Sparse Sampling for Inverse Problems with Tensors

Ortiz-Jimenez, Guillermo; Coutino, Mario; Chepuri, Sundeep Prabhakar; Leus, Geert

DOI
10.1109/TSP.2019.2914879

Publication date
2019

Document Version
Final published version

Published in
IEEE Transactions on Signal Processing

Citation (APA)
Ortiz-Jimenez, G., Coutino, M., Chepuri, S. P., & Leus, G. (2019). Sparse Sampling for Inverse Problems with Tensors. *IEEE Transactions on Signal Processing*, *67*(12), 3272-3286. Article 8705331. https://doi.org/10.1109/TSP.2019.2914879

Sparse Sampling for Inverse Problems With Tensors

Guillermo Ortiz-Jiménez, *Student Member, IEEE*, Mario Coutino, *Student Member, IEEE*, Sundeep Prabhakar Chepuri, *Member, IEEE*, and Geert Leus, *Fellow, IEEE*

***Abstract*—We consider the problem of designing sparse sampling strategies for multidomain signals, which can be represented using tensors that admit a known multilinear decomposition. We leverage the multidomain structure of tensor signals and propose to acquire samples using a Kronecker-structured sensing function, thereby circumventing the curse of dimensionality. For designing such sensing functions, we develop low-complexity greedy algorithms based on submodular optimization methods to compute near-optimal sampling sets. We present several numerical examples, ranging from multiantenna communications to graph signal processing, to validate the developed theory.**

***Index Terms*—Graph signal processing, multidimensional sampling, sparse sampling, submodular optimization, tensors.**

I. Introduction

In MANY engineering and scientific applications, we frequently encounter large volumes of multisensor data, defined over multiple domains, which are complex in nature. For example, in wireless communications, received data per user may be indexed in space, time, and frequency [2]. Similarly, in hyperspectral imaging, a scene measured in different wavelengths contains information from the three-dimensional spatial domain as well as the spectral domain [3]. And also, when dealing with graph data in a recommender system, information resides on multiple domains (e.g., users, movies, music, and so on) [4]. To process such multisensor datasets, *higher-order tensors* or *multiway arrays* have been proven to be extremely useful [5].

Manuscript received June 28, 2018; revised January 10, 2019; accepted April 4, 2019. Date of publication May 3, 2019; date of current version May 20, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yimin D. Zhang. This work was supported in part by the ASPIRE project (Project 14926 within the STW OTP programme), in part by the Netherlands Organization for Scientific Research, and in part by the KAUST-MIT-TUD consortium under Grant OSR-2015-Sensors-2700. The work of G. Ortiz-Jiménez was supported by a fellowship from Fundación Bancaria "la Caixa." The work of M. Coutino was supported by CONACYT. This paper was presented in part at the Sixth IEEE Global Conference on Signal and Information Processing, Anaheim, CA, November 2018 [1]. *(Corresponding author: Guillermo Ortiz-Jiménez.)*

G. Ortiz-Jiménez was with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft 2628 CD, The Netherlands. He is now with the Institute of Electrical Engineering, Ecole Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: guillermo.ortizjimenez@epfl.ch).

M. Coutino and G. Leus are with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft 2628 CD, The Netherlands (e-mail: m.coutino@tudelft.nl; g.j.t.leus@tudelft.nl).

S. P. Chepuri was with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft 2628 CD, The Netherlands. He is now with the Department of Electrical Communications Engineering, Indian Institute of Science, Bangalore 560012, India (e-mail: spchepuri@iisc.ac.in).

Digital Object Identifier 10.1109/TSP.2019.2914879

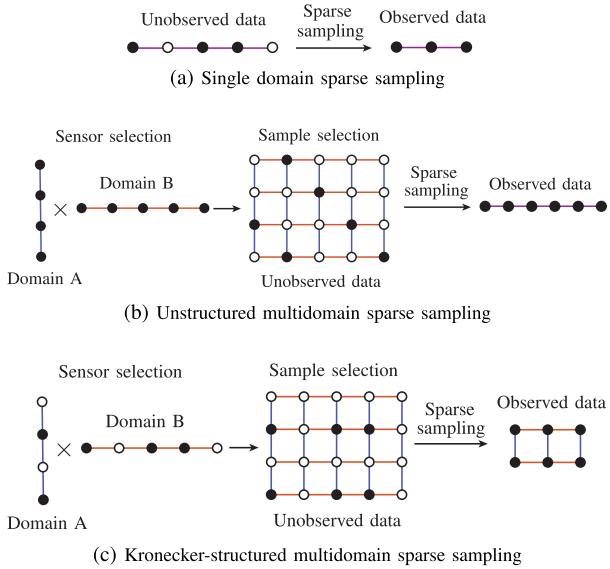


Fig. 1. Different sparse sensing schemes. Black (white) dots represent selected (unselected) measurement locations. Blue and red lines determine different domain directions, and a purple line means that data has a single-domain structure.

In practice, however, due to limited access to sensing resources, economic or physical space limitations, it is often not possible to measure such multidomain signals using every combination of sensors related to different domains. To cope with such issues, in this work, we propose *sparse sampling* techniques to acquire multisensor tensor data.

Sparse samplers can be designed to select a subset of measurements (e.g., spatial or temporal samples as illustrated in Fig. 1a) such that the desired inference performance is achieved. This subset selection problem is referred to as sparse sampling [6]. An example of this is field estimation, in which the measured field is related to the source signal of interest through a linear model. To infer the source signal, a linear inverse problem is solved. In a resource-constrained environment, since many measurements cannot be taken, it is crucial to carefully select the best subset of samples from a large pool of measurements. This problem is combinatorial in nature and extremely hard to solve in general, even for small-sized problems. Thus, most of the research efforts on this topic focus on finding suboptimal sampling strategies that yield good approximations of the optimal solution [6]–[16].

For signals defined over multiple domains, the dimensionality of the measurements grows much faster. An illustration of this "curse of dimensionality" is provided in Fig. 1b, wherein the

measurements now have to be systematically selected from an even larger pool of measurements. Typically used suboptimal sensor selection strategies are not useful anymore as their complexity is too high; or simply because they need to store very large matrices that do not fit in memory (see Section III for a more detailed discussion). Usually, selecting samples arbitrarily from a multidomain signal, requires that sensors are placed densely in every domain, which greatly increases the infrastructure costs. Hence, we propose an efficient *Kronecker-structured* sparse sampling strategy for gathering multidomain signals that overcomes these issues. In Kronecker-structured sparse sampling, instead of choosing a subset of measurements from all possible combined domain locations (as in Fig. 1b), we propose to choose a subset of sensing locations from each domain and then combine them to obtain multidimensional observations (as illustrated in Fig 1c). We will see later that taking this approach will allow us to define computationally efficient design algorithms that are useful in big data scenarios. In essence, the main question addressed in this paper is, *how to choose a subset of sampling locations from each domain to sample a multidomain signal so that its reconstruction has the minimum error?*

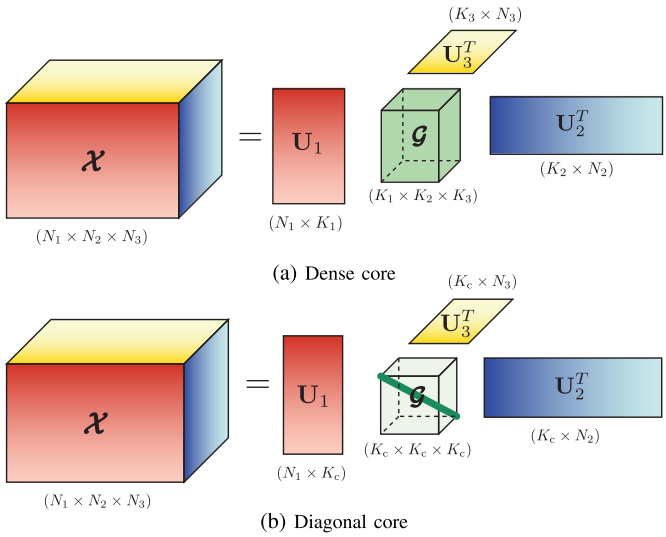


Fig. 2. Graphic representation of a multilinear system of equations for $R = 3$. Colors represent arbitrary values.

II. Preliminaries

In this section, we introduce the notation that will be used throughout the rest of the paper as well as some preliminary notions of tensor algebra and multilinear systems.

A. Notation

We use calligraphic letters such as $\mathcal{L}$ to denote sets, and $|\mathcal{L}|$ to represent its cardinality. Upper (lower) case boldface letters such as $\mathbf{X}$ ($\mathbf{x}$) are used to denote matrices (vectors). Bold calligraphic letters such as $\boldsymbol{\mathcal{X}}$ denote tensors. $(\cdot)^T$ represents transposition, $(\cdot)^H$ conjugate transposition, and $(\cdot)^\dagger$ the Moore-Penrose pseudoinverse. The trace and determinant operations on matrices are represented by $\mathrm{tr}\{\cdot\}$ and $\det\{\cdot\}$, respectively. $\lambda_{\min}\{\mathbf{A}\}$ denotes the minimum eigenvalue of matrix $\mathbf{A}$. We use $\otimes$ to represent the Kronecker product, $\odot$ to represent the Khatri-Rao or column-wise Kronecker product; and $\circ$ to represent the Hadamard or element-wise product between matrices. We write the ℓ_2-norm of a vector as $\|\cdot\|_2$ and the Frobenius norm of a matrix or tensor as $\|\cdot\|_F$. We denote the inner product between two elements of a Euclidean space as $\langle\cdot,\cdot\rangle$. The expectation operator is denoted by $\mathbb{E}\{\cdot\}$. All logarithms are considered natural. In general, we will denote the product of variables/sets using a tilde, i.e., $\tilde{N} = \prod_{i=1}^R N_i$, or $\tilde{\mathcal{N}} = \mathcal{N}_1 \times \cdots \times \mathcal{N}_R$; and drop the tilde to denote sums (unions), i.e., $N = \sum_{i=1}^R N_i$, or $\mathcal{N} = \bigcup_{i=1}^R \mathcal{N}_i$.

Some important properties of the Kronecker and the Khatri-Rao products that will appear throughout the paper are [17]: $(\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D}) = \mathbf{AC}\otimes\mathbf{BD}$; $(\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\odot\mathbf{D}) = \mathbf{AC}\odot\mathbf{BD}$; $(\mathbf{A}\odot\mathbf{B})^H(\mathbf{A}\odot\mathbf{B}) = \mathbf{A}^H\mathbf{A}\circ\mathbf{B}^H\mathbf{B}$; $(\mathbf{A}\otimes\mathbf{B})^\dagger = \mathbf{A}^\dagger\otimes\mathbf{B}^\dagger$; and $(\mathbf{A}\odot\mathbf{B})^\dagger = (\mathbf{A}^H\mathbf{A}\circ\mathbf{B}^H\mathbf{B})^\dagger(\mathbf{A}\odot\mathbf{B})^H$.

B. Tensors

A tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{N_1\times\cdots\times N_R}$ of order R can be viewed as a discretized multidomain signal, with each of its entries indexed over R different domains.

Using multilinear algebra two tensors $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{N_1\times\cdots\times N_R}$ and $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{K_1\times\cdots\times K_R}$ may be related by a multilinear system of equations as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \bullet_1 \mathbf{U}_1 \bullet_2 \cdots \bullet_R \mathbf{U}_R, \tag{1}$$

where $\{\mathbf{U}_i \in \mathbb{C}^{N_i\times K_i}\}_{i=1}^R$ represents a set of matrices that relates the ith domain of $\boldsymbol{\mathcal{X}}$ and the so-called *core tensor* $\boldsymbol{\mathcal{G}}$, and $\bullet_i$ represents the ith mode product between a tensor and a matrix [5]; see Fig. 2a. Alternatively, vectorizing (1), we have

$$\mathbf{x} = (\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)\,\mathbf{g}, \tag{2}$$

with $\mathbf{x} = \mathrm{vec}(\boldsymbol{\mathcal{X}}) \in \mathbb{C}^{\tilde{N}}$; $\tilde{N} = \prod_{i=1}^R N_i$, and $\mathbf{g} = \mathrm{vec}(\boldsymbol{\mathcal{G}}) \in \mathbb{C}^{\tilde{K}}$; $\tilde{K} = \prod_{i=1}^R K_i$.

When the core tensor $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{K_c\times\cdots\times K_c}$ is hyperdiagonal (as depicted in Fig. 2b), (2) simplifies to

$$\mathbf{x} = (\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)\,\mathbf{g}, \tag{3}$$

with $\mathbf{g}$ collecting the main diagonal entries of $\boldsymbol{\mathcal{G}}$. Note that $\mathbf{g}$ has different meanings in (2) and (3), which can always be inferred from the context.

Such a multilinear system is commonly seen with $R = 2$ and $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \bullet_1 \mathbf{U}_1 \bullet_2 \mathbf{U}_2 = \mathbf{U}_2\boldsymbol{\mathcal{G}}\,\mathbf{U}_1^T$, for instance, in image processing when relating an image to its 2-dimensional Fourier transform with $\boldsymbol{\mathcal{G}}$ being the spatial Fourier transform of $\boldsymbol{\mathcal{X}}$, and $\mathbf{U}_1$ and $\mathbf{U}_2$ being inverse Fourier matrices related to the row and column spaces of the image, respectively. When dealing with Fourier matrices (more generally, Vandermonde matrices) with $\mathbf{U}_1 = \mathbf{U}_2$ and a diagonal tensor core, $\boldsymbol{\mathcal{X}}$ will be a Toeplitz covariance matrix, for which the sampling sets may be designed using sparse covariance sensing [18], [19].

III. Problem Modeling

We are concerned with the design of optimal sampling strategies for an Rth order tensor signal $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{N_1\times\cdots\times N_R}$, which admits a multilinear parameterization in terms of a

core tensor $\boldsymbol{\mathcal{G}} \in \mathbb{C}^{K_1 \times \cdots \times K_R}$ (dense or diagonal) of smaller dimensionality. We assume that the set of system matrices $\{\mathbf{U}_i\}_{i=1}^R$ are perfectly known, and that each of them is tall, i.e., $N_i > K_i$ for $i = 1, \ldots, R$, and has full column rank.

Sparse sampling a tensor $\boldsymbol{\mathcal{X}}$ is equivalent to selecting entries of $\mathbf{x} = \text{vec}(\boldsymbol{\mathcal{X}})$. Let $\tilde{\mathcal{N}}$ denote the set of indices of $\mathbf{x}$. Then, a particular sample selection is determined by a subset of selected indices $\mathcal{L}_{\text{un}} \subseteq \tilde{\mathcal{N}}$ such that $|\mathcal{L}_{\text{un}}| = L_{\text{un}}$ (subscript "un" denotes unstructured). This way, we can denote the process of sampling $\boldsymbol{\mathcal{X}}$ as a multiplication of $\mathbf{x}$ by a selection matrix $\boldsymbol{\Theta}(\mathcal{L}_{\text{un}}) \in \{0,1\}^{L_{\text{un}} \times \tilde{N}}$ such that

$$\mathbf{y} = \boldsymbol{\Theta}(\mathcal{L}_{\text{un}})\mathbf{x} = \boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)\mathbf{g}, \tag{4}$$

for a dense core [cf. (2)], and

$$\mathbf{y} = \boldsymbol{\Theta}(\mathcal{L}_{\text{un}})\mathbf{x} = \boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)\mathbf{g}, \tag{5}$$

for a diagonal core [cf. (3)]. Here, $\mathbf{y}$ is a vector containing the L_{un} selected entries of $\mathbf{x}$ indexed by the set $\mathcal{L}_{\text{un}}$.

For each case, if $\boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)$ and $\boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)$ have full column rank, then knowing $\mathbf{y}$ allows to retrieve a unique least squares solution, $\hat{\mathbf{g}}$, as

$$\hat{\mathbf{g}} = [\boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)]^{\dagger}\,\mathbf{y}, \tag{6}$$

or

$$\hat{\mathbf{g}} = [\boldsymbol{\Theta}(\mathcal{L}_{\text{un}})(\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)]^{\dagger}\,\mathbf{y}, \tag{7}$$

depending on whether $\boldsymbol{\mathcal{G}}$ is dense or hyperdiagonal. Next, we estimate $\boldsymbol{\mathcal{X}}$ using either (2) or (3).

In many applications, such as transmitter-receiver placement in multiple input multiple output (MIMO) radar [20], it is not possible to perform sparse sampling in an unstructured manner by ignoring the underlying domains. For these applications, some unstructured sparse sample selections generally require using a dense sensor selection in each domain (as shown in Fig. 1b), which produces a significant increase in hardware cost. Also, there is no particular structure in (6) and (7) that may be exploited to compute the pseudo-inverses, thus leading to a high computational cost to estimate $\mathbf{x}$. Finally, in the multidomain case, the dimensionality grows rather fast making it difficult to store the matrix $(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)$ or $(\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)$ to perform row subset selection. For all these reasons, we will constrain ourselves to the case where the sampling matrix has a compatible Kronecker structure. In particular, we define a new sampling matrix

$$\boldsymbol{\Phi}(\mathcal{L}) := \boldsymbol{\Phi}_1(\mathcal{L}_1) \otimes \cdots \otimes \boldsymbol{\Phi}_R(\mathcal{L}_R), \tag{8}$$

where each $\boldsymbol{\Phi}_i(\mathcal{L}_i)$ represents a selection matrix for the ith factor of $\boldsymbol{\mathcal{X}}$, $\mathcal{L}_i \subseteq \mathcal{N}_i$ is the set of selected row indices from the matrix $\mathbf{U}_i$ for $i = 1, \ldots, R$, and $\mathcal{L} = \bigcup_{i=1}^R \mathcal{L}_i$ and $\mathcal{L}_i \cap \mathcal{L}_j = \varnothing$ for $i \neq j$.

We will use the notation $|\mathcal{L}_i| = L_i$ and $|\mathcal{L}| = \sum_{i=1}^R L_i = L$ to denote the number of selected sensors per domain and the total number of selected sensors, respectively; whereas $\tilde{\mathcal{L}} = \mathcal{L}_1 \times \cdots \times \mathcal{L}_R$ and $\tilde{L} = |\tilde{\mathcal{L}}| = \prod_{i=1}^R L_i$ denote the set of sample indices and the total number of samples acquired with the above Kronecker-structured sampler. In order to simplify the notation, whenever it will be clear, we will drop the explicit dependency of $\boldsymbol{\Phi}_i(\mathcal{L}_i)$ on the set of selected rows $\mathcal{L}_i$, from now on, and simply use $\boldsymbol{\Phi}_i$.

Imposing a Kronecker structure on the sampling scheme means that sampling can be performed independently for each domain. In the *dense core tensor* case [cf. (2)], we have

$$\begin{aligned}\mathbf{y} &= (\boldsymbol{\Phi}_1 \otimes \cdots \otimes \boldsymbol{\Phi}_R)\,(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)\,\mathbf{g} \\ &= (\boldsymbol{\Phi}_1\mathbf{U}_1 \otimes \cdots \otimes \boldsymbol{\Phi}_R\mathbf{U}_R)\,\mathbf{g} = \boldsymbol{\Psi}(\mathcal{L})\mathbf{g},\end{aligned} \tag{9}$$

whereas in the *diagonal core tensor* case [cf. (3)], we have

$$\begin{aligned}\mathbf{y} &= (\boldsymbol{\Phi}_1 \otimes \cdots \otimes \boldsymbol{\Phi}_R)\,(\mathbf{U}_1 \odot \cdots \odot \mathbf{U}_R)\,\mathbf{g} \\ &= (\boldsymbol{\Phi}_1\mathbf{U}_1 \odot \cdots \odot \boldsymbol{\Phi}_R\mathbf{U}_R)\,\mathbf{g} = \boldsymbol{\Psi}(\mathcal{L})\mathbf{g}.\end{aligned} \tag{10}$$

As in the unstructured case, whenever (9) or (10) are overdetermined, using least squares, we can estimate the core $\hat{\mathbf{g}} = \boldsymbol{\Psi}^{\dagger}(\mathcal{L})\mathbf{y}$ as

$$\hat{\mathbf{g}} = \left[(\boldsymbol{\Phi}_1\mathbf{U}_1)^{\dagger} \otimes \cdots \otimes (\boldsymbol{\Phi}_R\mathbf{U}_R)^{\dagger}\right]\mathbf{y}, \tag{11}$$

or

$$\begin{aligned}\hat{\mathbf{g}} = &\left[(\boldsymbol{\Phi}_1\mathbf{U}_1)^H\,(\boldsymbol{\Phi}_1\mathbf{U}_1) \circ \cdots \circ (\boldsymbol{\Phi}_R\mathbf{U}_R)^H\,(\boldsymbol{\Phi}_R\mathbf{U}_R)\right]^{\dagger} \\ &\times \left[(\boldsymbol{\Phi}_1\mathbf{U}_1)^H \odot \cdots \odot (\boldsymbol{\Phi}_R\mathbf{U}_R)^H\right]\mathbf{y},\end{aligned} \tag{12}$$

and then reconstruct $\hat{\mathbf{x}}$ using (2) or (3), respectively. Comparing (11) and (12) to (6) and (7) we can see that leveraging the Kronecker structure of the proposed sampling scheme allows to greatly reduce the computational complexity of the least-squares problem, as the pseudoinverses in (11) and (12) are taken on matrices of a much smaller dimensionality than in (6) and (7). An illustration of the comparison between unstructured sparse sensing and Kronecker-structured sparse sensing is shown in Fig. 3 for $R = 2$.

Suppose the measurements collected in $\mathbf{y}$ are perturbed by zero-mean white Gaussian noise with unit variance, then the least-squares solution has the inverse error covariance or the Fisher information matrix $\mathbf{T}(\mathcal{L}) = \mathbb{E}\{(\mathbf{g} - \hat{\mathbf{g}})(\mathbf{g} - \hat{\mathbf{g}})^H\} = \boldsymbol{\Psi}^H(\mathcal{L})\boldsymbol{\Psi}(\mathcal{L})$ that determines the quality of the estimators $\hat{\mathbf{g}}$. Therefore, we can use scalar functions of $\mathbf{T}(\mathcal{L})$ as a figure of merit to propose the sparse tensor sampling problem

$$\underset{\mathcal{L}_1,\ldots,\mathcal{L}_R}{\text{optimize}}\, f\left\{\mathbf{T}(\mathcal{L})\right\} \;\text{ s. to }\; \sum_{i=1}^R |\mathcal{L}_i| = L,\; \mathcal{L} = \bigcup_{i=1}^R \mathcal{L}_i, \tag{13}$$

where with "optimize" we mean either "maximize" or "minimize" depending on the choice of the scalar function $f\{\cdot\}$. Solving (13) is not trivial due to the cardinality constraints. Therefore, in the following, we will propose tight surrogates for typically used scalar performance metrics $f\{\cdot\}$ in design of experiments with which the above discrete optimization problem can be solved efficiently and near optimally.

Note that the cardinality constraint in (13) restricts the total number of selected sensors to L, without imposing any constraint on the total number of gathered samples $\tilde{L}$. Although the maximum number of samples can be constrained using the constraint $\sum_{i=1}^R \log|\mathcal{L}_i| \leq \tilde{L}$, the resulting near-optimal solvers

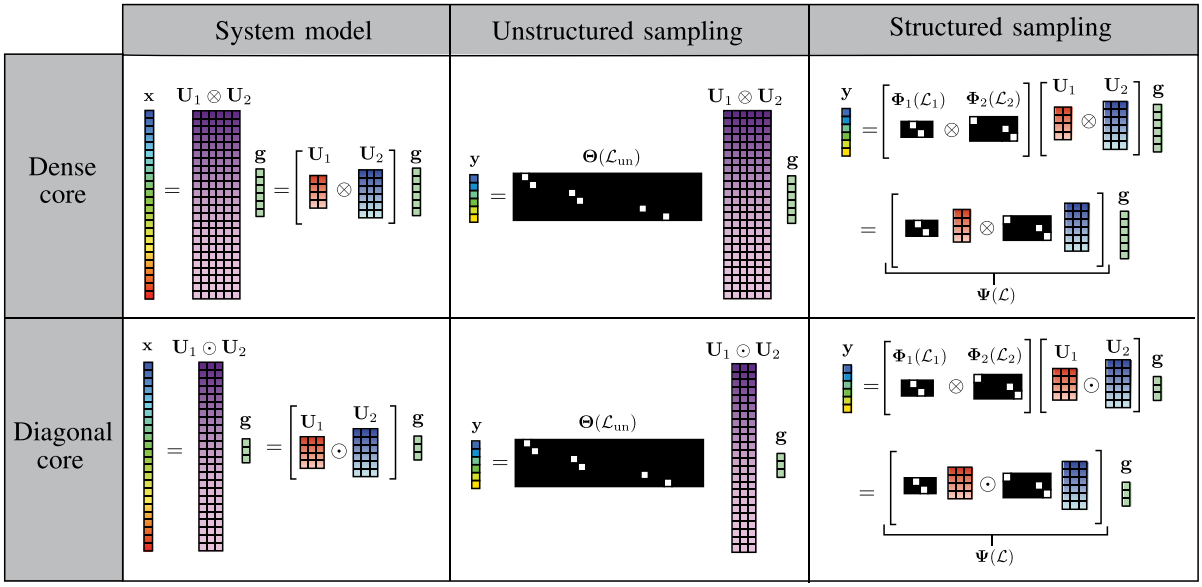


Fig. 3. Comparison between unstructured sampling and structured sampling ($R = 2$). Black (white) cells represent zero (one) entries, and colored cells represent arbitrary numbers.

are computationally intense with a complexity of about $\mathcal{O}(N^5)$ [21], [22]. Such heuristics are not suitable for the large-scale scenarios of interest.

Remark: For a given system model such as the ones in (2) or (3) designing an unstructured sampling set $\mathbf{\Theta}(\mathcal{L}_{\text{un}})$ would have a computational complexity of $\mathcal{O}(g(\tilde{N}, \tilde{L}, \tilde{K}))$, where g is a complexity measure function that would depend on the specific optimization method that is used to select the samples. On the other hand, as we will see later on, for the formulation in (13) and for certain performance measures the complexity of selecting a near-optimal sampling set becomes $\mathcal{O}(g'(N, L, K))$ with g' being another complexity measure function that would again depend on the sampling set design algorithm. For this reason, as long as g and g' scale similarly, as happens with our algorithms, we will achieve a major computational complexity reduction since $\tilde{N} \gg N$, $\tilde{L} \gg L$, and $\tilde{K} \gg K$.

A. Prior Art

Choosing the best subset of measurements from a large set of candidate sensing locations has received a lot of attention, particularly for $R = 1$, usually under the name of sensor selection/placement, which also is more generally referred to as sparse sensing [6]. Typically sparse sensing design is posed as a discrete optimization problem that finds the best sampling subset by optimizing a scalar function of the error covariance matrix. Some of the popular choices are, to minimize the mean squared error (MSE): $f\{\mathbf{T}(\mathcal{L})\} := \text{tr}\{\mathbf{T}^{-1}\}$ or the frame potential: $f\{\mathbf{T}(\mathcal{L})\} := \text{tr}\{\mathbf{T}^H\mathbf{T}\}$, or to maximize $f\{\mathbf{T}(\mathcal{L})\} := \lambda_{\min}\{\mathbf{T}\}$ or $f\{\mathbf{T}(\mathcal{L})\} := \log\det\{\mathbf{T}\}$. In this work, we will focus on the frame potential criterium as we will show later that this metric leads to very efficient sampler designs.

Depending on the strategy used to solve the optimization problem (13) we can classify the prior art in two categories: solvers based on convex optimization, and greedy methods that leverage submodularity. In the former category, [7] and [14] present several convex relaxations of the sparse sensing problem for different optimality criteria for inverse problems with linear and non-linear models, respectively. In particular, due to the Boolean nature of the sensor selection problem (i.e., a sensor is either selected or not), its related optimization problem is not convex. However, these constraints and the constraint on the number of selected sensors can be relaxed, and once the relaxed convex problem is solved, a thresholding heuristic (deterministic or randomized) can be used to recover a Boolean solution. Despite its good performance, the complexity of convex optimization solvers is rather high (cubic with the dimensionality of the signal). Therefore, the use of convex optimization approaches to solve the sparse sensing problem in large-scale scenarios, such as the sparse tensor sampling problem, gets even more computationally intense.

For high-dimensional scenarios, greedy methods (algorithms that select one sensor at a time) are more useful. The number of function evaluations performed by a greedy algorithm scales linearly with the number of sensors, and if one can prove submodularity of the objective function, its solution has a multiplicative near-optimality guarantee [23]. Several authors have followed this strategy and have proved submodularity of different optimality criteria such as *D*-optimality [9], mutual information [8], and frame potential [10], for the case $R = 1$. Later works illustrated a good performance of greedy methods on some easy to compute surrogate functions that have a positive correlation with some of the aforementioned performance measures (e.g. $\lambda_{\min}$) [11]. However, they fail to show theoretical near-optimality guarantees.

Besides parameter estimation, sparse sensing has also been studied for other common signal processing tasks, like detection [12], [15], [24] and filtering [13], [16], and has been expanded

to more general scenarios through the use of convex and matroid constraints that allow to robustify the sensing mechanisms by accounting for uncertainties in the system model [25] or some sensor functional constraints [20], [26]. Extensions to nonlinear system models have also been introduced in [27]. In [28], the authors propose an extension of the sparse sensing framework to a particular case of $R = 2$ (space-time signals): They suggest learning a spatio-temporal sampling pattern using different temporal selection matrices for every spatial node location. To obtain the sampling positions they use the model in [10] and optimize greedily the frame potential of every spatial location using a system model that was learnt using the aggregated data from previous iterations. However, although their methodology allows for the design of overall sparse space-time samplers, they do not impose individual domain sparsity in any of the domains, thus requiring a general dense deployment of space-time sensing resources (cf. Fig. 1b). In this sense, to the best of our knowledge, we are the first to extend the sparse sensing paradigm to the general case of $R \geq 1$ while enforcing sensor sparsity in all domains.

In a different context, the extension of compressed sensing (CS) to multidomain signals has been extensively studied [29]–[32]. CS is many times seen as a complementary sampling framework to sparse sensing [6], wherein CS the focus is on recovering a sparse signal rather than on designing a sparse measurement space.

B. Our Contributions

In this paper, we extend the sparse sampling framework to *multilinear inverse problems*. We refer to it as "sparse tensor sampling". We focus on two particular cases, depending on the structure of the core tensor $\boldsymbol{\mathcal{G}}$:

- *Dense core:* Whenever the core tensor is non-diagonal, sampling is performed based on (9). We will see that to ensure identifiability of the system, we need to select more entries in each domain than the rank of its corresponding system matrix, i.e., as a necessary condition we require $L \geq \sum_{i=1}^{R} K_i = K$ sensors, where $\{K_i\}_{i=1}^{R}$ are the dimensions of the core tensor $\boldsymbol{\mathcal{G}}$.
- *Diagonal core:* Whenever the core tensor is diagonal, sampling is performed based on (10). The use of the Khatri-Rao product allows for higher compression. In particular, under mild conditions on the entries of the factor matrices, we can guarantee identifiability of the sampled equations using $L \geq K_{\mathrm{c}} + R - 1$ sensors, where K_c is the length of the edges of the hypercubic core $\boldsymbol{\mathcal{G}}$.

For both the cases, we propose efficient greedy algorithms to compute a near-optimal sampling set.

C. Paper Outline

The remainder of the paper is organized as follows. In Sections IV and V, we develop solvers for the sparse tensor sampling problem with dense and diagonal core tensors, respectively. In Section VI, we provide a few examples to illustrate the developed framework. Finally, we conclude this paper by summarizing the results in Section VII.

IV. Dense Core Sampling

In this section, we focus on the most general situation when $\boldsymbol{\mathcal{G}}$ is an unstructured dense tensor. Our objective is to design the sampling sets $\{\mathcal{L}_i\}_{i=1}^{R}$ by solving the discrete optimization problem (13).

We formulate the sparse tensor sampling problem using the frame potential as a performance measure. Following the same rationale as in [10], but for multidomain signals, we will argue that the frame potential is a tight surrogate of the MSE. By doing so, we will see that when we impose a Kronecker structure on the sampling scheme, as in (8), the frame potential of $\boldsymbol{\Psi}$ can be factorized in terms of the frame potential of the different domain factors. This allows us to propose a low complexity algorithm for sampling tensor data.

Throughout this section, we will use tools from submodular optimization theory. Hence, we will start by introducing the main concepts related to submodularity in the next subsection.

A. Submodular Optimization

Submodularity is a notion based on the law of diminishing returns [33] that is useful to obtain heuristic algorithms with near-optimality guarantees for cardinality-constrained discrete optimization problems. More precisely, submodularity is formally defined as follows.

Definition 1 (Submodular function [33]): A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ defined over the subsets of $\mathcal{N}$ is submodular if, for every $\mathcal{X} \subseteq \mathcal{N}$ and $x, y \in \mathcal{N} \setminus \mathcal{X}$, we have

$$f(\mathcal{X} \cup \{x\}) - f(\mathcal{X}) \geq f(\mathcal{X} \cup \{x, y\}) - f(\mathcal{X} \cup \{y\}).$$

A function f is said to be supermodular if $-f$ is submodular.

Besides submodularity, many near-optimality theorems in discrete optimization require functions to be also monotone non-decreasing, and normalized.

Definition 2 (Monotonicity): A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is monotone non-decreasing if, for every $\mathcal{X} \subseteq \mathcal{N}$,

$$f(\mathcal{X} \cup \{x\}) \geq f(\mathcal{X}) \quad \forall x \in \mathcal{N} \setminus \mathcal{X}$$

Definition 3 (Normalization): A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is normalized if $f(\varnothing) = 0$.

In submodular optimization, matroids are generally used to impose constraints on an optimization, such as the ones in (13). A matroid generalizes the concept of linear independence in algebra to sets. Formally, a matroid is defined as follows.

Definition 4 (Matroid [34]): A finite matroid $\mathcal{M}$ is a pair $(\mathcal{N}, \mathcal{I})$, where $\mathcal{N}$ is a finite set and $\mathcal{I}$ is a family of subsets of $\mathcal{N}$ that satisfies: 1) The empty set is independent, i.e., $\varnothing \in \mathcal{I}$; 2) For every $\mathcal{X} \subseteq \mathcal{Y} \subseteq \mathcal{N}$, if $\mathcal{Y} \in \mathcal{I}$, then $\mathcal{X} \in \mathcal{I}$; and 3) For every $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{N}$ with $|\mathcal{Y}| > |\mathcal{X}|$ and $\mathcal{X}, \mathcal{Y} \in \mathcal{I}$ there exists one $x \in \mathcal{Y} \setminus \mathcal{X}$ such that $\mathcal{X} \cup \{x\} \in \mathcal{I}$.

In this paper we will deal with the following types of matroids.

Example 1 (Uniform matroid [34]): The subsets of $\mathcal{N}$ with at most K elements form a uniform matroid $\mathcal{M}_{\mathrm{u}} = (\mathcal{N}, \mathcal{I}_{\mathrm{u}})$ with $\mathcal{I}_{\mathrm{u}} = \{\mathcal{X} \subseteq \mathcal{N} : |\mathcal{X}| \leq K\}$.

Example 2 (Partition matroid [34]): If $\{\mathcal{N}_i\}_{i=1}^{R}$ form a partition of $\mathcal{N} = \bigcup_{i=1}^{R} \mathcal{N}_i$ then $\mathcal{M}_{\mathrm{p}} = (\mathcal{N}, \mathcal{I}_{\mathrm{p}})$ with $\mathcal{I}_{\mathrm{p}} = \{\mathcal{X} \subseteq \mathcal{N} : |\mathcal{X} \cap \mathcal{N}_i| \leq K_i \quad i = 1, \ldots, R\}$ defines a partition matroid.

Algorithm 1: Greedy Maximization Subject to T Matroid Constraints.

Require: $\mathcal{X} = \varnothing$, K, $\{\mathcal{I}_i\}_{i=1}^T$
1: **for** $k \leftarrow 1$ **to** K **do**
2: $\quad s^\star = \underset{s \notin \mathcal{X}}{\operatorname{argmax}} \{f(\mathcal{X} \cup \{s\}) : \mathcal{X} \cup \{s\} \in \bigcap_{i=1}^T \mathcal{I}_i\}$
3: $\quad \mathcal{X} \leftarrow \mathcal{X} \cup \{s^\star\}$
4: **end**
5: **return** $\mathcal{X}$

Example 3 (Truncated partition matroid [34]): The intersection of a uniform matroid $\mathcal{M}_\mathrm{u} = (\mathcal{N}, \mathcal{I}_\mathrm{u})$ and a partition matroid $\mathcal{M}_\mathrm{p} = (\mathcal{N}, \mathcal{I}_\mathrm{p})$ defines a truncated partition matroid $\mathcal{M}_\mathrm{t} = (\mathcal{N}, \mathcal{I}_\mathrm{p} \cap \mathcal{I}_\mathrm{u})$.

The matroid-constrained submodular optimization problem

$$\underset{\mathcal{X} \subseteq \mathcal{N}}{\text{maximize}}\, f(\mathcal{X}) \quad \text{subject to} \quad \mathcal{X} \in \bigcap_{i=1}^T \mathcal{I}_i \tag{14}$$

can be solved near optimally using Algorithm 1. This result is formally stated in the following theorem.

Theorem 1 (Matroid-constrained submodular maximization [35]): Let $f : 2^\mathcal{N} \to \mathbb{R}$ be a monotone non-decreasing, normalized, submodular set function, and $\{\mathcal{M}_i = (\mathcal{N}, \mathcal{I}_i)\}_{i=1}^T$ be a set of matroids defined over $\mathcal{N}$. Furthermore, let $\mathcal{X}^\star$ denote the optimal solution of (14), and let $\mathcal{X}_\text{greedy}$ be the solution obtained by Algorithm 1. Then

$$f(\mathcal{X}_\text{greedy}) \geq \frac{1}{T+1} f(\mathcal{X}^\star).$$

B. Greedy Method

The frame potential [36] of the matrix $\mathbf{\Psi}$ is defined as the trace of the Grammian matrix $\mathrm{FP}(\mathbf{\Psi}) := \mathrm{tr}\{\mathbf{T}^H\mathbf{T}\}$ with $\mathbf{T} = \mathbf{\Psi}^H\mathbf{\Psi}$. The frame potential can be related to the MSE, $\mathrm{MSE}(\mathbf{\Psi}(\mathcal{L})) = \mathrm{tr}\{\mathbf{T}^{-1}(\mathcal{L})\}$, using [10]

$$c_1 \frac{\mathrm{FP}\left(\mathbf{\Psi}(\mathcal{L})\right)}{\lambda^2_\text{max}\{\mathbf{T}(\mathcal{L})\}} \leq \mathrm{MSE}(\mathbf{\Psi}(\mathcal{L})) \leq c_2 \frac{\mathrm{FP}\left(\mathbf{\Psi}(\mathcal{L})\right)}{\lambda^2_\text{min}\{\mathbf{T}(\mathcal{L})\}}, \tag{15}$$

where c_1, and c_2 are constants that depend on the data model.

From the above bound, it is clear that by minimizing the frame potential of $\mathbf{\Psi}$ one can minimize the MSE, which is otherwise difficult to minimize as it is neither convex, nor submodular.

The frame potential of $\mathbf{\Psi}(\mathcal{L}) := \mathbf{\Psi}_1(\mathcal{L}_1) \otimes \cdots \otimes \mathbf{\Psi}_R(\mathcal{L}_R)$ can be expressed as the frame potential of its factors $\mathbf{\Psi}_i(\mathcal{L}_i) := \mathbf{\Phi}_i(\mathcal{L}_i)\mathbf{U}_i$. To show this, recall the definition of the frame potential as

$$\begin{aligned}\mathrm{FP}\left(\mathbf{\Psi}(\mathcal{L})\right) &= \mathrm{tr}\left\{\mathbf{T}^H(\mathcal{L})\mathbf{T}(\mathcal{L})\right\} \\ &= \mathrm{tr}\left\{\mathbf{T}_1^H\mathbf{T}_1 \otimes \cdots \otimes \mathbf{T}_R^H\mathbf{T}_R\right\}, \end{aligned} \tag{16}$$

where $\mathbf{T}_i = \mathbf{\Psi}_i^H\mathbf{\Psi}_i$. Now, using the fact that for any two matrices $\mathbf{A} \in \mathbb{C}^{K_A \times K_A}$ and $\mathbf{B} \in \mathbb{C}^{K_B \times K_B}$ we have $\mathrm{tr}\{\mathbf{A} \otimes \mathbf{B}\} = \mathrm{tr}\{\mathbf{A}\}\mathrm{tr}\{\mathbf{B}\}$, we can expand (16) as

$$\mathrm{FP}\left(\mathbf{\Psi}(\mathcal{L})\right) = \prod_{i=1}^R \mathrm{tr}\left\{\mathbf{T}_i^H\mathbf{T}_i\right\} = \prod_{i=1}^R \mathrm{FP}\left(\mathbf{\Psi}_i(\mathcal{L}_i)\right).$$

For brevity, we will write the above expression alternatively as an explicit function of the selection sets $\mathcal{L}_i$:

$$F(\mathcal{L}) := \mathrm{FP}\left(\mathbf{\Psi}(\mathcal{L})\right) = \prod_{i=1}^R F_i(\mathcal{L}_i) := \prod_{i=1}^R \mathrm{FP}\left(\mathbf{\Psi}_i(\mathcal{L}_i)\right). \tag{17}$$

Expression (17) shows again the advantage of working with a Kronecker-structured sampler: instead of computing every cross-product between the columns of $\mathbf{\Psi}$ to compute the frame potential, we can arrive to the same value using the frame potential of $\{\mathbf{\Psi}_i\}_{i=1}^R$.

1) Submodularity of $F(\mathcal{L})$: Function $F(\mathcal{L})$ as defined in (17) does not directly meet the conditions for the guarantees provided by Theorem 1, but it can be modified slightly to satisfy them. In this sense, we define the function $G : 2^\mathcal{N} \to \mathbb{R}$ on the subsets of $\mathcal{N}$ as

$$G(\mathcal{S}) := F(\mathcal{N}) - F(\mathcal{N} \backslash \mathcal{S}), \tag{18}$$

where $F(\mathcal{N}) = \prod_{i=1}^R F_i(\mathcal{N}_i)$, $F(\mathcal{N}\backslash\mathcal{S}) = \prod_{i=1}^R F_i(\mathcal{N}_i \backslash \mathcal{S}_i)$, and $\mathcal{S} = \bigcup_{i=1}^R \mathcal{S}_i$, $\mathcal{S}_i \cap \mathcal{S}_j = \varnothing$ for $i \neq j$. Therefore, $\{\mathcal{S}_i\}_{i=1}^R$ form a partition of $\mathcal{S}$.

It is clear that if we make the change of variables from $\mathcal{L}$ to $\mathcal{S}$ maximizing G over $\mathcal{S}$ is the same as minimizing the frame potential over $\mathcal{L}$. However, working with the complement set results in a set function that is submodular and monotone non-decreasing, as shown in the next theorem. Consequently, G satisfies the conditions for the results in Theorem 1.

Theorem 2: The set function $G(\mathcal{S})$ defined in (18) is a normalized, monotone non-decreasing, submodular function for all subsets of $\mathcal{N} = \bigcup_{i=1}^R \mathcal{N}_i$.

Proof: See Appendix A. ■

With this result we can now claim near-optimality of the greedy algorithm that solves the cardinality constrained maximization of $G(\mathcal{S})$. However, as we said, minimizing the frame potential only makes sense as long as (15) is tight. In particular, whenever $\mathbf{T}(\mathcal{L})$ is singular we know that the MSE is infinity, and hence (15) is meaningless. For this reason, next to the cardinality constraint in (13) that limits the total number of sensors, we need to ensure that $\mathbf{\Psi}(\mathcal{L})$ has full column rank, i.e., $L_i \geq K_i$ for $i = 1, \ldots, R$. In terms of $\mathcal{S}$, this is equivalent to

$$|\mathcal{S}_i| = |\mathcal{N}_i \setminus \mathcal{L}_i| \leq N_i - K_i \quad i = 1, \ldots, R, \tag{19}$$

where this set of constraints forms a partition matroid $\mathcal{M}_p = (\mathcal{N}, \mathcal{I}_p)$ [cf. Example 2 from Definition 4]. Hence, we can introduce the following submodular optimization problem as surrogate for the minimization of the frame potential

$$\underset{\mathcal{S} \subseteq \mathcal{N}}{\text{maximize}}\, G(\mathcal{S}) \;\text{ s. to }\; \mathcal{S} \in \mathcal{I}_\mathrm{u} \cap \mathcal{I}_p, \tag{20}$$

with $\mathcal{I}_\mathrm{u} = \{\mathcal{A} \subseteq \mathcal{N} : |\mathcal{A}| \leq N - L\}$ and $\mathcal{I}_p = \{\mathcal{A} \subseteq \mathcal{N} : |\mathcal{A} \cap \mathcal{N}_i| \leq N_i - K_i \;\; i = 1, \ldots, R\}$. Theorem 1 gives, therefore, all the ingredients to assess the near-optimality of Algorithm 1 applied on (20), for which the results are particularized in the following corollary.

Corollary 1: The greedy solution $\mathcal{S}_\text{greedy}$ to (20) obtained from Algorithm 1) is 1/2 near optimal, i.e., $G(\mathcal{S}_\text{greedy}) \geq \frac{1}{2} G(\mathcal{S}^\star)$.

Proof: Follows from Theorem 1, and since (20) has $T = 1$ (truncated-partition) matroid constraint. ■

Remark: Note that the alternative formulation of (20) that maximizes the number of removed sensors subject to some performance guarantee, i.e.,

$$\text{maximize}_{\mathcal{S}\subseteq\mathcal{N}} |\mathcal{S}| \text{ s. to. } F(\mathcal{N}\backslash\mathcal{S}) \leq \gamma \text{ and } \mathcal{S} \in \mathcal{I}_p, \quad (21)$$

can also be solved by the same greedy algorithm by stopping the iterations whenever $F(\mathcal{N} \setminus \mathcal{L}) \leq \gamma$ instead of when $|\mathcal{S}| = N - L$.

Next, we compute an explicit bound with respect to the frame potential of $\mathbf{\Psi}$, which is the objective function we initially wanted to minimize. This bound is given in the following theorem.

Theorem 3: The greedy solution $\mathcal{L}_{\text{greedy}}$ to (20) obtained from Algorithm 1 is near optimal with respect to the frame potential as $F(\mathcal{L}_{\text{greedy}}) \leq \gamma F(\mathcal{L}^\star)$ with $\gamma = \frac{1}{2}(\frac{K}{L_{\min}^2}\prod_{i=1}^R F_i(\mathcal{N}_i) + 1)$, and $L_{\min} = \min_{i\in\mathcal{L}} \|\mathbf{u}_i\|_2^2$, being $\mathbf{u}_i$ the ith row of $(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)$.

Proof: Obtained similar to the bound in [10], but specialized for (17) and 1/2-near-optimality; see [37] for details. ■

As with the $R = 1$ case in [10], γ is heavily influenced by the frame potential of $(\mathbf{U}_1 \otimes \cdots \otimes \mathbf{U}_R)$. Specifically, the approximation gets tighter when $F(\mathcal{N})$ is small or the core tensor dimensionality decreases.

2) Computational Complexity: The running time of Algorithm 1 applied to solve (20) can greatly be reduced by precomputing the inner products between the rows of every $\mathbf{U}_i$ before starting the iterations. This has a complexity of $\mathcal{O}(N_i^2 K_i)$ for each domain. Once these inner products are computed, in each iteration we need to find R times the maximum over $\mathcal{O}(N_i)$ elements. Since we run $N - L$ iterations, the complexity of all iterations is $\mathcal{O}(N_{\max}^2)$, with $N_{\max} = \max_i N_i$. Therefore, the total computational complexity of the greedy method is $\mathcal{O}(N_{\max}^2 K_{\max})$ with $K_{\max} = \max_i K_i$, whereas performing the sample selection in an unstructured manner requires $\mathcal{O}(\tilde{N}^2\tilde{K})$ computations, as described in [10], where we now need to find a subset of rows of a matrix of size $\tilde{N} \times \tilde{K}$. In light of this result, and considering that $N_{\max} \ll \tilde{N}$ and $K_{\max} \ll \tilde{K}$, we highlight once more the computational advantage of the structured approach which effectively puts a cap on the exponential increase in complexity with the tensor order.

3) Practical Considerations: Due to the characteristics of the greedy iterations, the algorithm tends to give solutions with a very unbalanced cardinality. In particular, for most situations, the algorithm chooses one of the domains in the first few iterations and empties that set till it hits the identifiability constraint of that domain. Then, it proceeds to another domain and empties it as well, and so on. This is due to the objective function, which is a product of smaller objectives. Indeed, if we are asked to minimize a product of two elements by subtracting a value from them, it is generally better to subtract from the smallest element. Hence, if this minimization is performed multiple times we will tend to remove always from the same element.

The consequences of this behavior are twofold. On the one hand, this greedy method tends to give a sensor placement that yields a very small number of samples $\tilde{L}$, as we will also see in the simulations. Therefore, when comparing this method to other sensor selection schemes that produce solutions with a larger $\tilde{L}$ it generally ranks worse in MSE for a given L. On the other hand, the solution of this scheme tends to be tight on the identifiability constraints for most of the domains, thus hampering the performance on those domains. This implication, however, has a simple solution. By introducing a small slack variable $\alpha_i > 0$ to the constraints, we can obtain a sensor selection which is not tight on the constraints. This amounts to solving the problem

$$\begin{aligned} &\underset{\mathcal{S}\subseteq\mathcal{N}}{\text{maximize}}\, G(\mathcal{S}) \\ &\text{s. to} \quad |\mathcal{S}| = N - L, |\mathcal{S} \cap \mathcal{N}_i| \leq N_i - K_i - \alpha_i, i = 1, \ldots, R. \end{aligned} \quad (22)$$

Tuning $\{\alpha_i\}_{i=1}^R$ allows to regularize the tradeoff between compression and accuracy of the greedy solution.

On the other hand, as explained in [10] the greedy minimization of the frame potential tends to select the rows with the highest energy first regardless of their direction. For this reason, we follow the advice in [10] and normalize all rows of $\{\mathbf{U}_i\}_{i=1}^R$ before starting the algorithm iterations.

We conclude this section with a remark on an alternative performance measure.

Remark: As a alternative performance measure, one can think on maximizing the set function $\log\det\{\mathbf{T}(\mathcal{L})\}$. Although this set function can be shown to be submodular over all subsets of $\mathcal{N}$ [37], the related greedy algorithm cannot be constrained to always result in an identifiable system after subsampling. Thus, its applicability is more limited than the frame potential formulation; see [37] for a more detailed discussion.

V. Diagonal Core Sampling

So far, we have focused on the case when $\boldsymbol{\mathcal{G}}$ is dense and has no particular structure. In that case, we have seen that we require at least $\sum_{i=1}^R K_i$ sensors to recover our signal with a finite MSE. In many cases of interest, $\boldsymbol{\mathcal{G}}$ admits a structure. In particular, in this section, we investigate the case when $\boldsymbol{\mathcal{G}}$ is a diagonal tensor. Under some mild conditions on the entries of $\{\mathbf{U}\}_{i=1}^R$, we can leverage the structure of $\boldsymbol{\mathcal{G}}$ to further increase the compression. As before, we develop an efficient and near-optimal greedy algorithm based on minimizing the frame potential to design the sampling set.

A. Identifiability Conditions

In contrast to the dense core case, the number of unknowns in a multilinear system of equations with a diagonal core does not increase with the tensor order, whereas for a dense core it grows exponentially. This means that when sampling signals with a diagonal core decomposition, one can expect a stronger compression.

To derive the identifiability conditions for (10), we present the result from [38] as the following theorem.

Theorem 4 (Rank of Khatri-Rao product [38]): Let $\mathbf{A} \in \mathbb{C}^{N\times K}$ and $\mathbf{B} \in \mathbb{C}^{M\times K}$ be two matrices with no all-zero column. Then,

$$\operatorname{rank}(\mathbf{A} \odot \mathbf{B}) \geq \max\{\operatorname{rank}(\mathbf{A}), \operatorname{rank}(\mathbf{B})\}.$$

Based on the above theorem, we can give the following sufficient conditions for identifiability of the system (10).

Corollary 2: Let z_i denote the maximum number of zero entries in any column of $\mathbf{U}_i$. If for every $\mathbf{\Psi}_i(\mathcal{L}_i)$ we have $|\mathcal{L}_i| > z_i$, and there is at least one $\mathbf{\Psi}_j$ with $\operatorname{rank}(\mathbf{\Psi}_j) = K_{\mathrm{c}}$, then $\mathbf{\Psi}(\mathcal{L})$ has full column rank.

Proof: Selecting $L_i > z_i$ rows from each $\mathbf{U}_i$ ensures that no $\mathbf{\Psi}_i$ will have an all-zero column. Then, if for at least one $\mathbf{\Psi}_j$ we have $\operatorname{rank}(\mathbf{\Psi}_j) = K_{\mathrm{c}}$, then due to Theorem 4 we have

$$\begin{aligned}\operatorname{rank}(\mathbf{\Psi}(\mathcal{L})) &\geq \max_{i=1,\ldots,R}\{\operatorname{rank}(\mathbf{\Psi}_i)\} \\ &= \max\left\{\operatorname{rank}(\mathbf{\Psi}_j), \max_{i\neq j}\{\operatorname{rank}(\mathbf{\Psi}_j)\}\right\} = K_{\mathrm{c}}.\end{aligned}$$

■

Therefore, in order to guarantee identifiability we need to select $L_j \geq \max\{K_{\mathrm{c}}, z_j + 1\}$ rows from any factor matrix j, and $L_i \geq \max\{1, z_i + 1\}$ from the other factors with $i \neq j$. In many scenarios, we usually have $\{z_i = 0\}_{i=1}^R$ since no entry in $\{\mathbf{U}_i\}_{i=1}^R$ will exactly be zero. In those situations we will require to select at least $L = \sum_{i=1}^R L_i \geq K_{\mathrm{c}} + R - 1$ elements.

B. Greedy Method

As we did for the case with a dense core, we start by finding an expression for the frame potential of a Khatri-Rao product in terms of its factors. The Grammian matrix $\mathbf{T}(\mathcal{L})$ of a diagonal core tensor decomposition has the form

$$\begin{aligned}\mathbf{T} = \mathbf{\Psi}^H\mathbf{\Psi} &= (\mathbf{\Psi}_1 \odot \cdots \odot \mathbf{\Psi}_R)^H (\mathbf{\Psi}_1 \odot \cdots \odot \mathbf{\Psi}_R) \\ &= \mathbf{\Psi}_1^H\mathbf{\Psi}_1 \circ \cdots \circ \mathbf{\Psi}_R^H\mathbf{\Psi}_R = \mathbf{T}_1 \circ \cdots \circ \mathbf{T}_R.\end{aligned}$$

Using this expression, the frame potential of a Khatri-Rao product becomes

$$\mathrm{FP}(\mathbf{\Psi}) = \operatorname{tr}\left\{\mathbf{T}^H\mathbf{T}\right\} = \|\mathbf{T}\|_F^2 = \|\mathbf{T}_1 \circ \cdots \circ \mathbf{T}_R\|_F^2. \quad (23)$$

For brevity, we will denote the frame potential as an explicit function of the selected set as

$$P(\mathcal{L}) := \mathrm{FP}(\mathbf{\Psi}(\mathcal{L})) = \|\mathbf{T}_1(\mathcal{L}_1) \circ \cdots \circ \mathbf{T}_R(\mathcal{L}_R)\|_F^2. \quad (24)$$

Unlike in the dense core case, the frame potential of a Khatri-Rao product cannot be separated in terms of the frame potential of its factors. Instead, (23) decomposes the frame potential using the Hadamard product of the Grammian of the factors.

1) Submodularity of $P(\mathcal{L})$: Since $P(\mathcal{L})$ does not directly satisfy the conditions for the guarantees provided by Theorem 1, we propose using the following set function $Q : 2^{\mathcal{N}} \to \mathbb{R}$ as a surrogate for the frame potential

$$Q(\mathcal{S}) := P(\mathcal{N}) - P(\mathcal{N}\backslash\mathcal{S}), \quad (25)$$

with $P(\mathcal{N}) = \|\mathbf{T}_1(\mathcal{N}_1) \circ \cdots \circ \mathbf{T}_r(\mathcal{N}_r)\|_F^2$ and $P(\mathcal{N}\backslash\mathcal{S}) = \|\mathbf{T}_1(\mathcal{N}_1\backslash\mathcal{S}_1) \circ \cdots \circ \mathbf{T}_R(\mathcal{N}_R \setminus \mathcal{S}_R)\|_F^2$.

Theorem 5: The set function $Q(\mathcal{S})$ defined in (25) is a normalized, monotone non-decreasing, submodular function for all subsets of $\mathcal{N} = \bigcup_{i=1}^R \mathcal{N}_i$.

Proof: See Appendix B. ■

Using Q and imposing the identifiability constraints defined in Section V-A, we can write the related optimization problem for the minimization of the frame potential as

$$\underset{\mathcal{S}\subseteq\mathcal{N}}{\text{maximize}}\, Q(\mathcal{S}) \quad \text{s. to} \quad \mathcal{S} \in \mathcal{I}_u \cap \mathcal{I}_p, \quad (26)$$

where $\mathcal{I}_u = \{\mathcal{A} \subseteq \mathcal{N} : |\mathcal{S}| \leq N - L\}$ and $\mathcal{I}_p = \{\mathcal{A} \subseteq \mathcal{N} : |\mathcal{A} \cap \mathcal{N}_i| \leq \beta_i \quad i = 1, \ldots, R\}$ with $\beta_j = N_j - \max\{K_{\mathrm{c}}, z_j\}$ and $\beta_i = N_i - \max\{1, z_i + 1\}$ for $i \neq j$. Here, the choice of the index j is arbitrary, and can be set depending on the application. For example, with some space-time signals it is more costly to sample space than time, and, in those cases, j is generally chosen for the temporal domain.

As a result, (26) is a submodular maximization problem with a truncated partition matroid constraint [cf. Example 2 from Definition 4]. Thus, from Theorem 1, we know that greedy maximization of (26) using Algorithm 1 has a multiplicative near-optimal guarantee. This result is made precise in the following corollary.

Corollary 3: The greedy solution $\mathcal{S}_{\text{greedy}}$ to (26) obtained using Algorithm 1 is $1/2$ near optimal, i.e., $Q(\mathcal{S}_{\text{greedy}}) \geq \frac{1}{2}Q(\mathcal{S}^\star)$. Here, $\mathcal{S}^\star$ denotes the optimal solution of (26).

Similar to the dense core case, we can also provide a bound on the near-optimality of of the greedy solution with respect to the frame potential.

Theorem 6: The solution set $\mathcal{L}_{\text{greedy}} = \mathcal{N} \setminus \mathcal{S}_{\text{greedy}}$ obtained from Algorithm 1 is near optimal with respect to the frame potential as $P(\mathcal{L}_{\text{greedy}}) \leq \gamma P(\mathcal{L}^\star)$, with $\gamma = 0.5(\|\mathbf{T}_1(\mathcal{N}_1) \circ \cdots \circ \mathbf{T}_R(\mathcal{N}_R)\|_F^2 K L_{\min}^{-2} + 1)$ and $\mathcal{L}^\star = \mathcal{N} \setminus \mathcal{S}^\star$.

Proof: Based on the proof of Theorem 3. The bound is obtained using (23) instead of (17) in the derivation. ■

2) Computational Complexity: The computational complexity of the greedy method is now governed by the complexity of computing the Grammian matrices $\mathbf{T}_i$. This can greatly be improved if before starting the iterations, one precomputes all the outer products in $\{\mathbf{T}_i\}_{i=1}^R$. Doing this has a computational complexity of $\mathcal{O}(N_{\max}K_{\mathrm{c}}^2)$. Then, in every iteration, the evaluation of $P(\mathcal{L})$ would only cost $\mathcal{O}(RK_{\mathrm{c}}^2)$ operations. Further, because in every iteration we need to query $\mathcal{O}(N_i)$ elements on each domain, and we run the algorithm for $N - L$ iterations, the total time complexity of the iterations is $\mathcal{O}(RN_{\max}^2K_{\mathrm{c}}^2)$. This term dominates over the complexity of the precomputations, and thus can be treated as the worst case complexity of the greedy method. On the other hand, the unstructured sample selection requires $\mathcal{O}(\tilde{N}^2K)$ computations, corresponding to the complexity of selecting rows from a matrix of size $\tilde{N} \times K$. Again, although more moderate than in the dense core case, the gain in computational complexity between the structured and unstructured case is very substantial.

3) Practical Considerations: The proposed scheme suffers from the same issues as in the dense core case. Namely, it tends to empty the domains sequentially, thus producing solutions which are tight on the identifiability constraints. Nevertheless, as we

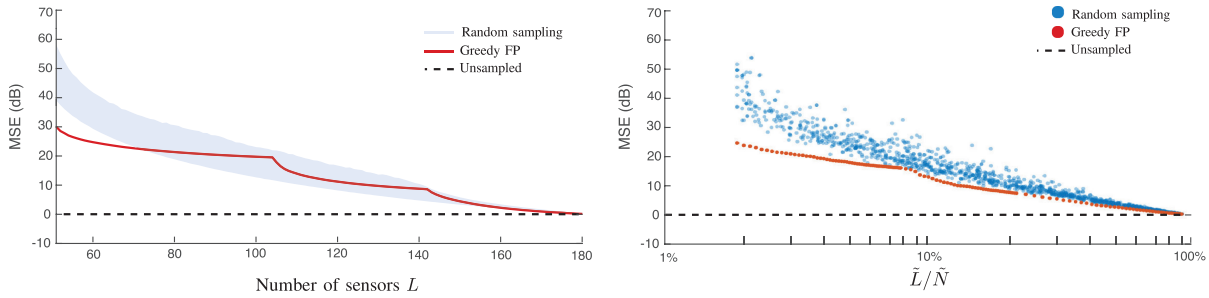


Fig. 4. Dense core with $R = 3$, $N_1 = 50$, $N_2 = 60$, $N_3 = 70$, $K_1 = 10$, $K_2 = 20$, $K_3 = 15$, and $\alpha_1 = \alpha_2 = \alpha_3 = 2$.

indicated for the dense core, the drop in performance associated with the proximity of the solutions to the constraints can be reduced by giving some slack to the constraints. We also normalize all rows of the system matrices before performing the sensor selection.

VI. Numerical Results

In this section[1], we will illustrate the developed framework through several examples. First, we will show some results obtained on synthetic datasets to compare the performance of the different near-optimal algorithms. Then, we will focus on large-scale real-world examples related to (i) graph signal processing: *sampling product graphs for active learning in recommender systems*, and (ii) array processing for wireless communications: *multiuser source separation*, to show the benefits of the developed framework.

A. Synthetic Example

1) Dense Core: We compare the performance in terms of the theoretical MSE of our proposed greedy algorithm (henceforth referred to as greedy-FP) to a random sampling scheme based on randomly selecting rows of $\mathbf{U}_i$ such that the resulting subset of samples also have a Kronecker structure. Only those selections that satisfy the identifiability constraints in (19) are considered valid. We note that the time complexity of evaluating M times the MSE for a Kronecker-structured sampler is $\mathcal{O}(MN_{\max}^2 K_{\max})$. For this reason, using many realizations (say, a large number M) of random sampling to obtain a good sparse sampler is computationally intense.

To perform this comparison, we perform 100 Monte-Carlo experiments, and in each experiment, random matrices with dimensions $N_1 = 50$, $N_2 = 60$, and $N_3 = 70$, as well as $K_1 = 10$, $K_2 = 20$, and $K_3 = 15$ are drawn with normal Gaussian distributed entries. For each Monte-Carlo trial, we solve (20) for a different number of sensors L using greedy-FP and compare its performance in terms of MSE to $M = 100$ random structured sparse samplers[2] obtained for each value of L. Fig. 4 shows the results of these experiments. The plot on the left shows the performance averaged over the different models against the number of sensors, wherein the blue shaded area represents the 10–90 percentile average interval of the random sampling scheme. The performance values, in dB scale, are normalized by the value of the unsampled MSE. Because the estimation performance is heavily influenced by its related number of samples $\tilde{L}$, and noting the fact that a value of L may lead to different $\tilde{L}$, we also present, in the plot on the right side of Fig. 4, the performance comparison for one model realization against the relative number of samples $\tilde{L}/\tilde{N}$ so that differences in the informative quality of the selections are highlighted.

The plots in Fig. 4 illustrate some important features of the proposed sparse sampling method. When comparing the performance against the number of sensors, we see that there are areas where greedy-FP performs as well as random sampling. However, when comparing the same results against the number of samples we see that greedy-FP consistently performs better than random sampling. The reason for this disparity is due to characteristics of greedy-FP that we introduced in Section IV-B. Namely, the tendency of greedy-FP to produce sampling sets with the minimum number of samples.

On the other hand, the performance curve of greedy-FP shows three bumps (recall that we use $R = 3$). Again, this is a consequence of greedy-FP trying to meet the identifiability constraints in (20) with equality. As we increase L, the solutions of greedy-FP increase in cardinality by adding more elements to a single domain until the constraints are met, and then proceed to the next domain. The bumps in Fig. 4 correspond precisely to these instances. Furthermore, as seen from the exponential decay after every plateau in Fig. 4, slightly increasing the number of elements in one domain beyond its identifiability value (on every plateau only one domain is being filled) induces a significant improvement on the reconstruction performance. This suggests that in practical applications α_i can be set to a very small value (e.g. $\alpha_i = 0.05N_i$) and still achieve a good balance between compression and accuracy.

2) Diagonal Core: We perform the same experiment for the diagonal core case, this time with dimensions $N_1 = 50$, $N_2 = 60$, $N_3 = 70$, and $K_c = 20$. The results are shown in Fig. 5. Again we see that the proposed algorithm outperforms random sampling, especially when collecting just a few samples. Furthermore, as happened in the dense core case, the performance curve of greedy-FP follows a stairway shape.

[1]The code to reproduce these experiments can be found at https://gitlab.com/gortizji/sparse_tensor_sensing

[2]The programmatic generation of structured random samplers from a uniform distribution that meet the identifiability constraints is not a trivial problem. In this work, we use a stars-and-bars algorithm [39] for the selection of the random $\{L_i\}$ set dimensions where we discard any sample that does not satisfy $L_i > K_i$ for all $i \in \{1, \ldots, R\}$.

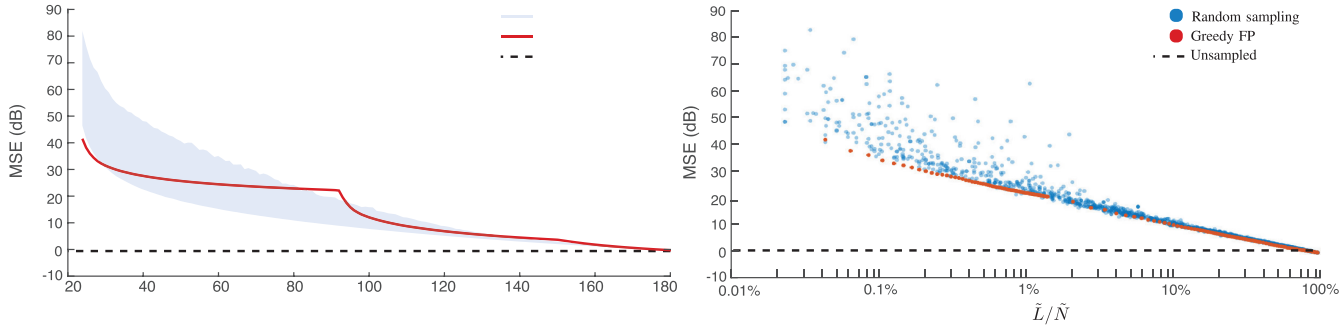
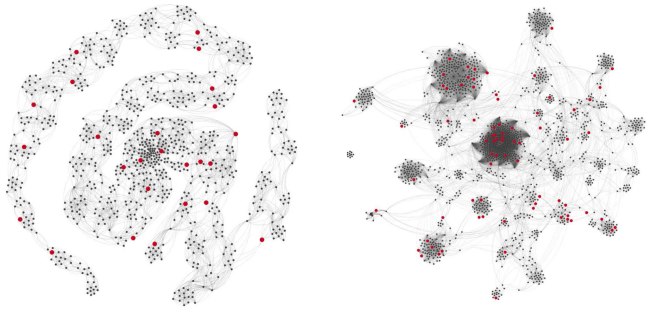


Fig. 5. Diagonal core with $R = 3$ with $N_1 = 50$, $N_2 = 60$, $N_3 = 70$, $K_c = 20$, $\beta_1 = \beta_2 = 1$, and $\beta_3 = 20$.



(a) User graph (b) Movie graph

Fig. 6. User and movie networks. The red (black) dots represent the observed (unobserved) vertices. Visualization obtained using Gephi [45].

B. Active Learning For Recommender Systems

Current recommendation algorithms seek solving an estimation problem of the form: given the past recorded preferences of a set of users, what is the *rating* that these would give to a set of products? In this paper, in contrast, we focus on the data acquisition phase of the recommender system, which is also referred to as active learning/sampling. In particular, we claim that by carefully designing which users to poll and on which items, we can obtain an estimation performance on par with the state-of-the-art methods, but using only a fraction of the data that current methods require, and using a simple least-squares estimator.

We showcase this idea on the MovieLens $100k$ dataset [40] that contains partial ratings of $N_1 = 943$ users over $N_2 = 1682$ movies which are stored in a second-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{N_1 \times N_2}$. At this point, we emphasize the need for our proposed framework, since it is obvious that designing an unstructured sampling set with about 1.5 million candidate locations is unfeasible with current computing resources.

A model of $\boldsymbol{\mathcal{X}}$ in the form of (1) can be obtained by viewing $\boldsymbol{\mathcal{X}}$ as a signal that lives on a graph. In particular, the first two modes of $\boldsymbol{\mathcal{X}}$ can be viewed as a signal defined on the Cartesian product of a user and movie graph, respectively. These two graphs, shown in Fig. 6, are provided in the dataset and are two 10-nearest-neighbors graphs created based on the user and movie features.

Based on the recent advances in graph signal processing (GSP) [41], [42], $\boldsymbol{\mathcal{X}}$ can be decomposed as $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_{\mathrm{f}} \bullet_1 \mathbf{V}_1 \bullet_2 \mathbf{V}_2$. Here, $\mathbf{V}_1 \in \mathbb{R}^{N_1 \times N_1}$ and $\mathbf{V}_2 \in \mathbb{R}^{N_2 \times N_2}$ are the eigenbases of the Laplacians of the user and movie graphs, respectively, and $\boldsymbol{\mathcal{X}}_{\mathrm{f}} \in \mathbf{C}^{N_1 \times N_2}$ is the so-called graph spectrum of $\boldsymbol{\mathcal{X}}$ [41], [42]. Suppose the energy of the spectrum of $\boldsymbol{\mathcal{X}}$ is concentrated in the first few K_1 and K_2 columns of $\mathbf{V}_1$ and $\mathbf{V}_2$, respectively, then $\boldsymbol{\mathcal{X}}$ admits a low-dimensional representation, or $\boldsymbol{\mathcal{X}}$ is said to be smooth or bandlimited with respect to the underlying graph [42]. This property has been exploited in [43], [44] to impute the missing entries in $\boldsymbol{\mathcal{X}}$. In contrast, we propose a scheme for sampling and reconstruction of signals defined on product graphs.

In our experiments, we set $K_1 = K_2 = 20$, and obtain the decomposition $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}} \bullet_1 \mathbf{U}_1 \bullet_2 \mathbf{U}_2$, where $\mathbf{U}_1 \in \mathbb{C}^{N_1 \times K_1}$ and $\mathbf{U}_2 \in \mathbb{C}^{N_2 \times K_1}$ consist of the first K_1 and K_2 columns of $\mathbf{V}_1$ and $\mathbf{V}_2$, respectively; and $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{X}}_{\mathrm{f}}(1 : K_1, 1 : K_2)$.

For the greedy algorithm we use $L = 100$ and $\alpha_1 = \alpha_2 = 5$, resulting in a selection of $L_1 = 25$ users and $L_2 = 75$ movies, i.e., a total of 1875 vertices in the product graph. Fig. 6, shows the sampled users and movies, i.e., users to be probed for movie ratings. The user graph [cf. Fig. 6a] is made out of small clusters connected in a chain-like structure, resulting in a uniformly spread distribution of observed vertices. On the other hand, the movies graph [cf. Fig. 6b] is made out of a few big and small clusters. Hence, the proposed active querying scheme assigns more observations to the bigger clusters and fewer observations to the smaller ones.

To evaluate the performance of our algorithm, we compute the RMSE of the estimated data using the test mask provided by the dataset. Nevertheless, since our active query method requires access to ground truth data (i.e., we need access to the samples at locations suggested by the greedy algorithm) which is not provided in the dataset, we use GRALS [4] to complete the matrix, and use its estimates when required. A comparison of our algorithm to the performance of the state-of-the-art methods run on the same dataset is shown in Table I. In light of these results, it is clear that a proper design of the sampling set allows to obtain top performance with significantly fewer ratings, i.e., about an order of magnitude, and using a much simpler non-iterative estimator.

C. Multiuser Source Separation

In multiple-input multiple-output (MIMO) communications [2], the use of rectangular arrays [48] allows to separate signals coming from different azimuth and elevation angles, and it is common that users transmit data using different spreading

TABLE I
PERFORMANCE ON MOVIELENS 100 K. BASELINE SCORES ARE TAKEN FROM [47]

Method	Number of samples	RMSE
GMC [43]	80,000	0.996
GRALS [4]	80,000	0.945
sRGCNN [46]	80,000	0.929
GC-MC [47]	80,000	**0.905**
Our method	**1,875**	0.9347

codes to reduce the interference from other sources. Reducing hardware complexity by minimizing the number of antennas and samples to be processed is an important concern in the design of MIMO receivers. This design can be seen as a particular instance of sparse tensor sampling.

We consider a scenario with K_c users located at different angles of azimuth (ϕ) and elevation (θ) transmitting using unique spreading sequences of length N_3. The receiver consists of a uniform rectangular array (URA) with antennas located on a $N_1 \times N_2$ grid. Each time instant, every antenna receives [48]

$$x(r,l,m,n) = \sum_{k=1}^{K_c} s_k(r)c_k(l)e^{j2\pi n\Delta_x \sin\theta_k} e^{j2\pi m\Delta_y \sin\phi_k} + w(r,l,m,n),$$

where $s_k(r)$ the symbol transmitted by user k in the rth symbol period; $c_k(l)$ the lth sample of the spreading sequence of the kth user; Δ_x and Δ_y the antenna separations in wavelengths of the URA in the x and y dimensions, respectively; and ϕ_k and θ_k the azimuth and elevation coordinates of user k, respectively; and where $w(r,l,m,n)$ represents an additive white Gaussian noise term with zero mean and variance σ^2. For the r-th symbol period, all these signals can be collected in a 3rd-order tensor $\boldsymbol{\mathcal{X}}(r) \in \mathbb{C}^{N_1\times N_2\times N_3}$ that can be decomposed as

$$\boldsymbol{\mathcal{X}}(r) = \boldsymbol{\mathcal{S}}(r) \bullet_1 \mathbf{U}_1 \bullet_2 \mathbf{U}_2 \bullet_3 \mathbf{U}_3 + \boldsymbol{\mathcal{W}}(r),$$

where $\mathbf{U}_1 \in \mathbb{C}^{N_1\times K_c}$ and $\mathbf{U}_2 \in \mathbb{C}^{N_2\times K_c}$ are the array responses for the x and y directions, respectively; $\mathbf{U}_3 \in \mathbb{C}^{N_3\times K_c}$ contains the spreading sequences of all users in its columns; and $\boldsymbol{\mathcal{S}}(r) \in \mathbb{C}^{K_c\times K_c\times K_c}$ is a diagonal tensor that stores the symbols of all users for the rth symbol period on its diagonal.

We simulate this setup using $K_c = 10$ users that transmit BPSK symbols with different random powers and that are equispaced in azimuth and elevation. We use a rectangular array with $N_1 = 50$ and $N_2 = 60$ for the ground set locations of the antennas, and binary random spreading sequences of length $N_3 = 100$. With these parameters, each $\boldsymbol{\mathcal{X}}(r)$ has 300,000 entries. We generate many realizations of these signals for different levels of signal-to-noise ratio (SNR) and sample the resulting tensors using the greedy algorithm for the diagonal core case with $L = 15$, resulting in a relative number of samples of 0.048%. The results are depicted in Fig. 7, where the blue shaded area represents the MSE obtained with the best and worst random samplers. As expected, the MSE of the reconstruction decreases exponentially with the SNR. For a given MSE, achieving maximum compression requires transmitting with a higher SNR of about 30 dB than the one needed for no compression. Besides, we see that our proposed greedy algorithm consistently performs as well as the best random sampling scheme.

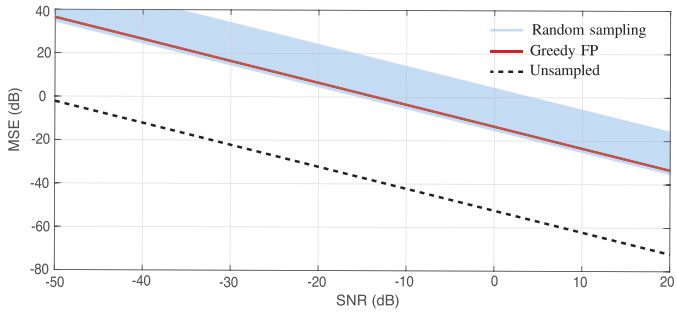


Fig. 7. MSE of symbol reconstruction. $N_1 = 50$, $N_2 = 60$, $N_3 = 100$, and $L = 15$.

VII. CONCLUSIONS

In this paper, we presented the design of sparse samplers for inverse problems with tensors. We have seen that by using samplers with a Kronecker structure we can overcome the curse of dimensionality, and design efficient subsampling schemes that guarantee a good performance for the reconstruction of multidomain tensor signals. We presented sparse sampling design methods for cases in which the multidomain signals can be decomposed using a multilinear model with a dense core or a diagonal core. For both cases, we have provided a near-optimal greedy algorithm based on submodular optimization methods to compute the sampling sets.

APPENDIX

A. Proof of Theorem 2

In order to simplify the derivations, let us introduce the notation $\bar{F}_i(\mathcal{S}_i) = F_i(\mathcal{N}_i \backslash \mathcal{S}_i)$, so that $G(\mathcal{S})$ can also be written

$$G(\mathcal{S}) := \prod_{i=1}^{R} F_i(\mathcal{N}_i) - \prod_{i=1}^{R} \bar{F}_i(\mathcal{S}_i). \tag{27}$$

From (27) it is evident that $G(\varnothing) = 0$. Thus, proving that G is normalized. To prove monotonicity, recall that the single domain frame potential terms $F_i(\mathcal{L}_i)$ are all non-negative, monotone non-decreasing functions for all $\mathcal{L}_i \subseteq \mathcal{N}_i$ [10]. Therefore, $\bar{F}_i(\mathcal{S}_i) = F_i(\mathcal{N} \setminus \mathcal{S}_i)$ will be non-negative, but monotone non-increasing. Let $\mathcal{S} \subseteq \mathcal{N}$ and $x \in \mathcal{N} \setminus \mathcal{S}$. Without loss of generality, let us assume $x \in \mathcal{N}_i$. Then, we have

$$G(\mathcal{S} \cup \{x\}) = \prod_{i=1}^{R} F_i(\mathcal{N}_i) - \bar{F}_i(\mathcal{S}_i \cup \{x\}) \prod_{j\neq i} \bar{F}_j(\mathcal{S}_j),$$

$$G(\mathcal{S}) = \prod_{i=1}^{R} F_i(\mathcal{N}_i) - \bar{F}_i(\mathcal{S}_i) \prod_{j\neq i} \bar{F}_j(\mathcal{S}_j).$$

Now, since $\bar{F}_i(\mathcal{S}_i) \geq \bar{F}_i(\mathcal{S}_i \cup \{x\})$, we know that $G(\mathcal{S} \cup \{x\}) \geq G(\mathcal{S})$. Hence, $G(\mathcal{S})$ is monotone non-decreasing.

To prove submodularity, recall that every $F_i(\mathcal{L}_i)$ is supermodular [10]. As taking the complement preserves (super)submodularity, $\bar{F}_i(\mathcal{L}_i) = F_i(\mathcal{N}_i \setminus \mathcal{L}_i)$ is also supermodular. Let $\mathcal{S} = \bigcup_{i=1}^{R} \mathcal{A}_i$, with $\mathcal{A}_i \subseteq \mathcal{N}_i$ for $i = 1, \ldots, R$, such that $\{\mathcal{A}_i\}_{i=1}^{R}$ forms a partition of $\mathcal{S}$. Now, recall from Definition 1 that for G to be submodular we require that $\forall x, y \in \mathcal{N} \setminus \mathcal{S}$

$$G(\mathcal{S} \cup \{x\}) - G(\mathcal{S}) \geq G(\mathcal{S} \cup \{x, y\}) - G(\mathcal{S} \cup \{y\}). \quad (28)$$

As the ground set is now partitioned into the union of several ground sets, there are two possible ways the elements x and y can be selected. Either they both belong to the same domain, or they belong to different domains. We next prove that (28) is satisfied for the aforementioned both cases.

Suppose $x, y \in \mathcal{N}_i$, then (28) can be developed as

$$\begin{aligned}\bar{F}_i(\mathcal{A}_i) \prod_{j \neq i} \bar{F}_j(\mathcal{A}_j) - \bar{F}_i(\mathcal{A}_i \cup \{x\}) \prod_{j \neq i} \bar{F}_j(\mathcal{A}_j) \\ \geq \bar{F}_i(\mathcal{A}_i \cup \{y\}) \prod_{j \neq i} \bar{F}_j(\mathcal{A}_j) - \bar{F}_i(\mathcal{A}_i \cup \{i, j\}) \prod_{j \neq i} \bar{F}_j(\mathcal{A}_j),\end{aligned}$$

which can be further simplified to

$$\bar{F}_i(\mathcal{A}_i \cup \{x\}) - \bar{F}_i(\mathcal{A}_i) \leq \bar{F}_i(\mathcal{A}_i \cup \{x, y\}) - \bar{F}_i(\mathcal{A}_i \cup \{y\}).$$

The above inequality is true since $\bar{F}_i$ is supermodular.

Next, suppose $x \in \mathcal{N}_i$ and $y \in \mathcal{N}_j$ with $i \neq j$, then (28) can be expanded as

$$\begin{aligned}\prod_{k \neq i,j} \bar{F}_k(\mathcal{A}_k) \left[\bar{F}_i(\mathcal{A}_i)\bar{F}_j(\mathcal{A}_j) - \bar{F}_i(\mathcal{A}_i \cup \{x\})\bar{F}_j(\mathcal{A}_j)\right] \\ \geq \prod_{k \neq i,j} \bar{F}_k(\mathcal{A}_k) \left[\bar{F}_i(\mathcal{A}_i)\bar{F}_j(\mathcal{A}_j \cup \{y\}) \right. \\ \left. -\bar{F}_i(\mathcal{A}_i \cup \{x\})\bar{F}_j(\mathcal{A}_j \cup \{y\})\right].\end{aligned}$$

Extracting the common factors

$$\left[\bar{F}_i(\mathcal{A}_i) - \bar{F}_i(\mathcal{A}_i \cup \{x\})\right]\left[\bar{F}_j(\mathcal{A}_j) - \bar{F}_j(\mathcal{A}_j \cup \{y\})\right] \geq 0. \quad (29)$$

Since $\bar{F}_i$ and $\bar{F}_j$ are non-increasing

$$\bar{F}_i(\mathcal{A}_i) - \bar{F}_i(\mathcal{A}_i \cup \{x\}) \geq 0; \quad \bar{F}_j(\mathcal{A}_j) - \bar{F}_j(\mathcal{A} \cup \{y\}) \geq 0.$$

Thus, (29) is always satisfied, thus proving that (28) is satisfied for any $\mathcal{S} \subseteq \mathcal{N}$ and $x, y \in \mathcal{N} \setminus \mathcal{S}$ and therefore G is submodular.

B. Proof of Theorem 5

We divide the proof in two parts. First, we derive some properties of the involved operations that are useful to simplify the proof. Then, we use this to derive the proof.

1) Preliminaries: First, note that the single-domain Grammian matrices satisfy the following lemma.

Lemma 1 (Grammian of disjoint union): Let $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{N}_i$ with $\mathcal{X} \cap \mathcal{Y} = \varnothing$. Then, the Grammian of $\mathcal{X} \cup \mathcal{Y}$ satisfies

$$\mathbf{T}_i(\mathcal{X} \cup \mathcal{Y}) = \mathbf{T}_i(\mathcal{X}) + \mathbf{T}_i(\mathcal{Y}).$$

Proof: Let $\mathbf{u}_{i,j}$ denote the jth row of $\mathbf{T}_i$. Then,

$$\mathbf{T}_i(\mathcal{X} \cup \mathcal{Y}) = \sum_{j \in \mathcal{X} \cup \mathcal{Y}} \|\mathbf{u}_{i,j}\|_2^2 = \sum_{j \in \mathcal{X}} \|\mathbf{u}_{i,j}\|_2^2 + \sum_{j \in \mathcal{Y}} \|\mathbf{u}_{i,j}\|_2^2.$$

■

Let us introduce the complement Grammian matrix

$$\bar{\mathbf{T}}_i(\mathcal{S}_i) := \mathbf{T}_i(\mathcal{N}_i \backslash \mathcal{S}_i) = \mathbf{T}_i(\mathcal{N}_i) - \mathbf{T}_i(\mathcal{S}_i), \quad (30)$$

which satisfies the following lemma.

Lemma 2 (Complement Grammian of disjoint union): Let $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{N}_i$ with $\mathcal{X} \cap \mathcal{Y} = \varnothing$. Then, $\bar{\mathbf{T}}_i(\mathcal{X} \cup \mathcal{Y}) = \bar{\mathbf{T}}_i(\mathcal{X}) - \mathbf{T}_i(\mathcal{Y})$.

Proof: From (30) and Lemma 1, we have

$$\bar{\mathbf{T}}_i(\mathcal{X} \cup \mathcal{Y}) = \mathbf{T}_i(\mathcal{N}_i) - [\mathbf{T}_i(\mathcal{X}) + \mathbf{T}_i(\mathcal{Y})] = \bar{\mathbf{T}}_i(\mathcal{X}) - \mathbf{T}_i(\mathcal{Y}).$$

■

Now, let us introduce an operator to compress the writing of the multidomain Hadamard product

$$\mathbb{T}(\mathcal{L}) := \mathbf{T}_1(\mathcal{L}_1) \circ \cdots \circ \mathbf{T}_R(\mathcal{L}_R),$$

or alternatively for the complement Grammian

$$\bar{\mathbb{T}}(\mathcal{S}) := \bar{\mathbf{T}}_1(\mathcal{S}_1) \circ \cdots \circ \bar{\mathbf{T}}_R(\mathcal{S}_R).$$

Furthermore, we write the Hadamard multiplication of all $\mathbf{T}_i$ with $i = 1, \ldots, R$, but j as

$$\begin{aligned}\mathbb{T}_{-j}(\mathcal{L}) := \mathbf{T}_1(\mathcal{L}_1) \circ \cdots \circ \mathbf{T}_{j-1}(\mathcal{L}_{j-1}) \\ \circ \mathbf{T}_{j+1}(\mathcal{L}_{j+1}) \circ \cdots \circ \mathbf{T}_R(\mathcal{L}_R).\end{aligned}$$

Similarly, for the complement Grammians, we will use $\bar{\mathbb{T}}_{-j}(\mathcal{S})$. We also make use of the following properties of the Hadamard product.

Property 1: The Hadamard product of two positive semidefinite matrices is always positive semidefinite.

Property 2: Let $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{N \times N}$. Then,

$$\|\mathbf{A} \circ \mathbf{B}\|_F^2 = \operatorname{tr}\left\{\mathbf{A}^{\circ 2}\left(\mathbf{B}^{\circ 2}\right)^T\right\} = \left\langle \mathbf{A}^{\circ 2}, \mathbf{B}^{\circ 2}\right\rangle,$$

where $\mathbf{A}^{\circ n}$ denotes the element-wise nth power of $\mathbf{A}$.

Let us introduce the notation

$$\mathbf{H}_i(\mathcal{S}) := \mathbf{T}_i^{\circ 2}(\mathcal{S}) \quad \text{and} \quad \bar{\mathbf{H}}_i(\mathcal{S}) := \bar{\mathbf{T}}_i^{\circ 2}(\mathcal{S}), \quad (31)$$

which satisfies the following lemma.

Lemma 3: Let $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{N}_i$ with $\mathcal{X} \cap \mathcal{Y} = \varnothing$. Then,

$$\begin{aligned}\mathbf{H}_i(\mathcal{X} \cup \mathcal{Y}) &= \mathbf{T}_i^{\circ 2}(\mathcal{X} \cup \mathcal{Y}) = (\mathbf{T}_i(\mathcal{X}) + \mathbf{T}_i(\mathcal{Y}))^{\circ 2} \\ &= \mathbf{H}_i(\mathcal{X}) + \mathbf{H}_i(\mathcal{Y}) + 2\mathbf{T}_i(\mathcal{X}) \circ \mathbf{T}_i(\mathcal{Y}).\end{aligned}$$

and

$$\begin{aligned}\bar{\mathbf{H}}_i(\mathcal{X} \cup \mathcal{Y}) &= \bar{\mathbf{T}}_i^{\circ 2}(\mathcal{X} \cup \mathcal{Y}) = \left(\bar{\mathbf{T}}_i(\mathcal{X}) - \mathbf{T}_i(\mathcal{Y})\right)^{\circ 2} \\ &= \bar{\mathbf{H}}_i(\mathcal{X}) + \mathbf{H}_i(\mathcal{Y}) - 2\bar{\mathbf{T}}_i(\mathcal{X}) \circ \mathbf{T}_i(\mathcal{Y}).\end{aligned}$$

Moreover, as we did with the Grammian matrices, we introduce the notation

$$\mathbb{H}(\mathcal{L}) := \mathbf{H}_1(\mathcal{L}_1) \circ \cdots \circ \mathbf{H}_R(\mathcal{L}_R),$$

and

$$\mathbb{H}_{-j}(\mathcal{L}) := \mathbf{H}_1(\mathcal{L}_1) \circ \cdots \circ \mathbf{H}_{j-1}(\mathcal{L}_{j-1}) \circ \mathbf{H}_{j+1}(\mathcal{L}_{j+1}) \circ \cdots \circ \mathbf{H}_R(\mathcal{L}_R),$$

with its analogue $\bar{\mathbb{H}}$, and $\bar{\mathbb{H}}_{-j}$. Due to Property 1, all these matrices are also positive semidefinite.

Finally, note that with the new notation we can simplify the definition of Q to

$$Q(\mathcal{S}) := \|\mathbb{T}(\mathcal{N})\|_F^2 - \left\|\bar{\mathbb{T}}(\mathcal{S})\right\|_F^2. \tag{32}$$

2) Derivation: Normalization is derived from the fact that $\bar{\mathbf{T}}_i(\varnothing) = \mathbf{T}_i(\mathcal{N})$. To prove monotonicity, let $\mathcal{S} \subseteq \mathcal{N}$ and $x \in \mathcal{N} \setminus \mathcal{S}$. Without loss of generality, assume $x \in \mathcal{N}_i$. We have

$$Q(\mathcal{S} \cup \{x\}) = \|\mathbb{T}(\mathcal{N})\|_F^2 - \left\|\bar{\mathbf{T}}_i(\mathcal{S}_i \cup \{x\}) \circ \bar{\mathbb{T}}_{-i}(\mathcal{S})\right\|_F^2,$$

$$Q(\mathcal{S}) = \|\mathbb{T}(\mathcal{N})\|_F^2 - \left\|\bar{\mathbf{T}}_i(\mathcal{S}_i) \circ \bar{\mathbb{T}}_{-i}(\mathcal{S})\right\|_F^2.$$

Monotonicity requires that $Q(\mathcal{S}) \leq Q(\mathcal{S} \cup \{x\})$, or

$$-\left\|\bar{\mathbf{T}}_i(\mathcal{S}_i) \circ \bar{\mathbb{T}}_{-i}(\mathcal{S})\right\|_F^2 \leq -\left\|\bar{\mathbf{T}}_i(\mathcal{S}_i \cup \{x\}) \circ \bar{\mathbb{T}}_{-i}(\mathcal{S})\right\|_F^2.$$

Using Property 2, we have

$$\left\langle \bar{\mathbf{T}}_i(\mathcal{S}_i), \bar{\mathbb{T}}_{-i}(\mathcal{S}) \right\rangle \geq \left\langle \bar{\mathbf{T}}_i(\mathcal{S}_i \cup \{x\}), \bar{\mathbb{T}}_{-i}(\mathcal{S}) \right\rangle.$$

Expanding the unions using Lemma 2, and due to the linearity of the inner product this becomes

$$0 \leq \left\langle \mathbf{T}_i(\mathcal{S}_i \cup \{x\}), \bar{\mathbb{T}}_{-i}(\mathcal{S}) \right\rangle,$$

which is always satisfied because the inner product between two positive semidefinite matrices is always greater or equal than zero.

To prove submodularity, let $\mathcal{S} = \bigcup_{i=1}^R \mathcal{A}_i$, with $\mathcal{A}_i \subseteq \mathcal{N}_i$ for $i = 1, \ldots, R$ such that $\{\mathcal{A}_i\}_{i=1}^R$, forms a partition of $\mathcal{S}$. For Q to be submodular we require that $\forall x, y \in \mathcal{N} \setminus \mathcal{S}$

$$Q(\mathcal{S} \cup \{x\}) - Q(\mathcal{S}) \geq Q(\mathcal{S} \cup \{x, y\}) - Q(\mathcal{S} \cup \{y\}). \tag{33}$$

As before, we have two different cases. Suppose $x, y \in \mathcal{N}_i$, then (33) can be developed as

$$\begin{aligned}\left\|\bar{\mathbb{T}}(\mathcal{A})\right\|_F^2 &- \left\|\bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{x\}) \circ \bar{\mathbb{T}}_{-i}(\mathcal{A})\right\|_F^2 \\ &\geq \left\|\bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{y\}) \circ \bar{\mathbb{T}}_{-i}(\mathcal{A})\right\|_F^2 \\ &\quad - \left\|\bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{x, y\}) \circ \bar{\mathbb{T}}_{-i}(\mathcal{A})\right\|_F^2.\end{aligned}$$

Rewriting this expression using Property 2, we can express the left hand side as

$$\left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle - \left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle,$$

and the right hand side as

$$\left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{y\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle - \left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x, y\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle.$$

Leveraging the linearity of the inner product we arrive at

$$\begin{aligned}&\langle \bar{\mathbf{H}}_i(\mathcal{A}_i) - \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \rangle \\ &\quad \geq \left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{y\}) - \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x, y\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle.\end{aligned} \tag{34}$$

Developing the matrices using Lemma 3, we can operate on both sides of this expression giving, for the left hand side

$$\left\langle -\mathbf{H}_i(\{x\}) + 2\bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle,$$

and for the right hand side

$$\left\langle -\mathbf{H}_i(\{x\}) + 2\bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{y\}) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle.$$

Substituting in (34), we get

$$\begin{aligned}&\left\langle \bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle \\ &\quad \geq \left\langle \bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{y\}) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle,\end{aligned}$$

and using Lemma 2 we finally arrive at

$$\left\langle \mathbf{T}_i(\{y\}) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle \geq 0, \tag{35}$$

which is always satisfied because the inner product of positive semidefinite matrices is always non-negative.

Next, suppose $x \in \mathcal{N}_i$ and $y \in \mathcal{N}_j$ with $i \neq j$, then (33) can be rewritten as

$$\begin{aligned}&\left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i) - \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle \\ &\geq \left\langle \bar{\mathbf{H}}_i(\mathcal{A}_i) - \bar{\mathbf{H}}_i(\mathcal{A}_i \cup \{x\}), \bar{\mathbf{H}}_j(\mathcal{A}_j \cup \{y\}) \circ \bar{\mathbb{H}}_{-(i,j)}(\mathcal{A}) \right\rangle.\end{aligned}$$

Using Lemma 3, we can further develop this expression into

$$\begin{aligned}&\left\langle -\mathbf{H}_i(\{x\}) + 2\bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}), \bar{\mathbb{H}}_{-i}(\mathcal{A}) \right\rangle \\ &\geq \left\langle -\mathbf{H}_i(\{x\}) + 2\bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}), \right. \\ &\quad \left. \bar{\mathbf{H}}_j(\mathcal{A}_j \cup \{y\}) \circ \bar{\mathbb{H}}_{-(i,j)}(\mathcal{A}) \right\rangle.\end{aligned}$$

Leveraging the linearity of the inner product this can be simplified as

$$\begin{aligned}&\left\langle -\mathbf{H}_i(\{x\}) + 2\bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}), \right. \\ &\quad \left. \bar{\mathbb{H}}_{-i}(\mathcal{A}) - \bar{\mathbf{H}}_j(\mathcal{A}_j \cup \{y\}) \circ \bar{\mathbb{H}}_{-(i,j)}(\mathcal{A}) \right\rangle \geq 0.\end{aligned} \tag{36}$$

Here, we can factorize the left entry of the inner product as

$$\begin{aligned}-\mathbf{H}_i(\{x\}) &+ 2\bar{\mathbf{T}}_i(\mathcal{A}_i) \circ \mathbf{T}_i(\{x\}) \\ &= \mathbf{T}_i(\{x\}) \circ \left[2\bar{\mathbf{T}}_i(\mathcal{A}_i) - \mathbf{T}_i(\{x\})\right] \\ &= \mathbf{T}_i(\{x\}) \circ \left[\bar{\mathbf{T}}_i(\mathcal{A}_i) + \bar{\mathbf{T}}_i(\mathcal{A}_i \cup \{x\})\right],\end{aligned} \tag{37}$$

which is positive semidefinite due to Property 1, and the fact that the set of positive semidefinite matrices is closed under matrix addition.

Similarly, the right entry of the inner product in (36) can be factorized as

$$\begin{aligned}\bar{\mathbb{H}}_{-i}(\mathcal{A}) &- \left(\bar{\mathbf{H}}_j(\mathcal{A}_j) + \mathbf{H}_j(\{y\}) - 2\bar{\mathbf{T}}_j(\mathcal{A}_j) \circ \mathbf{T}_j(\{y\})\right) \\ &\quad \circ \bar{\mathbb{H}}_{-(i,j)}(\mathcal{A}) \\ &= \left(-\mathbf{H}_j(\{y\}) + 2\bar{\mathbf{T}}_j(\mathcal{A}_j) \circ \mathbf{T}_j(\{y\})\right) \circ \bar{\mathbb{H}}_{-(i,j)}(\mathcal{A}).\end{aligned}$$

The expression inside the parenthesis is analagous to that in (37). Hence, the resulting matrix is positive semidefinite, and thus (36) is always satisfied, proving submodularity of Q for all cases.

REFERENCES

[1] G. Ortiz-Jiménez, M. Coutino, S. P. Chepuri, and G. Leus, "Sampling and Reconstruction of signals on product graphs," in *Proc. IEEE Global Conf. Signal, Inf. Process.*, Anaheim, CA, USA, Nov. 2018, pp. 713–717.

[2] F. Rusek *et al.*, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[3] P. Ghamisi *et al.*, "Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, Dec. 2017.

[4] N. Rao, H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon, "Collaborative filtering with graph information: Consistency and scalable methods," in *Proc. Neural Inf. Process. Syst.*, Montreal, QC, Canada, Dec. 2015, pp. 2107–2115.

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

[6] S. P. Chepuri and G. Leus, "Sparse sensing for statistical inference," *Foundations Trends Signal Process.*, vol. 9, no. 3/4, pp. 233–368, 2016.

[7] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, Feb. 2009.

[8] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, pp. 235–284, Feb. 2008.

[9] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *Proc. 49th IEEE Conf. Decis. Control*, Atlanta, GA, USA, Dec. 2010, pp. 2572–2577.

[10] J. Rainieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1135–1146, Mar. 2014.

[11] C. Jiang, Y. C. Soh, and H. Li, "Sensor placement by maximal projection on minimum eigenspace for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5595–5610, Nov. 2016.

[12] C.-T. Yu and P. K. Varshney, "Sampling design for Gaussian detection problems," *IEEE Trans. Signal Process.*, vol. 45, no. 9, pp. 2328–2337, Sep. 1997.

[13] E. Masazade, M. Fardad, and P. K. Varshney, "Sparsity-promoting extended Kalman filtering for target tracking in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 845–848, Dec. 2012.

[14] S. P. Chepuri and G. Leus, "Sparsity-promoting sensor selection for nonlinear measurement models," *IEEE Trans. Signal Process.*, vol. 63, no. 3, pp. 684–698, Feb. 2015.

[15] S. P. Chepuri and G. Leus, "Sparse sensing for distributed detection," *IEEE Trans. Signal Process.*, vol. 64, no. 6, pp. 1446–1460, Mar. 2016.

[16] S. P. Chepuri and G. Leus, "Sparsity-promoting adaptive sensor selection for non-linear filtering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Florence, Italy, May 2014, pp. 5080–5084.

[17] S. Liu and G. Trenkler, "Hadamard, khatri-rao, kronecker and other matrix products," *Int. J. Inf. Syst. Sci.*, vol. 4, no. 1, pp. 160–177, 2008.

[18] D. Romero, D. D. Ariananda, Z. Tian, and G. Leus, "Compressive covariance sensing: Structure-based compressive sensing beyond sparsity," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 78–93, Jan. 2016.

[19] S. P. Chepuri and G. Leus, "Graph sampling for covariance estimation," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 3, pp. 451–466, Sep. 2017.

[20] E. Tohidi, M. Coutino, S. P. Chepuri, H. Behroozi, M. M. Nayebi, and G. Leus, "Sparse antenna and pulse placement for colocated MIMO radar," *IEEE Trans. Signal Process.*, vol. 67, no. 3, pp. 579–593, Feb. 2019.

[21] R. K. Iyer and J. A. Bilmes, "Submodular optimization with submodular cover and submodular knapsack constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, Montreal, QC, Canada, Dec. 2013, pp. 2436–2444.

[22] M. Sviridenko, "A note on maximizing a submodular set function subject to a knapsack constraint," *Oper. Res. Lett.*, vol. 32, no. 1, pp. 41–43, Jan. 2004.

[23] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions–I," *Math. Program.*, vol. 14, no. 1, pp. 265–294, Dec. 1978.

[24] M. Coutino, S. P. Chepuri, and G. Leus, "Submodular sparse sensing for gaussian detection with correlated observations," *IEEE Trans. Signal Process.*, vol. 66, no. 15, pp. 4025–4039, Aug. 2018.

[25] V. Tzoumas, "Resilient submodular maximization for control and sensing," Ph.D. dissertation, Dept. Elect. Syst. Eng., Univ. Pennsylvania, Philadelphia, PA, USA, 2018.

[26] C. Rusu, J. Thompson, and N. M. Robertson, "Sensor scheduling with time, energy, and communication constraints," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 528–539, Jan. 2018.

[27] S. Rao, S. P. Chepuri, and G. Leus, "Greedy sensor selection for non-linear models," in *Proc. IEEE 6th Int. Workshop Comput. Adv. Multi Sensor Adaptive Process.*, Dec. 2015, pp. 241–244.

[28] Z. Chen, J. Ranieri, R. Zhang, and M. Vetterli, "DASS: Distributed adaptive sparse sensing," vol. 14, no. 5, pp. 2571–2583, May 2015.

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

[30] C. F. Caiafa and A. Cichocki, "Computing sparse representations of multidimensional signals using kronecker bases," *Neural Comput.*, vol. 25, no. 1, pp. 186–220, Jan. 2013.

[31] C. Caiafa and A. Cichocki, "Multidimensional compressed sensing and their applications," *Wiley Interdisciplinary Rev., Data Mining Knowl. Discovery*, vol. 3, no. 6, pp. 355–380, Oct. 2013.

[32] Y. Yu, J. Jin, F. Liu, and S. Crozier, "Multidimensional compressed sensing MRI using tensor decomposition-based sparsifying transform," *PLoS ONE*, vol. 9, no. 6, Jun. 2014, Art. no. e98441.

[33] S. Fujishige, *Submodular Functions and Optimization* (Annals of Discrete Mathematics, 58). New York, NY, USA: Elsevier, 2005.

[34] R. G. Parker and R. L. Rardin, "Polynomial algorithms–matroids," in *Discrete Optimization* (Computer Science and Scientific Computing), R. G. Parker and R. L. Rardin, Eds. San Diego, CA, USA: Academic, 1988, pp. 57–106.

[35] M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey, "An analysis of approximations for maximizing submodular set functions–II," in *Polyhedral Combinatorics*. Berlin, Germany: Springer, Dec. 1978, pp. 73–87.

[36] M. Fickus, D. G. Mixon, and M. J. Poteet, "Frame completions for optimally robust reconstruction," in *Wavelets and Sparsity XIV*, vol. 8138. Bellingham, WA, USA: Int. Soc. Opt. Eng., Sep, p. 81380Q.

[37] G. Ortiz-Jiménez, "Multidomain graph signal processing: Learning and sampling," Master's thesis, Dept. Microelectronics, Delft University of Technology, Delft, The Netherlands, Aug. 2018.

[38] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of n-way arrays," *J. Chemometrics*, vol. 14, no. 3, pp. 229–239, Jun. 2000.

[39] W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. 2, 2nd ed. Hoboken, NJ, USA: Wiley, 1971.

[40] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, Jan. 2016.

[41] A. Sandryhaila and J. M. F. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 80–90, Sep. 2014.

[42] A. Ortega, P. Frossard, J. Kovačević, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, no. 5, pp. 808–828, May 2018.

[43] V. Kalofolias, X. Bresson, M. Bronstein, and P. Vandergheynst, "Matrix completion on graphs," in *Proc. Neural Inf. Process. Syst., Workshop "Out Box, Robustness High Dimension,"* Montreal, QC, Canada, Dec. 2014.

[44] W. Huang, A. G. Marques, and A. Ribeiro, "Collaborative filtering via graph signal processing," in *Proc. Eur. Signal Process. Conf.*, Kos, Greece, Aug. 2017, pp. 1094–1098.

[45] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," in *Proc. Int. AAAI Conf. Weblogs Social Media*, San Jose, CA, USA, May 2009, pp. 361–362.

[46] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Montreal, QC, Canada, Dec. 2017, pp. 3700–3710.

[47] R. van den Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," in *Proc. ACM SIGKDD Conf. Knowledge Discovery Data Mining "Deep Learning Day"*, London, U.K., Aug. 2018, pp. 1–7.

[48] P. Larsson, "Lattice array receiver and sender for spatially orthonormal MIMO communication," in *Proc. IEEE Veh. Technol. Conf.*, Stockholm, Sweden, May 2005, vol. 1, pp. 192–196.

Guillermo Ortiz-Jiménez (S'18) received the B.Sc. degree (*Valedictorian*) in telecommunications engineering from Universidad Politécnica de Madrid (UPM), Madrid, Spain, in June 2015, and the M.Sc. degree (*Best Graduate, cum laude*) in electrical engineering from Delft University of Technology (TU Delft), Delft, The Netherlands, in August 2018. He is currently working toward the Ph.D. degree at cole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. He has held positions at the Microwaves and Radar Group of UPM during 2015–2016, and at Philips Healthcare Research, Hamburg, Germany, during summer 2017. His research interests include machine learning, signal processing and network theory. He is the recipient of the National Award for Excellence in Academic Performance from the Ministry of Education of Spain, a "la Caixa" Postgraduate Fellowship, and the Best Graduate award both at UPM and TU Delft.

Mario Coutino (S'17) received the M.Sc. degree (*cum laude*) in electrical engineering in July 2016 from Delft University of Technology, Delft, The Netherlands, where has been working toward the Ph.D. degree in signal processing since October 2016. He has held positions at Thales Nederlands, during summer 2015, and at Bang and Olufsen during 2015–2016. His research interests include signal processing on networks, submodular and convex optimization, and numerical lineal algebra. He was the recipient of the Best Student Paper Award for his publication at the CAMSAP 2017 conference in Curacao.

Sundeep Prabhakar Chepuri (M'16) received the M.Sc. degree (*cum laude*) in electrical engineering and the Ph.D. degree (*cum laude*) from Delft University of Technology, Delft, The Netherlands, in July 2011 and January 2016, respectively. He was a Postdoctoral Researcher (September 2015 to December 2018) with the Delft University of Technology, a Visiting Researcher with the University of Minnesota, USA, and a Visiting Lecturer with the Aalto University, Espoo, Finland. He has held positions at Robert Bosch, India, during 2007–2009, Holst Centre/imec-nl, The Netherlands, during 2010–2011. He is currently an Assistant Professor with the Electrical Communication Engineering Department, Indian Institute of Science, Bengaluru, India. His general research interests include mathematical signal processing, learning, and statistical inference applied to computational imaging, communication systems, and network sciences. He was the recipient of the Best Student Paper Award for his publication at the ICASSP 2015 conference in Australia. He is currently an Associate Editor of the *EURASIP Journal on Advances in Signal Processing* (JASP), and a member of the *EURASIP Signal Processing for Multisensor Systems* Special Area Team.

Geert Leus (F'12) received the M.Sc. and Ph.D. degrees in electrical engineering from KU Leuven, Leuven, Belgium, in June 1996 and May 2000, respectively. He is currently an "Antoni van Leeuwenhoek" Full Professor with the Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands. His research interests are in the broad area of signal processing, with a specific focus on wireless communications, array processing, sensor networks, and graph signal processing. He was the recipient of the 2002 IEEE Signal Processing Society Young Author Best Paper Award and the 2005 IEEE Signal Processing Society Best Paper Award. He is a Fellow of *EURASIP*. He was a Member-at-Large of the Board of Governors of the IEEE Signal Processing Society, the Chair of the IEEE Signal Processing for Communications and Networking Technical Committee, a Member of the IEEE Sensor Array and Multichannel Technical Committee, and the Editor in Chief of the *EURASIP Journal on Advances in Signal Processing*. He was also on the Editorial Boards of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE SIGNAL PROCESSING LETTERS, and the *EURASIP Journal on Advances in Signal Processing*. He is currently the Chair of the EURASIP Special Area Team on *Signal Processing for Multisensor Systems,* a Member of the IEEE Signal Processing Theory and Methods Technical Committee, a Member of the IEEE Big Data Special Interest Group, an Associate Editor for Foundations and Trends in Signal Processing, and the Editor in Chief of *EURASIP Signal Processing*.