Signal Processing 169 (2020) 107404

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Rethinking sketching as sampling: A graph signal processing approach☆,☆☆

Fernando Gama[a,*], Antonio G. Marques[b], Gonzalo Mateos[c], Alejandro Ribeiro[a]

[a] Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, USA
[b] Department of Signal Theory and Comms., King Juan Carlos University, Madrid, Spain
[c] Department of Electrical and Computer Engineering, University of Rochester, Rochester, USA

ARTICLE INFO

Article history:
Received 21 January 2019
Revised 25 November 2019
Accepted 26 November 2019
Available online 2 December 2019

Keywords:
Sketching
Sampling
Streaming
Linear transforms
Linear inverse problems
Graph signal processing

ABSTRACT

Sampling of signals belonging to a low-dimensional subspace has well-documented merits for dimensionality reduction, limited memory storage, and online processing of streaming network data. When the subspace is known, these signals can be modeled as bandlimited graph signals. Most existing sampling methods are designed to minimize the error incurred when reconstructing the original signal from its samples. Oftentimes these parsimonious signals serve as inputs to computationally-intensive linear operators. Hence, interest shifts from reconstructing the signal itself towards approximating the output of the prescribed linear operator efficiently. In this context, we propose a novel sampling scheme that leverages graph signal processing, exploiting the low-dimensional (bandlimited) structure of the input as well as the transformation whose output we wish to approximate. We formulate problems to jointly optimize sample selection and a sketch of the target linear transformation, so when the latter is applied to the sampled input signal the result is close to the desired output. Similar sketching as sampling ideas are also shown effective in the context of linear inverse problems. Because these designs are carried out off line, the resulting sampling plus reduced-complexity processing pipeline is particularly useful for data that are acquired or processed in a sequential fashion, where the linear operator has to be applied fast and repeatedly to successive inputs or response signals. Numerical tests showing the effectiveness of the proposed algorithms include classification of handwritten digits from as few as 20 out of 784 pixels in the input images and selection of sensors from a network deployed to carry out a distributed parameter estimation task.

1. Introduction

The complexity of modern datasets calls for new tools capable of analyzing and processing signals supported on irregular domains. A key principle to achieve this goal is to explicitly account for the intrinsic structure of the domain where the data resides. This is oftentimes achieved through a parsimonious description of the data, for instance modeling them as belonging to a known (or otherwise learnt) lower-dimensional subspace. Moreover, these data can be further processed through a linear transform to estimate a quantity of interest [3,4], or to obtain an alternative, more useful representation [5,6] among other tasks [7,8]. Linear models are ubiquitous in science and engineering, due in part to their generality, conceptual simplicity, and mathematical tractability. Along with heterogeneity and lack of regularity, data are increasingly high dimensional and this curse of dimensionality not only raises statistical challenges, but also major computational hurdles even for linear models [9]. In particular, these limiting factors can hinder processing of streaming data, where say a massive linear operator has to be repeatedly and efficiently applied to a sequence of input signals [10]. These Big Data challenges motivated a recent body of work collectively addressing so-termed *sketching* problems [11–13], which seek computationally-efficient solutions to a subset of (typically inverse) linear problems. The basic idea is to *draw a sketch* of the linear model such that the resulting linear transform is lower dimensional, while still offering quantifiable approximation error guarantees. To this end, a fat random projection matrix is designed

☆ Work in this paper is supported by NSF CCF 1750428, NSF ECCS 1809356, NSF CCF 1717120, ARO W911NF1710438 and Spanish MINECO grants No TEC2013-41604-R and TEC2016-75361-R.
☆☆ Part of the results in this paper were presented at the *2016 Asilomar Conference of Signals, Systems and Computers* [1] and the *2016 IEEE GobalSIP Conference* [2].
* Corresponding author.
E-mail addresses: fgama@seas.upenn.edu (F. Gama), antonio.garcia.marques@urjc.es (A.G. Marques), gmateosb@ece.rochester.edu (G. Mateos), aribeiro@seas.upenn.edu (A. Ribeiro).

https://doi.org/10.1016/j.sigpro.2019.107404
0165-1684/

to pre-multiply and reduce the dimensionality of the linear operator matrix, in such way that the resulting matrix sketch still captures the quintessential structure of the model. The input vector has to be adapted to the sketched operator as well, and to that end the same random projections are applied to the signal in a way often agnostic to the input statistics.

Although random projection methods offer an elegant dimensionality reduction alternative for several Big Data problems, they face some shortcomings: i) sketching each new input signal entails a nontrivial computational cost, which can be a bottleneck in streaming applications; ii) the design of the random projection matrix does not take into account any a priori information on the input (known subspace); and iii) the guarantees offered are probabilistic. Alternatively one can think of reducing complexity by simply retaining a few samples of each input. A signal belonging to a known subspace can be modeled as a *bandlimited* graph signal [14,15]. In the context of graph signal processing (GSP), sampling of bandlimited graph signals has been thoroughly studied [14,15], giving rise to several noteworthy sampling schemes [16–19]. Leveraging these advances along with the concept of stationarity [20–22] offers novel insights to design sampling patterns accounting for the signal statistics, that can be applied at negligible online computational cost to a stream of inputs. However, most existing sampling methods are designed with the objective of reconstructing the original graph signal, and do not account for subsequent processing the signal may undergo; see [23–25] for a few recent exceptions.

In this sketching context and towards reducing the online computational cost of obtaining the solution to a linear problem, we leverage GSP results and propose a novel sampling scheme for signals that belong to a *known* low-dimensional subspace. Different from most existing sampling approaches, our design explicitly accounts for the transformation whose output we wish to approximate. By exploiting the stationary nature of the sequence of inputs, we shift the computational burden to the off-line phase where both the sampling pattern and the sketch of the linear transformation are designed. After doing this only once, the online phase merely consists of repeatedly selecting the signal values dictated by the sampling pattern and processing this stream of samples using the sketch of the linear transformation.

In Section 2 we introduce the mathematical formulation of the direct and inverse linear sketching problems as well as the assumptions on the input signals. Then, we proceed to present the solutions for the direct and inverse problems in Section 3. In both cases, we obtain first a closed-form expression for the optimal reduced linear transform as a function of the selected samples of the signal. Then we use that expression to obtain an equivalent optimization problem on the selection of samples, that turns out to be a Semidefinite Program (SDP) modulo binary constraints that arise naturally from the sample (node) selection problem. Section 4 discusses a number of heuristics to obtain tractable solutions to the binary optimization. In Section 5 we apply this framework to the problem of estimating the graph frequency components of a graph signal in a fast and efficient fashion, as well as to the problems of selecting sensors for parameter estimation, classifying handwritten digits and attributing texts to their corresponding author. Finally, conclusions are drawn in Section 6.

Notation Generically, the entries of a matrix $\mathbf{X}$ and a (column) vector $\mathbf{x}$ will be denoted as X_{ij} and x_i. The notation ^{T} and ^{H} stands for transpose and transpose conjugate, respectively, and the superscript † denotes pseudoinverse; $\mathbf{0}$ is the all-zero vector and $\mathbf{1}$ is the all-one vector; and the ℓ_0 pseudo norm $\|\mathbf{X}\|_0$ equals the number of nonzero entries in $\mathbf{X}$. For a vector $\mathbf{x}$, $\text{diag}(\mathbf{x})$ is a diagonal matrix with the (i,i)th entry equal to x_i; when applied to a matrix, $\text{diag}(\mathbf{X})$ is a vector with the diagonal elements of $\mathbf{X}$. For vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ we adopt the partial ordering $\preceq$ defined with respect to the positive orthant $\mathbb{R}^n_+$ by which a $\mathbf{x} \preceq \mathbf{y}$ if and only if $x_i \leq y_i$ for all $i = 1, \ldots, n$. For symmetric matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{n \times n}$, the partial ordering $\preceq$ is adopted with respect to the semidefinite cone, by which $\mathbf{X} \preceq \mathbf{Y}$ if and only if $\mathbf{Y} - \mathbf{X}$ is positive semi-definite.

2. Sketching of bandlimited signals

Our results draw inspiration from the existing literature for sampling of bandlimited graph signals. So we start our discussion in Section 2.1 by defining sampling in the GSP context, and explaining how these ideas extend to more general signals that belong to lower-dimensional subspaces. Then in Sections 2.2 and 2.3 we present, respectively, the direct and inverse formulation of our sketching as sampling problems. GSP applications and motivating examples that involve linear processing of network data are presented in Section 5.

2.1. Graph signal processing

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ be a graph described by a set of n nodes $\mathcal{V}$, a set $\mathcal{E}$ of edges (i, j) and a weight function $\mathcal{W} : \mathcal{E} \to \mathbb{R}$ that assigns weights to the directed edges. Associated with the graph we have a shift operator $\mathbf{S} \in \mathbb{R}^{n \times n}$ which we define as a matrix sharing the sparsity pattern of the graph so that $[\mathbf{S}]_{i,j} = 0$ for all $i \neq j$ such that $(j, i) \notin \mathcal{E}$ [26]. The shift operator is assumed normal so that there exists a matrix of eigenvectors $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ and a diagonal matrix $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_n)$ such that

$$\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^H. \tag{1}$$

We consider realizations $\mathbf{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ of a random signal with zero mean $\mathbb{E}[\mathbf{x}] = \mathbf{0}$ and covariance matrix $\mathbf{R}_x := \mathbb{E}[\mathbf{x}\mathbf{x}^T]$. The random signal $\mathbf{x}$ is interpreted as being supported on $\mathcal{G}$ in the sense that components x_i of $\mathbf{x}$ are associated with node i of $\mathcal{G}$. The graph is intended as a descriptor of the relationship between components of the signal $\mathbf{x}$. The signal $\mathbf{x}$ is said to be stationary on the graph if the eigenvectors of the shift operator $\mathbf{S}$ and the eigenvectors of the covariance matrix $\mathbf{R}_x$ are the same [20–22]. It follows that there exists a diagonal matrix $\tilde{\mathbf{R}}_x \succeq \mathbf{0}$ that allows us to write

$$\mathbf{R}_x := \mathbb{E}[\mathbf{x}\mathbf{x}^T] = \mathbf{V}\tilde{\mathbf{R}}_x\mathbf{V}^H. \tag{2}$$

The diagonal entry $[\tilde{\mathbf{R}}_x]_{ii} = \tilde{r}_i$ is the eigenvalue associated with eigenvector $\mathbf{v}_i$. Without loss of generality we assume eigenvalues are ordered so that $\tilde{r}_i \geq \tilde{r}_j$ for $i \leq j$. Crucially, we assume that exactly $k \leq n$ eigenvalues are nonzero. This implies that if we define the subspace projection $\tilde{\mathbf{x}} := \mathbf{V}^H\mathbf{x}$, its last $n - k$ elements are almost surely zero. Therefore, upon defining the vector $\tilde{\mathbf{x}}_k := [\tilde{x}_1, \ldots, \tilde{x}_k]^T \in \mathbb{R}^k$ containing the first k elements of $\tilde{\mathbf{x}}$ it holds with probability 1 that

$$\tilde{\mathbf{x}} := \mathbf{V}^H\mathbf{x} = [\tilde{\mathbf{x}}_k; \mathbf{0}_{n-k}]^T. \tag{3}$$

Stationary graph signals can arise, e.g., when considering diffusion processes on the graph and they will often have spectral profiles with a few dominant eigenvalues [22,27,28]. Stationary graph signals are important in this paper because they allow for interesting interpretations and natural connections to the sampling of bandlimited graph signals [16–19,23–25] – see Section 3. That said, techniques and results apply for as long as (3) holds whether there is a graph that supports the signal or not.

Observe for future reference that since $\mathbf{V}\tilde{\mathbf{x}} = \mathbf{V}\mathbf{V}^H\mathbf{x} = \mathbf{x}$ we can undo the projection with multiplication by the eigenvector matrix $\mathbf{V}$. This can be simplified because, as per (3), only the first k entries of $\tilde{\mathbf{x}}$ are of interest. Define then the (tall) matrix $\mathbf{V}_k = [\mathbf{v}_1, \ldots, \mathbf{v}_k] \in \mathbb{C}^{n \times k}$ containing the first k eigenvectors of $\mathbf{R}_x$. With this definition we can write $\tilde{\mathbf{x}}_k = \mathbf{V}_k^H\mathbf{x}$ and

$$\mathbf{x} = \mathbf{V}_k\tilde{\mathbf{x}}_k = \mathbf{V}_k(\mathbf{V}_k^H\mathbf{x}). \tag{4}$$

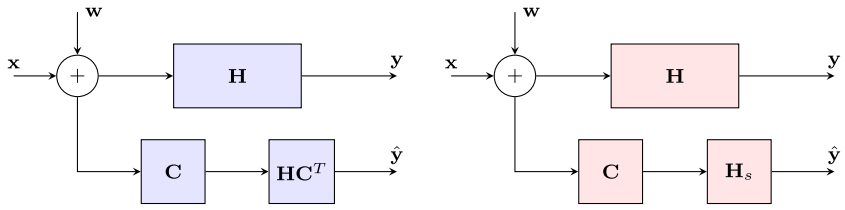


Fig. 1. Direct sketching problem. We observe realizations $\mathbf{x}+\mathbf{w}$ and want to estimate the output $\mathbf{y}=\mathbf{H}\mathbf{x}$ with a reduced complexity pipeline. (Left) We sample the incoming observation $\mathbf{x}_s=\mathbf{C}(\mathbf{x}+\mathbf{w})$, and multiply it by the corresponding columns $\mathbf{H}_s=\mathbf{H}\mathbf{C}^T$ [cf. (6)]; the appropriate samples for all incoming observations are determined by (7). (Right) Since the multiplication by $\mathbf{H}_s$ is the same for all incoming $\mathbf{x}_s$ [cf. (8)] we can design, off-line, an arbitrary matrix $\mathbf{H}_s$ that improves the performance, as per (9).

When a process is such that realizations lie in a k-dimensional subspace as specified in (3) we say that it is k-bandlimited or simply bandlimited if k is understood. We emphasize that in this definition $\mathbf{V}$ is an arbitrary orthonormal matrix which need not be associated with a shift operator as in (1) – notwithstanding the fact that it will be associated with a graph in some applications. Our goal here is to use this property to sketch the computation of linear transformations of $\mathbf{x}$ or to sketch the solution of linear systems of equations involving $\mathbf{x}$ as we explain in Sections 2.2 and 2.3.

2.2. Direct sketching

The direct sketching problem is illustrated in Fig. 1. Consider a noise vector $\mathbf{w}\in\mathbb{R}^n$ with zero mean $\mathbb{E}[\mathbf{w}]=\mathbf{0}$ and covariance matrix $\mathbf{R}_w:=\mathbb{E}[\mathbf{w}\mathbf{w}^T]$. We observe realizations $\mathbf{x}+\mathbf{w}$ and want to estimate the matrix-vector product $\mathbf{y}=\mathbf{H}\mathbf{x}$ for a matrix $\mathbf{H}\in\mathbb{R}^{m\times n}$. Estimating this product requires $O(mn)$ operations. The motivation for sketching algorithms is to devise alternative computation methods requiring a (much) smaller number of operations in settings where multiplication by the matrix $\mathbf{H}$ is to be carried out for a large number of realizations $\mathbf{x}$. In this paper we leverage the bandlimitedness of $\mathbf{x}$ to design sampling algorithms to achieve this goal.

Formally, we define binary selection matrices $\mathbf{C}$ of dimension $p\times n$ as those that belong to the set

$$\mathcal{C}:=\{\mathbf{C}\in\{0,1\}^{p\times n}:\ \mathbf{C}\mathbf{1}=\mathbf{1},\ \mathbf{C}^T\mathbf{1}\preceq\mathbf{1}\}. \tag{5}$$

The restrictions in (5) are such that each row of $\mathbf{C}$ contains exactly one nonzero element and that no column of $\mathbf{C}$ contains more than one nonzero entry. Thus, the product vector $\mathbf{x}_s:=\mathbf{C}\mathbf{x}\in\mathbb{R}^p$ samples (selects) p elements of $\mathbf{x}$ that we use to estimate the product $\mathbf{y}=\mathbf{H}\mathbf{x}$. In doing so we not only need to select entries of $\mathbf{x}$ but columns of $\mathbf{H}$. This is achieved by computing the product $\mathbf{H}_s:=\mathbf{H}\mathbf{C}^T\in\mathbb{R}^{m\times p}$ and we therefore choose to estimate the product $\mathbf{y}=\mathbf{H}\mathbf{x}$ with the product

$$\hat{\mathbf{y}}\ :=\ \mathbf{H}_s\mathbf{x}_s\ :=\ \mathbf{H}\mathbf{C}^T\mathbf{C}(\mathbf{x}+\mathbf{w}). \tag{6}$$

Implementing (6) requires $O(mp)$ operations. Adopting the mean squared error (MSE) as a figure of merit, the optimal sampling matrix $\mathbf{C}\in\mathcal{C}$ is the solution to the minimum (M)MSE problem comparing $\hat{\mathbf{y}}$ in (6) to the desired response $\mathbf{y}=\mathbf{H}\mathbf{x}$, namely

$$\mathbf{C}^*:=\operatorname*{argmin}_{\mathbf{C}\in\mathcal{C}}\mathbb{E}\Big[\big\|\mathbf{H}\mathbf{C}^T\mathbf{C}(\mathbf{x}+\mathbf{w})-\mathbf{H}\mathbf{x}\big)\big\|_2^2\Big]. \tag{7}$$

Observe that selection matrices $\mathbf{C}\in\mathcal{C}$ have rank p and satisfy $\mathbf{C}\mathbf{C}^T=\mathbf{I}$, the p-dimensional identity matrix. It is also readily verified that $\mathbf{C}^T\mathbf{C}=\operatorname{diag}(\mathbf{c})$ with the vector $\mathbf{c}\in\{0,1\}^n$ having entries $c_i=1$ if and only if the ith column of $\mathbf{C}$ contains a nonzero entry. Thus, the product $\mathbf{C}^T\mathbf{C}$ is a diagonal matrix in which the ith entry is nonzero if and only if the ith entry of $\mathbf{x}$ is selected in the sampled vector $\mathbf{x}_s:=\mathbf{C}\mathbf{x}$. In particular, there is a bijective correspondence between the matrix $\mathbf{C}$ and the vector $\mathbf{c}$ modulo an arbitrary ordering of the rows of $\mathbf{C}$. In turn, this implies that choosing $\mathbf{C}\in\mathcal{C}$ in (7) is equivalent to choosing $\mathbf{c}\in\{0,1\}^n$ with $\mathbf{c}^T\mathbf{1}=p$. We take advantage of this fact in the algorithmic developments in Sections 3 and 4.

In (6), the sampled signal $\mathbf{x}_s$ is multiplied by the matrix $\mathbf{H}_s:=\mathbf{H}\mathbf{C}^T$. The restriction to have the matrix $\mathbf{H}_s$ be a sampled version of $\mathbf{H}$ is unnecessary in cases where it is possible to compute an arbitrary matrix off line. This motivates an alternative formulation where we estimate $\mathbf{y}$ as

$$\hat{\mathbf{y}}\ :=\ \mathbf{H}_s\mathbf{x}_s\ :=\ \mathbf{H}_s\mathbf{C}(\mathbf{x}+\mathbf{w}). \tag{8}$$

The matrices $\mathbf{H}_s$ and $\mathbf{C}$ can now be jointly designed as the solution to the MMSE optimization problem

$$\{\mathbf{C}^*,\mathbf{H}_s^*\}:=\operatorname*{argmin}_{\mathbf{C}\in\mathcal{C},\mathbf{H}_s}\mathbb{E}\Big[\big\|\mathbf{H}_s\mathbf{C}(\mathbf{x}+\mathbf{w})-\mathbf{H}\mathbf{x}\big\|_2^2\Big]. \tag{9}$$

To differentiate (7) from (9) we refer to the former as *operator* sketching since it relies on sampling the operator $\mathbf{H}$ that operates on the vector $\mathbf{x}$. In (8) we refer to $\mathbf{H}_s$ as the sketching matrix which, as per (9), is jointly chosen with the sampling matrix $\mathbf{C}$. In either case we expect that sampling $p\approx k$ entries of $\mathbf{x}$ should lead to good approximations of $\mathbf{y}$ given the assumption that $\mathbf{x}$ is k-bandlimited. We will see that $p=k$ suffices in the absence of noise (Proposition 1) and that making $p>k$ helps to reduce the effects of noise otherwise (Proposition 2). We point out that solving (7) or (9) is intractable because of their nonconvex objectives and the binary nature of the matrices $\mathbf{C}\in\mathcal{C}$. Heuristics for approximate solution with manageable computational cost are presented in Section 4. While tractable, these heuristics still entail a significant computation cost. This is justified when solving a large number of estimation tasks. See Section 5 for concrete examples.

2.3. Inverse sketching

The inverse sketching problem seeks to solve a least squares estimation problem with reduced computational cost. As in the case of the direct sketching problem we exploit the bandlimitedness of $\mathbf{x}$ to propose the sampling schemes illustrated in Fig. 2. Formally, we consider the system $\mathbf{x}=\mathbf{H}^T\mathbf{y}$ and want to estimate $\mathbf{y}$ from observations of the form $\mathbf{x}+\mathbf{w}$. As in Section 2.2, the signal $\mathbf{x}\in\mathbb{R}^n$ is k-bandlimited as described in (2)-(4) and the noise $\mathbf{w}\in\mathbb{R}^n$ is zero mean with covariance $\mathbf{R}_w:=\mathbb{E}[\mathbf{w}\mathbf{w}^T]$. The signal of interest is $\mathbf{y}\in\mathbb{R}^m$ and the matrix that relates $\mathbf{x}$ to $\mathbf{y}$ is $\mathbf{H}\in\mathbb{R}^{m\times n}$. The solution to this least squares problem is to make $\hat{\mathbf{y}}=(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}(\mathbf{x}+\mathbf{w})$ which requires $O(mn)$ operations if the matrix $\mathbf{A}_{\text{LS}}=(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}$ is computed off line or $O(m^2n)$ operations if the matrix $\mathbf{H}\mathbf{H}^T$ is computed online.

To reduce the cost of computing the least squares estimate we consider sampling matrices $\mathbf{C}\in\mathcal{C}$ as defined in (5). Thus, the sampled vector $\mathbf{x}_s:=\mathbf{C}(\mathbf{x}+\mathbf{w})$ is one that selects p entries of the observation $\mathbf{x}+\mathbf{w}$. Sampling results in a reduced observation model and leads to the least square estimate

$$\hat{\mathbf{y}}\ :=\ \mathbf{H}_s\mathbf{x}_s\ :=\ (\mathbf{H}\mathbf{C}^T\mathbf{C}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{C}^T\mathbf{C}(\mathbf{x}+\mathbf{w}) \tag{10}$$

where we have defined the estimation matrix $\mathbf{H}_s:=(\mathbf{H}\mathbf{C}^T\mathbf{C}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{C}^T$. The computational cost of implementing (10) is $O(mp)$ operations if the matrix $\mathbf{H}_s$ is computed off line or $O(m^2p)$

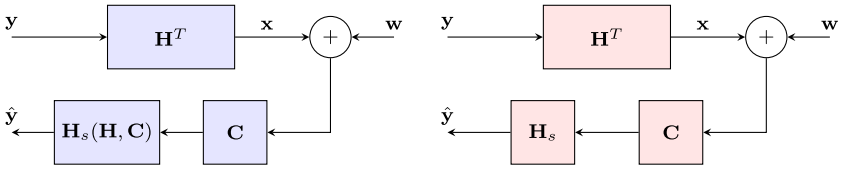


Fig. 2. Inverse sketching problem. Reduce the computational cost of solving a least squares estimation problem $\mathbf{x}=\mathbf{H}^T\mathbf{y}$. (Left) Given observations of $\mathbf{x}+\mathbf{w}$, we sample them $\mathbf{x}_s=\mathbf{C}(\mathbf{x}+\mathbf{w})$ and solve the linear regression problem of the reduced system $\mathbf{C}\mathbf{x}=\mathbf{C}\mathbf{H}^T\mathbf{y}$ by computing the least squares solution $\mathbf{H}_s(\mathbf{H},\mathbf{C})=(\mathbf{H}\mathbf{C}^T\mathbf{C}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{C}^T$ and multiplying it by the sampled observations [cf. (10)]; the optimal samples $\mathbf{C}$ are designed by solving (11). (Right) Likewise, instead of solving the least squares problem on the reduced matrix, we can design an entirely new smaller matrix $\mathbf{H}_s$ that acts on the sampled observations [cf. (12)], jointly designing the sampling pattern $\mathbf{C}$ and the sketch $\mathbf{H}_s$ as per (13).

operations if the matrix $\mathbf{H}_s$ is computed online. We seek the optimal sampling matrix that minimizes the MSE

$$\mathbf{C}^* := \underset{\mathbf{C}\in\mathcal{C}}{\operatorname{argmin}}\, \mathbb{E}\Big[\big\|\mathbf{H}^T(\mathbf{H}\mathbf{C}^T\mathbf{C}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{C}^T\mathbf{C}(\mathbf{x}+\mathbf{w})-\mathbf{x}\big\|_2^2\Big]. \tag{11}$$

As in (7), restricting $\mathbf{H}_s$ in (10) to be a sampled version of $\mathbf{H}$ is unnecessary if $\mathbf{H}_s$ is to be computed off line. In such case we focus on estimates of the form

$$\hat{\mathbf{y}} \;:=\; \mathbf{H}_s\mathbf{x}_s \;:=\; \mathbf{H}_s\mathbf{C}(\mathbf{x}+\mathbf{w}) \tag{12}$$

where the matrix $\mathbf{H}_s\in\mathbb{R}^{m\times p}$ is an arbitrary matrix that we select jointly with $\mathbf{C}$ to minimize the MSE,

$$\{\mathbf{C}^*,\mathbf{H}_s^*\} := \underset{\mathbf{C}\in\mathcal{C},\mathbf{H}_s}{\operatorname{argmin}}\, \mathbb{E}\Big[\big\|\mathbf{H}^T\mathbf{H}_s\mathbf{C}(\mathbf{x}+\mathbf{w})-\mathbf{x}\big\|_2^2\Big]. \tag{13}$$

We refer to (10)-(11) as the inverse *operator* sketching problem and to (12)-(13) as the inverse sketching problem. They differ in that the matrix $\mathbf{H}_s$ is arbitrary and jointly optimized with $\mathbf{C}$ in (13) whereas it is restricted to be of the form $\mathbf{H}_s:=(\mathbf{H}\mathbf{C}^T\mathbf{C}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{C}^T$ in (11). Inverse sketching problems are studied in Section 3. We will see that in the absence of noise we can choose $p=k$ for k-bandlimited signals to sketch estimates that are as good as least square estimates (Proposition 1). In the presence of noise choosing $p>k$ helps in reducing noise (Proposition 2). The computation of optimal sampling matrices $\mathbf{C}$ and optimal estimation matrices $\mathbf{H}_s$ is intractable due to nonconvex objectives and binary constraints in the definition of the set $\mathcal{C}$ in (5). Heuristics for its solution are discussed in Section 4. As in the case of direct sketching, these heuristics still entail significant computation cost that is justifiable when solving a stream of estimation tasks. See Section 5 for concrete examples.

Remark 1 (Sketching and sampling)**.** This paper studies sketching as a sampling problem to reduce the computational cost of computing the linear transformations of a vector $\mathbf{x}$. In its original definition sketching is not necessarily restricted to sampling, is concerned with the inverse problem only, and does not consider the joint design of sampling and estimation matrices [12,13]. Sketching typically refers to the problem in (10)-(11) where the matrix $\mathbf{C}$ is not necessarily restricted to be a sampling matrix – although it often is – but any matrix such that the product $\mathbf{C}(\mathbf{x}+\mathbf{w})$ can be computed with low cost. Our work differs in that we are trying to exploit a bandlimited model for the signal $\mathbf{x}$ to design optimal sampling matrices $\mathbf{C}$ along with, possibly, optimal computation matrices $\mathbf{H}_s$. Our work is also different in that we consider not only the inverse sketching problem of Section 2.3 but also the direct sketching problem of Section 2.2.

3. Direct and inverse sketching as signal sampling

In this section we delve into the solutions of the direct and inverse sketching problems stated in Section 2. We start with the simple case where the observations are noise free (Section 3.1). This will be useful to gain insights on the solutions to the noisy formulations studied in Section 3.2, and to establish links with the literature of sampling graph signals. Collectively, these results will also inform heuristic approaches to approximate the output of the linear transform in the direct sketching problem, and the least squares estimate in the inverse sketching formulation (Section 4). We consider operator sketching constraints in Section 3.3.

3.1. Noise-free observations

Since in this noiseless scenario we have that $\mathbf{w}=\mathbf{0}$ (cf. Figs. 1 and 2), then the desired output for the direct sketching problem is $\mathbf{y}=\mathbf{H}\mathbf{x}$ and the reduced-complexity approximation is given by $\hat{\mathbf{y}}=\mathbf{H}_s\mathbf{C}\mathbf{x}$. In the inverse problem we instead have $\mathbf{x}=\mathbf{H}^T\mathbf{y}$, but assuming $\mathbf{H}$ is full rank then we can exactly invert the aforementioned relationship as $\mathbf{y}:=\mathbf{A}_{\text{LS}}\mathbf{x}=(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{x}$. Accordingly, in the absence of noise we can equivalently view the inverse problem as a direct one whereby $\mathbf{H}=\mathbf{A}_{\text{LS}}$.

Next, we formalize the intuitive result that asserts that perfect estimation, namely that $\hat{\mathbf{y}}=\mathbf{y}$, in the noiseless case is possible if $\mathbf{x}$ is a k-bandlimited signal [cf. (3)] and the number of samples is $p\geq k$. To aid readability, the result is stated as a proposition.

Proposition 1. *Let $\mathbf{x}\in\mathbb{R}^n$ be a k-bandlimited signal and let $\mathbf{H}\in\mathbb{R}^{m\times n}$ be a linear transformation. Let $\mathbf{H}_s\in\mathbb{R}^{m\times p}$ be a reduced-input dimensionality sketch of $\mathbf{H}$, $p\leq n$ and $\mathbf{C}\in\mathcal{C}$ be a selection matrix. In the absence of noise ($\mathbf{w}=\mathbf{0}$), if $p=k$ and $\mathbf{C}^*$ is designed such that* $\operatorname{rank}\{\mathbf{C}^*\mathbf{V}_k\}=p=k$, *then $\hat{\mathbf{y}}=\mathbf{H}_s^*\mathbf{C}^*\mathbf{x}=\mathbf{y}$ provided that the sketching matrix $\mathbf{H}_s^*$ is given by*

$$\mathbf{H}_s^* = \begin{cases} \mathbf{H}\mathbf{V}_k(\mathbf{C}^*\mathbf{V}_k)^{-1}, & \text{Direct sketching,}\\ \mathbf{A}_{\text{LS}}\mathbf{V}_k(\mathbf{C}^*\mathbf{V}_k)^{-1}, & \text{Inverse sketching.}\end{cases} \tag{14}$$

The result follows immediately from e.g., the literature of sampling and reconstruction of bandlimited graph signals via selection sampling [16,18]. Indeed, if $\mathbf{C}^*$ is chosen such that $\operatorname{rank}\{\mathbf{C}^*\mathbf{V}_k\}=p=k$, then one can *perfectly reconstruct* $\mathbf{x}$ from its samples $\mathbf{x}_s:=\mathbf{C}^*\mathbf{x}$ using the interpolation formula

$$\mathbf{x}=\mathbf{V}_k(\mathbf{C}^*\mathbf{V}_k)^{-1}\mathbf{x}_s. \tag{15}$$

The sketches $\mathbf{H}_s^*$ in (14) follow after plugging (15) in $\mathbf{y}=\mathbf{H}\mathbf{x}$ (or $\mathbf{y}=\mathbf{A}_{\text{LS}}\mathbf{x}$ for the inverse problem), and making the necessary identifications in $\hat{\mathbf{y}}=\mathbf{H}_s^*\mathbf{C}^*\mathbf{x}$. Notice that forming the inverse-mapping sketch $\mathbf{H}_s^*$ involves the (costly) computation of $(\mathbf{H}\mathbf{H}^T)^{-1}$ within $\mathbf{A}_{\text{LS}}$, but this is carried out entirely off line.

In the absence of noise the design of $\mathbf{C}$ *decouples* from that of $\mathbf{H}_s$. Towards designing $\mathbf{C}^*$, the $O(p^3)$ complexity techniques proposed in [16] for finding a subset of p rows of $\mathbf{V}_k$ that are linearly independent can be used here. Other existing methods to determine the most informative samples are relevant as well [24].

3.2. Noisy observations

Now consider the general setup described in Sections 2.2 and 2.3, where the noise vector signal $\mathbf{w}\in\mathbb{R}^n$ is random and independent of $\mathbf{x}$, with $\mathbb{E}[\mathbf{w}]=\mathbf{0}$, $\mathbf{R}_w=\mathbb{E}[\mathbf{w}\mathbf{w}^T]\in\mathbb{R}^{n\times n}$ and $\mathbf{R}_w\succ\mathbf{0}$. For

the direct sketching formulation, we have $\mathbf{y} = \mathbf{H}(\mathbf{x} + \mathbf{w})$ and $\hat{\mathbf{y}} = \mathbf{H}_s\mathbf{C}(\mathbf{x} + \mathbf{w})$ (see Fig. 1). In the inverse problem we want to approximate the least squares estimate $\mathbf{A}_{\text{LS}}(\mathbf{x} + \mathbf{w})$ of $\mathbf{y}$, with an estimate of the form $\hat{\mathbf{y}} = \mathbf{H}_s\mathbf{C}(\mathbf{x} + \mathbf{w})$ as depicted in Fig. 2. Naturally, the joint design of $\mathbf{H}_s$ and $\mathbf{C}$ to minimize (9) or (13) must account for the noise statistics.

Said design will be addressed as a two-stage optimization that guarantees global optimality and proceeds in three steps. First, the optimal sketch $\mathbf{H}_s$ is expressed as a *function* of $\mathbf{C}$. Second, such a function is substituted into the MMSE cost to yield a problem that depends only on $\mathbf{C}$. Third, the optimal $\mathbf{H}_s^*$ is found using the function in step one and the optimal value of $\mathbf{C}^*$ found in the step two. The result of this process is summarized in the following proposition; see the ensuing discussion and Appendix A.1 for a short formal proof.

Proposition 2. *Consider the direct and inverse sketching problems in the presence of noise [cf. (9) or (13)]. Their solutions are* $\mathbf{C}^*$ *and* $\mathbf{H}_s^* = \mathbf{H}_s^*(\mathbf{C}^*)$*, where*

$$\mathbf{H}_s^*(\mathbf{C}) = \begin{cases} \mathbf{H}\mathbf{R}_x\mathbf{C}^T\left(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T\right)^{-1}, & \text{Direct sketching,} \\ \mathbf{A}_{\text{LS}}\mathbf{R}_x\mathbf{C}^T\left(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T\right)^{-1}, & \text{Inverse sketching.} \end{cases} \tag{16}$$

For the direct sketching problem (9), the optimal sampling matrix $\mathbf{C}^*$ *can be obtained as the solution to the problem*

$$\min_{\mathbf{C}\in\mathcal{C}} \ \text{tr}\left[\mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{H}\mathbf{R}_x\mathbf{C}^T\left(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{R}_x\mathbf{H}^T\right] \tag{17}$$

Likewise, for the inverse sketching problem (13), $\mathbf{C}^*$ *is the solution of*

$$\min_{\mathbf{C}\in\mathcal{C}} \ \text{tr}\left[\mathbf{R}_x - \mathbf{H}^T\mathbf{A}_{\text{LS}}\mathbf{R}_x\mathbf{C}^T\left(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T\right)^{-1}\mathbf{C}\mathbf{R}_x\mathbf{A}_{\text{LS}}^T\mathbf{H}\right]. \tag{18}$$

For the sake of argument, consider now the direct sketching problem. The optimal sketch $\mathbf{H}_s^*$ in (16) is tantamount to $\mathbf{H}$ acting on a preprocessed version of $\mathbf{x}$, using the matrix $\mathbf{R}_x\mathbf{C}^T(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T)^{-1}$. What this precoding entails is, essentially, choosing the samples of $\mathbf{x}$ with the optimal tradeoff between the signal in the factor $\mathbf{R}_x\mathbf{C}^T$ and the noise in the inverse term $(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T)^{-1}$. This is also natural from elementary results in linear MMSE theory [29]. Specifically, consider forming the MMSE estimate of $\mathbf{x}$ given observations $\mathbf{x}_s = \mathbf{C}\mathbf{x}_s + \mathbf{w}_s$, where $\mathbf{w}_s := \mathbf{C}\mathbf{w}$ is a zero-mean sampled noise with covariance matrix $\mathbf{R}_{w_s} = \mathbf{C}\mathbf{R}_w\mathbf{C}^T$. Said estimator is given by

$$\hat{\mathbf{x}} = \mathbf{R}_x\mathbf{C}^T(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T)^{-1}\mathbf{x}_s. \tag{19}$$

Because linear MMSE estimators are preserved through linear transformations such as $\mathbf{y} = \mathbf{H}\mathbf{x}$, then the sought MMSE estimator of the response signal is $\hat{\mathbf{y}} = \mathbf{H}\hat{\mathbf{x}}$. Hence, the expression for $\mathbf{H}_s^*$ in (16) follows. Once more, the same argument holds for the inverse sketching problem after replacing $\mathbf{H}$ with the least squares operator $\mathbf{A}_{\text{LS}} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}$. Moreover, note that (18) stems from minimizing $\mathbb{E}[\|\mathbf{x} - \mathbf{H}^T\hat{\mathbf{y}}\|_2^2]$ as formulated in (13), which is the standard objective function in linear regression problems. Minimizing $\mathbb{E}[\|\mathbf{y} - \hat{\mathbf{y}}\|_2^2]$ for the inverse problem is also a possibility, and one obtains solutions that closely resemble the direct problem. Here we opt for the former least squares estimator defined in (13), since it is the one considered in the sketching literature [13]; see Section 5 for extensive performance comparisons against this baseline.

Proposition 2 confirms that the optimal selection matrix $\mathbf{C}^*$ is obtained by solving (17) which, after leveraging the expression in (16), only requires knowledge of the given matrices $\mathbf{H}$, $\mathbf{R}_x$ and $\mathbf{R}_w$. The optimal sketch is then found substituting $\mathbf{C}^*$ into (16) as $\mathbf{H}_s^* = \mathbf{H}_s^*(\mathbf{C}^*)$, a step incurring $O(mnp + p^3)$ complexity. Naturally, this two-step solution procedure resulting in $\{\mathbf{C}^*, \mathbf{H}_s^*(\mathbf{C}^*)\}$ entails no loss of optimality [30, Section 4.1.3], while effectively reducing the dimensionality of the optimization problem. Instead of solving a problem with $p(m+n)$ variables [cf. (9)], we first solve a problem with pn variables [cf. (17)] and, then, use the closed-form in (16) for the remaining pm unknowns. The practicality of the approach relies on having a closed-form for $\mathbf{H}_s^*(\mathbf{C})$. While this is possible for the quadratic cost in (9), it can be challenging for other error metrics or nonlinear signal models. In those cases, schemes such as alternating minimization, which solves a sequence of problems of size pm (finding the optimal $\mathbf{H}_s$ given the previous $\mathbf{C}$) and pn (finding the optimal $\mathbf{C}$ given the previous $\mathbf{H}_s$), can be a feasible way to bypass the (higher dimensional and non-convex) joint optimization.

The next proposition establishes that (17) and (18), which yield the optimal $\mathbf{C}^*$, are equivalent to *binary* optimization problems with a linear objective function and subject to linear matrix inequality (LMI) constraints.

Proposition 3. *Let* $\mathbf{c} \in \{0, 1\}^n$ *be the binary vector that contains the diagonal elements of* $\mathbf{C}^T\mathbf{C}$*, i.e.* $\mathbf{C}^T\mathbf{C} = \text{diag}(\mathbf{c})$*. Then, in the context of Proposition 2, the optimization problem (17) over* $\mathbf{C}$ *is equivalent to*

$$\min_{\substack{\mathbf{c}\in\{0,1\}^n, \\ \mathbf{Y},\bar{\mathbf{C}}_\alpha}} \ \text{tr}[\mathbf{Y}] \tag{20}$$

$$\text{s. t. } \bar{\mathbf{C}}_\alpha = \alpha^{-1}\text{diag}(\mathbf{c}) \quad \mathbf{c}^T\mathbf{1}_n = p$$

$$\begin{bmatrix} \mathbf{Y} - \mathbf{H}\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T & \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha \\ \bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T & \bar{\mathbf{R}}_\alpha^{-1} + \bar{\mathbf{C}}_\alpha \end{bmatrix} \succeq \mathbf{0}$$

where $\bar{\mathbf{R}}_\alpha = (\mathbf{R}_x + \mathbf{R}_w - \alpha\mathbf{I}_n)$*,* $\bar{\mathbf{C}}_\alpha$ *and* $\mathbf{Y} \in \mathbb{R}^{m\times m}$ *are auxiliary variables and* $\alpha > 0$ *is any scalar satisfying* $\bar{\mathbf{R}}_\alpha \succ \mathbf{0}$*.*

Similarly, (18) is equivalent to a problem identical to (20) except for the LMI constraint that should be replaced with

$$\begin{bmatrix} \mathbf{Y} - \mathbf{R}_x + \mathbf{H}^T\mathbf{A}_{\text{LS}}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{A}_{\text{LS}}^T\mathbf{H} & \mathbf{H}^T\mathbf{A}_{\text{LS}}\mathbf{R}_x\bar{\mathbf{C}}_\alpha \\ \bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{A}_{\text{LS}}^T\mathbf{H} & \bar{\mathbf{R}}_\alpha^{-1} + \bar{\mathbf{C}}_\alpha \end{bmatrix} \succeq \mathbf{0}. \tag{21}$$

See Appendix A.2 for a proof. Problem (20) is an SDP optimization modulo the binary constraints on vector $\mathbf{c}$, which can be relaxed to yield a convex optimization problem (see Remark 2 for comments on the value of α). Once the relaxed problem is solved, the solution can be binarized again to recover a feasible solution $\mathbf{C} \in \mathcal{C}$. This convex relaxation procedure is detailed in Section 4.

Remark 2 (Numerical considerations)**.** While there always exists $\alpha > 0$ such that $\bar{\mathbf{R}}_\alpha = (\mathbf{R}_x + \mathbf{R}_w - \alpha\mathbf{I}_n)$ is invertible, this value of α has to be smaller than the smallest eigenvalue of $\mathbf{R}_x + \mathbf{R}_w$. In low-noise scenarios this worsens the condition number of $\bar{\mathbf{R}}_\alpha$, creating numerical instabilities when inverting said matrix (especially for large graphs). Alternative heuristic solutions to problems (17) and (20) (and their counterparts for the inverse sketching formulation) are provided in Section 4.1.

3.3. Operator sketching

As explained in Sections 2.2 and 2.3, there may be setups where it is costly (or even infeasible) to freely design the new operator $\mathbf{H}_s$ with entries that do not resemble those in $\mathbf{H}$. This can be the case in distributed setups where the values of $\mathbf{H}$ cannot be adapted, or when the calculation (or storage) of the optimal $\mathbf{H}_s$ is impossible. An alternative to overcome this challenge consists in forming the sketch by sampling p *columns* of $\mathbf{H}$, i.e., setting $\mathbf{H}_s = \mathbf{H}\mathbf{C}^T$ and optimizing for $\mathbf{C}$ in the sense of (7) or (11). Although from an MSE performance point of view such an *operator sketching* design is suboptimal [cf. (16)], numerical tests carried out in Section 5 suggest it can sometimes yield competitive performance. The optimal *sampling* strategy for sketches within this restricted class is given in

Table 1
Computational complexity of the heuristic methods proposed in Section 4 to solve optimization problems in Section 3: (i) Number of operations required; (ii) Time (in seconds) required to solve the problem in Section 5.4, see Fig. 7. Also, for the sake of comparison, we have included a row containing the cost of the optimal solution, meaning the cost of solving the optimization problems exactly with no relaxation on the binary constraints.

Method	Number of operations	Time [s]
Optimal solution	$O\left(\binom{n}{p}\right)$	$> 10^6$
Convex relaxation (SDP)	$O\big((m+n)^{3.5}\big)$	25.29
Noise-blind heuristic	$O(n \log n + nm)$	0.46
Noise-aware heuristic	$O\big(n^3 + nm\big)$	21.88
Greedy approach	$O\big(mn^2p^2 + np^4\big)$	174.38

the following proposition; see Appendix A.3 for a sketch of the proof.

Proposition 4. *Let* $\mathbf{H}_s = \mathbf{H}\mathbf{C}^T$ *be constructed from a subset of* p *columns of* $\mathbf{H}$*. Then, the optimal sampling matrix* $\mathbf{C}^*$ *defined in (7) can be recovered from the diagonal elements* $\mathbf{c}^*$ *of* $(\mathbf{C}^*)^T\mathbf{C}^* = \text{diag}(\mathbf{c}^*)$*, where* $\mathbf{c}^*$ *is the solution to the following problem*

$$\min_{\substack{\mathbf{c}\in\{0,1\}^n,\\ \mathbf{Y},\bar{\mathbf{C}}}} \text{tr}[\mathbf{Y}] \tag{22}$$

$$\text{s.t. } \bar{\mathbf{C}} = \text{diag}(\mathbf{c}) \;\; \mathbf{c}^T\mathbf{1}_n = p$$

$$\begin{bmatrix} \mathbf{Y} - \mathbf{H}\mathbf{R}_x\mathbf{H}^T + 2\mathbf{H}\bar{\mathbf{C}}\mathbf{R}_x\mathbf{H}^T & \mathbf{H}\bar{\mathbf{C}} \\ \bar{\mathbf{C}}\mathbf{H}^T & (\mathbf{R}_x + \mathbf{R}_w)^{-1} \end{bmatrix} \succeq \mathbf{0}.$$

Likewise, the optimal sampling matrix $\mathbf{C}^*$ *defined in (11) can be recovered from the solution to a problem identical to (22) except for the LMI constraint that should be replaced with*

$$\begin{bmatrix} \mathbf{Y} - \mathbf{R}_x + 2\mathbf{H}^T\mathbf{H}\bar{\mathbf{C}}\mathbf{R}_x & \mathbf{H}^T\mathbf{H}\bar{\mathbf{C}} \\ \bar{\mathbf{C}}\mathbf{H}^T\mathbf{H} & (\mathbf{R}_x + \mathbf{R}_w)^{-1} \end{bmatrix} \succeq \mathbf{0}. \tag{23}$$

In closing, we reiterate that solving the optimization problems stated in Propositions 3 and 4 is challenging because of the binary decision variables $\mathbf{c} \in \{0, 1\}^n$. Heuristics for approximate solution with manageable computational cost are discussed next.

4. Heuristic approaches

In this section, several heuristics are outlined for tackling the linear sketching problems described so far. The rationale is that oftentimes the problems posed in Section 3 can be intractable, ill-conditioned, or, just too computationally expensive even if carried out off line. In fact, the optimal solution $\mathbf{C}^*$ to (17) or (18) can be obtained by evaluating the objective function in each one of the $\binom{n}{p}$ possible solutions. Table 1 lists the complexity of each of the proposed methods. Additionally, the time (in seconds) taken to run the simulation related to Fig. 7 is also included in the table for comparison. In all cases, after obtaining $\mathbf{C}^*$, forming the optimal value of $\mathbf{H}_s^*$ in (16) entails $O(mnp + p^3)$ operations.

4.1. Convex relaxation (SDP)

Recall that the main difficulty when solving the optimization problems in Propositions 3 and 4 are the *binary* constraints that render the problems non-convex and, in fact, NP-hard. A standard alternative to overcome this difficulty is to relax the binary constraint $\mathbf{c} \in \{0, 1\}^n$ on the sampling vector as $\mathbf{c} \in [0, 1]^n$. This way, the optimization problem (20), or alternatively, with LMI constraints (21), becomes convex and can be solved with polynomial complexity in $O((m+n)^{3.5})$ operations as per the resulting SDP formulation [30].

Once a solution to the relaxed problem is obtained, two ways of recovering a binary vector $\mathbf{c}$ are considered. The first one consists in computing $\mathbf{p}_c = \mathbf{c}/\|\mathbf{c}\|_1$, which can be viewed as a probability distribution over the samples (SDP-Random). These samples are then drawn at random from this distribution; see [31]. This should be done once, off line, and the same selection matrix used for every incoming input (or output). The second one is a deterministic method referred to as thresholding (SDP-Thresh.), which simply consists in setting the largest p elements to 1 and the rest to 0. Since the elements in $\mathbf{c}$ are non-negative, note that the constraint $\mathbf{c}^T\mathbf{1}_n = p$ considered in the optimal sampling formulation can be equivalently rewritten as $\|\mathbf{c}\|_1 = p$. Using a dual approach, this implies that the objective of the optimization problem is implicitly augmented with an ℓ_1-norm penalty $\lambda\|\mathbf{c}\|_1$, whose regularization parameter λ corresponds to the associated Lagrange multiplier. In other words, the formulation is implicitly promoting sparse solutions. The adopted thresholding is a natural way to approximate the sparsest ℓ_0-(pseudo) norm solution with its convex ℓ_1-norm surrogate. An alternative formulation of the convex optimization problem can thus be obtained by replacing the constraint $\mathbf{c}^T\mathbf{1}_n = p$ with a penalty $\lambda\|\mathbf{c}\|_1$ added to the objective, with a hyperparameter λ. While this approach is popular, for the simulations carried out in Section 5 we opted to keep the constraint $\mathbf{c}^T\mathbf{1}_n = p$ to have explicit control over the number of selected samples p, and to avoid tuning an extra hyperparameter λ.

4.2. Noise-aware heuristic (NAH)

A heuristic that is less computationally costly than the SDP in Section 4.1 can be obtained as follows. Consider for instance the objective function of the direct problem (17)

$$\text{tr}\Big[\mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{H}\mathbf{R}_x\mathbf{C}^T\big(\mathbf{C}(\mathbf{R}_x + \mathbf{R}_w)\mathbf{C}^T\big)^{-1}\mathbf{C}\mathbf{R}_x\mathbf{H}^T\Big] \tag{24}$$

where we note that the noise imposes a tradeoff in the selection of samples. More precisely, while some samples are very important in contributing to the transformed signal, as determined by the rows of $\mathbf{R}_x\mathbf{H}^T$, those same samples might provide very noisy measurements, as determined by the corresponding submatrix of $(\mathbf{R}_x + \mathbf{R}_w)$. Taking this tradeoff into account, the proposed noise-aware heuristic (NAH) consists of selecting the rows of $(\mathbf{R}_x + \mathbf{R}_w)^{-1/2}\mathbf{R}_x\mathbf{H}^T$ with highest ℓ_2 norm. This polynomial-time heuristic entails $O(n^3)$ operations to compute the inverse square root of $(\mathbf{R}_x + \mathbf{R}_w)$ [32] and $O(mn)$ to compute its multiplication with $\mathbf{R}_x\mathbf{H}^T$ as well as the norm. Note that $(\mathbf{R}_x + \mathbf{R}_w)^{-1/2}\mathbf{R}_x\mathbf{H}^T$ resembles a signal-to-noise ratio (SNR), and thus the NAH is attempting to maximize this measure of SNR.

4.3. Noise-blind heuristic (NBH)

Another heuristic that incurs an even lower computational cost can also be obtained by inspection of (17). Recall that the complexity of the NAH is dominated by the computation of $(\mathbf{R}_x + \mathbf{R}_w)^{-1/2}$. An even faster heuristic solution can thus be obtained by simply ignoring this term, which accounted for the noise present in the chosen samples. Accordingly, in what we termed the noise-blind heuristic (NBH) we simply select the p rows of $\mathbf{R}_x\mathbf{H}^T$ that have maximum ℓ_2 norm. The resulting NBH is straightforward to implement, entails $O(mn)$ operations for computing the norm of $\mathbf{R}_x\mathbf{H}^T$ and $O(n\log n)$ operations for the sorting algorithm [33]. It is shown in Section 5 to yield satisfactory performance, especially if the noise variance is low or the linear transform has favorable structure. In summary, the term noise-blind stems from the fact that we are selecting the samples that yield the highest output energy as measured by $\mathbf{R}_x\mathbf{H}^T$, while being completely agnostic to the noise corrupting those same samples.

The analysis of both the NAH (Section 4.2) and NBH (Section 4.3) can be readily extended to the linear inverse problem by inspecting the objective function in (18).

4.4. Greedy approach

Another alternative to approximate the solution of (17) and (18) over $\mathcal{C}$, is to implement an iterative greedy algorithm that adds samples to the sampling set incrementally. At each iteration, the sample that reduces the MSE the most is incorporated to the sampling set. Considering problem (17) as an example, first, one-by-one all n samples are tested and the one that yields the lowest value of the objective function in (17) is added to the sampling set. Then, the sampling set is augmented with one more sample by choosing the one that yields the lowest optimal objective among the remaining $n-1$ ones. The procedure is repeated until p samples are selected in the sampling set. This way only $n+(n-1)+\cdots+(n-(p-1)) < np$ evaluations of the objective function (17) are required. Note that each evaluation of (17) entails $O(mnp+p^3)$ operations, so that the overall cost of the greedy approach is $O(mn^2p^2+np^4)$. Greedy algorithms have well-documented merits for sample selection, even for non-submodular objectives like the one in (17); see [25,34].

5. Numerical examples

To demonstrate the effectiveness of the sketching methods developed in this paper, five numerical test cases are considered. In Sections 5.1 and 5.2 we look at the case where the linear transform to approximate is a graph Fourier transform (GFT) and the signals are bandlimited on a given graph [14,15]. Then, in Section 5.3 we consider an (inverse) linear estimation problem in a wireless sensor network. The transform to approximate is the (fat) linear estimator and choosing samples in this case boils down to selecting the sensors acquiring the measurements [24, Section VI-A]. For the fourth test, in Section 5.4, we look at the classification of digits from the MNIST Handwritten Digit database [35]. By means of principal component analysis (PCA), we can accurately describe these images using a few coefficients, implying that they (approximately) belong to a lower dimensional subspace given by a few columns of the covariance matrix. Finally, in Section 5.5 we consider the problem of authorship attribution to determine whether a given text belongs to some specific author or not, based on the stylometric signatures dictated by word adjacency networks [36].

Throughout, we compare the performance in approximating the output of the linear transform of the sketching-as-sampling technique presented in this paper (implemented by each of the five heuristics introduced) and compare it to that of existing alternatives. More specifically, we consider:

a) The sketching-and-sampling algorithms proposed in this paper, namely: a1) the random sampling scheme based on the convex relaxation (SDP-Random, Section 4.1); a2) the thresholding sampling schemed based on the convex relaxation (SDP-Thresh., Section 4.1); a3) the noise-aware heuristic (NAH, Section 4.2); a4) the noise-blind heuristic (NBH, Section 4.3); a5) the greedy approach (Section 4.4); and a6) the operator sketching methods that directly sample the linear transform (SLT) [cf. Section 3.3] by solving the problems in Proposition 4 using the methods a1)-a5).

b) Algorithms for sampling bandlimited signals [cf. (3)], namely: b1) the experimental design sampling (EDS) method proposed in [37]; and b2) the spectral proxies (SP) greedy-based method proposed in [19, Algorithm 1]. In particular, we note that the EDS method in [37] computes a distribution where the probability of selecting the ith element of $\mathbf{x}$ is proportional to the norm of the ith row of the matrix $\mathbf{V}$ that determines the subspace basis. This gives rise to three different EDS methods depending on the norm used: EDS-1 when using the ℓ_1 norm, EDS-2 when using the ℓ_2 norm [31] and EDS-∞ when using the ℓ_∞ norm [37].

c) Traditional sketching algorithms, namely Algorithms 2.4, 2.6 and 2.11 described in (the tutorial paper) [13], and respectively denoted in the figures as Sketching 2.4, Sketching 2.6 and Sketching 2.11. Note that these schemes entail different (random) designs of the matrix $\mathbf{C}$ in (12) that do not necessarily entail sampling (see Remark 1). They can only be used for the inverse problem (13), hence they will be tested only in Sections 5.1 and 5.3.

d) The optimal linear transform $\hat{\mathbf{y}} = \mathbf{A}^*(\mathbf{x}+\mathbf{w})$ that minimizes the MSE, operating on the entire signal, without any sampling nor sketching. Exploiting the fact that the noise $\mathbf{w}$ and the signal $\mathbf{x}$ are uncorrelated, these estimators are $\mathbf{A}^* = \mathbf{H}$ for the direct case and $\mathbf{A}^* = \mathbf{A}_{\text{LS}} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}$ for the inverse case. We use these method as the corresponding baselines (denoted as Full). Likewise, for the classification problems in Sections 5.4 and 5.5, the baseline is the corresponding classifier operating on the entire signal, without any sampling (denoted as SVM).

To aid readability and reduce the number of curves on the figures presented next, only the best-performing among the three EDS methods in b1) and the best-performing of the three sketching algorithms in c) are shown.

5.1. Approximating the GFT as an inverse problem

Obtaining alternative data representations that would offer better insights and facilitate the resolution of specific tasks is one of the main concerns in signal processing and machine learning. In the particular context of GSP, a k-bandlimited graph signal $\mathbf{x} = \mathbf{V}_k\tilde{\mathbf{x}}_k$ can be described as belonging to the k-dimensional subspace spanned by the k eigenvectors $\mathbf{V}_k = [\mathbf{v}_1, \ldots, \mathbf{v}_k] \in \mathbb{C}^{k\times n}$ of the graph shift operator $\mathbf{S}$ [cf. (3), (4)]. The coefficients $\tilde{\mathbf{x}}_k \in \mathbb{C}^k$ are known as the GFT coefficients and offer an alternative representation of $\mathbf{x}$ that gives insight into the modes of variability of the graph signal with respect to the underlying graph topology [15].

Computing the GFT coefficients of a bandlimited signal following $\mathbf{x} = \mathbf{V}_k\tilde{\mathbf{x}}_k$ can be modeled as an inverse problem where we have observations of the output $\mathbf{x}$, which are a transformation of $\mathbf{y} = \tilde{\mathbf{x}}_k$ through a linear operation $\mathbf{H}^T = \mathbf{V}_k$ (see Section 2.3). We can thus reduce the complexity of computing the GFT of a sequence of graph signals, by adequately designing a sampling pattern $\mathbf{C}$ and sketching matrix $\mathbf{H}_s$ that operate only on $p \ll n$ samples of each graph signal $\mathbf{x}$, instead of solving $\mathbf{x} = \mathbf{V}_k\tilde{\mathbf{x}}_k$ for the entire graph signal $\mathbf{x}$ (cf. Fig. 2). This reduces the computational complexity by a factor of n/p, serving as a fast method of obtaining the GFT coefficients.

In what follows, we set the graph shift operator $\mathbf{S}$ to be the adjacency matrix of the underlying undirected graph $\mathcal{G}$. Because the shift is symmetric, it can be decomposed as $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. So the GFT to approximate is $\mathbf{V}^T$, a real orthonormal matrix that projects signals onto the eigenvector space of the adjacency of $\mathcal{G}$. With this notation in place, for this first experiment we assume to have access to 100 noisy realizations of this k-bandlimited graph signal $\{\mathbf{x}_t + \mathbf{w}_t\}_{t=1}^{100}$, where $\mathbf{w}_t$ is zero-mean Gaussian noise with $\mathbf{R}_w = \sigma_w^2\mathbf{I}_n$. The objective is to compute $\{\tilde{\mathbf{x}}_{k,t}\}_{t=1}^{100}$, that is, the k active GFT coefficients of each one of these graph signals.

To run the algorithms, we consider two types of undirected graphs: a stochastic block model (SBM) and a small-world (SW) network. Let $\mathcal{G}_{\text{SBM}}$ denote a SBM network with n total nodes and c communities with n_b nodes in each community, $b = 1, \ldots, c$,

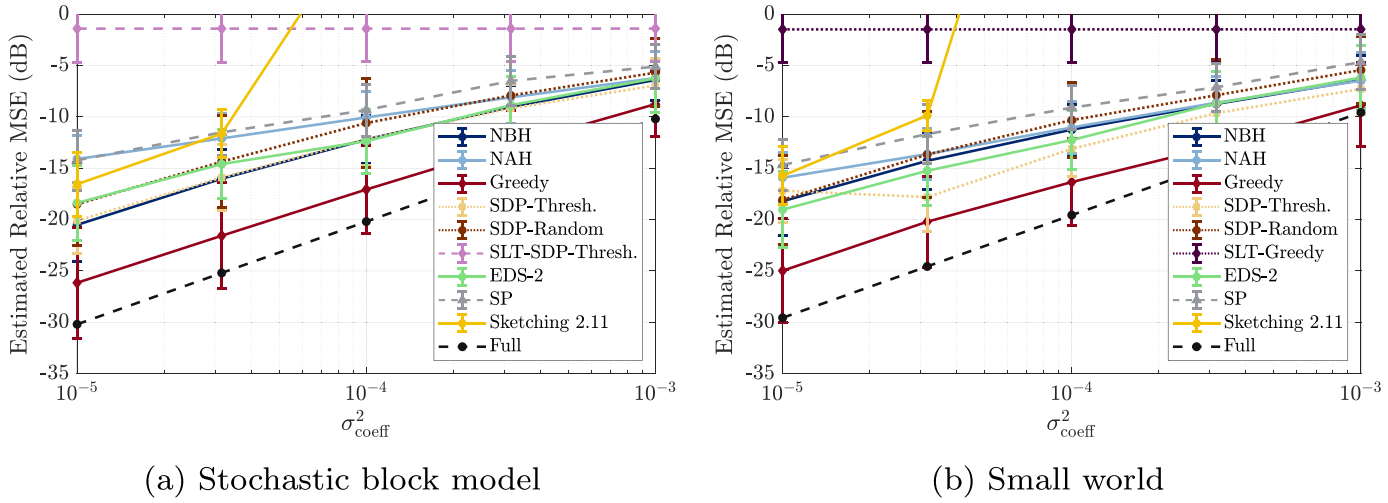


Fig. 3. Approximating the GFT as an inverse sketching problem. Relative estimated MSE as a function of noise. Legends: SDP-Random (scheme in a1); SDP-Thresh. (scheme in a2); NAH (scheme in a3), NBH (scheme in a4); Greedy (scheme in a5); SLT, EDS-1, EDS-2 (schemes in b1), SP (scheme in b2); Sketching 2.6, Sketching 2.11 (schemes in c); and baseline using the full signal (no sampling).

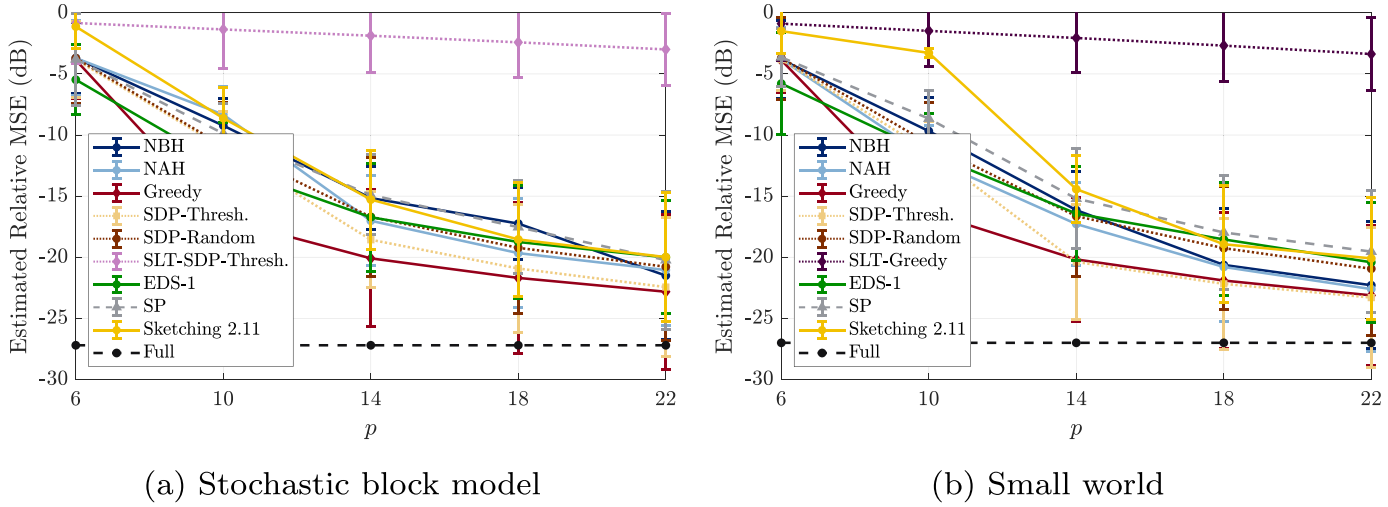
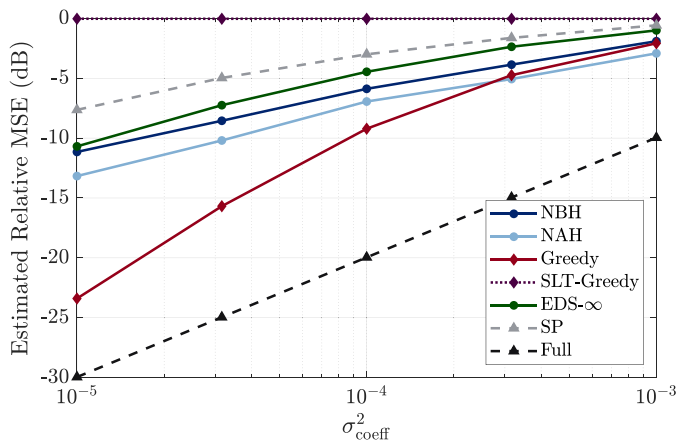


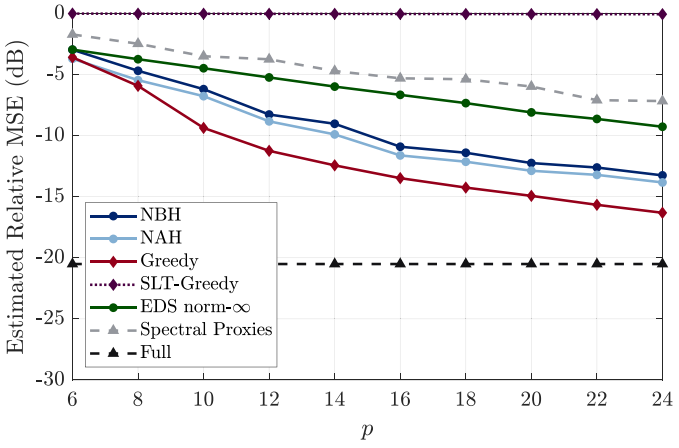
Fig. 4. Approximating the GFT as an inverse sketching problem. Relative estimated MSE as a function of the number of samples used. Legends: SDP-Random (scheme in a1); SDP-Thresh. (scheme in a2); NAH (scheme in a3), NBH (scheme in a4); Greedy (scheme in a5); SLT, EDS-1, EDS-2 (schemes in b1), SP (scheme in b2); Sketching 2.6, Sketching 2.11 (schemes in c); optimal solution using the full signal.

$\sum_{b=1}^{c} n_b = n$ [38]. The probability of drawing an edge between nodes within the same community is p_b and the probability of drawing an edge between any node in community b and any node in community b' is $p_{bb'}$. Similarly, $\mathcal{G}_{\text{SW}}$ describes a SW network with n nodes characterized by parameters p_e (probability of drawing an edge between nodes) and p_r (probability of rewiring edges) [39]. In the first experiment, we study signals $\mathbf{x} \in \mathbb{R}^n$ supported on either the SBM or the SW networks. In each of the test cases presented, $\mathcal{G} \in \{\mathcal{G}_{\text{SBM}}, \mathcal{G}_{\text{SW}}\}$ denotes the underlying graph, and $\mathbf{A} \in \{0, 1\}^{n \times n}$ its associated adjacency matrix. The simulation parameters are set as follows. The number of nodes is $n = 96$ and the bandwidth is $k = 10$. For the SBM, we set $c = 4$ communities of $n_b = 24$ nodes in each, with edge probabilities $p_b = 0.8$ and $p_{bb'} = 0.2$ for $b \neq b'$. For the SW case, we set the edge and rewiring probabilities as $p_e = 0.2$ and $p_r = 0.7$. The metric to assess the reconstruction performance is the relative mean squared error (MSE) computed as $\mathbb{E}[\|\hat{\mathbf{y}} - \tilde{\mathbf{x}}_k\|_2^2]/\mathbb{E}[\|\tilde{\mathbf{x}}_k\|_2^2]$. We estimate the MSE by simulating 100 different sequences of length 100, totaling 10,000 signals. Each of these signals is obtained by simulating $k = 10$ i.i.d. zero-mean, unit-variance Gaussian random variables, squaring them to obtain the GFT $\tilde{\mathbf{x}}_k$, and finally computing $\mathbf{x} = \mathbf{V}_k \tilde{\mathbf{x}}_k$. We repeat this simulation for 5 different random graph realizations. For the methods that use the covariance matrix $\mathbf{R}_x$ as input, we estimate $\mathbf{R}_x$ from 500 realizations of $\mathbf{x}$, which we regarded as training samples and are not used for estimating the relative MSE. Finally, for the methods in which the sampling is random, i.e., a1), b1), and c), we perform the node selection 10 different times and average the results.

For each of the graph types (SBM and SW), we carry out two parametric simulations. In the first one, we consider the number of selected samples to be fixed to $p = k = 10$ and consider different noise power levels $\sigma_w^2 = \sigma^2_{\text{coeff}} \cdot \mathbb{E}[\|\mathbf{x}\|^2]$ by varying σ^2_{coeff} from 10^{-5} to 10^{-3} and where $\mathbb{E}[\|\mathbf{x}\|^2]$ is estimated from the training samples, see Fig. 3 for results. For the second simulation, we fix the noise to $\sigma^2_{\text{coeff}} = 10^{-4}$ and vary the number of selected samples p from 6 to 22, see Fig. 4.



(a) Function of the noise variance



(b) Function of the sample size

Fig. 5. Approximating the GFT as a direct sketching problem for a large network. Relative estimated MSE $\|\hat{\mathbf{y}} - \tilde{\mathbf{x}}_k\|^2 / \|\tilde{\mathbf{x}}_k\|^2$ for the problem of estimating the $k = 10$ frequency components of a bandlimited graph signal from noisy observations, supported on an ER graph with $p = 0.1$ and size $n = 10,000$. 5 a As a function of σ^2_{coeff} for fixed $p = k = 10$. 5 b As a function of the p for fixed $\sigma^2_{\text{coeff}} = 10^{-4}$.

First, Fig. 3a and b show the estimated relative MSE as a function of σ^2_{coeff} for fixed $p = k = 10$ for the SBM and the SW graph supports, respectively. We note that, for both graph supports, the greedy approach in a5) outperforms all other methods and is, at most, 5dB worse than the baseline which computes the GFT using the full signal, while saving 10 times computational cost in the online stage. Then, we observe that the SDP-Thresh. method in a2) is the second best method, followed closely by the EDS-2 in b1). We observe that both NAH and NBH heuristics of a3) and a4) have similar performance, with the NBH working considerably better in the low-noise scenario. This is likely due to the suboptimal account of the noise carried out by the NAH (see Section 4.2). NBH works as well as the SDP-Thresh. method in the SBM case. With respect to off-line computational complexity, we note that the EDS-2 method incurs in an off-line cost of $O(n^3 + n^2 + n\log(n))$ which, in this problem, reduces to $O(10^6)$, while the greedy scheme has a cost of $O(10^7)$; however, the greedy approach performs almost one order of magnitude better in the low-noise scenario. Alternatively, the NBH heuristic has a comparable performance to the EDS-2 on both graph supports, but incurs in an off-line cost of $O(10^3)$.

Second, the relative MSE as a function of the number of samples p for fixed noise σ^2_{coeff} for the SBM and SW networks is plotted in Fig. 4a and b, respectively. In this case, we note that the greedy approach of a5) outperforms all other methods, but its relative gain in terms of MSE is inferior. This would suggest that less computationally expensive methods like the NBH are preferable, especially for higher number of samples. Additionally, we observe that for $p < k$, the EDS graph signal sampling technique works better. For $p = 22$ selected nodes, the best performing sketching-as-sampling technique (the greedy approach) performs 5dB worse than the baseline, but incurring in only 23% of the online computational cost.

5.2. Approximating the GFT in a large-scale network

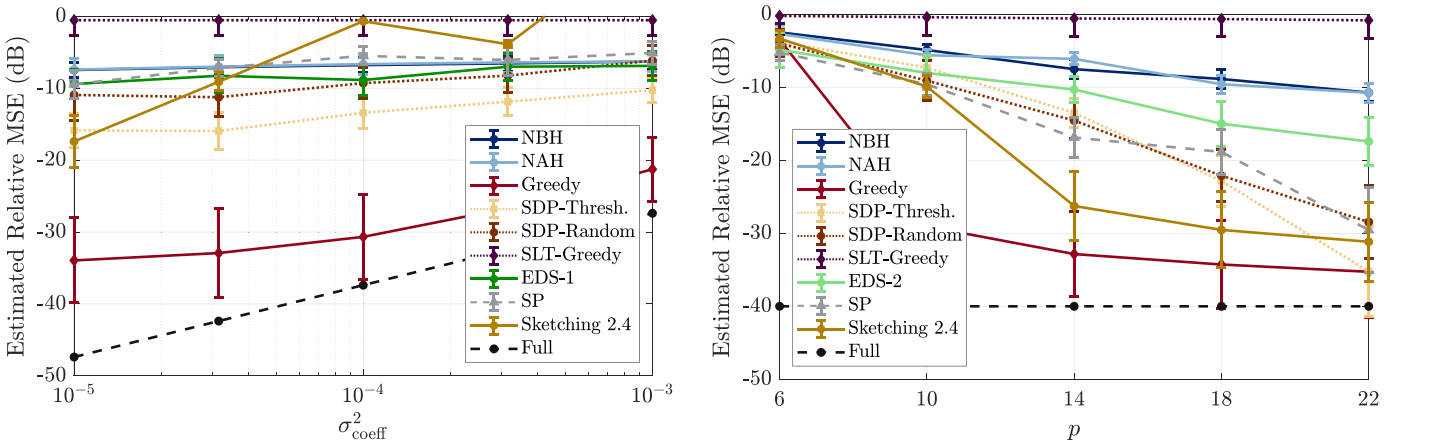
The GFT coefficients can also be computed via a matrix multiplication $\tilde{\mathbf{x}}_k = \mathbf{V}_k^H \mathbf{x}$. We can model this operation as a direct problem, where the input is given by $\mathbf{x}$, the linear transform is $\mathbf{H} = \mathbf{V}_k^H$ and the output is $\mathbf{y} = \tilde{\mathbf{x}}_k$. We can thus proceed to reduce the complexity of this operation by designing a sampling pattern $\mathbf{C}$ and a matrix sketch $\mathbf{H}_s$ that compute an approximate output operating only on a subset of $p \ll n$ samples of $\mathbf{x}$ [cf. (8)]. This way, the computational complexity is reduced by a factor of p/n when compared to computing the GFT using $\mathbf{V}_k^H$ directly on the entire graph signal $\mathbf{x}$.

We consider a substantially larger problem with an Erdős-Rényi (ER) graph $\mathcal{G}_{\text{ER}}$ of $n = 10,000$ nodes and where edges connecting pairs of nodes are drawn independently with $p_{\text{ER}} = 0.1$. The signal under study is bandlimited with $k = 10$ frequency coefficients, generated in the same way as in Section 5.1. To solve the problem (17) we consider the NAH in a3), the NBH in a4) and the greedy approach in a5). We also consider the case in which the linear transform is sampled directly (6), solving (7) by the greedy approach. Comparisons are carried out against all the methods in b).

In Fig. 5 we show the relative MSE for each method, in two different simulations. First, in Fig. 5a the simulation was carried out as a function of noise σ^2_{coeff} varying from 10^{-5} to 10^{-3}, for a fixed number of samples $p = k = 10$. We observe that the greedy approach in a5) performs best. The NAH in a3) and the NBH in a4) are the next best performers, achieving a comparable MSE. The EDS-∞ in b1) yields the best results among the competing methods. We see that for the low-noise case ($\sigma^2_{\text{coeff}} = 10^{-5}$) we can obtain a performance that is 7dB worse than the baseline, but using only $p = 10$ nodes out of $n = 10,000$, thereby *reducing the online computational cost of computing the GFT coefficients by a factor of 1,000*.

For the second simulation, whose results are shown in Fig. 5b, we fixed the noise at $\sigma^2_{\text{coeff}} = 10^{-4}$ and varied the number of samples from $p = 6$ to $p = 24$. Again, the greedy approach in a5) outperforms all the other methods, and the NAH in a3) and the NBH in a4) as the next best performers. The EDS-∞ in b1) performs better than the SP method in b2). We note that, for $p < k$, the EDS-∞ method has a performance very close to the NBH in a4), but as p grows larger, the gap between them widens, improving the relative performance of the NBH. When selecting $p = 24$ nodes, the greedy approach achieves an MSE that is 4dB worse than the baseline, but incurring in only 0.24% of the online computational cost.

With respect to the off-line cost, the EDS-∞ method incurs in $O(n^3 + n^2 \log n + n \log n)$ which, in this setting is $O(10^{12})$, while the greedy cost is $O(10^{11})$; yet, the MSE of the greedy approach is over



(a) Function of the noise variance

(b) Function of the sample size

Fig. 6. Sensor selection for parameter estimation. Sensor are distributed uniformly over the $[0, 1]^2$ region of the plane. Sensor graph is built using a Gaussian kernel over the Euclidean distance between sensors, and keeping only 4 nearest neighbors. Relative estimated MSE as: 6 a A function of σ^2_{coeff} and 6 b A function of p. Legends: SDP-Random (scheme in a1); SDP-Thresh. (scheme in a2); NAH (scheme in a3); NBH (scheme in a4); Greedy (scheme in a5); SLT, EDS-1 and EDS-2 (schemes in b1); SP (scheme in b2); Sketching 2.6, Sketching 2.11 (schemes in c); optimal solution using the full signal.

an order of magnitude better than that of EDS-∞. Likewise, the NAH and the NBH yield a lower, but comparable performance, with an off-line cost of $O(10^9)$ and $O(10^6)$, respectively.

5.3. Sensor selection for distributed parameter estimation

Here we address the problem of sensor selection for communication-efficient distributed parameter estimation [24]. The model under study considers the measurements $\mathbf{x} \in \mathbb{R}^n$ of each of the n sensors to be given as a linear transform of some unknown parameter $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{x} = \mathbf{H}^T\mathbf{y}$, where $\mathbf{H}^T \in \mathbb{R}^{n\times m}$ is the observation matrix; refer to [24, Section II] for details.

We consider $n = 96$ sensors, $m = 12$ unknown parameters, and that the bandwidth of the sensor measurements is $k = 10$. Following [24, Section VI-A], matrix $\mathbf{H}^T$ is random where each element is drawn independently from a zero-mean Gaussian distribution with variance $1/\sqrt{n}$. The underlying graph support $\mathcal{G}_{\text{U}}$ is built as follows. Each sensor is positioned at random, according to a uniform distribution, in the region $[0, 1]^2$ of the plane. With $d_{i,j}$ denoting the Euclidean distance between sensors i and j, their corresponding link weight is computed as $w_{ij} = \alpha e^{-\beta d^2_{i,j}}$, where α and β are constants selected such that the minimum and maximum weights are 0.01 and 1. The network is further sparsified by keeping only the edges in the 4-nearest neighbor graph. Finally, the resulting adjacency matrix is used as the graph shift operator $\mathbf{S}$.

For the simulations in this setting we generate a collection $\{\mathbf{x}_t + \mathbf{w}_t\}_{t=1}^{100}$ of sensor measurements. For each one of these measurements, 100 noise realizations are drawn to estimate the relative MSE defined as $\mathbb{E}[\|\mathbf{y} - \hat{\mathbf{y}}\|^2]/\mathbb{E}[\|\mathbf{y}\|^2]$. Recall that $\hat{\mathbf{y}} = \mathbf{H}^*_s\mathbf{C}^*(\mathbf{x} + \mathbf{w})$ [cf. (12)], with $\mathbf{H}^*_s$ given in Proposition 2 and $\mathbf{C}^*$ designed according to the methods under study. We run the simulations for 5 different sensor networks and average the results, which are shown in Fig. 6.

For the first simulation, the number of samples is fixed as $p = k = 10$ and the noise coefficient σ^2_{coeff} varies from 10^{-5} to 10^{-3}. The estimated relative MSE is shown in Fig. 6b. We observe that the greedy approach in a5) performs considerably better than any other method. The solution provided by SDP-Thresh. in a2) exhibits the second best performance. With respect to the methods under comparison, we note that EDS-1 in b1) is the best performer and outperforms NAH in a3) and the NBH in a4), but is still worse than SDP-Random in a1), although comparable. We also observe that while traditional Sketching 2.4 in c) yields good results for low-noise scenarios, its performance quickly degrades as the noise increases. We conclude that using the greedy approach it is possible to estimate the parameter $\mathbf{y}$ with a 4dB loss, with respect to the optimal, baseline solution, but taking measurements from only 10 out of the 96 deployed sensors.

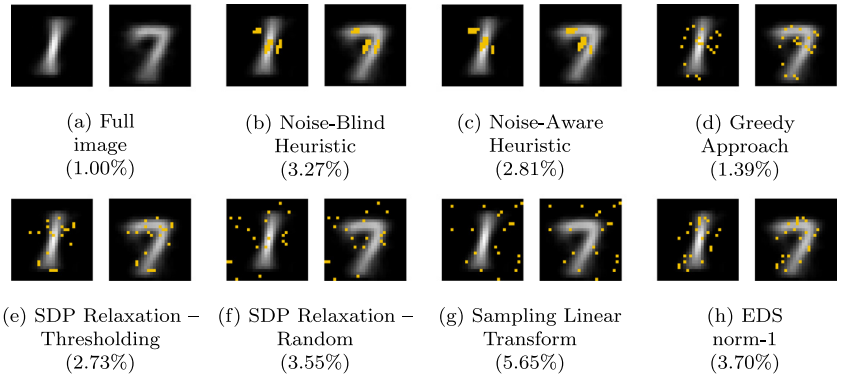
In the second simulation, we fixed the noise given by $\sigma^2_{\text{coeff}} = 10^{-4}$ and varied the number of samples from $p = 6$ to $p = 22$. The estimated relative MSE is depicted in Fig. 6b. We observe that the greedy approach in a5) outperforms all other methods. We also note that, in this case, the Sketching 2.4 algorithm in c) has a very good performance. It is also worth pointing out that the SP method in b2) outperforms the EDS scheme in b1), and that the performance of the SDP relaxations in a1) and a2) improves considerably as p increases.

5.4. MNIST handwritten digits classification

Images are another example of signals that (approximately) belong to a lower-dimensional subspace. In fact, principal component analysis (PCA) shows that only a few coefficients are enough to describe an image [40,41]. More precisely, if we vectorize an image, compute its covariance matrix, and project it onto the eigenvectors of this matrix, then the resulting vector (the PCA coefficients) would have most of its components almost zero. This shows that natural images are approximately bandlimited in the subspace spanned by the eigenvectors of the covariance matrix, and thus are suitable for sampling.

We focus on the problem of classifying images of handwritten digits from the MNIST database [35]. To do so, we use a linear support vector machine (SVM) classifier [42], trained to operate on a few of the PCA coefficients of each image. We can model this task as a direct problem where the linear transform to apply to each (vectorized) image is the cascade of the projection onto the covariance matrix eigenvectors (the PCA transform), followed by the linear SVM classifier.

To be more formal, let $\mathbf{x} \in \mathbb{R}^n$ be the vectorized image, where $n = 28 \times 28 = 784$ is the total number of pixels. Each element of this vector represents the value of a pixel. Denote by $\mathbf{R}_{\text{x}} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ the covariance matrix of $\mathbf{x}$. Then, the PCA coefficients are com-



(a) Full image (1.00%) (b) Noise-Blind Heuristic (3.27%) (c) Noise-Aware Heuristic (2.81%) (d) Greedy Approach (1.39%)

(e) SDP Relaxation – Thresholding (2.73%) (f) SDP Relaxation – Random (3.55%) (g) Sampling Linear Transform (5.65%) (h) EDS norm-1 (3.70%)

Fig. 7. Selected pixels to use for classification of the digits according to each strategy. The percentage error (ratio of errors to total number of images) is also shown. 7 a Images showcasing the average of all images in the test set labeled as a 1 (left) and as a 7 (right). Using all pixels to compute $k = 20$ PCA coefficients and feeding them to a linear SVM classifier yields 1.00% error.

puted as $\mathbf{x}^{\text{PCA}} = \mathbf{V}^T\mathbf{x}$ [cf. (3)]. Typically, there are only a few non-negligible PCA coefficients which we assume to be the first $k \ll n$ elements. These elements can be directly computed by $\mathbf{x}_k^{\text{PCA}} = \mathbf{V}_k^T\mathbf{x}$ [cf. (4)]. Then, these k PCA coefficients are fed into a linear SVM classifier, $\mathbf{A}_{\text{SVM}} \in \mathbb{R}^{m\times k}$ where m is the total number of digits to be classified (the total number of classes). Lastly, $\mathbf{y} = \mathbf{A}_{\text{SVM}}\mathbf{x}_k^{\text{PCA}} = \mathbf{A}_{\text{SVM}}\mathbf{V}_k^T\mathbf{x}$ is used to determine the class (typically, by assigning the image to the class corresponding to the maximum element in $\mathbf{y}$). This task can be cast as a direct sketching problem (8), by assigning the linear transform to be $\mathbf{H} = \mathbf{A}_{\text{SVM}}\mathbf{V}_k^T \in \mathbb{R}^{m\times n}$, with $\mathbf{y} \in \mathbb{R}^m$ being the output and the vectorized image $\mathbf{x} \in \mathbb{R}^n$ being the input.

In this experiment, we consider the classification of c different digits. The MNIST database consists of a training set of 60,000 images and a test set of 10,000. We select, uniformly at random, a subset $\mathcal{T}$ of training images and a subset $\mathcal{S}$ of test images, containing only images of the c digits to be classified. We use the $|\mathcal{T}|$ images in the training set to estimate the covariance matrix $\mathbf{R}_x$ and to train the SVM classifier [43]. Then, we run on the test set $\mathcal{S}$ the classification using the full image, as well as the results of the sketching-as-sampling method implemented by the five different heuristics in a), and compute the classification error $e = |\#\text{misclassified images}|/|\mathcal{S}|$ as a measure of performance. The baseline method, in this case, stems from computing the classification error obtained when operating on the entire image, without using any sampling or sketching. For all simulations, we add noise to the collection of images to be classified $\{\mathbf{x}_t + \mathbf{w}_t\}_{t=1}^{|\mathcal{S}|}$, where $\mathbf{w}_t$ is zero-mean Gaussian noise with variance $\mathbf{R}_w = \sigma_w^2\mathbf{I}_n$, with $\sigma_w^2 = \sigma_{\text{coeff}}^2\mathbb{E}[\|\mathbf{x}\|^2]$ (where the expected energy is estimated from the images in the training set $\mathcal{T}$). We assess performance as function of σ_{coeff}^2 and p.

We start by classifying digits $\{1, 7\}$, that is $c = 2$. We do so for fixed $p = k = 20$ and $\sigma_{\text{coeff}}^2 = 10^{-4}$. The training set has $|\mathcal{T}| = 10,000$ images (5,000 of each digit) and the test set contains $|\mathcal{S}| = 200$ images (100 of each). Fig. 7 illustrates the averaged images of both digits (averaged across all images of each given class in the test set $\mathcal{S}$) as well as the selected pixels following each different selection technique. We note that, when using the full image, the percentage error obtained is 1%, which means that 2 out $|\mathcal{S}| = 200$ images were misclassified. The greedy approach (Fig. 7d) has a percentage error of 1.5% which entails 3 misclassified images, only one more than the full image SVM classifier, but using only $p = 20$ pixels instead of the $n = 784$ pixels of the image. Remarkably, even after reducing the online computational cost by a factor of 39.2, we only incur a marginal performance degradation (a single additional misclassified image). Moreover, solving the direct sketching-as-sampling problem with any of the proposed heuristics in a) outperforms the selection matrix obtained by EDS-1. When considering the computationally simpler problem of using the same sampling matrix for the signal and the linear transform [cf. (6)], the error incurred is of 5.65%. Finally, we note that the sketching-as-sampling techniques tend to select pixels for classification that are different in each image (pixels that are black in the image of one digit and white in the image of the other digit, and vice versa), i.e., the most discriminative pixels.

For the first parametric simulation, we consider the same two digits $\{1, 7\}$ under the same setting as before, but, in one case, for fixed $p = k = 20$ and varying noise σ_{coeff}^2 (Fig. 8a) and for fixed $\sigma_{\text{coeff}}^2 = 10^{-4}$ and varying p (Fig. 8b). We carried out these simulations for 5 different random dataset train/test splits. The greedy approach in a5) outperforms all other solutions in a), and performs comparably to the SVM classifier using the entire image. The NAH and NBH also yield satisfactory performance. In the most favorable situation, the greedy approach yields the same performance as the SVM classifier on the full image, but using only 20 pixels, the worst case difference is of 0.33% (1 image) when using only $p = 16$ pixels (49% reduction in computational cost).

For the second parametric simulation, we consider $c = 10$ digits: $\{0, 1, \ldots, 9\}$ (Fig. 9). We observe that the greedy approach in a5) is always the best performer, although the relative performance with respect to the SVM classifier on the entire image worsens. We also observe that the NAH and NBH are more sensitive to noise, being outperformed by the EDS-1 in b1) and the SDP relaxations. These three simulations showcase the tradeoff between faster computations and performance.

5.5. Authorship attribution

As a last example of the sketching-as-sampling methods, we address the problem of authorship attribution where we want to determine whether a text was written by a given author or not [36]. To this end, we collect a set of texts that we know have been written by the given author (the training set), and build a word adjacency network (WAN) of function words for each of these texts. Function words, as defined in linguistics, carry little lexical meaning or have ambiguous meaning and express grammatical relationships among other words within a sentence, and as such, cannot be attributed to a specific text due to its semantic content. It has been found that the order of appearance of these function words, together with their frequency, determine a stylometric signature of the author. WANs capture, precisely, this fact. More specifically, they determine a relationship between words using a mutual information measure based on the order of appearance of these words

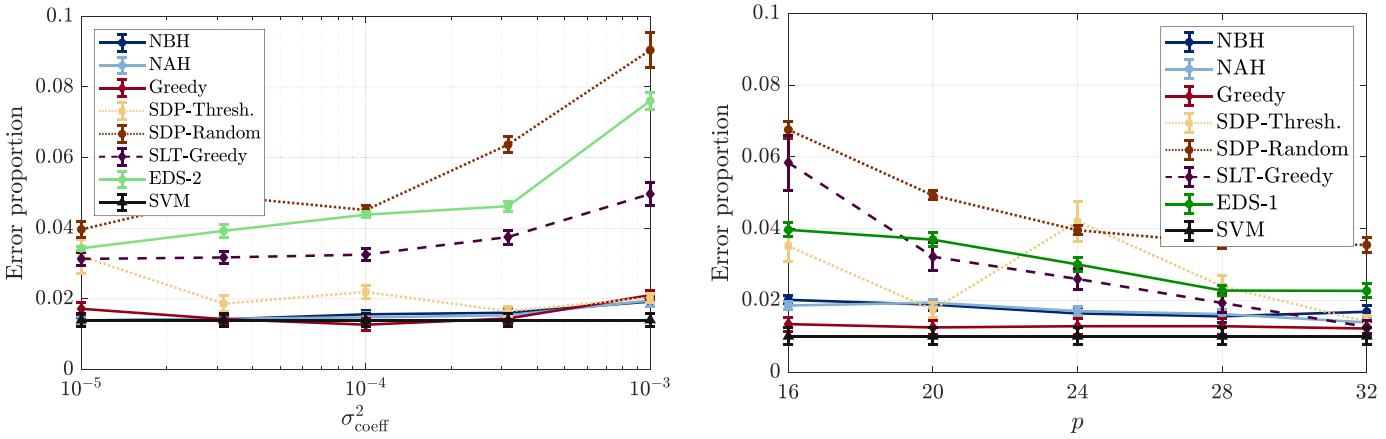


Fig. 8. MNIST Digit Classification for digits 1 and 7. Error proportion as 8 a function of noise σ^2_{coeff} and as 8 b a function of the number of samples p. We note that in both cases the greedy approach in a5) works best. The worst case difference between the greedy approach and the SVM classifier using the entire image is 0.33%, which happens when attempting classification with just 16 pixels out of 784.

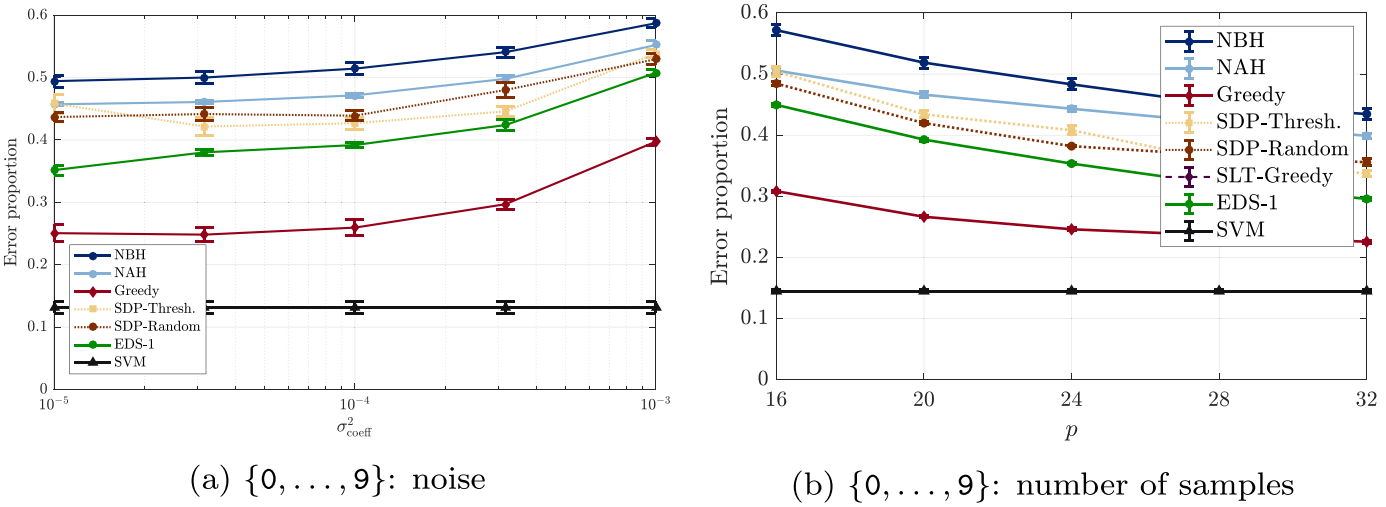


Fig. 9. MNIST Digit Classification for all ten digits. Error proportion as 9 a function of noise σ^2_{coeff} and as 9 b a function of the number of samples p. We note that in both cases the greedy approach in a5) works best. The worst case difference between the greedy approach and the SVM classifier using the entire image is 16.36%.

(how often two function words appear together and how many other words are usually in between them). For more details on function words and WAN computation, please refer to [36].

In what follows, we consider the corpus of novels written by Jane Austen. Each novel is split in fragments of around 1,000 words, leading to 771 texts. Of these, we take 617 at random to be part of the training set, and 154 to be part of the test set. For each of the 617 texts in the training set we build a WAN considering 211 function words, as detailed in [36]. Then we combine these WANs to build a single graph, undirected, normalized and connected (usually leaving around 190 function words for each random partition, where some words were discarded to make the graph connected). An illustration of one realization of a resulting graph can be found in Fig. 10. The adjacency matrix of the resulting graph is adopted as the shift operator $\mathbf{S}$.

Now that we have built the graph representing the stylometric signature of Jane Austen, we proceed to obtain the corresponding graph signals. We obtain the word frequency count of the function words on each of the 771 texts, respecting the split 617 for training and 154 for test set, and since each function word represents a node in the graph, the word frequency count can be modeled as a graph signal. The objective is to exploit the relationship between the graph signal (word frequency count) and the underlying graph support (WAN) to determine whether a given text was authored by Jane Austen or not. To do so, we use a linear SVM classifier (in a similar fashion as in the MNIST example covered in Section 5.4). The SVM classifier is trained by augmenting the training set with another 617 graph signals (word frequency count) belonging to texts written by other contemporary authors, including Louisa May Alcott, Emily Brontë, Charles Dickens, Mark Twain, Edith Wharton, among others. The 617 graph signals corresponding to texts by Jane Austen are assigned a label 1 and the remaining 617 samples are assigned a label 0. This labeled training set of 1,234 samples is used to carry out supervised training of a linear SVM. The trained linear SVM serves as a linear transform on the incoming graph signal, so that it becomes $\mathbf{H}$ in the direct model problem. We then apply the sketching-as-sampling methods discussed in this work to each of the texts in the test set (which has been augmented to include 154 texts of other contemporary authors, totaling 308 samples). To assess performance, we evaluate the error rate of determining whether the texts in the test set were authored by Jane Austen or not, that is achieved by the *sketched*

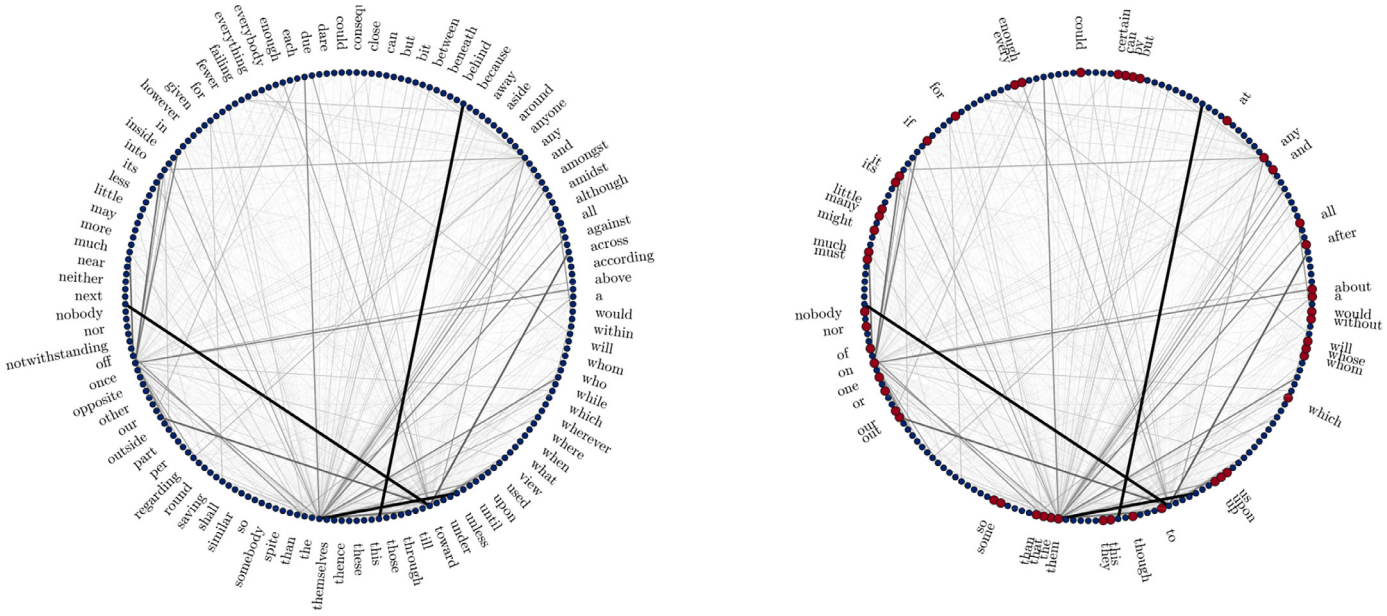


Fig. 10. Word adjacency networks (WANs). 10 a Example of a WAN built from the training set of texts written by Jane Austen; to avoid clutter only one every other word are shown as node labels. 10 b Highlighted in red are the words selected by the greedy approach in a5). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

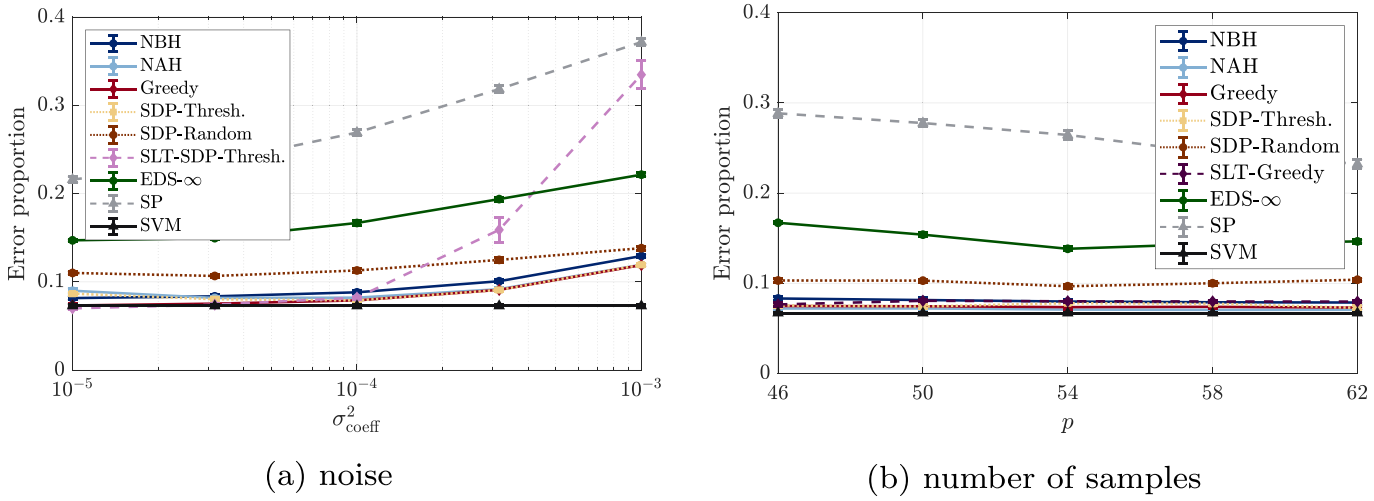


Fig. 11. Authorship attribution for texts written by Jane Austen. Error proportion as 11 a function of noise σ^2_{coeff} and as 11 b a function of the number of samples p (number of words selected). We note that in both cases the greedy approach in a5), the noise-aware heuristic (NAH) in and the SDP relaxation with thresholding (SDP-Thresh.) in offer similar, best performance. The baseline error for an SVM operating on the full text is 7.27% in 11 a and 6.69% in 11 b.

linear classifiers operating on only a subset of function words (selected nodes).

For the experiments, we thus considered sequences $\{\mathbf{x}_t\}_{t=1}^{308}$ of the 308 test samples, where each $\mathbf{x}_t$ is the graph signal representing the word frequency count of each text t in the test set. After observing the graph frequency response of these graph signals over the graph (given by the WAN), we note that signals are approximately bandlimited with $k = 50$ components. We repeat the experiment for 5 different realizations of the random train/test set split of the corpus. To estimate $\mathbf{R}_x$ we use the sample covariance of the graph signals in the training set. The classification error is computed as the proportion of mislabeled texts out of the 308 test samples, and is averaged across the random realizations of the dataset split. We run simulations for different noise levels, computed as $\sigma_w^2 = \sigma^2_{\text{coeff}} \cdot \mathbb{E}[\|\mathbf{x}\|^2]$ for a fixed number of $p = k = 50$ selected nodes (function words), and also for different number of selected nodes p for a fixed noise coefficient $\sigma^2_{\text{coeff}} = 10^{-4}$. Results are shown in Fig. 11. Additionally, Fig. 10 illustrates an example of the selected words for one of the realizations.

In Fig. 11a we show the error rate as a function of noise for all the methods considered. First and foremost, we observe a baseline error rate of 7.27% corresponding to the SVM acting on all function words. As expected, observe that the performance of all the methods degrades as more noise is considered (classification error in-

creases). In particular, the greedy approach attains the lowest error rate, with a performance matched by both the SDP-relaxation with thresholding in a2) and the NAH in a3). It is interesting to observe that the technique of directly sampling the linear transform and the signal (method a6) performs as well as the greedy approach in the low-noise scenario, but then degrades rapidly. Finally, we note that selecting samples irrespective of the linear classifier exhibits a much higher error-rate than the sketching-as-sampling counterparts. This is the case for the EDS-∞ shown in Fig. 11a (the best performer among all methods in b).

In the second experiment for fixed noise $\sigma^2_{\text{coeff}} = 10^{-4}$ and varying number of selected samples p, we show the resulting error rate in Fig. 11b. The baseline for the SVM classifier using all the function words is a 6.69% error rate. Next, we observe that the error rate is virtually unchanged as more function words are selected, suggesting that the performance of the methods in this example is quite robust. Among all the competing methods, we see that the best performing one is the NAH in a3), incurring in an error rate of 7%, but with comparable performance from the greedy approach in a5) as well as the SDP relaxation with thresholding in a2). The selection of words with complete disregard for the classification task (i.e. selecting words only for reconstruction) significantly degrades the performance, as evidenced by the error rate achieved by the EDS-∞ (the best performer among all methods described in b).

6. Conclusions

In this work, we proposed a novel framework for sampling signals that belong to a lower-dimensional subspace. By viewing these signals as bandlimited graph signals, we leveraged notions of sampling drawn from the GSP framework. The objective is to significantly reduce the cost of approximating the output of a linear transform of the given signal. This can be achieved by leveraging knowledge of both the signal statistics as well as the said linear operator.

Under the working principle that the linear transform has to be applied to a sequence of streaming signals, we jointly designed a sampling pattern together with a sketch of the linear operator so that this sketch can be affordably applied to a few samples of each incoming signal. The joint design problem was formulated as a two-stage optimization. We first found the expression for the optimal sketching matrix as a function of feasible sampling strategies, and then substituted this expression in the original MSE cost to solve for the optimal sampling scheme. Overall, the resulting *sketched* output offers a good approximation to the output of the full linear transform, but at a much lower online computational cost.

We proposed five different heuristics to (approximately) solve the MSE minimization problems, which are intractable due to the (combinatorial) sample selection constraints. We then tested these heuristics, together with traditional sketching approaches and other sampling methods (that select samples for reconstruction) in four different scenarios, namely: computation of GFT coefficients, distributed estimation in a sensor network, handwritten digit classification and authorship attribution. All these GSP applications involve signals that belong to lower-dimensional subspaces. In general terms, we observed that the greedy heuristic works best, even though at a higher off-line computational cost. We also noted that the noise-blind and noise-aware heuristics offer reasonable performance at a very reduced cost, while the SDP relaxation performance is highly dependent on the application and does not scale gracefully for large problems. Any of these heuristics, however, outperforms both traditional sketching methods as well as sampling-for-reconstruction schemes in the recent GSP literature.

Declaration of Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

For concreteness, here we give the proofs for the direct sketching problem. The arguments for the inverse problem are similar; see [2] for details.

A1. Proof of Proposition 2

The objective function in (9) can be rewritten as

$$\begin{aligned}\mathbb{E}\big[\|\mathbf{y}-\hat{\mathbf{y}}\|_2^2\big] &= \mathbb{E}\big[\|\mathbf{H}\mathbf{x}-\mathbf{H}_s\mathbf{C}(\mathbf{x}+\mathbf{w})\|_2^2\big] \\ &= \text{tr}\big[\mathbf{H}\mathbf{R}_x\mathbf{H}^T - 2\mathbf{H}_s\mathbf{C}\mathbf{R}_x\mathbf{H}^T + \mathbf{H}_s\mathbf{C}(\mathbf{R}_x+\mathbf{R}_w)\mathbf{C}^T\mathbf{H}_s^T\big]\end{aligned} \tag{25}$$

since $\mathbf{x}$ and $\mathbf{w}$ are assumed independent. Optimizing with respect to $\mathbf{H}_s$ first, yields

$$\mathbf{H}_s^*(\mathbf{C}) = \mathbf{H}\mathbf{R}_x\mathbf{C}^T\big(\mathbf{C}(\mathbf{R}_x+\mathbf{R}_w)\mathbf{C}^T\big)^{-1} \tag{26}$$

establishing the first branch in (16). Matrix $\mathbf{C}\in\mathcal{C}$ is full rank since it selects p *distinct* samples, then $\mathbf{C}(\mathbf{R}_x+\mathbf{R}_w)\mathbf{C}^T$ has rank p and thus it is invertible [44]. Substituting (26) into (25), yields (17). □

A2. Proof of Proposition 3

The inverse in the objective of (17) can be written as [24,30]

$$\begin{aligned}(\mathbf{C}(\mathbf{R}_x+\mathbf{R}_w)\mathbf{C}^T)^{-1} &= \big(\alpha\mathbf{I}_p - \alpha\mathbf{I}_p + \mathbf{C}(\mathbf{R}_x+\mathbf{R}_w)\mathbf{C}^T\big)^{-1} \\ &= \alpha^{-1}\mathbf{I}_p - \alpha^{-2}\mathbf{C}(\bar{\mathbf{R}}_\alpha^{-1}+\alpha^{-1}\mathbf{C}^T\mathbf{C})^{-1}\mathbf{C}^T\end{aligned} \tag{27}$$

where $\alpha\neq 0$ is a rescaling parameter, and in obtaining the second equality we used the Woodbury Matrix Identity [45]. Note that α has to be such that $\bar{\mathbf{R}}_\alpha = (\mathbf{R}_x+\mathbf{R}_w-\alpha\mathbf{I}_n)$ is still invertible. Substituting (27) into (17) and recalling that $\mathbf{C}^T\mathbf{C} = \text{diag}(\mathbf{c})$, we have that

$$\begin{aligned}\min_{\mathbf{c}\in\{0,1\}^n,\bar{\mathbf{C}}_\alpha}\ & \text{tr}\Big[\mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\big(\bar{\mathbf{R}}_\alpha^{-1}+\bar{\mathbf{C}}_\alpha\big)^{-1}\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T\Big] \\ \text{s.t. }\ & \bar{\mathbf{C}}_\alpha = \alpha^{-1}\text{diag}(\mathbf{c})\ \ \mathbf{c}^T\mathbf{1}_n = p.\end{aligned} \tag{28}$$

Note that in (28) we optimize over a binary vector $\mathbf{c}\in\mathbb{R}^n$ with exactly p nonzero entries, instead of a binary matrix $\mathbf{C}\in\mathcal{C}$. The p nonzero elements in $\mathbf{c}$ indicate the samples to be taken. Problem (28) can be reformulated as

$$\min_{\substack{\mathbf{c}\in\{0,1\}^n,\\ \mathbf{Y},\bar{\mathbf{C}}_\alpha}} \text{tr}[\mathbf{Y}] \tag{29}$$

$$\begin{aligned}\text{s.t. }\ & \mathbf{H}\mathbf{R}_x\mathbf{H}^T - \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\mathbf{R}_x\bar{\mathbf{C}}_\alpha\big(\bar{\mathbf{R}}_\alpha^{-1}+\bar{\mathbf{C}}_\alpha\big)^{-1}\bar{\mathbf{C}}_\alpha\mathbf{R}_x\mathbf{H}^T \preceq \mathbf{Y} \\ & \bar{\mathbf{C}}_\alpha = \alpha^{-1}\text{diag}(\mathbf{c})\ \ \mathbf{c}^T\mathbf{1}_n = p\end{aligned}$$

where $\mathbf{Y}\in\mathbb{R}^{m\times m}$, $\mathbf{Y}\succeq\mathbf{0}$ is an auxiliary optimization variable. Using the Schur-complement lemma for *positive definite* matrices [30], (29) can be written as (20). Hence, to complete the proof we need to show that $\bar{\mathbf{R}}_\alpha^{-1}+\bar{\mathbf{C}}_\alpha\succeq\mathbf{0}$ so that the aforementioned lemma can be invoked. To that end, suppose first that $\alpha<0$. Then we have that $\bar{\mathbf{R}}_\alpha\succ\mathbf{0}$ and $\bar{\mathbf{C}}_\alpha = \alpha^{-1}\text{diag}(\mathbf{c})\preceq\mathbf{0}$, so that $\bar{\mathbf{R}}_\alpha^{-1}+\bar{\mathbf{C}}_\alpha\succeq\mathbf{0}$ may not be positive definite. Suppose now that $\alpha>0$. Then $\bar{\mathbf{C}}_\alpha\succeq\mathbf{0}$ and there always exists a sufficiently small positive α such that $\bar{\mathbf{R}}_\alpha\succ\mathbf{0}$ since $\mathbf{R}_w\succ\mathbf{0}$. This implies that if α is chosen such that $\alpha>0$ and $\bar{\mathbf{R}}_\alpha = \mathbf{R}_x+\mathbf{R}_w-\alpha\mathbf{I}_n\succ\mathbf{0}$ (i.e., the conditions stated in the proposition), then $\bar{\mathbf{R}}_\alpha^{-1}+\bar{\mathbf{C}}_\alpha$ is positive definite and problems (28) and (20) are equivalent. □

A3. Proof of Proposition 4

Defining matrix $\bar{\mathbf{C}} = \text{diag}(\mathbf{c})$, the estimated output is

$$\hat{\mathbf{y}} = \mathbf{H}\mathbf{C}^T\mathbf{C}(\mathbf{x}+\mathbf{w}) = \mathbf{H}\text{diag}(\mathbf{c})(\mathbf{x}+\mathbf{w}) = \mathbf{H}\bar{\mathbf{C}}(\mathbf{x}+\mathbf{w}).$$

Since the desired output is $\mathbf{y} = \mathbf{H}\mathbf{x}$, the MSE simplifies to

$$\mathbb{E}\big[\|\mathbf{H}\mathbf{x} - \mathbf{H}\bar{\mathbf{C}}(\mathbf{x}+\mathbf{w})\|_2^2\big] = \text{tr}\big[\mathbf{H}\mathbf{R}_x\mathbf{H}^T - 2\mathbf{H}\bar{\mathbf{C}}\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\bar{\mathbf{C}}(\mathbf{R}_x+\mathbf{R}_w)\bar{\mathbf{C}}\mathbf{H}^T\big].$$

Introducing the auxiliary variable $\mathbf{Y} \in \mathbb{R}^{m\times m}$, $\mathbf{Y} \succeq \mathbf{0}$, minimizing the MSE with respect to $\mathbf{C} \in \mathcal{C}$ is equivalent to solving

$$\min_{\substack{\mathbf{c}\in\{0,1\}^n,\\ \mathbf{Y},\bar{\mathbf{C}}}} \text{tr}[\mathbf{Y}] \tag{30}$$

$$\text{s.t. } \bar{\mathbf{C}} = \text{diag}(\mathbf{c}) \quad \mathbf{c}^T\mathbf{1}_n = p$$

$$\mathbf{H}\mathbf{R}_x\mathbf{H}^T - 2\mathbf{H}\bar{\mathbf{C}}\mathbf{R}_x\mathbf{H}^T + \mathbf{H}\bar{\mathbf{C}}(\mathbf{R}_x+\mathbf{R}_w)\bar{\mathbf{C}}\mathbf{H}^T \preceq \mathbf{Y}.$$

Finally, using the Schur complement-based lemma for positive semidefiniteness [30], then (30) can be shown equivalent to (22), competing the proof. □

CRediT authorship contribution statement

Fernando Gama: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Writing - review & editing, Visualization. **Antonio G. Marques:** Conceptualization, Methodology, Formal analysis, Investigation, Writing - original draft, Writing - review & editing. **Gonzalo Mateos:** Conceptualization, Methodology, Formal analysis, Investigation, Writing - original draft, Writing - review & editing. **Alejandro Ribeiro:** Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing - review & editing, Visualization, Supervision.

References

[1] F. Gama, A. G. Marques, G. Mateos, A. Ribeiro, Rethinking sketching as sampling: Linear transforms of graph signals, in: 50th Asilomar Conf. Signals, Systems and Comput., IEEE, Pacific Grove, CA, 2016, pp. 522–526.

[2] F. Gama, A. G. Marques, G. Mateos, A. Ribeiro, Rethinking sketching as sampling: Efficient approximate solution to linear inverse problems, in: 2016 IEEE Global Conf. Signal and Inform. Process., IEEE, Washington, DC, 2016, pp. 390–394.

[3] E. Moulines, P. Duhamel, J.-F. Cardoso, S. Mayrargue, Subspace methods for blind identification of multichannel FIR filters, IEEE Trans. Signal Process. 43 (2) (1995) 516–525.

[4] W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing signal reconstruction, IEEE Trans. Inf. Theory 55 (2–5) (2009) 2230–2249.

[5] H. Akçay, Subspace-based spectrum estimation in frequency-domain by regularized nuclear norm minimization, Signal Process. 88 (2014) 69–85.

[6] P.V. Giampouras, A.A. Rontogiannis, K.E. Themelis, K.D. Koutroumbas, Online sparse and low-rank subspace learning from incomplete data: a bayesian view, Signal Process. 137 (2017) 199–212.

[7] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 29 (1) (2007) 40–51.

[8] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.

[9] K. Slavakis, G.B. Giannakis, G. Mateos, Modeling and optimization for big data analytics: (statistical) learning tools for our era of data deluge, IEEE Signal Process. Mag. 31 (5) (2014) 18–31.

[10] D.K. Berberidis, V. Kekatos, G.B. Giannakis, Online censoring for large-scale regressions with application to streaming big data, IEEE Trans. Signal Process. 64 (15) (2016) 3854–3867.

[11] C. Boutsidis, M.W. Mahoney, P. Drineas, An improved approximation algorithm for the column subset selection problem, in: 20th ACM-SIAM Symp. Discrete Algorithms, SIAM, New York, NY, 2009, pp. 968–977.

[12] M.W. Mahoney, Randomized algorithms for matrices and data, Found. Trends® Mach. Learn. 3 (2) (2011) 123–224.

[13] D.P. Woodruff, Sketching as a tool for numerical linear algebra, Found. Trends® Theoret. Comput. Sci. 10 (1–2) (2014) 1–157.

[14] D.I. Shuman, S.K. Narang, P. Frossard, A. Ortega, P. Vandergheynst, The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains, IEEE Signal Process. Mag. 30 (3) (2013) 83–98.

[15] A. Sandryhaila, J.M.F. Moura, Discrete signal processing on graphs: frequency analysis, IEEE Trans. Signal Process. 62 (12) (2014) 3042–3054.

[16] S. Chen, R. Varma, A. Sandryhaila, J. Kovačević, Discrete signal processing on graphs: sampling theory, IEEE Trans. Signal Process. 63 (24) (2015) 6510–6523.

[17] A. G. Marques, S. Segarra, G. Leus, A. Ribeiro, Sampling of graph signals with successive local aggregations, IEEE Trans. Signal Process. 64 (7) (2016) 1832–1843.

[18] M. Tsitsvero, S. Barbarossa, P. Di Lorenzo, Signals on graphs: uncertainty principle and sampling, IEEE Trans. Signal Process. 64 (18) (2016) 4845–4860.

[19] A. Anis, A. Gadde, A. Ortega, Efficient sampling set selection for bandlimited graph signals using graph spectral proxies, IEEE Trans. Signal Process. 64 (14) (2016) 3775–3789.

[20] B. Girault, P. Gonçalves, E. Fleury, Translation and Stationarity for Graph Signals, Research Report, École Normale Supérieure de Lyon, Inria Rhône-Alpes, 2015.

[21] N. Perraudin, P. Vandergheynst, Stationary signal processing on graphs, IEEE Trans. Signal Process. 65 (13) (2017) 3462–3477.

[22] A. G. Marques, S. Segarra, G. Leus, A. Ribeiro, Stationary graph processes and spectral estimation, IEEE Trans. Signal Process. 65 (22) (2017) 5911–5926.

[23] R.M. Gray, Quantization in task-driven sensing and distributed processing, in: 31st IEEE Int. Conf. Acoust., Speech and Signal Process., IEEE, Toulouse, France, 2006, pp. 1049–1052.

[24] S. Liu, S.P. Chepuri, M. Fardad, E. Masazade, G. Leus, P.K. Varshney, Sensor selection for estimation with correlated measurement noise, IEEE Trans. Signal Process. 64 (13) (2016) 3509–3522.

[25] S.P. Chepuri, G. Leus, Subsampling for graph power spectrum estimation, in: 2016 IEEE Sensor Array Multichannel Signal Process. Workshop, IEEE, Rio de Janeiro, Brazil, 2016, pp. 1–5.

[26] M. Püschel, J.M.F. Moura, Algebraic signal processing theory: foundation and 1-d time, IEEE Trans. Signal Process. 56 (8) (2008) 3575–3585.

[27] F. Gama, A. Ribeiro, Ergodicity in stationary graph processes: a weak law of large numbers, IEEE Trans. Signal Process. 67 (10) (2019) 2761–2774.

[28] F. Gama, E. Isufi, A. Ribeiro, G. Leus, Controllability of bandlimited graph processes over random time varying graphs, 2019. arXiv: 1904.10089v2 [cs.SY].

[29] S. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory, Prentice-Hall Signal Processing Series, Prentice Hall, Upper Saddle River, NJ, 1993.

[30] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, UK, 2004.

[31] G. Puy, N. Tremblay, R. Gribonval, P. Vandergheynst, Random sampling of bandlimited graph signals, Appl. Comput. Harmonic Anal. 44 (2) (2018) 446–475.

[32] A. Frommer, B. Hashemi, Verified computation of square roots of a matrix, SIAM J. Matrix Anal. Appl. 31 (3) (2010) 1279–1302.

[33] D.E. Knuth, The Art of Computer Programming, 3: Sorting and Searching, 2nd, Addison-Wesley, Upper Saddle River, NJ, 1998.

[34] L.F.O. Chamon, A. Ribeiro, Greedy sampling of graph signals, IEEE Trans. Signal Process. 66 (1) (2018) 34–47.

[35] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[36] S. Segarra, M. Eisen, A. Ribeiro, Authorship attribution through function word adjacency networks, IEEE Trans. Signal Process. 63 (20) (2015) 5464–5478.

[37] R. Varma, S. Chen, J. Kovačević, Spectrum-blind signal recovery on graphs, in: 2015 IEEE Int. Workshop Comput. Advances Multi-Sensor Adaptive Process., IEEE, Cancún, México, 2015, pp. 81–84.

[38] A. Decelle, F. Krzakala, C. Moore, L. Zdeborová, Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications, Phys. Rev. E 84 (6) (2011) 066106(1–19).

[39] D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks, Nature 393 (6684) (1998) 440–442.

[40] J.E. Jackson, A User's Guide to Principal Components, John Wiley and Sons, New York, NY, 1991.

[41] N. Shahid, N. Perraudin, G. Puy, P. Vandergheynst, Compressive PCA for low-rank matrices on graphs, IEEE Trans. Signal, Inform. Process. Networks 3 (4) (2017) 695–710.

[42] A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall Inform. Syst. Sci. Series, Prentice Hall, Englewood Cliffs, NJ, 1989.

[43] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second, John Wiley and Sons, New York, NY, 2001.

[44] R.A. Horn, C.R. Johnson, Matrix Analysis, Cambridge University Press, Cambridge, UK, 1985.

[45] G.H. Golub, C.F. van Loan, Matrix Computations, third, Johns Hopkins University Press, Baltimore, MD, 1996.