where the inter-domain interferences are treated in combination with the additive noise term. On the contrary, the tensor model from (30) and (25) can be employed to design transceivers jointly across domains such as users, sub carriers, and antennas, using tools from tensor algebra. We present several numerical examples in sections VI-B and VII illustrating the advantages of the joint domain processing for multi-domain communication systems. Note that an alternate rep-
representation of (29) can also be achieved via a large matrix channel based model, where the input and output signals are concatenated across users, sub carrier and antenna domains to form vectors. However, such representation will obscure the indices $u'_1, n_t, q$ and $u, n_r, p$ into single indices with no well defined physical meaning. Such a lower order representation of high order entities makes it difficult to incorporate domain specific constraints such as those explained in section V-A which are well handled using tensors. As a prelude to the capacity of tensor channels, we first present some basic information theoretic notions for tensors in the next section.

IV. INFORMATION THEORETIC NOTIONS FOR TENSORS

A tensor is said to be random if its components are random variables. Expectation of a random tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ denoted by $\mathbb{E}[\mathbf{X}] \in \mathbb{C}^{I_1 \times \ldots \times I_N}$, is a tensor with each component consisting of the expected value of the corresponding component of $\mathbf{X}$.

Covariance and Pseudo-covariance

Covariance of a random tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ can be represented using a tensor $\mathbf{Q} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ as $\mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ] \otimes \mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ]^*$. It can be shown that a covariance tensor is Hermitian, i.e. $\mathbf{Q} = \mathbf{Q}^H$. The pseudo-covariance of $\mathbf{X}$ is also an order $2N$ tensor defined as $\mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ] \circ \mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ]$. A random tensor is called proper if its pseudo-covariance is an all zero tensor [14].

Tensor Gaussian Distribution

The probability density function (pdf) of a random tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ is a scalar function of all its elements. Hence the pdf of a tensor having proper complex Gaussian entries can be specified by vectorizing the tensor and using the pdf of a proper complex Gaussian vector. However, using the properties of the Einstein product, the pdf can also be expressed without vectorizing as explained in [14]. The pdf of a proper complex Gaussian distributed tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ is given as :

$$p_{\mathbf{X}}(\mathbf{X}) = \exp \left\{ - \frac{1}{2} \left( \mathbf{X} - \mathbb{E}[\mathbf{X}] \right) \otimes \mathbb{E}[\mathbf{X}]^* - \mathbb{E}[\mathbf{X}] \otimes \mathbf{X}^* \right\}$$

(31)

where $\mathbb{E}[\mathbf{X}]$ is the order-N mean tensor and $\mathbf{Q} = \mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ] \circ \mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] ]$ is the order-2N covariance tensor.

A. DIFFERENTIAL ENTROPY OF A CIRCULAR COMPLEX GAUSSIAN DISTRIBUTED TENSOR

A complex random tensor $\mathbf{X}$ is defined as circular if it is rotationally invariant, i.e. $\mathbf{X}$ and $\mathbf{Y} = e^{j\alpha} \mathbf{X}$ have the same probability distribution for any given real $\alpha$. A complex Gaussian random vector is circular if and only if it is zero mean and proper [70]. This statement can be extended to tensor case also : 

Lemma 2. A complex Gaussian random tensor is circular if and only if it is zero mean and proper.

The proof of this lemma directly follows from the definition of proper and circular tensors. The distribution of a circular symmetric complex Gaussian tensor $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ with covariance $\mathbf{Q}$ is given by (31), with $\mathbb{M} = 0$. Subsequently, the differential entropy of such a tensor is given by :

$$H(\mathbf{X}) = \mathbb{E} \left[ - \log \ p_{\mathbf{X}}(\mathbf{X}) \right]$$

$$= \mathbb{E} \left[ - \log \left\{ \frac{\exp \left\{ - \frac{1}{2} \left( \mathbf{X} - \mathbb{E}[\mathbf{X}] \right) \otimes \mathbb{E}[\mathbf{X}]^* - \mathbb{E}[\mathbf{X}] \otimes \mathbf{X}^* \right\}}{\pi^{I_1 \times \ldots \times I_N} \det(\mathbf{Q})} \right\} \right]$$

$$= \log \left( \frac{\pi^{I_1 \times \ldots \times I_N} \det(\mathbf{Q})}{\det(\mathbf{Q})} \right) + \log \left( \frac{\mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] \otimes \mathbb{E}[\mathbf{X}]^* - \mathbb{E}[\mathbf{X}] \otimes \mathbf{X}^*]}{\mathbb{E}[\mathbf{X} - \mathbb{E}[\mathbf{X}] \otimes \mathbb{E}[\mathbf{X}]^* - \mathbb{E}[\mathbf{X}] \otimes \mathbf{X}^*]} \right)$$

(32)

The following lemma regarding the differential entropy of a circular Gaussian random tensor is proven in [45] :

Lemma 3. Let $\mathbf{X} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ be a circularly symmetric complex Gaussian random tensor with covariance tensor $\mathbf{Q}$. Let $\mathbf{Y} \in \mathbb{C}^{I_1 \times \ldots \times I_N}$ be another zero-mean random tensor with same covariance tensor. Then, $H(\mathbf{X}) \geq H(\mathbf{Y})$, i.e. for a given covariance tensor, a circular symmetric Gaussian distribution is the entropy maximizer.

B. MUTUAL INFORMATION

For the system model defined in (16), the output covariance tensor is

$$\mathbf{Q}_y = \text{cov}(\mathbf{Y}) = \mathbb{E}[\mathbf{Y} \circ \mathbf{Y}^*] = \mathbb{E}[\mathbb{E}[\mathbf{X} \circ \mathbf{N} + \mathbb{N}] \circ (\mathbb{E}[\mathbf{X} \circ \mathbf{N}]^* + \mathbb{N}^*)]$$

(33)

$$\mathbf{Q}_y = \mathbb{E}\left[ (\mathbb{E}[\mathbf{X} \circ \mathbf{N}] \circ (\mathbb{E}[\mathbf{X} \circ \mathbf{N}]^*) + \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*] + \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*] \right]$$

$$+ \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*]$$

(34)

Assuming $\mathbf{X}$ and $\mathbf{N}$ are zero mean and independent, it can be shown that the cross terms are zero. Based on commutativity rule of (9), we get $\mathbb{E}[\mathbf{X} \circ \mathbf{N}] = \mathbb{X} \circ \mathbb{N} = \mathbb{X}^T \otimes \mathbb{N}$. Using the associativity rule (8), we get :

$$\mathbf{Q}_y = \mathbb{E}[\mathbb{E}[\mathbf{X} \circ \mathbf{N}] \circ (\mathbb{E}[\mathbf{X} \circ \mathbb{N}])^* + \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*] + \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*] + \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*]$$

$$+ \mathbb{E}[\mathbb{N} \circ \mathbb{N}^*]$$

(35)

where $\mathbb{Q}_\mathbf{X}$ and $\mathbb{Q}_\mathbf{N}$ are the input and noise covariance tensors respectively. The following two lemmas can be proven using the definition of circular symmetric complex Gaussian tensors :

Lemma 4. If $\mathbf{X} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ is a circular symmetric complex Gaussian tensor, then so is $\mathbf{Y} = \mathbb{H} \circ \mathbf{X}$ for any

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
deterministic tensor, \( \mathcal{H} \in \mathbb{C}^{K_1 \times K_2 \times \ldots \times K_P \times I_1 \times I_2 \times \ldots \times I_N} \).

**Lemma 5.** If \( \mathbf{X} \) and \( \mathbf{Y} \) are independent circular symmetric complex Gaussian tensors of same order and size, then \( \mathbf{Z} = \mathbf{X} + \mathbf{Y} \) is also circular symmetric complex Gaussian.

Assuming noise to be a zero-mean Gaussian distributed tensor with covariance \( Q_N \) that is independent of the input tensor \( \mathbf{X} \), we can write the mutual information between input and output tensors as:

\[
I(\mathbf{X}; \mathbf{Y}) = \mathcal{H}(\mathbf{Y}) - \mathcal{H}(\mathbf{Y} | \mathbf{X}) = \mathcal{H}(\mathbf{Y}) - \mathcal{H}(\mathcal{N})
\]

\[
= \mathcal{H}(\mathbf{Y}) - \log \left( (\pi e)^{J_1 J_2 \cdots J_M} \det(Q_N) \right) (36)
\]

Based on Lemma 3 and received covariance derived in (35), we can write:

\[
\mathcal{H}(\mathbf{Y}) \leq \log \left( (\pi e)^{J_1 J_2 \cdots J_M} \det(Q_N \ast Q_N \ast H + Q_N) \right)
\]

\[
\Rightarrow I(\mathbf{X}; \mathbf{Y}) \leq \log \left[ \frac{\det(Q_N \ast Q_N \ast H + Q_N)}{\det(Q_N)} \right] (37)
\]

where equality is achieved only if \( \mathbf{Y} \) is Gaussian.

**V. CAPACITY OF A FIXED TENSOR CHANNEL**

Finding the Shannon capacity of the tensor channel requires the maximization of the mutual information between the input and the output tensors over input distributions under possible constraints. In this work, we assume that the channel tensor is known and the noise tensor is zero-mean Gaussian distributed having independent components with variance \( \sigma^2 \), and hence the noise covariance tensor is given by \( Q_N = \sigma^2 I_M \). For simplicity we assume \( \sigma^2 = 1 \). Now let us consider the mutual information inequality of (38) where equality is achieved only if \( \mathbf{Y} \) is Gaussian. In our tensor channel model (16), since \( \mathbf{N} \) is zero-mean circular symmetric complex Gaussian, then using Lemma 4 and 5 implies that \( \mathbf{Y} \) will also be zero-mean circular symmetric complex Gaussian if \( \mathbf{X} \) is so. Hence for the purpose of maximizing mutual information, we take \( \mathbf{X} \) as zero mean circular symmetric complex Gaussian with covariance \( Q_X = \mathbb{Q} \). Thus with noise covariance tensor as identity tensor \( I_M \), we get

\[
I(\mathbf{X}; \mathbf{Y}) = \log \left[ \det (Q_N \ast Q_N \ast H + I_M) \right] (39)
\]

and the capacity is given by:

\[
\max_{Q} \left( \log \left[ \det (Q_N \ast Q_N \ast H + I_M) \right] \right) \\
\text{s.t.} \quad f(Q) \leq 0, \quad Q \succeq 0.
\]

where the inequality constraint \( f(Q) \leq 0 \) can represent a family of power constraints.

**A. FAMILY OF POWER CONSTRAINTS**

In a practical system, the transmit power constraints can span multiple domains. For instance, in a transmission scheme employing the space, time and frequency domains, instead of imposing power across the tensor symbol, the power constraint might be on each antenna, or each antenna and time slot, or each antenna, time slot and frequency bin. In our framework, we have the flexibility of mathematically representing any such power constraint.

First we introduce some notations for a simplification. We denote the sequence of indices \((i_1, i_2, \ldots, i_N)\) as \(\mathbf{i}\). Let \(\mathbf{L}_c\) denotes the sequence of indices indicating tensor symbol elements under power constraint and let \(\mathbf{L}_r\) represents the rest of the indices in \(\mathbf{i}\) which are not in \(\mathbf{L}_c\). For example, in an order 5 tensor of size \(I_1 \times I_2 \times I_3 \times I_4 \times I_5\), we have \(\mathbf{i} = (i_1, i_2, i_3, i_4, i_5)\). Let the domains which are under individual power constraints be 2 and 3, then \(\mathbf{L}_c = (i_2, i_3)\) and \(\mathbf{L}_r = (i_1, i_4, i_5)\). With this choice of notations, we will write \(\sum_{i_2 = 1}^{I_2} \sum_{i_3 = 1}^{I_3} Q_{i_{1,\ldots,i_{11}},i_{4\ldots,i_{14}}} \) as simply \(\sum_{\mathbf{L}_c}\). Notations corresponding to cases when either \(\mathbf{L}_c\) or \(\mathbf{L}_r\) is empty are explained in Table 1.

**TABLE 1: Simplified Notation for indices**

| Cases | \(\sum_{\mathbf{L}_c} Q_{i_{1,\ldots,i_{11}},i_{4\ldots,i_{14}}} \) | \(P_{L_r}\) denotes | Interpretation |
|-------|---------------------------------|------------------------|-----------------|
| \(\mathbf{L}_c\) is empty and \(\mathbf{L}_r = i\) | \(\sum_{\mathbf{L}_c} Q_{i_{1,\ldots,i_{11}},i_{4\ldots,i_{14}}} = P_{\mathbf{L}_c}\) | \(\mathbf{L}_c\) | sum power constraint |
| \(\mathbf{L}_r\) is empty and \(\mathbf{L}_c = i\) | \(Q_{i_{1,\ldots,i_{11}},i_{4\ldots,i_{14}}} = P_{\mathbf{L}_r}\) | \(\mathbf{L}_r\) | per element constraints |

Using these simplified notations, we will now describe a family of optimization problems to find capacity which can cover different types of power constraint, as follows:

\[
\max_{Q} \left( \log \left[ \det (Q_N \ast Q_N \ast H + I_M) \right] \right) (41)
\]

\[
\text{s.t.} \quad \sum_{\mathbf{L}_c} Q_{i_{1,\ldots,i_{11}},i_{4\ldots,i_{14}}} \leq P_{\mathbf{L}_c}, \quad \forall \mathbf{L}_c, \quad \mathbf{Q} \succeq 0. (42)
\]

To understand how the above framework represents a large variety of constraints, let us consider a few specific cases. The case \(\mathbf{L}_c = \mathbf{L}_r\) hence \(\mathbf{L}_c\) is empty, will represent the situation where we have per element power constraints with \(P_{\mathbf{L}_c} = P_{i_1,\ldots,i_N}\) and (42) becomes:

\[
Q_{i_1,\ldots,i_N} \leq P_{i_1,\ldots,i_N}, \quad \forall i_1, i_2, \ldots, i_N (44)
\]

When \(\mathbf{L}_c = i_K\), where \(K \leq N\), we have the case with per domain element constraint for \(K\)th domain where each element \(i_K\) has a different budget of \(P_{i_K}\), i.e. \(P_{\mathbf{L}_c} = P_{i_K}\) and (42) becomes:

\[
\sum_{i_1 = 1}^{I_1} \ldots \sum_{i_{K-1} = 1}^{I_{K-1}} \sum_{i_{K+1} = 1}^{I_{K+1}} \ldots \sum_{i_N = 1}^{I_N} Q_{i_1,\ldots,i_K,\ldots,i_N} \leq P_{i_K}, \quad \forall i_K (45)
\]

Now let’s assume we have power constraints on two domains \(K\) and \(L\) such that \(K < L \leq N\). Then \(\mathbf{L}_c = (i_K, i_L)\) and \(\mathbf{L}_r = (i_1, \ldots, i_{K-1}, i_{K+1}, \ldots, i_{L-1}, i_{L+1}, \ldots, i_N)\). In this case \(P_{\mathbf{L}_c} = P_{i_K,i_L}\) and (42) becomes:
Similarly, we can represent constraints on any number of domains. Lastly, let’s assume that \( \mathcal{L} \) is empty, hence \( \mathcal{L} = \emptyset \) and the power constraint translates to the sum power constraint, i.e. \( P_L = P \) and we get:

\[
\sum_{i_1=1}^{I_1} \ldots \sum_{i_N=1}^{I_N} Q_{i_1,\ldots,i_N} = \text{tr}(Q) \leq P
\]  

(47)

All these power constraints are linear, and the objective function in (41) is concave. Notice that the feasible set for this optimization problem is the set of positive semi-definite tensors satisfying the given power constraints which are linear. Hence the feasible set is convex. Thereby (41), (42) and (43) represent a family of convex optimization problems which can be solved using the Karush-Kuhn-Tucker (KKT) conditions [71]. Furthermore, since \( P_L \) are finite and non-negative, a simple choice of covariance tensor belonging to the feasible set could be a pseudo-diagonal tensor with all non-negative entries such that they satisfy the power constraints. So the feasible set is a non-empty convex set, and hence by Slater’s condition [71], strong duality holds and the optimal solution always exist. Next we will find the optimal solution using the KKT conditions.

### B. SOLUTION USING KKT CONDITIONS

Let \( \mathcal{M} \geq 0 \) be the Lagrange multiplier tensor for the positive semi-definite constraint from (43) of size \( I_1 \times \ldots \times I_N \). Let \( \mu_{\mathcal{L}} \geq 0 \) for all \( \mathcal{L} \) be the Lagrange multipliers corresponding to all the linear constraints from (42). Then the Lagrangian functional can be defined as:

\[
\mathcal{L}(Q, \mu_{\mathcal{L}}, \mathcal{M}) = -\log(\det(\mathcal{M}^{\ast} \mathcal{Q}^{\ast} \mathcal{M}^{\mathcal{H}} + \mathcal{J}_M)) + \sum_{\mathcal{L}} \mu_{\mathcal{L}} (\text{tr}(\mathcal{L}^{\mathcal{H}}) - P_L) - \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}) \]  

(48)

We arrange the values \( \{\mu_{\mathcal{L}}\} \) in a pseudo-diagonal tensor \( \mathcal{B} \) of same size as the input covariance such that its non-zero entries are \( \mathcal{B}_{i_1,\ldots,i_N} = \mu_{\mathcal{L}}, \forall \mathcal{L} \). For instance, if \( \mathcal{L} = (i_1, i_2) \), then \( \mathcal{B}_{i_1,\ldots,i_N, i_{N-1},\ldots,i_1} = \mu_{i_1,i_2} \) for all \( (i_3, \ldots, i_N) \). Then we get:

\[
\sum_{\mathcal{L}} \mu_{\mathcal{L}} \cdot \sum_{\mathcal{L}} Q_{i_1,\ldots,i_N} = \sum_{\mathcal{L}} \sum_{\mathcal{L}} \mu_{\mathcal{L}} = \sum_{\mathcal{L}} \mathcal{B}_{i_1,\ldots,i_N} \cdot Q_{i_1,\ldots,i_N} = \text{tr}(\mathcal{B}^{\ast} \mathcal{Q}) \]  

(49)

Based on (49), we can re-write the Lagrangian from (48) as:

\[
\mathcal{L}(Q, \mu_{\mathcal{L}}, \mathcal{M}) = -\log(\det(\mathcal{M}^{\ast} \mathcal{Q}^{\ast} \mathcal{M}^{\mathcal{H}} + \mathcal{J}_M)) - \sum_{\mathcal{L}} \mu_{\mathcal{L}} P_L + \text{tr}(\mathcal{B}^{\ast} \mathcal{Q}) - \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}) \]  

(50)

Based on (14), (15) and the definition of tensor gradient as presented in section II-B, the gradient of Lagrangian with respect to \( Q \) can be written as:

\[
\nabla_Q \mathcal{L} = -\mathcal{H}^{\mathcal{H}} \ast_{\mathcal{M}} (\mathcal{M}^{\mathcal{H}} + \mathcal{J}_M)^{-1} \mathcal{H}^{\ast} \mathcal{M} + \mathcal{B} - \mathcal{M}.
\]

(51)

Equating \( \nabla_Q \mathcal{L} \) from (51) to \( 0 \), we get the first KKT condition as:

\[
\mathcal{H}^{\mathcal{H}} \ast_{\mathcal{M}} (\mathcal{M}^{\mathcal{H}} + \mathcal{J}_M)^{-1} \mathcal{H} = \mathcal{B}. \]

(52)

The KKT equations also include complementary slackness condition corresponding to each constraint and its associated Lagrange multiplier [71]. For the linear constraints in (42), the definition of complementary slackness leads to:

\[
\mu_{\mathcal{L}} (\sum_{\mathcal{L}} Q_{i_1,\ldots,i_N} - P_L) = 0, \quad \forall \mathcal{L}.
\]

(53)

For the constraint in (43), based on the approach taken for semi-definite programming [71], the complementary slackness can be written as \( \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}) = 0 \). Since \( \mathcal{M}, \mathcal{Q} \geq 0 \), similar to matrix case as presented in [72], we have \( \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}) = 0 \Rightarrow \mathcal{M}, \mathcal{Q} = 0 \). Also, since \( \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}) = \text{tr}(\mathcal{M}^{\ast} \mathcal{Q}^{1/2}) \), the complementary slackness for the positive semi-definite constraint is written as:

\[
\mathcal{Q}^{1/2} \ast_{\mathcal{M}} \mathcal{Q}^{1/2} = 0 \]  

(54)

The tensor KKT conditions for the problem in (41)-(43) are given by (52), (53) and (54).

Notice that all the entries of \( \mathcal{B} \), i.e. \( \mu_{\mathcal{L}} \), will be strictly greater than 0 at optimum because the inequality constraint must be met with equality at optimum. So \( \mathcal{B} \) is a positive definite tensor, i.e. \( \mathcal{B} \succ 0 \) and hence invertible. Also since \( \mu_{\mathcal{L}} > 0 \), (53) can be written as:

\[
\sum_{\mathcal{L}} Q_{i_1,\ldots,i_N} - P_L = 0, \quad \forall \mathcal{L}.
\]

(55)

Let us define a tensor \( \tilde{\mathcal{H}} \in \mathbb{C}^{I_1 \times \ldots \times I_N} \) and its tensor EVD as:

\[
\tilde{\mathcal{H}} \triangleq \mathcal{B}^{-1/2} \ast_{\mathcal{N}} (\mathcal{H}^{\mathcal{H}} \ast_{\mathcal{M}} \mathcal{H}) \ast_{\mathcal{N}} \mathcal{B}^{-1/2} = \mathcal{V} \ast_{\mathcal{N}} \mathcal{D} \ast_{\mathcal{N}} \mathcal{V}^{\mathcal{H}}.
\]

(56)

We can use the result of Theorem 1 from [73] by extending it to tensor case to solve (52) and (54) subject to \( \mathcal{Q} \geq 0, \mathcal{M} \geq 0 \) and \( \mathcal{B} \succ 0 \) to obtain the optimal \( \mathcal{Q} \). The tensor version of Theorem 1 from [73] along with its proof has been included in Appendix A. Based on results from Appendix A, the optimal \( \mathcal{Q} \) is given by:

\[
\mathcal{Q}_{\text{opt}} = \mathcal{B}^{-1/2} \ast_{\mathcal{N}} \mathcal{V} \ast_{\mathcal{N}} (\mathcal{J}_N + \mathcal{D}^{-1}) \ast_{\mathcal{N}} \mathcal{V}^{\mathcal{H}} \ast_{\mathcal{N}} \mathcal{B}^{-1/2}
\]

(57)

where \( \mathcal{D} \) and \( \mathcal{V} \) are obtained through the tensor EVD of \( \tilde{\mathcal{H}} \) (56) and (2) denotes a pseudo-diagonal tensor whose all the pseudo-diagonal entries are non-negative, i.e. \( (\mathcal{Z}_{i_1,\ldots,i_N})^+ = \max\{0, \mathcal{Z}_{i_1,\ldots,i_N}\} \). Hence we can calculate the capacity as:

\[
C = \log \{ \det (\mathcal{H}^{\mathcal{H}} \ast_{\mathcal{M}} \mathcal{H}^{\ast} + \mathcal{J}_M) \}
\]
Let $N_{\mu}$ denote the number of values that $\bar{I}_r$ can take. For example, if $\bar{I}_r = (i_1, \ldots, i_N)$, then $N_{\mu} = 1 \cdot I_1 \cdot I_L$; if $\bar{I}_r = (i_K, i_L)$, then $N_{\mu} = I_K \cdot I_L$. Since $B$ contains $\mu_{\bar{I}}$ on its pseudo-diagonal with each $\mu_{\bar{I}}$ appearing exactly $N_{\mu}$ number of times, from (63) we can write:

$$N_{\mu} \cdot \mu_{\bar{I}}^{-1} - \sum_{\bar{I}} ((C_{\mu}^H \cdot \mathcal{H})^{-1})_{\bar{I}} \bar{I} = P_{\mu} \quad \forall \bar{I}_r,$$

which gives us all the entries of $B$. The proposed solution in (65), (61) and (60) assumes high SNR as we have ignored the $(\cdot)^+$ operation. To extend the solution for any SNR, we now present a scaling approach to approximate the input covariance tensor at any SNR setting, and verify that the resulting covariance satisfies the constraints. If the covariance $\Omega$ obtained from (60) has negative eigenvalues, then we force the negative eigenvalues of $\Omega$ to be zero. If the tensor EVD of $\Omega$ is given as $\mathcal{U} \ast \mathcal{D} \ast \mathcal{U}_H$, the pseudo-diagonal elements of $\Omega$ can be written as, $\Omega_{\bar{I}} = \sum_{\bar{I}} U_{\bar{I}} U_{\bar{I}}^H D_{\bar{I}} D_{\bar{I}}^H U_{\bar{I}}^H$. Thus the brute force approach of setting the negative values in $D$ to zero can result into larger values at the pseudo-diagonal of $\Omega$. This in turn can make the solution infeasible, i.e. the power constraint $P_{\bar{I}}$ from (42) might not be met and we may get a different power allotted say $P'_{\bar{I}}$ where $P'_{\bar{I}} \geq P_{\bar{I}}$. Note that

$$P_{\bar{I}}' = \sum_{\bar{I}} \Omega_{\bar{I}}' \bar{I} \quad \forall \bar{I}_r.$$

So we scale the resulting $\Omega$ using another pseudo-diagonal tensor $\mathcal{S}$ such that the power constraints remain satisfied, i.e. $\Omega_{\text{scaled}} = \mathcal{S} \ast \Omega \ast \mathcal{S}^H$ where the pseudo-diagonal entries of scaling tensor $\mathcal{S}$ are given as:

$$S_{\bar{I}} \bar{I} = \sqrt{P_{\bar{I}}'}, \quad \forall \bar{I}_r.$$

such that the pseudo-diagonal entries of $\Omega_{\text{scaled}}$ become:

$$(\Omega_{\text{scaled}})_{\bar{I}} \bar{I} = \Omega_{\bar{I}}' \bar{I} \cdot \frac{P_{\bar{I}}}{P_{\bar{I}}'}.$$

Hence, based on (66) we have

$$\sum_{\bar{I}} (\Omega_{\text{scaled}})_{\bar{I}} \bar{I} = \sum_{\bar{I}} \Omega_{\bar{I}}' \bar{I} \cdot \frac{P_{\bar{I}}}{P_{\bar{I}}'} = P_{\bar{I}} \quad \forall \bar{I}_r.$$

Thus the choice of scaling operation ensures that $\Omega_{\text{scaled}}$ satisfies the power constraints. A similar technique has been used for matrix-field water filling in [74] where the diagonal entries of the covariance matrix are scaled to ensure that the resulting matrix is positive semi-definite. Such a scaling approach simplifies the computation of the input covariance but also makes it sub-optimal and hence leads to an approximation of the capacity. However, this approximation gets better as SNR grows and is exact at sufficiently high SNR. The procedure is systematically presented in Algorithm 1. Finding the capacity is a convex optimization problem.
hence can be solved using software tools such as CVX [75], which can be compared with the capacity obtained via the approximation in Algorithm 1 to assess the validity of the proposed approach. We present such a comparison in the numerical examples (Figure 9) to illustrate the accuracy of the proposed approach.

Algorithm 1 Finding the Input Covariance tensor

1: Input $\mathcal{H}, P_L, N_L$
2: Initialise $\text{flag} \leftarrow 0$
3: Find pseudo-diagonal elements of $\mathcal{B}$ by calculating $\mu_{\mathcal{L}}$ using (65)
4: Calculate $\mathcal{Q}$ using (60).
5: Perform tensor EVD of $\mathcal{Q} = U \ast N \mathcal{D} \ast N U^H$
6: for all $i_1, i_2, \ldots, i_N$
7: if $\mathcal{D}_{i_1, \ldots, i_N} < 0$
8: flag $\leftarrow 1$
9: end if
10: end for
11: if $\text{flag} == 1$
12: Update $\mathcal{Q} \leftarrow U \ast N \mathcal{D} \ast N U^H$
13: Calculate $P_U^Q$ using (66)
14: Find pseudo-diagonal tensor $\mathcal{S}$ using (67)
15: Update $\mathcal{Q} \leftarrow \mathcal{S} \ast N \mathcal{Q} \ast N \mathcal{S}^H$
16: end if
17: return $\mathcal{Q}$

C. COMPLEXITY ANALYSIS OF ALGORITHM 1

Algorithm 1 essentially approximates the optimal input covariance tensor using (60) and a scaling process which ensures a feasible solution. The algorithm requires fixed amount of computational resources and can be deployed off line. Thus it is of interest to analyze the computational resources required to execute this algorithm. Modern day computational strategies often employ cloud services provided by suppliers such as AWS and Azure. Such cloud services often provide both the infrastructure and the platform/software as services on demand [76], [77], where the cost depends on the amount of required computational resources and time of execution. Given that Algorithm 1 requires extensive tensor operations which scale with the tensor size, using cloud services to implement it is an appealing option. Such cloud services provide several parallel and distributed computing infrastructures for faster and efficient computations [78]. Thus depending on the platform being considered and the amount of multi-core processors being employed for such implementation, the time of execution of the algorithm can significantly differ. However, independent of the computing infrastructure available, a suitable measure of the computational complexity can be specified in terms of the required number of mathematical operations to be performed in a given algorithm, as used in [79], [80]. Hence in this section, we analyze the computational complexity of Algorithm 1 in terms of the required number of flops for a given step. A flop is defined as a single floating point operation such as an addition, multiplication, subtraction, division, comparison ($>, <, ==$), or a scalar square root etc.

The complexity of Algorithm 1 primarily depends on finding the tensor inversion and the tensor EVD for a tensor of size $I_1 \times \ldots \times I_N \times I_1 \times \ldots \times I_N$. Such tensor operations can be performed using various algorithms by employing the Einstein product as discussed in [18], [44], [49], [81]. Consider the definition of Einstein product across $N$ modes between tensor $\mathcal{A}$ of size $I_1 \times \ldots \times I_P \times K_1 \times \ldots \times K_N$ and tensor $\mathcal{B}$ of size $K_1 \times \ldots \times K_N \times J_1 \times \ldots \times J_N$ from (3). Based on (3), computing a single element in $\mathcal{A} \ast \mathcal{B}$ requires $K_1 \cdot K_2 \cdot \ldots \cdot K_N$ multiplications and also $K_1 \cdot K_2 \cdot \ldots \cdot K_N - 1$ additions. There are total $I_1 \cdot I_2 \cdot \ldots \cdot I_P \cdot J_1 \cdot J_2 \cdot \ldots \cdot J_M$ elements in the tensor $\mathcal{A} \ast \mathcal{B}$. For ease of notation, we use $I = \prod_{p=1}^P I_p$, $J = \prod_{j=1}^M J_j$, and $K = \prod_{k=1}^N K_k$. So there are total $IJ$ elements in $\mathcal{A} \ast \mathcal{B}$ where each element is computed using $K$ multiplications and $K - 1$ additions. Subsequently finding all the elements of $\mathcal{A} \ast \mathcal{B}$ requires $IJK$ multiplications and $(K - 1)IJ$ additions. Hence a total of $IJK + (K - 1)IJ = (K - 1)IJ$ flops are required for computing all the elements in $\mathcal{A} \ast \mathcal{B}$. Note that for any operation where the number of required flops is polynomial in its size $n$, i.e. the operation requires $an^p + a_1n^{p-1} + \ldots + a_p$ flops (for fixed constants $a_0, a_1, \ldots, a_p$), we represent the complexity only using the highest power in $n$ as $O(n^p)$. Hence the complexity of the Einstein product which requires $(2K - 1)IJ$ flops is given as $O((IKJ)^3)$ which is same as $O((I_1 \cdot \ldots \cdot I_P) \cdot (K_1 \cdot \ldots \cdot K_N) \cdot (J_1 \cdot \ldots \cdot J_M))$. Thus for the specific case where both $\mathcal{A}$ and $\mathcal{B}$ are order $2N$ tensors of size $I_1 \times \ldots \times I_N \times I_1 \times \ldots \times I_N$ each, the complexity of $\mathcal{A} \ast \mathcal{B}$ is given as $O((I_1 \cdot \ldots \cdot I_N)^3)$.

Now assume channel $\mathcal{H}$ is of size $J_1 \times \ldots \times J_M \times I_1 \times \ldots \times I_N$ and $N_L$ denotes the number of values that $\mathcal{L}$ can take. In Algorithm 1, the first two steps are input initialization. Step 3 requires computing the Einstein product over the common $M$ modes of $\mathcal{H}^H$ and $\mathcal{H}$, which based on the discussion in the previous paragraph has a complexity of $O((I_1 \cdot \ldots \cdot I_N)^2(J_1 \cdot \ldots \cdot J_M))$. Further within step 3, it is required to find the inverse of $\mathcal{H}^H \ast \mathcal{H}$ which is an order $2N$ tensor. The inverse of an order $2N$ tensor can be calculated using the higher order bi-conjugate gradient (HOBG) method described in [18] or Newton’s method (NM) which solves an iterative equation involving Einstein product between tensors of size $I_1 \times \ldots \times I_N \times I_1 \times \ldots \times I_N$. Thus each iteration in NM has a computational cost of $O((I_1 \cdot \ldots \cdot I_N)^3)$ [81]. It is important to note that the complexity of such iterative methods depends on the number of iterations, which in turn depends on the desired accuracy level set to achieve convergence. It was shown in [81] that NM requires a lower number of iterations as compared to HOBG to reach the same accuracy. The NM requires $O(\log(I_1 \cdot \ldots \cdot I_N))$ iterations to converge to a fixed error bound [14]. Furthermore, [14] shows a method to reduce the per-iteration complexity of NM, and thus perform tensor inversion in $O(\log^2(I_1 \cdot \ldots \cdot I_N))$ parallel time units. Also, non-iterative methods such as Gauss elimination based
on triangular decomposition of tensors [49] can be used for tensor inversion which requires a computational complexity of \( O(I_1 \cdots I_N)^3 \). Hence, the worst case complexity of tensor inversion without any use of parallel processors is \( O(I_1 \cdots I_N)^3 \). Eventually step 3 calculates each \( \mu_\ell \) using (65) which requires \( N_\ell + 1 \) additions and 1 division, and this needs to be done for all the \( N_\ell \) values that \( \ell \) can take. Thus this step requires \( (N_\ell + 2) \cdot N_\ell \) flops and its complexity is \( O(N_\ell \cdot N_\ell) \). Note that since \( N_\ell \cdot N_\ell = I_1 \cdots I_N \), thus complexity of finding \( \mu_\ell \) can be written as \( O(I_1 \cdots I_N) \). Step 4 which finds \( Q \) using (60) subtracts an order 2\( N \) tensor from a pseudo-diagonal tensor. Since the number of pseudo-diagonal elements are \( I_1 \cdots I_N \), step 4 performs \( I_1 \cdots I_N \) subtractions and thus has a complexity of \( O(I_1 \cdots I_N) \). Further step 5 finds the tensor EVD of an order \( 2 N \) tensor \( Q \). The complexity of finding the EVD of a tensor of size \( I_1 \times \cdots \times I_N \times I_1 \times \cdots \times I_N \) using the Einstein product properties is \( O((I_1 \cdots I_N)^3) \) [16]. Algorithms which generalize the matrix eigen decomposition approaches to tensor EVD using the Einstein product properties can be found in [44], [49], [16, Algorithm C.2]. Steps 6 to 11 essentially perform the operation \( max(0, D_{ij...n}) \) on each of the \( I_1 \cdots I_N \) pseudo-diagonal elements of the tensor \( D \). Hence it has a complexity of \( O(I_1 \cdots I_N) \). Step 12 is just a single scalar comparison, and step 13 updates \( \mathcal{Q} \) using the Einstein product between tensors of order \( 2N \) with size \( I_1 \times \cdots \times I_N \times I_1 \times \cdots \times I_N \) for which the complexity is \( O((I_1 \cdots I_N)^3) \). Step 14 finds \( P^\ell_\ell \) for all the \( N_\ell \) values of \( \ell \). Thus it performs \( N_\ell \) additions for each of the \( N_\ell \) values that \( \ell \) can take. Hence step 14 requires \( N_\ell \cdot N_\ell \) flops and thus has a complexity of \( O(N_\ell \cdot N_\ell) \) which is same as \( O(I_1 \cdots I_N) \). Step 15 calculates the scaling factor for all \( \ell \), thus performs \( N_\ell \) divisions and square roots. Hence it has a complexity of \( O(N_\ell) \). Finally, step 16 updates \( \mathcal{Q} \) using the Einstein product between tensors of order \( 2 N \) and thus has a complexity of \( O((I_1 \cdots I_N)^3) \).

### TABLE 2: Computational Complexity of Algorithm 1

| Step | Operation | Complexity |
|------|-----------|------------|
| 3    | Find \( \mathcal{H} * M \mathcal{H} \) | \( O((I_1 \cdots I_N)^2 \cdot J_1 \cdots J_M) \) |
| 3    | Find \( \mathcal{H} * M \mathcal{H}^{-1} \) | \( O((I_1 \cdots I_N)^3) \) |
| 3    | Find \( \mu_\ell \) using (65) | \( O(I_1 \cdots I_N) \) |
| 4    | Find \( \mathcal{Q} \) using (60) | \( O(I_1 \cdots I_N) \) |
| 5    | EVD of \( \mathcal{Q} \) | \( O((I_1 \cdots I_N)^3) \) |
| 6-11 | Check \( \mathcal{D} \) | \( O(I_1 \cdots I_N) \) |
| 13   | Update \( \mathcal{Q} \) | \( O(I_1 \cdots I_N)^3 \) |
| 14   | Find \( P^\ell_\ell \) using (66) | \( O(I_1 \cdots I_N) \) |
| 15   | Find \( S \) using (67) | \( O(N_\ell) \) |
| 16   | Update \( \mathcal{Q} \) | \( O((I_1 \cdots I_N)^3) \) |

Table 2 summarizes these step by step computational complexity cost of Algorithm 1. The first column in Table 2 indicates the step number from Algorithm 1, second column describes the operation and third column states the complexity. We can observe that all the entries in complexity column of Table 2, have a complexity order of 3 (cubic) or less in \( I_1 \cdots I_N \) except the first operation which has a complexity of \( O((I_1 \cdots I_N)^2 \cdot J_1 \cdots J_M) \). Hence on summing all the entries of the third column in Table 2, we see that the overall complexity of Algorithm 1 is given as \( O((I_1 \cdots I_N)^2 \cdot J_1 \cdots J_M) + O((I_1 \cdots I_N)^3) \). Further in the case when \( J_1 \cdots J_M \leq I_1 \cdots I_N \), the complexity of Algorithm 1 can be written as \( O((I_1 \cdots I_N)^3) \).

Note that the steps in Algorithm 1 with complexity of \( O((I_1 \cdots I_N)^3) \) primarily rely on the Einstein product operation. However, since this algorithm can be executed on multi-core computer systems, the time complexity of performing all the operations in Einstein product can be significantly reduced by making use of parallel processing. Several flops in the Einstein product can be executed simultaneously on a parallel computing platform. To see this, assume tensors \( \mathcal{X} \) and \( \mathcal{Y} \) of size \( I_1 \times \cdots \times I_N \times I_1 \times \cdots \times I_N \) each and \( \mathcal{Z} = \mathcal{X} \ast_N \mathcal{Y} \). Then all the elements of \( \mathcal{Z} \) can be computed using multiple processors as shown in Figure 1.

The white rectangular nodes in Figure 1 correspond to the individual multiplication operations. All the white nodes need to be added to compute a single element (the gray rectangular node) in tensor \( \mathcal{Z} \). The addition of white nodes to generate the gray nodes can be done using a binary tree approach as shown in Figure 1 where all the addition operations at a given level of the tree are performed simultaneously. The figure illustrates this process for a single gray node, but further all the gray nodes can be computed simultaneously using similar binary tree approach if multiple processors are available.

For a specific gray node, the number of nodes in a binary tree at each level starting from root (gray node) are \( 2^0, 2^1, 2^2, \ldots, 2^h \) where \( h \) is the depth of the tree and \( 2^h \) is the number of leaf nodes. We can have similar binary trees for each of the gray nodes. Since in Figure 1, the number of leaf nodes (white nodes) are \( I_1 \cdots I_N \), we get that the depth of the tree corresponding to each gray node is \( \lceil \log(I_1 \cdots I_N) \rceil \). We use the ceil operator as \( I_1 \cdots I_N \) may not always be a power of 2. Hence we can say that the height of the tree is \( O(\log(I_1 \cdots I_N)) \). Since all the gray nodes can be computed simultaneously, all the individual elements of \( \mathcal{X} \ast_N \mathcal{Y} \) can be calculated in \( O(\log(I_1 \cdots I_N)) \) parallel time units. Such a parallel processing method can significantly reduce the time complexity of calculating the Einstein product, and subsequently other tensor operations which rely on the Einstein product such as tensor inversion and EVD. The other steps in Algorithm 1 which have a complexity of \( O(I_1 \cdots I_N) \) or less (step 15), can also be performed faster using parallel processors. For example in step 3, all the \( \mu_\ell \) of (65) for different \( \ell \) can be calculated simultaneously on parallel processors. Similarly all the \( I_1 \cdots I_N \) operations in steps 4 and 6-11, can be performed simultaneously. In steps 14 and 15, the \( P^\ell_\ell \) from (66) and the scaling factors from (67) for all the \( \ell \) can also be computed simultaneously. Hence Algorithm 1 can be suitably adapted to run on parallel processing multi-core computer systems depending on the number of processors available.
D. COMPARING DIFFERENT CONSTRAINTS

Let the set of all possible positive semi-definite tensors of size $I_1 \times \ldots \times I_N \times I_1 \times \ldots \times I_N$ be represented by $Q$. Let the feasible set for the optimization problem in (41)-(43) for two different settings $\mathcal{L}_1$ and $\mathcal{L}_2$ be $Q_1$ and $Q_2$ respectively. Assume $\mathcal{L}_1$ is a subsequence of $\mathcal{L}_2$. For instance, let $\mathcal{L}_2 = (i_1, i_2, i_3)$, and $\mathcal{L}_1 = (i_1, i_2)$. So $Q_1$ represents a set of all positive semi-definite tensors $Q$ which satisfies

$$\sum_{i_3, i_4, \ldots, i_N} Q_{i_1, \ldots, i_N, i_1, \ldots, i_N} \leq P_{i_1, i_2} \ \forall (i_1, i_2)$$

(70)

and $Q_2$ represents a set of all positive semi-definite tensors $Q$ which satisfies

$$\sum_{i_4, \ldots, i_N} Q_{i_1, \ldots, i_N, i_1, \ldots, i_N} \leq P_{i_1, i_2, i_3} \ \forall (i_1, i_2, i_3).$$

(71)

In (70), the first two domains are under power constraints, whereas in (71), the power constraints span the third domain as well with $\sum_{i_3} P_{i_1, i_2, i_3} = P_{i_1, i_2}$. Summing over $i_3$ in (71) gives (70). Hence every $Q$ that satisfies (71) will also satisfy (70), showing that the set of $Q$ satisfying (71) is a subset of the set of $Q$ satisfying (70), i.e. $Q_2 \subseteq Q_1$.

Let the optimal value of the objective function in the optimization problem in (41)-(43) for set $Q_1$ be $C_1$ and for set $Q_2$ be $C_2$. From the basic principles of convex optimization [71], it is known that a globally optimal point is also locally optimal. Hence if $C_1$ is the maximum of the objective function over the set of constraints $Q_1$, then $C_1$ is also the maximum of the objective function over $Q_2$ since $Q_2 \subseteq Q_1$. Hence $C_2 \leq C_1$, where equality is possible if the optimal $Q$ lies in the set $Q_2$. This holds for any configuration of $\mathcal{L}_1$ and $\mathcal{L}_2$ so far as $\mathcal{L}_1$ is a subsequence of $\mathcal{L}_2$. Essentially, as more domains are put under constraints, the feasible set for the optimization problem shrinks, possibly lowering the achievable capacity.

For instance, consider a $2 \times 2$ input where the two domains are antenna and time slots. Let the capacity achieved under total power constraint $P$, be $C_1$. The capacity achieved under per antenna power constraints of $P_1$ for antenna 1 and $P_2$ for antenna 2 such that $P_1 + P_2 = P$ be $C_2$, then since the set of feasible solution with per antenna power constraints is a subset of the set of feasible solution with total power constraint we have $C_2 \leq C_1$. Similarly, capacity achieved under power constraints per element, i.e. $P_{11}, P_{12}, P_{21}, P_{22}$ where $P_{1,j}$ represents power budget on $j$th antenna and $j$th time slot, such that $P_{11} + P_{12} = P_1, P_{21}, P_{22} = P_2$ be $C_3$, then $C_3 \leq C_2$. This has also been shown in [88] for a MIMO channel where capacity under per antenna power constraint is always smaller than the capacity under sum power constraint.

E. CAPACITY UNDER SUM POWER CONSTRAINTS

Under the sum power constraint of (47), $\mathcal{L}_c$ is empty and hence there is a single Lagrange multiplier $\mu$ associated with the constraint (47). Hence the tensor $B$, which contains the Lagrange multipliers on its pseudo-diagonal, will be a scaled identity tensor. Substituting $B = \mu I_N$ in (56) gives $\mathcal{H} = \mu^{-1} \cdot (\mathcal{H}^H * M \mathcal{H}) = \mathcal{V} * N (\mu^{-1} \cdot \mathcal{D}) * N \mathcal{V}^H$, and

FIGURE 1: Parallel execution of Einstein product
subsequently (57) becomes

\[ Q = \mathcal{V}^{*}N \left( \mu^{-1}J_{N} - D^{-1} \right)^{+} \mathcal{V}^{H}. \]  

(72)

Substituting \( \bar{d}_{1i,\ldots,i_{N}} = d_{1i,\ldots,i_{N}}/\mu \) in (59) gives:

\[
C = \sum_{i_{1},\ldots,i_{N}} \log \left[ \left( 1 - \frac{1}{d_{1i,\ldots,i_{N}}} \right)^{+} \cdot d_{1i,\ldots,i_{N}} + 1 \right]
= \sum_{i_{1},\ldots,i_{N}} \left( \log \left( \frac{d_{1i,\ldots,i_{N}}}{\mu} \right) \right)^{+} \tag{73}
\]

where \( a^{+} \) denotes \( \max \{0, a\} \), \( d_{1i,\ldots,i_{N}} \) are the non-zero eigenvalues of \( \mathcal{H}^{H} \mathcal{M} \mathcal{H} \) and \( 1/\mu \) is chosen to satisfy (55)

\[
\text{tr}(Q) = \sum_{i_{1},\ldots,i_{N}} \left( \frac{1}{\mu} - \frac{1}{d_{1i,\ldots,i_{N}}} \right)^{+} = P. \tag{74}
\]

The optimum covariance derived in (72) is a generalization of water-filling solution for the MIMO matrix channel to multiple domains. Hence under sum power constraint, we perform the tensor EVD of \( \mathcal{H}^{H} \mathcal{M} \mathcal{H} \) to get \( \mathcal{V} \) and \( \mathcal{D} \). Further, we use (74) to find \( \mu \) and subsequently use (73) to find the capacity.

### F. MULTIPLEXING GAIN

We can also characterize the capacity contribution by each constrained domain separately and the multiplexing gain under various power constraints. For a fixed tensor channel, the tensor \( \mathcal{B}^{-1} \) is pseudo-diagonal with entries \( \mu_{i_{1}}^{-1} \), hence

\[
\det(\mathcal{B}^{-1}) = \prod_{i} \mu_{I_{i}}^{-1} = \prod_{i} \left( \prod_{i_{1}} \mu_{I_{i_{1}}}^{-1} \right)^{N_{I_{i}}}. \tag{75}
\]

Also \( \det(\mathcal{H}^{H} \mathcal{M} \mathcal{H}) = \prod_{i} d_{I_{i}} \) where \( d_{I_{i}} \) are eigenvalues of \( \mathcal{H}^{H} \mathcal{M} \mathcal{H} \). Hence using (61) we have

\[
C = \log \left[ \prod_{I_{i}} \frac{d_{I_{i}}}{\mu_{I_{i}}} \right] = \sum_{I_{i}} \log \frac{d_{I_{i}}}{\mu_{I_{i}}} = \sum_{I_{i}} \left( \sum_{i_{2},\ldots,i_{N}} \log \frac{d_{I_{i}}}{\mu_{I_{i}}} \right) = \sum_{I_{i}} C_{I_{i}} \tag{76}
\]

where \( C_{I_{i}} = \sum_{i_{2},\ldots,i_{N}} \log \frac{d_{I_{i}}}{\mu_{I_{i}}} \) can be seen as the contribution of the \( i_{i} \)th element of the constrained domain to the overall capacity. For instance if the first domain refers to space domain, then \( C_{I_{1}} \) is the capacity contribution of the \( i_{1} \)th antenna. For any general case where \( \mathcal{L} \) contains the indices of domains under constraint, we can write:

\[
C = \log \left[ \prod_{\mathcal{L} \in \mathcal{L}_{I_{i}}} \frac{d_{\mathcal{L}}}{\mu_{\mathcal{L}}} \right] = \sum_{\mathcal{L}} \left( \sum_{\mathcal{L}} \log \frac{d_{\mathcal{L}}}{\mu_{\mathcal{L}}} \right) = \sum_{\mathcal{L}} C_{\mathcal{L}} \tag{77}
\]

where \( C_{\mathcal{L}} = \sum_{\mathcal{L}} \log \frac{d_{\mathcal{L}}}{\mu_{\mathcal{L}}} \) can be seen as the contribution of the \( \mathcal{L}_{i} \)th element of the constrained domains to the overall capacity. Substituting \( \mu_{\mathcal{L}} \) from (65) into \( C_{\mathcal{L}} \), we can further write:

\[
C_{\mathcal{L}}(P_{\mathcal{L}}) = \sum_{\mathcal{L}} \log \frac{d_{\mathcal{L}}}{\mu_{\mathcal{L}}} \tag{78}
= \sum_{\mathcal{L}} \log \left( \frac{d_{\mathcal{L}}}{P_{\mathcal{L}}} + \sum_{\mathcal{L}} (\mathcal{H}^{H} \mathcal{M} \mathcal{H})^{-1}_{\mathcal{L}} \right) \tag{79}
= \sum_{\mathcal{L}} \log \left( P_{\mathcal{L}} + \sum_{\mathcal{L}} (\mathcal{H}^{H} \mathcal{M} \mathcal{H})^{-1}_{\mathcal{L}} \right) + \log \left( \frac{d_{\mathcal{L}}}{N_{\mathcal{L}}} \right) \tag{80}
\]

We can write the multiplexing gain provided by \( \mathcal{L}_{i} \)th constrained domain as:

\[
\chi_{\mathcal{L}} = \lim_{P_{\mathcal{L}} \to \infty} \frac{C_{\mathcal{L}}(P_{\mathcal{L}})}{\log(P_{\mathcal{L}})} \tag{81}
\]

Since for large \( P_{\mathcal{L}} \), we have \( \log \left( P_{\mathcal{L}} + \sum_{\mathcal{L}} (\mathcal{H}^{H} \mathcal{M} \mathcal{H})^{-1}_{\mathcal{L}} \right) \approx \log(P_{\mathcal{L}}) \), hence using (81) and (80) we get

\[
\chi_{\mathcal{L}} = \lim_{P_{\mathcal{L}} \to \infty} \sum_{\mathcal{L}} \left[ \log \left( \frac{P_{\mathcal{L}}}{N_{\mathcal{L}}} \right) + \log \left( \frac{d_{\mathcal{L}}}{N_{\mathcal{L}}} \right) \right] \tag{82}
= \lim_{P_{\mathcal{L}} \to \infty} \sum_{\mathcal{L}} \log \left( \frac{P_{\mathcal{L}}}{N_{\mathcal{L}}} \right) + \log \left( \frac{d_{\mathcal{L}}}{N_{\mathcal{L}}} \right) \tag{83}
= \sum_{\mathcal{L}} \frac{C_{\mathcal{L}}}{\log(P_{\mathcal{L}})} \tag{84}
\]

In general since \( N_{\mathcal{L}} \) represents the product of dimensions of the domains not under individual constraints, it increases exponentially in the number of unconstrained domains. As a result, for tensor channels the multiplexing gain increases exponentially with the increase in the number of unconstrained domains. Note that in deriving the multiplexing gain, for simplicity we have assumed that all the eigenvalues of \( \mathcal{H}^{H} \mathcal{M} \mathcal{H} \) are non-zero, which may not always be the case. In general, the capacity is a function of the given channel’s specific singular values, some of which may be zero. In that case depending on which and how many singular values are zero, the multiplexing gain will be different and may not increase exponentially with increase in domains.

Based on (84), the multiplexing gain achieved under sum power constraint is \( I_{1} \cdot I_{2} \cdot \cdots \cdot I_{N} \). However (84) assumes that inverse of \( (\mathcal{H}^{H} \mathcal{M} \mathcal{H}) \) exists which will be the case if all its eigenvalues are non-zero. If that is not the case, then an approximation of the inverse \( (\mathcal{H}^{H} \mathcal{M} \mathcal{H}) \) can be calculated using the Higher order Bi-conjugate Gradient method described in [18]. Next we analyze the multiplexing gain, also known as pre-log, associated with a tensor channel under sum power constraints. Note that the number of non-zero eigenvalues of \( (\mathcal{H}^{H} \mathcal{M} \mathcal{H}) \) will be same as the number of non-zero singular values of \( \mathcal{H} \). Let \( I_{1} \cdot I_{2} \cdot \cdots \cdot I_{N} = I \) and
J_1 \cdot J_2 \cdots J_M = J$, then the number of non-zero singular values $R$ are less than or equal to $I$ and $J$, i.e. $R \leq \min\{I, J\}$ where equality is met if all the the singular values of $\mathcal{H}$ are non-zero.

The pre-log, denoted by $\chi$, is calculated at a very high SNR, for which water-filling reduces to uniform power allocation over the non-zero eigen channels, i.e. $\left(\frac{1}{\mu} - \frac{1}{d_{i_1, \ldots, i_N}}\right) \approx P/R$. Hence using (73), the capacity becomes

$$C = \sum_{i_1, \ldots, i_N} \left( \log \left(1 + \frac{P}{R}d_{i_1, \ldots, i_N}\right) \right)$$

As $P \to \infty$, (85) simplifies to:

$$C = \sum_{i_1, \ldots, i_N} \log \left(\frac{P}{R}d_{i_1, \ldots, i_N}\right)$$

$$= \sum_{i_1, \ldots, i_N} \left[ \log(P) + \log\left(\frac{d_{i_1, \ldots, i_N}}{R}\right) \right]$$

$$= R \log(P) + \sum_{i_1, \ldots, i_N} \log\left(\frac{d_{i_1, \ldots, i_N}}{R}\right)$$

$$\Rightarrow \chi = \lim_{P \to \infty} \frac{C}{\log(P)} = R$$

(89)

Assuming all channel singular values are non-zero, we have $R = \min\{I_1, I_2 \cdots I_N, J_1, J_2 \cdots J_M\}$. For a conventional MIMO matrix channel, capacity pre-log is known to be less than or equal to the minimum of the number of transmit and receive antennas [21] which is the specific case of the tensor pre-log. For MIMO matrix channel $N = M = 1$, and we have $\chi = \min\{I_1, J_1\}$ where $I_1$ and $J_1$ are the number of transmit and receive antennas respectively. For a tensor case, it is interesting to see that assuming equal dimension values on each domain i.e. $I_1 = I_2 = \cdots = I_N = J_1 = J_2 \cdots J_M = L$, then the capacity pre-log can be given as:

$$\chi = \min\{I_1, I_2 \cdots I_N, J_1, J_2 \cdots J_M\} = \min\{L^N, L^M\} = L^{\min\{N,M\}}$$

(90)

which is exponential in the number of domains.

VI. NUMERICAL EXAMPLES AND APPLICATIONS

In this section, we present numerical examples to illustrate previous results. We also present an example of a MIMO GFDM system modelled using the tensor framework.

A. EXAMPLES OF ERGODIC CAPACITY WHEN CHANNEL REALIZATIONS ARE KNOWN

Our results assume that the channel is deterministic. For the numerical examples, rather than using a specific channel tensor, we generate the channel using the Rayleigh model as in [21], [89], [90]. The channel tensor consists of realizations of i.i.d. circular complex Gaussian entries of zero mean and unit variance such that $\mathbb{E}[\mathcal{H}_{J_1, \ldots, J_M; i_1, \ldots, i_N}^2] = 1$ [89]. These channel realizations are known at the transmitter and receiver.

Let the capacity of a deterministic tensor channel $\mathcal{H}$ be denoted as $C(\mathcal{H})$. Assume now that we have capacities of $K$ such channels denoted by $C(\mathcal{H}^{(k)})$ for $k = 1, \ldots, K$ where the tensor $\mathcal{H}^{(k)}$ consist of realizations of complex Gaussian random variables. The average capacity of $K$ such deterministic channels is given by $C_K = \frac{1}{K} \sum_{k=1}^{K} C(\mathcal{H}^{(k)})$. Due to law of large numbers, as $K \to \infty$, we have $C_K \to \mathbb{E}[C(\mathcal{H})]$ where $\mathcal{H}$ is a tensor of Gaussian random variables. All the numerical results presented in this section present $C_K$ for $K = 100$, which can be interpreted as the ergodic capacity of a random tensor channel when its realizations are known at the transmitter and the receiver. The SNR is defined as $P/\sigma^2$ as used in [21], where $P$ is the sum power constraint or the total transmit power and the noise tensor contains i.i.d. circular complex Gaussian entries with zero mean and variance $\sigma^2 = 1$.

1) Comparison with Uniform Power Allocation

If the channel is not known then the ergodic capacity of a random tensor channel, $\mathbb{E}[C(\mathcal{H})]$ will be a function of the pdf of $\mathcal{H}$. For a MIMO matrix channel, capacity when the channel is Rayleigh distributed with channel state information available at only receiver is achieved by uniform power allocation at the transmitter [21]. In this paper we present a comparison between uniform power allocation and the tensor water-filling power allocation for a tensor channel. Figure 2 presents capacity in bits per tensor symbol under sum power constraint for 2 different sizes of tensor channel with uniform power allocation across all the transmit tensor elements and optimal tensor water-filling approach. Similar to the MIMO matrix case, it can be seen that uniform power allocation is sub optimal but at high SNR, it gives similar capacity as achieved by optimal power allocation. Also, it can be seen that the total capacity is higher when we increase the number of input and output domains. Next, we present examples to illustrate the impact of different domains and their dimensions on tensor channel capacity.

2) Capacity for different sizes of Channel Tensor

In Figure 3 we present the channel capacity in bits/tensor symbol for a fourth order channel tensor under a sum power constraint at SNR of 10 dB. Input and output are order-2 tensors of size $X \times Y$ each, corresponding to a $X \times Y \times X \times Y$ tensor channel. It is seen that increasing $X$ and $Y$ individually causes an increase in total capacity. To understand the effect of different types of channel, we also find capacity when the channel is normalized such that the total receive average power is identical to the transmitted power. Such a normalization has been suggested in [91], [92] for the MIMO channel case. In tensor channels, such a normalization is achieved when the individual entries of the channel tensor $\mathcal{H} \in \mathbb{C}^{J_1 \times \cdots \times J_M \times I_1 \times \cdots \times I_N}$ are generated as circular complex Gaussian with zero mean and variance $1/(J_1 \cdots J_M)$.
Figure 4 shows the capacity of such a normalized tensor channel of size $X \times Y \times X \times Y$ with both input and output of size $X \times Y$ each, under sum power constraints corresponding to an SNR of 10 dB. On increasing the size of input and output tensors, the rate of increase in capacity is lower as compared to Figure 3, and the capacity tends to reach a saturation for large values of $X$ and $Y$. For the case where channel gain is unity, and hence transmit and receive signal power are same, capacity is plotted against received SNR for a $2 \times 2 \times 2 \times 2$ tensor channel in Figure 5. A comparison with corresponding scalar and matrix channels with the same received signal power is also presented. The gain in capacity achieved by moving from scalar to a tensor channel can be attributed to the multiplexing gain provided by tensor channels which increases with the number of domains.

For the rest of the numerical examples, the more widely used model from [21], [89] where the channel entries are circular complex Gaussian with zero mean and unit variance is employed, as for Figure 3. In Figure 6, we present the capacity for a fourth order tensor channel under sum power constraints where the output is fixed as a $2 \times 2$ tensor and input is $X \times Y$ tensor at 10 dB SNR, under sum power constraints. The rate of increase in capacity for $X$ and $Y$ is slower as compared to Figure 3 for higher values of $X$ and $Y$ as the output tensor size is fixed as $2 \times 2$. So the capacity pre-log which is bounded by $\min\{X \cdot Y, 2 \cdot 2\} = 4$, does not increase with increasing $X$ and $Y$. We see that increasing the size of individual domains of the input tensor does not provide significant gain if the number and size of
corresponding domains of the output tensor is fixed.

3) Capacity under different domain power constraints

In this section we compare different possible power constraints. In order to approximate the optimum input covariance and thereby capacity, under per domain element and per element power constraints, we use Algorithm 1.

Figure 7 illustrates the capacity under sum power constraint and per domain element power constraint for a $2 \times 2 \times 2 \times 2$ channel with power constraint on input tensor of size $2 \times 2$. The power budgets on one of the domains of the input tensor are $P_1 = x \cdot P$ and $P_2 = (1 - x) \cdot P$. To put it in context, let the two domains be space and time. Then this constraint reflects that the power on first time slot for both the antennas is $P_1$ and on second slot for both the antennas is $P_2$ with total available power $P_1 + P_2 = P$. The plot in Figure 7 is presented for capacity against $x = P_1 / P$ at 10dB SNR. The flat line represents the capacity under sum power constraint which shows no variation with $x$, and the curved line shows the capacity with per domain element power constraints. As can be observed from Figure 7, the capacity under per domain element constraint is always lower than the capacity under sum power constraint, and these become very close to each other when $x \approx 0.5$, i.e. uniform power is allotted to the elements of the constrained domain. Note that such a behaviour is observed over an average of 100 channel realizations. For a given specific realization, the two curves may not meet at $x = 0.5$. For the MIMO case, a similar numerical result has been presented in [88] under per antenna power constraints.

In Figure 8 we present the capacity under per element power constraints and compare it with sum power constraint. If total available power is $P$, then as before $P_1 = x \cdot P$ and $P_2 = (1 - x) \cdot P$. Further, $P_{11} = y \cdot P_1, P_{12} = (1 - y) \cdot P_1, P_{21} = y \cdot P_2$ and $P_{22} = (1 - y) \cdot P_2$. Thus, with $0 \leq x, y \leq 1$, $P_{11}, P_{12}, P_{21}, P_{22}$ denote the individual power constraints on all the four elements of input tensor such that $P_{11} + P_{12} + P_{21} + P_{22} = P$. With different choices of $x$ and $y$, we achieve different per element power constraints such that total power remains $P$. The capacity with per element power constraints against $x$ and $y$ at SNR of 15 dB is presented in Figure 8. The flat surface represents capacity under sum power constraint which shows no variation with $x$ and $y$, and the curved surface shows capacity under per element power constraints. It can be seen that for different values of $x$ and $y$, the capacity achieved under per element constraints can be significantly lower than the capacity achieved under sum power constraint.

Note that Algorithm 1 only approximates the optimum input covariance and thus does not provide the exact capacity at low SNRs. However, since the problem at hand is a convex optimization problem, several standard software tools for numerical optimization can be used to find the capacity. To analyze how well the scaling approximation in Algorithm 1 works, we present a comparison between the capacity calculated through the convex optimization software tool CVX [75], and the capacity approximated through Algorithm 1. Figure 9 presents the capacity for sum power constraint, per domain element power constraints where a single domain
of dimension 2 is constrained with power budgets $P_1, P_2$, and per element power constraints where all the four elements have different power budgets $P_{11}, P_{12}, P_{21}, P_{22}$ against SNR for $x = y = 0.1$. We present such results for capacity calculated by using two approaches. First approach uses Algorithm 1 and the graphs are presented using solid curves. Second approach uses CVX and the graphs are presented using dashed curves. As can be seen, the capacity calculated via the approximation of Algorithm 1 matches very closely to the one calculated via CVX, and is almost indistinguishable at moderate to high SNR. This shows that Algorithm 1 provides a reasonably good approximation to the optimal solution at low SNR, while providing an exact solution at high SNR. Furthermore, it can be seen that capacity under per domain element and per element constraints is always upper bounded by capacity under sum power constraint. The capacity increases with SNR for all three cases, but the performance difference also gradually increases between the three solid curves, with sum power constraint performing the best, followed by per domain element and lastly, the per element constraint.

Figure 10 compares capacity under sum power and per domain element power constraints with $x = 0.1$ for different $N$ where both input and output are order $N$. The channel is an order $2N$ tensor and the size of each domain is 2. Capacity increases exponentially with $N$ in both the cases, however capacity under per domain element constraint is always upper bounded by capacity under sum power constraint.
where (95) to (96) follow from matrix Kronecker product properties [95, Corollary 4]. Hence the correlation matrix of the vectorized tensor is given in terms of the Kronecker product of different mode-\(n\) factor correlation matrices denoted by \(\Psi^{(n)}\). Such a model is called separable and it is considered for real random variables in [96]. While (97) expresses the correlation as a matrix by vectorizing \(\mathbf{K}_{\text{corr}}\), it is shown in [96, Proposition 2.1] that the correlation of \(\mathbf{K}_{\text{corr}}\) from (91) can also be expressed as an order 2\(N\) tensor obtained via the outer product of the factor matrices \(\Psi^{(n)}\) defined as \(\mathbf{R} = \Psi^{(1)} \circ \cdots \circ \Psi^{(N)}\). Note that the correlation tensor when defined as \(\mathbf{R} = \mathbb{E}[\mathbf{K}_{\text{corr}} \circ \mathbf{K}_{\text{corr}}^\ast]\) is just a permuted version of \(\mathbf{R}\), where \(\mathbf{R}_i,\ldots,i_N \ast \mathbf{R}_i,\ldots,i_N^\ast = \mathbb{E}[\mathbf{K}_{\text{corr}}^\ast \mathbf{K}_{\text{corr}}^\ast] = \Psi^{(1)} \ast \cdots \ast \Psi^{(N)}\). Hence the separable model implies that each element in the correlation tensor can be written in terms of product of the elements of the factor matrices. The proof in [96] is for real tensors, but can be easily extended to complex tensors as well.

The well known MIMO matrix Kronecker correlation model forms a special case of (91) where the tensor \(\mathbf{K}\) is of order 2 and the factor matrices \(A^{(1)}\) and \(A^{(2)}\) represent the square root of row and column correlation matrices respectively [97]. The MIMO Kronecker model may not be very accurate in all scenarios, but has been widely used because of its tractable analytic form, see for example [98]–[101].

Next consider an order 4 tensor channel of size \(3 \times 3 \times 3 \times 3\) corresponding to an order 2 input and order 2 output. We generate such a channel \(\mathbf{K}_{\text{corr}}\) with correlated elements using (91), where the entries of \(\mathbf{K}\) are i.i.d zero mean complex Gaussian with unit variance. For numerical illustration, we consider the correlation matrices generated using the exponential model with different correlation factor \(\rho\), where the elements of the correlation matrix \(\Psi^{(n)}\) are defined as \(\Psi_{ij}^{(n)} = \rho^{i-j}\) for \(\rho \in [0,1]\) [102]. The four correlation matrices \(\Psi^{(1)},\Psi^{(2)},\Psi^{(3)}\) and \(\Psi^{(4)}\) are generated using the exponential model with correlation coefficients \(\rho_1,\rho_2,\rho_3\) and \(\rho_4\) respectively. Assuming that the channel realization is known at both transmitter and receiver, we find the capacity of such a channel with correlated elements under sum power constraint. Figure 11 presents capacity at 10 dB SNR for different values of correlation coefficients where the receive domains are correlated with \(\rho_1 = \rho_2 = \rho_3\) and the transmit domains are correlated with \(\rho_3 = \rho_4 = \rho_T\). The plot shows that capacity decreases with increase in \(\rho_T\) and \(\rho_R\), and it is least when \(\rho_T\) and \(\rho_R\) approach 1. Capacity is largest when both \(\rho_T\) and \(\rho_R\) approach zero in which case the correlation matrices are identity and the channel has only uncorrelated entries across all the domains.

Next we investigate the impact of correlation on the tensor channel capacity when the correlation spans over a variable number of domains. Figure 12 presents the capacity against SNR for different number of domains having correlated entries. Capacity is lowest when \(\rho_n\) is non-zero (0.7 in the figure) for all the domains (i.e. for \(n = 1,2,3,4\)) and is highest when \(\rho_n\) is zero for all \(n\), i.e. all entries are

FIGURE 10: Capacity [bits/tensor symbol] vs input order comparing sum power and per domain element power constraints with \(P_i/P = 0.1\).
uncorrelated. Further it can be observed that the capacity difference between the various cases presented in Figure 12 is more significant at higher SNR. It is seen that the capacity decreases with increase in the number of domains having non-zero correlation factor. Such a loss of capacity with increase in the domains having correlation is further illustrated in Figure 13 for various tensor channel order.

Figure 13 presents the capacity against SNR for order $2M$ correlated tensor channels with order $M$ input and output having individual dimensions of 3. The factor correlation matrices along each of the $2M$ modes are based on exponential model with correlation factor $\rho$. The plot is presented for $M = 2, 3, 4, 5$ and $\rho = 0.4, 0.7$. Note that different values of $M$ lead to different tensor channel sizes with order $4, 6, 8,$ and $10$. Hence for a meaningful comparison of the impact of correlation, the capacity plotted in Figure 13 is normalized with respect to the number of elements in the transmit tensor symbol which is $3^M$. As can be seen in Figure 13, for a fixed $\rho$, as the number of domains of the tensor channel with all correlated elements increases, the capacity per element decreases. Further, this loss is more significant for higher values of $\rho$ (increased correlation) as can be seen by comparing the plots for $\rho = 0.7$ and 0.4.

**B. TENSOR BASED MIMO GFDM SYSTEM**

Generalized Frequency Division Multiplexing (GFDM) is a multi-carrier modulation scheme where each GFDM symbol consists of complex valued data symbols $d_{k,m}$ distributed across $K$ sub-carriers and $M$ timeslots known as sub-symbols. Thus each GFDM block consists of

FIGURE 11: Capacity [bits/tensor symbol] vs Correlation Coefficients for tensor channel with correlated entries

FIGURE 12: Capacity [bits/tensor symbol] vs SNR for tensor channel with correlated entries

FIGURE 13: Capacity [bits/tensor element] vs SNR for different order tensor channels with correlated entries
where the matrices corresponding to each antenna form slices of the third order input tensor. The covariance of the input tensor \( \mathbf{D} \) can be written as a sixth order tensor \( \mathbf{Q} \) of size \( S \times K \times M \times S \times K \times M \). We simulate such a system with two transmit and two receive antennas, considering three different cases:

- **case 1**: the entire transmit tensor is under sum power constraint \( P \), i.e. \( \text{tr}(\mathbf{Q}) \leq P \). We obtain the optimal covariance \( \mathbf{Q}^{(\text{opt1})} \) in this case which performs power allocation and precoding at the transmitter using: \( \mathbf{Q}^{(\text{opt1})} = \mathbf{V}_3 \ast (\mu^{-1} \mathbf{I} - \mathbf{D}_3^{\ast})^\dagger \ast \mathbf{V}_3^H \), where \( \mathbf{V}_3 \) and \( \mathbf{D}_3^{\ast} \) are obtained from the tensor EVD of \( (\mathbf{Q}^H \ast \mathbf{Q}) \), and \( \mathbf{I} \) is an identity tensor of size \( S \times K \times M \times S \times K \times M \). The coefficient \( \mu \) is calculated using tensor water-filling to ensure \( \text{tr}(\mathbf{Q}^{(\text{opt1})}) \leq P \).

- **case 2**: the two transmit antennas have different power budgets, \( P_1 = x \cdot P \) and \( P_2 = (1-x) \cdot P \). Hence the constraint on the covariance can be written as \( \sum_{k,m} Q_{1,k,m,1,k,m} \leq P_1 \) and \( \sum_{k,m} Q_{2,k,m,2,k,m} \leq P_2 \). We obtain the input covariance \( \mathbf{Q}^{(\text{opt2})} \) in this case using Algorithm 1 with the channel tensor \( \mathbf{H} \), power budgets \( P_1 \) and \( P_2 \), and \( N_{\text{c}} = K \cdot M \) as inputs.

- **case 3**: the constraint is same as case 2, but now we perform per antenna power allocation and pre-coding. The optimum covariance in this case is determined by independently calculating the covariance for the data transmitted on the two antennas. We find \( \mathbf{Q}_{1}^{(\text{opt3})} \) and \( \mathbf{Q}_{2}^{(\text{opt3})} \) as:\n
\[
\mathbf{Q}_{1}^{(\text{opt3})} = \frac{1}{2} \mathbf{I}, \quad \mathbf{Q}_{2}^{(\text{opt3})} = \frac{1}{2} \mathbf{I}.
\]
the received power is the same as the transmit power $P$. Hence the received SNR per bit is calculated as $P/b$ where $b$ is the number of bits in each received tensor symbol, obtained as $b = \text{(Number of elements in each tensor)} \times \text{(bits per element)}$. The noise added is AWGN with zero mean and unit variance such that the noise covariance is the identity tensor.

The capacity of the channel $\mathcal{H}$ from (98) under three different cases of input covariance with $x = 0.1$ for case 2 and 3 is shown in Figure 15. As can be seen the capacity achieved for case 1, i.e. sum power constraint is always higher than the other two cases. For case 2, where individual antennas have different power budgets and joint domain processing is used to generate the optimal covariance outperforms case 3 where separate covariances are obtained for antenna 1 and 2 independently.

We also test the BER performance when the input is generated from a 4QAM constellation for the three different cases and the results are shown in Figure 16 where BER is plotted against received SNR per bit in dB. At the transmitter, the covariance of the input tensor $\mathcal{D} \in \mathbb{C}^{2 \times 8 \times 5}$ before precoding is given by an identity tensor. The input $\mathcal{D}$ is contracted with the square root of the optimum covariance as $\mathcal{Q}^{\text{opt}}_{1/2} \hat{\mathcal{D}}$. This ensures that the covariance of the input tensor after precoding, defined as $\mathbb{E}[(\mathcal{Q}^{\text{opt}})^{1/2} \hat{\mathcal{D}} \circ (\mathcal{Q}^{\text{opt}})^{1/2} \hat{\mathcal{D}}^*]$ is given by $\mathcal{Q}^{(opt)} \in \mathbb{C}^{2 \times 8 \times 5}$ before precoding. Hence, based on (98), we get $\mathcal{D} = \mathcal{H} \ast_3 \mathcal{Q}^{(opt)} \ast_3 \hat{\mathcal{D}} + \mathcal{N}$. To assess the BER performance, we assume a multi-linear minimum mean
A. MULTIPLE ACCESS CHANNELS

Consider a multi-user MIMO network where a base station (BS) equipped with \( N_R \) antennas is receiving information from \( K \) users with \( N_T \) antennas each. If we denote the uplink channel matrix between the \( k \)th user and the base station by \( H^{(k)} \in \mathbb{C}^{N_R \times N_T} \), then the discrete time received signal \( y \in \mathbb{C}^{N_R \times 1} \) at the BS can be written as [42]:

\[
y = \sum_{k=1}^{K} H^{(k)} x^{(k)} + n
\]

where \( x^{(k)} \in \mathbb{C}^{N_T \times 1} \) is the signal transmitted by the \( k \)th user and \( n \in \mathbb{C}^{N_R \times 1} \) is the received noise vector which is assumed circularly symmetric complex Gaussian with identity covariance matrix. In such a multiple access channel (MAC), each user \( k \) is subject to an individual power constraint \( P_k \). If the transmit covariance matrix of user \( k \) is denoted as \( Q^{(k)} \), then the constraint is represented as \( \text{tr}(Q^{(k)}) \leq P_k \) for \( k = 1, \ldots, K \).

A multi-user channel with \( K \) users is characterized by a \( K \)-dimensional achievable rate region \( C_R \), known as capacity region [42], where each point in the region \( (R_1, R_2, \ldots, R_K) \) represents the achievable rates \( R_k \) at which user \( k \) can send information with arbitrarily low error probability. We assume that all the channel matrices are known to the receiver and all the transmitters. To denote the convex hull of the union of sets we use the symbol \( \bigcup \). With power constraints \( (P_1, P_2, \ldots, P_K) \), the capacity region of MIMO MAC is

FIGURE 15: Capacity [bits/tensor symbol] vs transmit power for MIMO GFDM under sum power constraint, and per antenna power constraint with and without joint domain processing

FIGURE 16: BER plot for MIMO GFDM under sum power constraint, and per antenna power constraint with and without joint domain processing

VII. MULTI-USER MIMO CAPACITY

In this section, we consider the application of the tensor framework to multi-user MIMO channels. We present numerical examples comparing the tensor approach with other results found in literature. We consider \( K \)-user Gaussian Multiple Access Channels and Interference Channels.
given by [42]:

\[
C_R = \bigcup_{\text{tr}(Q(k)) \leq P_k, \forall k} \left\{ (R_1, \ldots, R_K) : \right. \\
0 \leq \sum_{k \in S} R_k \leq \log \det \left( I + \sum_{k \in S} H^{(k)}Q^{(k)}H^{(k)H} \right) \\
\forall S \subseteq \{1, \ldots, K\} \bigg\} 
\] (101)

where \(S\) denotes a subset of the set of users. Each set of covariance matrices \((Q^{(1)}, \ldots, Q^{(K)})\) satisfying the power constraints corresponds to a \(K\) dimensional polyhedron [104]. The capacity region is the convex hull of all such polyhedrons. Note that for Gaussian MIMO MAC, the capacity region is defined using only the union of rate regions and the convex hull is not needed [42]. It is shown in [105] that for Gaussian MIMO MAC, the boundary points of the capacity region can be characterized by maximizing a weighted sum rate \(\sum_k \nu_k R_k\) for all non-negative \(\nu_k\) such that \(\sum_k \nu_k = 1\), and thus finding boundary points can be cast into a convex optimization problem.

Essentially, (101) represents a set of bounds on individual rates \(R_1, R_2, \ldots, R_K\), and combination of rates such as \(R_1 + R_2, R_1 + R_3, R_2 + R_3, R_1 + R_2 + R_3\) and so on, including the sum rate \(R_1 + R_2 + \cdots + R_K\). For a two users case, this can be represented by (102) (at the top of next page). For a given choice of covariance matrices \(Q^{(1)}\) and \(Q^{(2)}\) which satisfy the power constraints, (102) represents an upper bound on \(R_1, R_2\) and \(R_1 + R_2\). The maximum of \(\log \det(I + H^{(1)}Q^{(1)}H^{(1)H})\) and \(\log \det(I + H^{(2)}Q^{(2)}H^{(2)H})\) from (102) are the individual achievable capacities by user 1 and user 2 respectively assuming the other user is silent. The maximum of \(\log \det(I + H^{(1)}Q^{(1)}H^{(1)H} + H^{(2)}Q^{(2)}H^{(2)H})\) is the sum capacity achievable when both users are transmitting. Note that the choice of covariance matrices \(Q^{(1)}\) and \(Q^{(2)}\) satisfying the power constraints which achieves the sum capacity may not achieve the individual capacities. Similarly the choice of \(Q^{(1)}\) and \(Q^{(2)}\) which achieves the individual capacities may not achieve the sum capacity. In general, different transmit choices lead to different pairs of covariance matrices \((Q^{(1)}, Q^{(2)})\) such that the capacity region is given by the convex hull of the union of an infinite number of rate regions each corresponding to a different set \((Q^{(1)}, Q^{(2)})\). The optimal choice of \(Q^{(1)}\) and \(Q^{(2)}\) which achieves the sum capacity is found by an iterative water-filling approach [105] which sequentially finds covariance for each user with the single-user classical water-filling method assuming interference from other users as noise. A detailed step by step algorithm can be found in [105].

Such an approach, however, assumes that different users transmit independently. The iterative water-filling algorithm treats the information available about other users’ interfering channels as noise. In the presence of complete channel state information, a better transmit strategy which would provide higher achievable rates would be to allow users to coordinate for transmission. Hence a joint signal transmit strategy can expand the capacity region. Using the tensor framework, we can find the capacity region assuming user coordination. First, let us represent (100) using the tensor system model.

We define the multi-user MIMO tensor channel as a third order tensor \(H \in \mathbb{C}^{N_T \times N_T \times K}\) where \(H_{J::k} = H^{(k)}\). The input signal can be defined as a matrix \(X \in \mathbb{C}^{N_T \times K}\), where each \(X^{(k)}\) of (100) forms a column of the matrix \(X\). Hence the system model in (100) can be represented as:

\[
y = H \ast_2 X + n
\] (103)

The input covariance is represented as an order 4 tensor \(Q \in \mathbb{C}^{N_T \times K \times N_T \times K}\). Assuming the noise to be circular symmetric complex Gaussian with identity covariance matrix, denoted by I, the output covariance can be written as \((H \ast_2 Q \ast_2 H^H + I)\). Subsequently, the sum capacity of such a system with user coordination can be calculated from the following optimization problem:

\[
\max_Q \log \left[ \det (H \ast_2 Q \ast_2 H^H + I) \right] \\
s.t. \sum_{n=1}^{N_T} Q_{n,k,n,k} \leq P_k \quad \forall k, \\
Q \succeq 0.
\] (104-106)

where \(\sum_{n=1}^{N_T} Q_{n,k,n,k} \leq P_k\) represents the individual power constraints for different users. The optimal \(Q\) that achieves capacity can be approximated using Algorithm 1. Note the difference in this tensor formulation and the iterative water-filling approach used in vector formulation is that the latter assumes different users transmit independently despite having perfect channel state information. With independent transmissions, the iterative water-filling maximizes the function \(\log \det(I + \sum_{k=1}^{K} H^{(k)}Q^{(k)}H^{(k)H})\) subject to \(\text{tr}(Q(k)) \leq P_k\) and \(Q^{(k)} \succeq 0\) for \(k = 1, \ldots, K\) [105]. Note that this objective function is same as the upper bound on the sum rate from (101) for \(S' = \{1, \ldots, K\}\). The vector based iterative water-filling treats inter-user interference as noise since it attempts to optimize the sum rate over all choices of separate covariance matrix for each user as shown in [105]. On the other hand, the tensor approach solves the problem in (104)-(106) and aims to find a joint covariance across all the users. Thereby, the tensor approach suggests a joint transmit scheme for all the users wherein the inter-user interference is not treated as noise since the interference term also carries signal information. The maximum sum rate achieved by all the \(K\) users given in (104) is the sum capacity of the \(K\) users MIMO MAC under user coordination. Similarly the sum capacity achieved by a subset \(S\) of the all the users \(U = \{1, \ldots, K\}\) is given as the maximum of \(\log \det(I + \sum_{k \in S} H^{(k)}Q^{(k)}H^{(k)H})\) over the choice of positive semi-definite \(Q^{(S)}\) which satisfy the power constraints. The tensor \(Q^{(S)} \in \mathbb{C}^{N_T \times |S| \times N_T \times |S|}\) with \(|S|\) denoting the cardinality of \(S\), contains matrices of size \(N_T \times N_T\) as slices corresponding to only those users which are included in \(S\). Similarly \(Q^{(S)} \in \mathbb{C}^{N_T \times |S| \times N_T \times |S|}\) is the covariance tensor of \(X^{(S)} \in \mathbb{C}^{N_T \times |S|}\) which contains columns of only those
users which are included in $S$. Hence the tensor framework allows to define a capacity region with user coordination as:

$$C_R = \bigcup_{\mathbf{p}^{(S)} \in \mathbb{C}^{N_i \times N_T \times 1}} \left\{ (R_1, \ldots, R_K) : \begin{array}{l}
0 \leq \sum_{k \in S} R_k \leq \log \det \left( I + H^{(S)} \odot Q^{(S)} \odot 2H^{(S)H} \right) \\
\forall S \subseteq \{1, \ldots, K\} \end{array} \right\} $$

(107)

where $S$ contains the list of users being considered. The vector $\mathbf{p}^{(S)}$ contains the power budgets of the users included in $S$, with its components $p^{(S)}_i$ denoting the power budget of the $i$th user in set $S$ for $i = 1, \ldots, |S|$. Note that the expression $\log \det(I + H^{(S)} \odot Q^{(S)} \odot 2H^{(S)H})$ in (107) is same as the objective function in (104) when $S$ contains all the users, i.e. $S = \{1, \ldots, K\}$.

As an example, let us consider a three users scenario in which case $S$ can assume the following sets : $\{1\}$, $\{2\}$, $\{3\}$, $\{1, 2\}$, $\{2, 3\}$, $\{1, 2, 3\}$. Subsequently, we can expand (107) as in (108) (at the top of page). For the first bound on $R_1$, we have $S = \{1\}$, i.e. we consider only first user and find what is the maximum rate that user 1 can transmit given the power constraint on user 1 and that all other users are silent. In this case, $H^{(1)} = \mathbb{C}^{N_i \times N_T \times 1}$ is essentially the matrix $H^{(1)}$ between the user 1 and base station, and $Q^{(1)} = \mathbb{C}^{N_T \times 1 \times N_T}$ is the covariance matrix $Q^{(1)}$ of user 1. Hence this reduces to single user MIMO channel where the optimal $Q^{(1)}$ which achieves maximum rate can be found using classical water-filling. Similarly, conditions two and three in (108) correspond to the bounds on rates achieved by user 2 and user 3 ($R_2$ and $R_3$) respectively when all other users are silent. Hence, the first three conditions are same as the one derived from the vector case (101). Condition four corresponds to $S = \{1, 2\}$, thus gives a bound on the sum rate that user 1 and user 2 can together achieve given that user 3 is silent. In this case $H^{(1,2)} = \mathbb{C}^{N_i \times N_T \times 2}$ is a sub-tensor of $H$ as $H^{(1,2)} = H^{(1:1,2)}$. The covariance tensor $Q^{(1,2)} = \mathbb{C}^{N_T \times 2 \times N_T \times 2}$ is a sub-tensor of $Q$ given by $Q^{(1,2)} = Q^{(1:1,2)}$. The power constraints are defined as $\sum_n Q_{n,1,1}^{(1,2)} \leq P_1$ and $\sum_n Q_{n,2,2}^{(1,2)} \leq P_2$ where $P_1, P_2$ are power budgets for user 1 and 2 respectively. Note that the bound on sum rate $R_1 + R_2$ achieved using this method assumes that user 1 and 2 perform a joint transmission. Hence the sum rate achieved using the tensor approach will be different than the one obtained from iterative water filling which assumes independent transmission. Similarly, condition five represents the bound on sum rate $R_2 + R_3$ that can be achieved when user 2 and 3 transmit together keeping user 1 silent. In this case $H^{(2,3)} = \mathbb{C}^{N_T \times N_T \times 2}$ is a sub-tensor of $H$ given by $H^{(2,3)} = H^{(2:2,3)}$. The covariance tensor $Q^{(2,3)} = \mathbb{C}^{N_T \times 2 \times N_T \times 2}$ is a sub-tensor of $Q$ given by $Q^{(2,3)} = Q^{(2:2,3)}$. The power constraints are defined as $\sum_n Q_{n,1,1}^{(2,3)} \leq P_2$ and $\sum_n Q_{n,2,2}^{(2,3)} \leq P_3$ where $P_2, P_3$ are power budgets for user 2 and 3 respectively. Similarly condition six represents the bound on sum rate $R_1 + R_3$ that can be achieved when user 1 and 3 transmit together keeping user 2 silent. The last condition represents the bound on sum rate when all the three users are transmitting. Note that the covariance tensors satisfying the power constraint of all the equations in (108) can be seen as sub-tensors of the covariance tensor $Q$. For instance, $Q_{1,1,1} = Q^{(1)}$ represents the covariance matrix of user 1. But the optimal choice of covariance tensor $Q$ that achieves the sum capacity under user cooperation can not be obtained from only individual covariance tensors $Q^{(i)}$ for different users. For instance, the optimal $Q^{(1)}$ that maximizes $\log \det(I + H^{(1)}Q^{(1)}H^{(1)H})$ may not be the sub-tensor of the optimal $Q$ that maximizes $\log \det(I + H^{(2)}Q^{(2)}H^{(2)H})$. The capacity region under user coordination is thus given by the convex hull of the union of all the rate regions over all the choices of covariance tensors which satisfy the power constraints.

With additional constraints on the covariance tensor

$$Q^{(S)}_{n,i,n',i'} = 0, \quad \text{for } i \neq i', \quad (109)$$

we can use (107) to represent the capacity region without user coordination as in (101). To show this, we consider the expression $H^{(S)} \odot 2Q^{(S)} \odot 2H^{(S)H}$ from (107) for any given $S$ and show that with (109) it reduces to $\sum_{k \in S} H^{(k)}(K_kH^{(k)H})$ as in (101). We can write:

$$\left(H^{(S)} \odot 2Q^{(S)} \odot 2H^{(S)H}\right)_{j,j'} = \sum_{n,i,n',i'} \sum_{n,i,n',i'} Q^{(S)}_{j,j,n,i,n',i'}J^{(S)H}_{n,i,n',i',j,j'} \quad (110)$$

With (109), entries of $Q^{(S)}$ are zero for all $i \neq i'$. Hence we can write (110) as

$$\left(H^{(S)} \odot 2Q^{(S)} \odot 2H^{(S)H}\right)_{j,j'} = \sum_{i=1}^{\big|S\big|} \sum_{n' = 1}^{N_T} \sum_{n = 1}^{N_T} J^{(S)}_{j,j,n,n'}Q^{(S)}_{n,n',i,i'}J^{(S)H}_{n',i',j,j'} \quad (111)$$

Note that each user in the set of all users $U = \{1, \ldots, K\}$ is known by its index $k$, i.e. first user, second user, $k$th user and so on. The variable $i$ denotes the index of a user in set $S$ where $S \subseteq U$. For instance if $S = \{3, 4, \ldots, K\}$, then the third user ($k = 3$) is at index $i = 1$ in $S$. Hence we replace
the index $i$ with user number $k$. The entities $J_{i,i;:i;i}^{(S)}$ and $Q_{i,i;:i;i}^{(S)}$ are the channel sub-tensor and covariance matrix of the user $i$ in set $S$, and equivalently of the $k$th user in set of all users. Hence we get $J_{i,i;:i;i}^{S,k} = H_{j,n;k}$ for $k \in S$, and the covariance as $Q_{n,i,n',i} = Q_{n,n'}^{(k)}$ for $k \in S$. Thus we can write (111) as:

$$(3J_{i,j}^{(S)} Q_{j,j}^{(S)} H_{j,ij}^{(S)})_{j,j'} = \sum_{k \in S} \left( \sum_{n'=1}^{N_T} \sum_{n=1}^{N_T} H_{j,n}^{(k)} Q_{n,n'}^{(k)} H_{n,j}^{(k)H} \right) (112)$$

Substituting (113) with the additional constraints (109) into (107) gives us the capacity region from (101) which assumes no user coordination.

Next we present a few numerical examples to illustrate the concepts. We consider a multi-user MIMO MAC scenario where $K$ users with $N_T = 2$ antennas each are communicating with a base station having $N_R = 10$ antennas. The noise vector at the receiver is circular symmetric complex Gaussian with zero mean and identity covariance matrix. The channel entries are realizations of circular symmetric complex Gaussian random variables with zero mean and unit variance and these realizations are known at the transmitters and receiver. The results presented are averaged over 100 different channel realizations. Each user has an individual power constraint $P_k$. The total transmit power is $P = \sum_k P_k$. We assume all the users have the same power constraint, i.e., $P_k = P/K$ and plot the sum capacity obtained through the tensor approach achieved by $K$ users against total power in Figure 17. The sum capacity can be found by solving the optimization problem in (104)-(106). We apply the proposed solution from Algorithm 1 to approximate the optimal input covariance tensor which achieves the sum capacity. The covariance tensor obtained from Algorithm 1 is then used to approximate the sum capacity given by $\log(3J_{*} Q_{*} H_{*}^{H} + 1)$. This approach assumes that all the users coordinate for transmission. It can be seen that for a fixed number of users, the sum capacity increases with an increase in the total transmit power. Furthermore, for a fixed total transmit power, the sum capacity increases when the number of users increases. Especially at higher transmit powers, the increased number of users lead to a significant increase in the sum capacity.

The results of Figure 17 can be contrasted with the iterative water-filling approach from literature where different users despite having channel state information of other users transmit independently. Figure 18 presents the sum capacity obtained with coordinated users and independent users against the number of users for two different values of total power. The sum capacity under independent users is calculated using the iterative water filling approach from [105, Algorithm 1]. It can be observed that as the number of users increases, there is a significant difference in achievable rate of coordinated users as compared to independent users. This shows that user cooperation captured in the input covariance tensor structure can improve the sum capacity substantially.

Further the capacity region of a 2 users MIMO MAC scenario is presented in Figure 19 with transmit power budgets $P_1 = P_2 = 5$ dB. We consider $K = 2$ where the base station
has $N_R = 8$ antennas and 2 different settings for transmit antennas $N_T = 4$ and 8. The capacity region obtained from tensor formulation for two users can be written as:

$$C_R = \bigcup_{\sum_{n=1}^{N_T} \sum_{k=1}^{2} q_{n,k,n,k} \leq P_k, k=1,2} \left\{ \begin{array}{l}
R_1 \leq \log \det \left( I + H^{(1)} Q^{(1)} H^{(1)H} \right) \\
R_2 \leq \log \det \left( I + H^{(2)} Q^{(2)} H^{(2)H} \right) \\
R_1 + R_2 \leq \log \det \left( I + \mathcal{H} *_2 Q *_2 \mathcal{H}^H \right) \end{array} \right\}$$

(114)

For the two users case, finding the optimum covariance that maximizes the achievable rate of user 1 assuming user 2 is silent reduces to a single user MIMO scenario. Hence the bounds on individual rates $R_1$ and $R_2$ in (114) and (102) are the same. Note however that the bound on the sum rate $R_1 + R_2$ differs. With the additional constraint on covariance from (109), the bound on the sum rate in (114) translates to:

$$R_1 + R_2 \leq \log \det \left( I + \mathcal{H} *_2 Q *_2 \mathcal{H}^H \right)$$

(115)

$$= \log \det \left( I + H^{(1)} Q^{(1)} H^{(1)H} + H^{(2)} Q^{(2)} H^{(2)H} \right)$$

(116)

where (115) to (116) follow from (113). Hence, with additional constraint on covariance $Q$ as defined by (109), the sum rate bound in (114) reduces to the sum rate bound in (102), and thus all three bounds in (114) depend only on $Q^{(1)}, Q^{(2)}$. The additional constraint corresponds to the transmit scheme where each user acts independent of the other user. In such cases, the capacity region is characterized by a pair of covariance matrices $(Q^{(1)}, Q^{(2)})$ which satisfies

the power constraints. In general, the capacity region of the two users case corresponding to (114) is characterized by a triplet $(Q^{(1)}, Q^{(2)}, Q)$ where $Q$ is the transmit covariance tensor which prescribes a joint transmission scheme. The transmit covariance matrix of individual users $Q^{(1)}, Q^{(2)}$ form the sub-tensors of $Q$. However the optimal $Q^{(2)}$ which maximizes $R_1, R_2$ respectively may not be the sub-tensors of optimal $Q$ which maximizes $R_1 + R_2$. The optimal $Q^{(1)}, Q^{(2)}$ are found assuming the other user to be silent for transmission, whereas the optimal $Q$ is found assuming joint transmission by both the users.

The rate regions in (114) and (102) forms a pentagon on a two dimensional $R_1, R_2$ plane. The capacity region is determined by the convex hull of the union of all such pentagons obtained through different choices of covariances which satisfy the constraints. In Figure 19, the solid line (case 1) represents the pentagon corresponding to (102) where $(Q^{(1)}, Q^{(2)})$ are obtained via iterative water-filling from [105, Algorithm 1] to maximize the sum rate with independent transmissions. The dotted line (case 2) represents the pentagon corresponding to (102) where $(Q^{(1)}, Q^{(2)})$ are obtained via conventional water filling for single user MIMO to maximize $R_1$ and $R_2$ individually. Similarly, different $(Q^{(1)}, Q^{(2)})$ will correspond to different pentagons based on (102). The capacity region with independent users is obtained as a convex hull of the union of all such pentagons associated with different covariances which satisfy the constraints in (102). Thus the convex hull of the regions of case 1 and 2 gives an inner bound to the capacity region with independent user transmission. The capacity region signifies the bounds on the rate of transmission by each user.

The dashed line (case 3) represents the pentagon corresponding to (114) where $Q$ is approximated using Algorithm

FIGURE 18: Sum Capacity vs Users of MU MIMO MAC

FIGURE 19: Capacity Region of a 2 user MIMO MAC
1 to maximize the sum rate, and \( Q^{(1)}, Q^{(2)} \) are the sub-tensors of the \( Q \) obtained from Algorithm 1. However such a choice of \( Q^{(1)}, Q^{(2)} \) may not maximize the individual rates of user 1 and 2. This is reflected in Figure 19 as the dotted line shows a larger bound on individual rates as compared to other cases in both the vertical and horizontal segments, A and B. Case 2 can be seen as another scenario for (114) where \( Q^{(1)}, Q^{(2)} \) are chosen to maximize the individual rates and the joint covariance tensor \( Q \) has structure \( Q_{i:j:i:j} = Q^{(1)} \) for \( i = j \) and 0 for \( i \neq j \) with \( i, j = 1, 2 \). Such a covariance tensor does not maximize the sum rate but only individual rates and still satisfies all the constraints in (114). The capacity region with user coordination is given by the convex hull of the union of pentagons associated with case 3 and case 2 along with every pentagon associated with different covariances which satisfy the constraints in (114). Thus, a convex hull of regions of case 2 and 3 gives an inner bound to the capacity region with user cooperation. The bound on the sum achievable rate indicated by segment C in Figure 19 is lowest in case 2, followed by the case when transmit covariances are chosen via iterative water-filling (case 1), and is largest when covariance is obtained using the tensor approach assuming user coordination (case 3). Moreover, this difference increases as \( N_T \) increases from 4 to 8, as observed in the figure. Hence it is seen that the boundary of the capacity region expands in segment C with user cooperation as opposed to the independent user transmissions.

**B. MIMO INTERFERENCE CHANNEL (IC)**

Let us now consider the general multi-user MIMO interference channel (IC) where both transmit and receive side have user separation. Consider \( K \) transmit devices having \( N_T \) antennas each that are communicating with their respective \( K \) receive devices having \( N_R \) antennas each. Such a channel model assumes that all transmitting devices are communicating with their respective receivers while generating interference to all other receivers. Finding the exact capacity region of a general \( K \) user interference channel is still an ongoing effort [106], [107]. For the two users scenario, capacity bounds have been discussed in [108], [109] and references within, under various assumptions regarding interference such as \( Z \) interference channels (where one of the two receive users does not experience interference), strong interference, and noisy interference. The term ‘noisy interference’ refers to conditions where the sum capacity can be achieved by treating interference as noise [110]. For a two users MIMO IC, [110, Theorem 2] presents a set of conditions involving the channel and transmit covariance matrices which are sufficient for interference to be treated as noise. Such conditions ensure that the indirect links are much weaker than the direct links, and power allocation is such that the received power of the message received via the indirect link at each user is very low compared to the received power of the message via the direct link. Further, it is shown in [111] that under strong interference, each receiver can jointly decode the signal and the interference to achieve the sum capacity. As an extension of two users case, [112] derives the conditions under which treating noise as interference can achieve the sum capacity for a \( K \) users IC. Most of these works assume no coordination among the source transmitters or the destinations. If different transmitting users coordinate for transmission, and receivers coordinate for reception then interference can be treated as information bearing entity which can be extracted using the tensor framework.

The system model for a MIMO interference network is given by [113]:

\[
Y^{(k)} = H^{(k,k)} X^{(k)} + \sum_{u \neq k} H^{(k,u)} X^{(u)} + N^{(k)}
\]

for \( k = 1, \ldots, K \) and \( X^{(k)} \in \mathbb{C}^{N_T \times 1} \) is the vector transmitted by source \( k \). Also, \( Y^{(k)}, X^{(k)} \in \mathbb{C}^{N_R \times 1} \) are the received signal and noise vectors at destination \( k \). Matrix \( H^{(k,k)} \in \mathbb{C}^{N_R \times N_T} \) is the direct channel between source \( k \) and destination \( k \) and \( H^{(k,u)} \in \mathbb{C}^{N_R \times N_T} \) is the cross-channel matrix between source \( u \) and destination \( k \). For each transmitting source, there is an individual power constraint defined as \( \text{tr}(Q^{(k)}) \leq P_k \), where \( Q^{(k)} \in \mathbb{C}^{N_T \times N_T} \) is the covariance matrix of vector \( x^{(k)} \) and \( P_k \) denotes the power budget. Such interference networks can be thought of as a tensor communication link. The input can be represented as a matrix \( X \in \mathbb{C}^{N_T \times K} \) where each \( x^{(k)} \) forms a column of the matrix \( X \). Similarly the received signal and noise can be represented using matrices \( Y, N \in \mathbb{C}^{N_R \times K} \) where each \( y^{(k)} \) and \( n^{(k)} \) form columns of the matrices \( Y \) and \( N \). The overall channel between such an input and output can be represented as a fourth order tensor \( \mathcal{H} \in \mathbb{C}^{N_R \times K \times N_T \times K} \) where \( \mathcal{H}_{u,k,n,k} = H^{(k,u)} \). Subsequently the interference network system model can be represented in tensor form as :

\[
Y = \mathcal{H} \ast_2 X + N.
\]

Note that (118) is different from MIMO MAC specified by (103) in the sense that in (103) the channel is a third order tensor and thus the output is a vector. On the other hand in (118) the channel is a fourth order tensor to account for user separation at the receiver side as well and thus the output is a matrix. The power constraints on the input can be defined in a similar way as for (103). Assuming the noise covariance is an identity tensor \( \mathcal{I} \) of size \( N_R \times K \times N_R \times K \), the tensor formulation can be used to specify the channel capacity as

\[
\max_{Q} \log \det (\mathcal{H} \ast_2 Q \ast_2 H^{T} + \mathcal{I}) \quad \text{subject to} \quad \sum_{n=1}^{N_T} Q_{n,k,n,k} \leq P_k \quad \forall k, \quad Q \succeq 0.
\]

Note that capacity obtained from such a tensor formulation assumes that all the sources cooperate for transmission and all the destinations cooperate for reception.

Now we consider an example with two users interference channel and compare the sum rate achieved using the tensor framework which assumes user cooperation, with the upper
bound on rate suggested in [108] while treating interference as noise. We consider the same example from [108] consisting of a system composed of two transmitters and two receivers, equipped with $N_T$ antennas each. The model introduces a positive scalar $\alpha \geq 0$ to control the interference power:

$$
\begin{align*}
\hat{x}^{(1)} &= H^{(1,1)}x^{(1)} + \alpha \cdot H^{(1,2)}x^{(2)} + \mathbf{n}^{(1)}, \\
\hat{x}^{(2)} &= \alpha \cdot H^{(2,1)}x^{(1)} + H^{(2,2)}x^{(2)} + \mathbf{n}^{(2)}.
\end{align*}
$$

(121) \tag{121} \tag{122}

Such a system of equations can be equivalently represented using (118) where the channel $\mathcal{H}$ is a tensor of size $N_T \times 2 \times N_T \times 2$ and the input, output and noise are matrices of size $N_T \times 2$ each. We find the capacity of such MIMO Interference channels with user coordination using the tensor framework where the optimal input covariance is approximated using Algorithm 1.

For our example, in (121), (122) we take $\alpha = 1/\sqrt{3}$ and the channel entries are i.i.d. zero mean unit variance Gaussian random variables as in [108]. The results are averaged over 100 different channel realizations. These channel realizations are known at the transmitters and receivers. We find the sum rate achieved via the tensor framework assuming user cooperation, and denote it using $R_T$. We compare $R_T$ with the capacity of $K$ parallel non-interfering channels found using standard water-filling approach and denote it as $R_{U_i}$. In [108], $R_{U_i}$ has been used as an upper bound on the achievable sum rate while treating interference as noise. Figures 20-24 presents a comparison between $R_T$ and $R_{U_i}$. To calculate $R_{U_i}$, we find the transmit covariance matrix $Q^{(k)}$ for each user input $x^{(k)}$ based on MIMO water filling corresponding to the channel $\mathcal{H}^{(k,k)}$, and set $R_{U_i} = \sum_k \log \det(I + H^{(k,k)}Q^{(k)}H^{(k,k)}H)$. Also, $R_T$ is calculated using the objective function in (119) where the optimal covariance tensor $\Omega$ is approximated using Algorithm 1.

Figure 20 compares $R_T$ and $R_{U_i}$ for a 2 users case with different values of power constraints $P_1 = P_2 = P$. The achievable sum rates with the two approaches are plotted against the number of antennas $N_T$. The sum rates for both the cases increase as $N_T$ increases. The achievable rate with the tensor approach, $R_T$ is higher than the upper bound on sum rate, $R_{U_i}$ from [108]. It can also be observed in Figure 20 that as $P$ increases, the gap between $R_T$ and $R_{U_i}$ increases as well. The tensor approach shows that the presence of interference can in fact give higher achievable sum rates if the transmit and receive operations are performed jointly by all the transmitting and receiving users respectively. Hence the sum rate achieved via the tensor approach, that allows cooperation at the transmitter and receiver sides, can be higher than the sum rate achieved in the absence of interference.

This feature is further observed in Figure 21 which presents the sum rate against the number of users $K$ for a multi-user MIMO interference scenario from (117) with $N_T = 2$. Each users’ receive signal contains a desired signal and information from $K - 1$ interfering links whose power is controlled by a scalar factor $\alpha$ as described in the two users case. The result presented is for $\alpha = 1/\sqrt{3}$ and for two different total power budgets $P = 5, 10$ dB with individual power constraints as $P_k = P/K$. It can be seen that $R_T$ is always larger than $R_{U_i}$. Furthermore, as the number of users grows, the difference between $R_T$ and $R_{U_i}$ also widens, which shows the advantage of considering interference as information bearing term rather than noise, through the tensor framework.

To further elaborate on the role of interference, Figure 22 presents the sum rate $R_T$ of the two users scenario achieved using the tensor framework for different interference power and compares it with $R_{U_i}$. The solid lines represent $R_T$ and dashed lines represent $R_{U_i}$. The result is plotted against the interference coupling power gain defined as $G_1 = 10 \log_{10} a^2$ dB [108] for different number of antennas $N_T$ at each device, and with power constraints $P_k = P/K$ for $k = 1, 2$ where $K = 2$ and $P$ is set to 10 dB. It is seen clearly that higher interference leads to higher sum rate $R_T$ using the tensor approach. Also, $R_T$ increases with $N_T$ and the gap between sum rate for different number of antennas widens with increasing interference. However, $R_{U_i}$ is always lower than $R_T$ and does not change with $G_1$. Since $R_{U_i}$ is calculated by assuming zero interference, it does not vary with changing the interference power. At very low interference power, we see that $R_T$ and $R_{U_i}$ are almost same. The difference between $R_T$ and $R_{U_i}$ starts to be significant (more than 1 bit/channel use) at an interference power $G_1$ of approximately -2 dB when $N_T = 2$, and around -6 dB when $N_T = 8$. 

![Figure 20: Achievable sum rate vs number of antennas for two users MIMO IC](image-url)
Achievable sum rate in bits/channel use

Sum rate in bits/channel use

rate increasing strength of the interfering links, we get higher sum low when the interference power of both links is weak. With N user MIMO IC with R and can be made in Figure 24 which represents G and R Y FIGURE such a system model can be represented using (118) with Y, X and N as NT × 3 matrices each and X as an NT × 3 × NT × 3 tensor. The number of antennas at each device is denoted by NT. Each destination user receives signals from a direct link and 2 interfering links whose power is controlled by scalar factors a and b respectively. In Figure 23 we present RT and RU against interference power with three users for NT = 2. The power constraints for each user is same, i.e. PT = P/K where K = 3, and P is set to 5 dB. In Figure 23 we have G1 = 10 log10 a2 dB and G2 = 10 log10 b2 dB, where G1, G2 represent the strength of the 2 interfering links for each user. It can be seen that RT is low when the interference power of both links is weak. With increasing strength of the interfering links, we get higher sum rate RT. The curve for RU does not vary with change in G1 and G2 and is always lower than RT. Similar observation can be made in Figure 24 which represents RT and RU for NT = 4. On comparing Figures 23 and 24, we see that RT and RU increases as NT increases from 2 to 4. Notice that difference in RT for NT = 2 and 4 is wider for large values of G1, G2, i.e. when the strengths of the interfering links increase. The tensor framework allows to treat interfering terms as information bearing components, resulting in higher rates with increasing G1, G2 and NT.

C. COMPLEXITY COMPARISON

The matrix based methods used for estimating the sum capacity for MIMO MAC and IC often treats interference as noise. In most cases, the objective of such simplification is a reduction in the computational complexity of the problem. Hence it is important to compare the complexity of the matrix based methods with the tensor method presented in this paper to highlight the relevance of both the approaches.

In section V-C, we presented the complexity analysis of employing Algorithm 1 for any given tensor channel of size J1 × J2 × ... × JM × I1 × I2 × ... × IN. In this section, we will use the results from section V-C and compare the computational complexity of finding capacity for the specific cases of MIMO MAC and IC channels using the tensor method with the matrix based methods.

1) Comparison for MIMO MAC

Consider the system model for K users MIMO MAC from (100) where \( H(k) \) denotes the uplink channel matrix of size NR × NT between the BS equipped with NR antennas and the kth user which has NT transmit antennas. The iterative water-filling approach of finding the sum capacity computes a separate covariance matrix for each user while treating the interference from all other users as noise. A standard water-filling approach is used in finding the individual covariance matrices. Furthermore, all the K covariance matrices are
iteratively updated until a convergence criteria is met. The dominant step in a standard water-filling approach is the singular value decomposition of the channel matrix. The computational complexity of the SVD of an $m \times n$ matrix is given by $O(m \cdot n \cdot \min(m, n))$ [114], [115]. Thus, the computational complexity of computing the water-filling corresponding to the matrix $H^{(k)} \in \mathbb{C}^{N_R \times N_T}$ would be given as $O(N_R \cdot N_T^2)$ (assuming $N_R > N_T$). For the MIMO MAC, since the water-filling is computed for each of the $K$ users iteratively, the complexity scales linearly with the number of users $K$, and the number of required iterations per user, denoted by $I$. Thus the overall complexity is given as $O(K \cdot I \cdot N_R \cdot N_T^2)$.

On the contrary, the tensor approach in Algorithm 1 is a non-iterative method to approximate the joint covariance of all the users, which also takes into account the interference as information bearing entities. Consider the tensor model for MIMO MAC from (103) where the channel is defined as a third order tensor of size $N_R \times N_T \times K$. Thus based on the discussion in section V-C, the complexity of employing Algorithm 1 for the MIMO MAC channel is given by $O((K \cdot N_T)^3 \cdot N_R + (K \cdot N_T)^3)$. Assuming that $K \cdot N_T \geq N_R$, we can write the computational complexity as $O((K \cdot N_T)^3)$. The complexity of the matrix and tensor approaches are listed in Table 3 for a clear comparison.

**TABLE 3: Computational Complexity for MIMO MAC**

| Approach                              | Complexity          |
|--------------------------------------|---------------------|
| Matrix based iterative water filling | $O(K \cdot I \cdot N_R \cdot N_T^2)$ |
| Tensor based Algorithm 1             | $O((K \cdot N_T)^3)$ |

It can be seen that the complexity of tensor approach is cubic in the number of users as opposed to the matrix based iterative water-filling where the complexity is linear in the number of users. Although it is to be noted that the matrix based approach also depends on the number of iterations. Within each such iteration, a standard water-filling algorithm for a matrix channel is executed. In contrast, the tensor approach presented in this paper is non-iterative, i.e. a single iteration is required irrespective of the number of users and antennas.

However, clearly the advantage of the matrix approach should not be dismissed as it offers a low complexity solution. Despite the iterative nature of the matrix approach, the complexity is linear in the number of users as opposed to the tensor approach where it is cubic. However, as the number of users grows the advantage of tensor based approach becomes significant in terms of the achievable sum rate possible due to user coordination. This is evident in Figure 18 where the sum capacity of the coordinated users shows a substantial increase as compared with the independent users, when the number of users grow. For instance, at total transmit power of 10 dB, the sum capacity for 10 users is 40 bits/channel use as opposed to around 32 bits/channel use for the independent users as seen in Figure 18. To better quantify this gap, for the example of Figure 18 we plot the difference between...
the sum rates achievable with tensor framework and the matrix approach in Figure 25 against number of users and different values of $N_T$, at $P = 10$ dB. We can see that the difference increases substantially with increasing users, and also increasing number of transmit antennas. To summarize, Table 3 shows the advantage of the matrix approach in terms of lower complexity, and Figure 25 shows the advantage of the tensor approach in terms of higher rates due to user coordination.

2) Comparison for MIMO IC

Now let us consider the case of MIMO IC for $K$ users from (117). To calculate $R_U$, which is the upper bound on the achievable sum rate while treating interference as noise, $K$ separate transmit covariance matrices have to be calculated corresponding to each user. Thus, the complexity in this case is also linear in the number of users. Each user solves a water-filling solution corresponding to a matrix channel $H^{(k,k)}$ of size $N_R \times N_T$. Assuming $N_R = N_T = N$, the complexity is given as $O(K \cdot N^3)$. The corresponding tensor model for $K$ users MIMO IC is given by (118) where the channel is a fourth order tensor of size $N \times K \times N \times K$. The process of calculating $R_T$, which denotes the achievable sum rate using user coordination, requires Algorithm 1 to be executed for the fourth order tensor channel. Thus, based on the discussion in section V-C, its computational complexity is given by $O((K \cdot N)^3)$. The complexity of the matrix and the tensor models are listed in Table 4 for a comparison.

Ignoring the interference leads to a degenerate situation where a higher order channel is reduced to $K$ non-interfering matrix channels. Thus, the computational complexity of calculating $R_U$ is lower when compared with the tensor method used to find $R_T$, as seen in Table 4. However, as shown in Figure 21, the difference between $R_T$ and $R_U$ can be very large with increasing the number of users. For instance, for 10 users at 10 dB total power, $R_T$ is around 19 bits/channel use as opposed to $R_U$ which is only around 10 bits/channel use as observed in Figure 21. To better quantify the difference between $R_T$ and $R_U$, we consider the example from Figure 21 for $P = 10$ dB, and plot the difference $R_T - R_U$ against number of users for different values of $N$ in Figure 26. It can be seen in Figure 26 that this difference increases significantly as the number of users, or the number of antennas increases. Thus, Figure 26 clearly shows the advantage of the tensor framework in providing higher achievable sum rates.

3) Advantage of the tensor approach

In both the MIMO MAC and MIMO IC, the achievable sum rates are interference limited using the matrix approaches. Hence as the number of users or antennas grow which leads to higher interference, the performance degrades in terms of achievable sum rates. It is clear that the tensor method should be preferred when user coordination is allowed leading to

![Figure 25: Difference in sum capacity achieved via tensor approach (coordinated users) and matrix approach (independent users) vs number of users for MIMO MAC.](image)

![Figure 26: Difference in achievable sum rate, $R_T - R_U$ at $P = 10$ dB vs number of users for $K$ users MIMO IC.](image)

TABLE 4: Computational Complexity for MIMO IC

| Approach                        | Complexity          |
|--------------------------------|---------------------|
| Calculating $R_U$ using water-filling | $O(K \cdot N^3)$   |
| Calculating $R_T$ using Algorithm 1 | $O((K \cdot N)^3)$ |
a joint transmission across all users, whereas the matrix-based methods can be used when users are constrained to act independently and transmit on a per user basis. The tensor method while admitting higher complexity, as seen in Tables 3 and 4, leads to much higher achievable sum rates also, as seen in Figures 25 and 26. Thus there is an inherent performance/complexity trade off where at the cost of higher complexity, much significant gain in sum rates can be achieved especially in the presence of strong inter-domain interference.

The representation of such multi-domain systems using matrix channels and vector signals often gets more complex as the number of domains or dimensions within a domain increases. It is important to note that the tensor method does not necessarily yields lower complexity for such high complex situations. Its main feature is that it provides a structured mathematically accurate framework to treat high complex systems, while having the capability to naturally scale back to lower complexity when the problem is of reduced complexity. For example, for a single user MIMO, the tensor approach scales back to a conventional matrix approach. For the MU MIMO MAC with additional constraint in (109), the tensor framework reduces to the conventional MU MIMO MAC model. Hence the tensor method can be seen as an umbrella framework to treat high complex situations. Several examples of such systems were included in this paper to demonstrate the usefulness of the proposed tensor method as specific cases with additional constraints imposed for simplifying the problem.

VIII. CONCLUSIONS

In this paper, we introduced a unified framework using tensors to represent a multi-domain communication system. The proposed framework can be used to model many of the currently used multi-domain approaches in communications by leveraging the multi-linear structure of the system. Several examples of such systems were included in this paper to demonstrate the usefulness of the proposed tensor framework. The information-theoretic analysis for a fixed tensor channel presented in this paper suggests an exponential increase in Shannon capacity with increase in number of domains, if the channel is known at both transmit and receive side. In particular, the tensor framework’s ability to mathematically represent a family of power constraints is utilized. This allows to characterize the capacity of multi-user MIMO systems having per user power constraints for any number of users. A tensor representation of the channel and its information theoretic analysis using the proposed framework leads to a joint transmission scheme across all the domains. An example of MIMO GFDM system was presented to illustrate that such a joint domain processing can lead to better BER performance as compared to per domain processing. It was also shown that capacity of tensor channels decreases as the correlation among the channel components increases. Furthermore, for the same total power as the number of domain elements under individual power constraints increases, the capacity decreases. In case of multi-user MAC and IC channels, it was shown that the tensor approach leads to higher achievable sum rates with user-coordination as compared to the sum rates achieved with independent users. The tensor framework allows to capture the user cooperation in the form of a joint covariance tensor across all the domains. Our results show that such an improvement in achievable rates through the tensor approach becomes even more significant as the number of users grow or the power of the interfering links increases. With independent user transmissions, the interference is treated as noise, as opposed to the tensor approach where interference is treated as information bearing entity. The performance advantages resulting from using the tensor framework come very often at the expense of a complexity increase. However, tensors provide structured and mathematically accurate tools of handling such complexity increase.

APPENDIX A SOLVING THE EQUATIONS DERIVED FROM KKT CONDITIONS FOR THE OPTIMAL COVARIANCE TENSOR

In this appendix, we present the solution to the equations obtained through KKT conditions for finding the optimal transmit covariance tensor \( \Omega \). The results presented here are a generalization of Theorem 1 from [73] to a tensor setting. In this appendix \( \mathcal{H} \) represents the channel tensor, \( \mathcal{M} \) represents the Lagrange multiplier corresponding to the semi-definite constraint on covariance and \( \mathcal{B} \) represents the tensor whose pseudo-diagonal entries are the Lagrange multipliers corresponding to other constraints on transmit covariance (such as power). For sum power constraint, \( \mathcal{B} \) is an identity tensor and the Lagrange multiplier is a scalar \( \mu > 0 \). So for \( \mathcal{M} \geq 0, \mathcal{B} > 0 \), our objective is to find \( \mathcal{Q} \geq 0 \), that satisfies (126) and (127):

\[
\mathcal{H}^H \mathcal{M} (\mathcal{H} \ast_N \mathcal{Q} \ast_N \mathcal{H}^H + \mathcal{J}_M)^{-1} \ast_M \mathcal{H} = \mu \mathcal{B} - \mathcal{M} \tag{126}
\]

\[
\mathcal{Q}^{1/2} \ast_N \mathcal{M} \ast_N \mathcal{Q}^{1/2} = 0 \mathcal{T} \tag{127}
\]

Contracting along \( N \) consecutive modes in (126) on left and right by \( \mathcal{Q}^{1/2} \) results in

\[
\mathcal{Q}^{1/2} \ast_N \mathcal{H}^H \mathcal{M} (\mathcal{H} \ast_N \mathcal{Q} \ast_N \mathcal{H}^H + \mathcal{J}_M)^{-1} \ast_M \mathcal{H} \ast_N \mathcal{Q}^{1/2}
\]

\[
= \mu \mathcal{Q}^{1/2} \ast_N \mathcal{B}^{1/2} \ast_N \mathcal{B}^{1/2} \ast_N \mathcal{Q}^{1/2} - \mathcal{Q}^{1/2} \ast_N \mathcal{M} \ast_N \mathcal{Q}^{1/2}
\]

\[
= \mathcal{Q}^{1/2} \ast_N \mathcal{M} \ast_N \mathcal{Q}^{1/2}
\]

\[
\Rightarrow \mathcal{A} \triangleq \mathcal{B}^{1/2} \ast_N \mathcal{Q}^{1/2} \Rightarrow \mathcal{Q}^{1/2} = \mathcal{B}^{-1/2} \ast_N \mathcal{A} \tag{129}
\]

Since \( \mathcal{Q} \) is Hermitian, we have \( \mathcal{Q}^{1/2} = (\mathcal{Q}^{1/2})^H = \mathcal{A}^H \ast_N \mathcal{B}^{-1/2} \), which gives

\[
\mathcal{Q} = \mathcal{B}^{-1/2} \ast_N \mathcal{A} \ast_N \mathcal{A}^H \ast_N \mathcal{B}^{-1/2} \tag{130}
\]

Substituting (130) in (126) and contracting both sides along \( N \) modes of \( \mathcal{B}^{-1/2} \) results in
Substituting (129) and (130) into (128), we get

$$\begin{align*}
\mathcal{A}^H N \mathcal{B}^{-1/2} N \mathcal{H} + \mathcal{M}^{-1} M \mathcal{H} + \mathcal{M}^{-1} M \mathcal{N} N \mathcal{H} = \mathcal{A}^H N \mathcal{B}^{-1/2} N \mathcal{A} + \mathcal{M}^{-1} M \mathcal{A} + \mathcal{M}^{-1} M \mathcal{N} N \mathcal{A}
\end{align*}$$

Substituting tensor SVD of $\mathcal{A}$ and $\mathcal{P} \triangleq \mathcal{H} \mathcal{B}^{-1/2} \mathcal{N} \mathcal{A} = \mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H$ into (132),

$$\begin{align*}
\mathcal{V}_{D} \mathcal{P} \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H & = \mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H
\end{align*}$$

Since $\mathcal{V}_{D}$ and $\mathcal{V}_{B}$ are unitary tensors, and the middle quantities on both side of (133) are pseudo-diagonal, both the right and left side represent the tensor EVD of two equal tensors. From the uniqueness of tensor EVD, (133) implies that $\mathcal{V}_{D} = \mathcal{V}_{B}$. Also,

$$\begin{align*}
\mathcal{P} \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H & = \mathcal{A}^H N \mathcal{B}^{-1/2} N \mathcal{H} + \mathcal{M}^{-1} M \mathcal{A} + \mathcal{M}^{-1} M \mathcal{N} N \mathcal{A}
\end{align*}$$

Since $\mathcal{V}_{A} = \mathcal{V}_{B}$ we see that (135) represents the tensor EVD of $\mathcal{P} \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H$. Hence the middle term in (135) is pseudo-diagonal. So $\mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H$ must also be a pseudo-diagonal tensor. Let $S \triangleq \mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H$, then since $\mathcal{U}_A$ and $\mathcal{V}_{B}$ are unitary, we get

$$\begin{align*}
S^H N S & = (\mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H) H \mathcal{N} (\mathcal{U}_A \mathcal{D}_{B \mathcal{N}} \mathcal{D}_N \mathcal{V}_{B}^H)
\end{align*}$$

Since $S$ is pseudo-diagonal, (136) implies $S = \mathcal{J}_N \Rightarrow \mathcal{U}_A = \mathcal{V}_{B}$, proving the proposition.

From the tensor SVD of $\mathcal{A}$ and $\mathcal{K}$, and Proposition A.1, the left-hand side of (131) can be written as (137) (equation written at the top of the next page). From the EVD, $\mathcal{B}^{-1/2} \mathcal{N} \mathcal{M} \mathcal{B}^{-1/2} = \mathcal{U} \mathcal{D}_{B \mathcal{M}} \mathcal{N} \mathcal{U}^H$, the right-hand side in (131) becomes

$$\begin{align*}
\mu \mathcal{J}_N \mathcal{B}^{-1/2} \mathcal{N} \mathcal{M} \mathcal{B}^{-1/2} = \mathcal{U} \mathcal{D}_{B \mathcal{M}} \mathcal{N} \mathcal{U}^H.
\end{align*}$$

Equations (137) and (138) represent the tensor EVD of the left and right hand side of (131), hence from uniqueness of tensor EVD we get $\mathcal{U} = \mathcal{V}_{K}$ and:

$$\begin{align*}
\mathcal{D}_{B \mathcal{M}} (\mathcal{D}_K \mathcal{N} \mathcal{D}_A \mathcal{N} \mathcal{D}_H \mathcal{N} \mathcal{D}_K \mathcal{N} \mathcal{D}_H + \mathcal{J})^{-1} M \mathcal{D}_K = \mu \mathcal{J}_N - \mathcal{D}_{B \mathcal{M}}
\end{align*}$$

Let the pseudo-diagonal elements of $\mathcal{D}_{A}, \mathcal{D}_{K}$ and $\mathcal{D}_{B \mathcal{M}}$ be $a_{i_1,...,i_N}, k_{i_1,...,i_N}$ and $m_{i_1,...,i_N}$ respectively. Pseudo-diagonal elements of $\mathcal{D}_H$ and $\mathcal{D}_K$ will also be $a_{i_1,...,i_N}$ and $k_{i_1,...,i_N}$ respectively as these are real values. Since both sides of (139) are pseudo-diagonal, hence (139) can be written component-wise as:

$$\begin{align*}
\frac{k_{i_1,...,i_N}^2}{1 + a_{i_1,...,i_N}^2 k_{i_1,...,i_N}^2} = \mu - m_{i_1,...,i_N}
\end{align*}$$

$$\begin{align*}
a_{i_1,...,i_N} = \frac{1}{\mu - m_{i_1,...,i_N}} - \frac{k_{i_1,...,i_N}^2}{1 + a_{i_1,...,i_N}^2 k_{i_1,...,i_N}^2}
\end{align*}$$

Substituting (129) into (127) gives $\mathcal{A}^H N \mathcal{B}^{-1/2} N \mathcal{M} \mathcal{N} \mathcal{B}^{-1/2} N \mathcal{A} = 0$ which can be written element-wise as $a_{i_1,...,i_N}^2 + m_{i_1,...,i_N}^2 = 0$. Since $B > 0$ and $M > 0$, we know $m_{i_1,...,i_N} = 0$. So $a_{i_1,...,i_N}^2 = 0$ when $m_{i_1,...,i_N} > 0$, otherwise it is given by (141) with $m_{i_1,...,i_N} = 0$. Together it can be written as

$$\begin{align*}
a_{i_1,...,i_N}^2 = \left( \frac{1}{\mu} - \frac{1}{k_{i_1,...,i_N}^2} \right)^+ \tag{143}
\end{align*}$$

where $(z)^+ = \max\{0, z\}$. From (143) and Proposition A.1 we get

$$\begin{align*}
\mathcal{A} \mathcal{A}^H = \mathcal{U}_A \mathcal{D}_{B \mathcal{M}} \mathcal{N} \mathcal{U}_A^H = \mathcal{V}_{K} \mathcal{A} \mathcal{A}^H \mathcal{V}_{K}^H
\end{align*}$$

where $\mathcal{V}_{K}$ and $\mathcal{D}$ are obtained from tensor EVD of $\mathcal{K}^H N \mathcal{K} = \mathcal{V}_{K} \mathcal{D} \mathcal{V}_{K}^H$. Based on the tensor SVD of $\mathcal{K}$ in Proposition A.1, we have $\mathcal{D} = \mathcal{D}_K^H N \mathcal{D}_K$. Substituting (144) into (130), we can conclude that

$$\begin{align*}
\mathcal{Q} = \mathcal{B}^{-1/2} N \mathcal{V}_{K} \mathcal{N} \mathcal{V}_{K}^H \mathcal{B}^{-1/2} \mathcal{N}
\end{align*}$$

REFERENCES

[1] T. G. Kolda and B. W. Bader, “Tensor Decompositions and Applications,” SIAM Review, vol. 51, no. 3, pp. 455–500, 2009. [Online]. Available:
\[
\mathbf{V}_K \ast \mathbf{H}^{*} \mathbf{U}^{*} + \mathbf{M}^{\dagger} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \ast \mathbf{D}^{*} \mathbf{H}^{*} \ast \mathbf{M}^{\dagger} \mathbf{U}^{*} = \mathbf{V}_K \ast \mathbf{H}^{*} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \ast \mathbf{D}^{*} \mathbf{H}^{*} \ast \mathbf{M}^{\dagger} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \ast \mathbf{D}^{*} \mathbf{H}^{*} \ast \mathbf{M}^{\dagger} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \ast \mathbf{D}^{*} \mathbf{H}^{*} \ast \mathbf{M}^{\dagger} \mathbf{U}^{*} \ast \mathbf{D}^{*} \mathbf{H}^{*} \mathbf{U}^{*} \sum_{K=1}^{N} \mathbf{V}_K^{\ast} \mathbf{D}^{\ast} \mathbf{H}^{*} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \ast \mathbf{D}^{*} \mathbf{H}^{*} \ast \mathbf{M}^{\dagger} \mathbf{U}^{*} \ast \mathbf{M}^{\dagger} \mathbf{D}^{*} \mathbf{A} \\
\text{pseudo-diagonal}
\]
N. Michailow, M. Matthé, I. S. Gaspar, A. N. Caldevilla, L. L. Mendes, X. Shang, B. Chen, G. Kramer, and H. V. Poor, “Capacity regions for the K-user MIMO interference channel,” IEEE Transactions on Information Theory, vol. 61, no. 4, pp. 1753–1767, 2015.

A. Ghasemi, A. S. Motahari, and A. K. Khandani, “Interference alignment for the K user MIMO interference channel,” in 2010 IEEE International Symposium on Information Theory, 2010, pp. 360–364.

V. Vaseudevan and M. Ramakrishna, “A hierarchical singular value decomposition algorithm for low rank matrices,” arXiv preprint arXiv:1710.02812, 2017.

J. W. Demmel, Applied Numerical Linear Algebra. Society for Industrial and Applied Mathematics, 1997. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611941446

Divyanshu Pandey received the B.Tech degree in Communication and Computer Engineering from the LNM Institute of Information Technology, Jaipur, India, in 2011 and the M.S. degree in Electrical Engineering from University of Minnesota, Twin Cities, USA, in 2014. He worked as a Systems Engineer in WLAN PHY R&D team at Marvell Semiconductors Inc., Santa Clara, CA, USA from February 2015 to August 2017. He is currently pursuing the Ph.D. degree in Electrical Engineering from McGill University, Montreal, QC, Canada. His research interests include Information Theory, Wireless Communications, and Tensor Algebra with applications to communications and signal processing.

HARRY LEIB received the B.Sc. (cum laude) and M.Sc. degrees in Electrical Engineering from the Technion - Israel Institute of Technology, Israel in 1977 and 1984 respectively. In 1987 he received the Ph.D. degree in Electrical Engineering from the University of Toronto, Canada.

During 1977-1984 he was with the Israel Ministry of Defense, working in Communication Systems. After completing his Ph.D. studies, he was with the University of Toronto as a Post-doctoral Research Associate and as an Assistant Professor. Since September 1989 he has been with the Department of Electrical and Computer Engineering at McGill University, where he is now a Full Professor. His current research activities are in the areas of Digital Communications, Wireless Communication Systems, Global Navigation Satellite Systems, Detection, Estimation, and Information Theory.

Dr. Leib was Editor for the IEEE Transactions on Communications 2000-2013, and Associate Editor for the IEEE Transactions on Vehicular Technology 2001-2007. He was a guest co-editor for special issues of the IEEE Journal on Selected Areas in Communication on “Differential and Noncoherent Wireless Communication” 2003-2005, and on “Spectrum and Energy Efficient Design of Wireless Communication Networks” 2012-2013. Since 2017 he has been the founding Editor-in-Chief of AIMS Electronics and Electrical Engineering Journal.

38