# SAMPLING AND RECONSTRUCTION OF GRAPH SIGNALS VIA WEAK SUBMODULARITY AND SEMIDEFINITE RELAXATION

*Abolfazl Hashemi*[†]*, Rasoul Shafipour*[‡]*, Haris Vikalo*[†]*, and Gonzalo Mateos*[‡]

[†]Dept. of Electrical and Computer Engineering, University of Texas at Austin, Austin, TX, USA
[‡]Dept. of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA

## ABSTRACT

We study the problem of sampling a bandlimited graph signal in the presence of noise, where the objective is to select a node subset of prescribed cardinality that minimizes the signal reconstruction mean squared error (MSE). To that end, we formulate the task at hand as the minimization of MSE subject to binary constraints, and approximate the resulting NP-hard problem via semidefinite programming (SDP) relaxation. Moreover, we provide an alternative formulation based on maximizing a monotone weak submodular function and propose a randomized-greedy algorithm to find a sub-optimal subset. We then derive a worst-case performance guarantee on the MSE returned by the randomized greedy algorithm for general non-stationary graph signals. The efficacy of the proposed methods is illustrated through numerical simulations on synthetic and real-world graphs. Notably, the randomized greedy algorithm yields an order-of-magnitude speedup over state-of-the-art greedy sampling schemes, while incurring only a marginal MSE performance loss.

***Index Terms***— graph signal processing, sampling, weak submodularity, semidefinite programming, randomized algorithms

## 1. INTRODUCTION

Consider a network represented by a graph $\mathcal{G}$ consisting of a node set $\mathcal{N}$ of cardinality $N$ and a weighted adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ whose $(i,j)$ entry, $A_{ij}$, denotes weight of the edge connecting node $i$ to node $j$. A *graph signal* $\mathbf{x} \in \mathbb{R}^N$ can be viewed as a vertex-valued network process that can be represented by a vector of size $N$ supported on $\mathcal{N}$, where its $i^{\text{th}}$ component denotes the signal value at node $i$. Under the assumption that properties of the network process relate to the underlying graph, the goal of graph signal processing (GSP) is to generalize traditional signal processing tasks and develop algorithms that fruitfully exploit this relational structure [1, 2].

A keystone generalization which has drawn considerable attention in recent years pertains to sampling and reconstruction of graph signals [3–10]. The task of finding an exact sampling set to perform reconstruction with minimal information loss is known to be NP-hard. Conditions for exact reconstruction of graph signals from noiseless samples were put forth in [3–6]. Sampling of noise-corrupted signals using randomized schemes including uniform and leverage score sampling is studied in [11], for which optimal sampling distributions and performance bounds are derived. In [7, 10], reconstruction of graph signals and their power spectrum density is studied and greedy schemes are developed. However, the performance guarantees in [10, 11] are restricted to the case of stationary graph signals, i.e., the covariance matrix in the nodal or spectral domains has certain structure (e.g., diagonal; see also [12–14]).

Work in this paper was supported by the NSF award CCF-1750428.

In this paper, we study the problem of sampling and reconstruction of graph signals and propose two algorithms that solve it approximately. First, we develop a semidefinite programming (SDP) relaxation that finds a near-optimal sampling set in polynomial time. Then, we formulate the sampling task as that of maximizing a monotone weak submodular function and propose an efficient randomized greedy algorithm motivated by [15]. We analyze the performance of the randomized greedy algorithm and in doing so, we show that the MSE associated with the selected sampling set is on expectation a constant factor away from that of the optimal set. Moreover, we prove that the randomized greedy algorithm achieves the derived approximation bound with high probability for every sampling task. In contrast to prior work, our results do not require stationarity of the signal. Finally, in simulation studies we illustrate the superiority of the proposed schemes over state-of-the-art randomized and greedy algorithms [10, 11] in terms of running time, accuracy, or both. [1]

**Notation.** $\mathbf{A}_{ij}$ denotes the $(i,j)$ entry of matrix $\mathbf{A}$, $\mathbf{a}_j$ is the $j^{\text{th}}$ row of $\mathbf{A}$, $\mathbf{A}_{S,r}$ ($\mathbf{A}_{S,c}$) is a submatrix of $\mathbf{A}$ that contains rows (columns) indexed by the set $S$, and $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ represent the maximum and minimum eigenvalues of $\mathbf{A}$, respectively. $\mathbf{I}_N \in \mathbb{R}^{N\times N}$ denotes the identity matrix and $[N] := \{1, 2, \ldots, N\}$.

## 2. PRELIMINARIES AND PROBLEM STATEMENT

Let $\mathbf{x}$ be a zero-mean, random graph signal which is $k$-bandlimited in a given basis $\mathbf{V} \in \mathbb{R}^{N\times N}$. This means that the signal's so-called graph Fourier transform (GFT) $\bar{\mathbf{x}} = \mathbf{V}^\top \mathbf{x}$ is $k$-sparse. There are several choices for $\mathbf{V}$ in the literature with most aiming to decompose a graph signal into different modes of variation with respect to the graph topology. For instance, $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_N]$ can be defined via the Jordan decomposition of the adjacency matrix [16, 17], through the eigenvectors of the Laplacian when $\mathcal{G}$ is undirected [1], or it can be obtained as the result of an optimization procedure [18, 19]. We also assume that the signal in not necessarily stationary with respect to $\mathcal{G}$, and that $\bar{\mathbf{x}}$ is a zero-mean random vector with (generally non-diagonal) covariance matrix $\mathbb{E}[\bar{\mathbf{x}}\bar{\mathbf{x}}^\top] = \mathbf{P}$. Recall that since $\mathbf{x}$ is bandlimited, $\bar{\mathbf{x}}$ is sparse with at most $k$ nonzero entries. Let $K$ be the support set of $\bar{\mathbf{x}}$, where $|K| = k$. Then, one can write $\mathbf{x} = \mathbf{U}\bar{\mathbf{x}}_K$, where $\mathbf{U} = \mathbf{V}_{K,c}$.

Suppose that only a few (possibly noisy) entries of $\mathbf{x}$ can be observed, corresponding to taking measurements from a subset of nodes in $\mathcal{N}$. The goal of sampling is to select the subset that enables reconstruction of the original signal with the smallest possible distortion. Formally, let $\mathbf{y} = \mathbf{x} + \mathbf{n}$ be the noise-corrupted signal, where $\mathbf{n} \in \mathbb{R}^N$ is the zero-mean noise vector with covariance matrix $\mathbb{E}[\mathbf{n}\mathbf{n}^\top] = \sigma^2\mathbf{I}_N$. Let $S \subseteq \mathcal{N}$ be a sampling subset of nodes and let

[1] MATLAB implementations of the proposed algorithms are available at `https://github.com/realabolfazl/GS-sampling`.

$\hat{\mathbf{x}}$ be the reconstructed graph signal based on the measurements (i.e., the entries of $\mathbf{y}$) indexed by $S$. Since the signal is $k$-bandlimited, $\bar{\mathbf{x}}$ has at most $k$ nonzero entries. Therefore, we assume that the least mean square estimator of $\bar{\mathbf{x}}$ has at most $k$ nonzero entries. This in turn imposes the constraint $|S| \leq k$. Now, since $\mathbf{x} = \mathbf{U}\bar{\mathbf{x}}_K$, the samples $\mathbf{y}_S$ and the non-zero frequency components of $\mathbf{x}$ are related via the Bayesian linear model

$$\mathbf{y}_S = \mathbf{U}_{S,r}\bar{\mathbf{x}}_K + \mathbf{n}_S. \tag{1}$$

Hence, in order to find $\hat{\mathbf{x}}$ it suffices to estimate $\bar{\mathbf{x}}_K$ based on $\mathbf{y}_S$. The least mean square estimator of $\bar{\mathbf{x}}_K$, denoted by $\hat{\hat{\mathbf{x}}}_{K_{\text{lms}}}$, satisfies the Bayesian counterparts of the normal equations in the Gauss-Markov theorem (see e.g., [20, Ch. 10]). Accordingly, it is given by

$$\hat{\hat{\mathbf{x}}}_{K_{\text{lms}}} = \sigma^{-2}\bar{\mathbf{\Sigma}}_S\mathbf{U}_{S,r}^{\top}\mathbf{y}_S, \tag{2}$$

where

$$\bar{\mathbf{\Sigma}}_S = \left(\mathbf{P}^{-1} + \sigma^{-2}\mathbf{U}_{S,r}^{\top}\mathbf{U}_{S,r}\right)^{-1} \tag{3}$$

is the error covariance matrix of $\hat{\hat{\mathbf{x}}}_{K_{\text{lms}}}$. Therefore, $\hat{\mathbf{x}} = \mathbf{U}\hat{\hat{\mathbf{x}}}_{K_{\text{lms}}}$ and its error covariance matrix can be obtained as $\mathbf{\Sigma}_S = \mathbf{U}\bar{\mathbf{\Sigma}}_S\mathbf{U}^{\top}$.

The problem of sampling for near-optimal reconstruction can now be formulated as the task of choosing $S$ so as to minimize the mean square error (MSE) of the estimator $\hat{\mathbf{x}}$. Since the MSE is defined as the trace of the error covariance matrix, we obtain the following optimization problem,

$$\min_{S} \quad \text{Tr}\left(\mathbf{\Sigma}_S\right) \quad \text{s.t.} \quad S \subseteq \mathcal{N}, \;\; |S| \leq k. \tag{4}$$

Using trace properties and the fact that $\mathbf{U}^{\top}\mathbf{U}$ is a Hermitian positive semidefinite matrix, (4) simplifies to

$$\min_{S} \quad \text{Tr}\left(\bar{\mathbf{\Sigma}}_S\right) \quad \text{s.t.} \quad S \subseteq \mathcal{N}, \;\; |S| \leq k. \tag{5}$$

The optimization problem (5) is NP-hard and evaluating all $\binom{N}{k}$ possibilities to find an exact solution makes it intractable even for relatively small graphs. In the next section we propose two alternatives to find near-optimal solutions in polynomial time.

## 3. NEW SCHEMES FOR SAMPLING GRAPH SIGNALS

Here we resort to two approximation methods to find a near-optimal solution $S$ of (5). Our proposed algorithms are based on semidefinite and weak submodular optimization techniques that have recently shown superior performance in applications such as sensor selection [21], graph sketching [9], wireless sensor networks [22], Kalman filtering [23], and sparse signal recovery [24, 25].

### 3.1. Sampling via SDP relaxation

We first develop an SDP relaxation for problem (5). Our proposed scheme is motivated by the framework of [21] developed in the context of sensor scheduling. However, our focus is on sampling and reconstruction of graph signals which entails a different objective function, i.e., MSE. Let $z_i \in \{0, 1\}$ indicate whether the $i^{\text{th}}$ node of $\mathcal{N}$ is included in the sampling set $S$ and define $\mathbf{z} = [z_1, z_2, \ldots, z_N]^{\top}$. Then, (3) can alternatively be written as

$$\bar{\mathbf{\Sigma}}_z = \left(\mathbf{P}^{-1} + \sigma^{-2}\sum_{i=1}^{n} z_i\mathbf{u}_i\mathbf{u}_i^{\top}\right)^{-1}. \tag{6}$$

Therefore, by relaxing the binary constraint $z_i \in \{0, 1\}$ one can obtain a convex relaxation of (5),

$$\min_{\mathbf{z}} \quad \text{Tr}\left(\bar{\mathbf{\Sigma}}_z\right) \quad \text{s.t.} \quad 0 \leq z_i \leq 1, \;\; \sum_{i=1}^{N} z_i \leq k. \tag{7}$$

**Algorithm 1** SDP Relaxation for Graph Sampling

1: **Input:** $\mathbf{P}$, $\mathbf{U}$, $k$.
2: **Output:** Subset $S \subseteq \mathcal{N}$ with $|S| = k$.
3: Find $\mathbf{z}$, the minimizer of the SDP relaxation problem in (10)
4: Set $S$ to contain nodes corresponding to top $k$ entries of $\mathbf{z}$
5: **return** $S$.

In order to obtain an SDP in standard form, let $\mathbf{C}$ be a positive semidefinite matrix such that $\mathbf{C} \succeq \bar{\mathbf{\Sigma}}_z$. Then, (7) is equivalent to

$$\min_{\mathbf{z},\mathbf{C}} \quad \text{Tr}\left(\mathbf{B}\right) \quad \text{s.t.} \quad 0 \leq z_i \leq 1, \;\; \sum_{i=1}^{n} z_i \leq k, \;\; \mathbf{C} - \bar{\mathbf{\Sigma}}_z \succeq \mathbf{0}. \tag{8}$$

The last constraint in (8), i.e., $\mathbf{C} - \bar{\mathbf{\Sigma}}_z \succeq \mathbf{0}$, can be thought of as being the Schur complement [26] of the block matrix

$$\mathbf{B} = \begin{bmatrix} \mathbf{C} & \mathbf{I} \\ \mathbf{I} & \bar{\mathbf{\Sigma}}_z^{-1} \end{bmatrix}. \tag{9}$$

Note that the Schur complement of $\mathbf{B}$ is positive semidefinite if and only if $\mathbf{B} \succeq \mathbf{0}$ [26]. Therefore, replacing the last constraint in (8) with the positive semidefiniteness constraint on $\mathbf{B}$ results in the following SDP relaxation:

$$\min_{\mathbf{z},\mathbf{B}} \quad \text{Tr}\left(\mathbf{B}\right) \quad \text{s.t.} \quad 0 \leq z_i \leq 1, \;\; \sum_{i=1}^{n} z_i \leq k, \;\; \mathbf{B} \succeq \mathbf{0}. \tag{10}$$

An exact solution to (10) can be obtained by means of existing SDP solvers; see, e.g., [27, 28]. However, the solution $\hat{\mathbf{z}}$ contains real-valued entries and hence a rounding procedure is needed to obtain a binary solution. Here, we propose to use the rounding procedure introduced in [21] and accordingly select the nodes of $\mathcal{N}$ corresponding to the $k$ $z_i$'s with largest values. The proposed SDP relaxation sampling scheme is summarized as Algorithm 1.

### 3.2. Sampling via a randomized greedy scheme

The computational complexity of the SDP approach developed in Section 3.1 might be challenging in applications dealing with large graphs. Hence, we now propose an iterative randomized greedy algorithm for the task of sampling and reconstruction of graph signals by formulating (5) as the problem of maximizing a monotone weak submodular set function. First, we define the notion of monotonicity, submodularity, and curvature that will be useful in the subsequent analysis.

A set function $f : 2^X \to \mathbb{R}$ is submodular if

$$f(S \cup \{j\}) - f(S) \geq f(T \cup \{j\}) - f(T) \tag{11}$$

for all subsets $S \subseteq T \subset X$ and $j \in X \backslash T$. The term $f_j(S) = f(S \cup \{j\}) - f(S)$ is the marginal value of adding element $j$ to set $S$. Moreover, the function is monotone if $f(S) \leq f(T)$ for all $S \subseteq T \subseteq X$.

The concept of submodularity can be generalized by the notion of curvature or submodularity ratio [24] that quantifies how close a set function is to being submodular. Specifically, the maximum element-wise curvature of a set function $f$ is defined as

$$\mathcal{C}_{\max} = \max_{1 \leq l < n} \max_{(S,T,i) \in \mathcal{X}_l} f_i(T)/f_i(S), \tag{12}$$

with $\mathcal{X}_l = \{(S, T, i) | S \subset T \subset X, i \in X \backslash T, |T \backslash S| = l, |X| = n\}$. Note that a set function is submodular if and only if $\mathcal{C}_{\max} \leq 1$. Set

functions with $\mathcal{C}_{\max} > 1$ are called weak or approximate submodular functions [24].

Next, similar to [10], we formulate (5) as a set function maximization task. Let $f(S) = \mathrm{Tr}(\mathbf{P} - \bar{\mathbf{\Sigma}}_S)$. Then, (5) can equivalently be written as

$$\max_{S} \quad f(S) \quad \text{s.t.} \quad S \subseteq \mathcal{N}, \quad |S| \leq k. \tag{13}$$

In Proposition 1 below, by applying the matrix inversion lemma [29] we establish that $f(S)$ is monotone and weakly submodular. Moreover, we derive an efficient recursion to find the marginal gain of adding a new node to the sampling set $S$.

**Proposition 1.** *$f(S) = \mathrm{Tr}(\mathbf{P} - \bar{\mathbf{\Sigma}}_S)$ is a weak submodular, monotonically increasing set function, $f(\emptyset) = 0$, and for all $j \in \mathcal{N} \backslash S$*

$$f(S \cup \{j\}) - f(S) = \frac{\mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S^2 \mathbf{u}_j}{\sigma^2 + \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S \mathbf{u}_j}, \ \textit{and} \tag{14}$$

$$\bar{\mathbf{\Sigma}}_{S\cup\{j\}} = \bar{\mathbf{\Sigma}}_S - \frac{\bar{\mathbf{\Sigma}}_S \mathbf{u}_j \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S}{\sigma^2 + \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S \mathbf{u}_j}. \tag{15}$$

Proposition 1 enables efficient construction of the sampling set in an iterative fashion. To further reduce the computational cost, we propose a randomized greedy algorithm that performs the task of sampling set selection in the following way. Starting with $S = \emptyset$, at iteration $(i+1)$ of the algorithm, a subset $R$ of size $s$ is sampled uniformly at random and without replacement from $\mathcal{N} \backslash S$. The marginal gain of each node in $R$ is found using (14), and the one corresponding to the highest marginal gain is added to $S$. Then, the algorithm employs the recursive relation (15) to update $\bar{\mathbf{\Sigma}}_S$ for the subsequent iteration. This procedure is repeated until some stopping criteria, such as condition on the cardinality of $S$ is met. Regarding $s$, we follow the suggestion in [15] and set $s = \frac{N}{k} \log \frac{1}{\epsilon}$, where $e^{-k} \leq \epsilon < 1$ is a predetermined parameter that controls trade-off between the computational cost and MSE of the reconstructed signal; randomized greedy algorithm with smaller $\epsilon$ produces sampling solutions with lower MSE while the one with larger $\epsilon$ requires lower computational costs. Note that if $\epsilon = e^{-k}$, the randomized greedy algorithm in each iteration considers all the available nodes and hence matches the greedy scheme proposed in [10]. However, as we illustrate in our simulation studies, the proposed randomized greedy algorithm is significantly faster than the method in [10] for larger $\epsilon$ while returning essentially the same sampling solution. The randomized greedy algorithm is formalized as Algorithm 2.

**Performance guarantees.** Here we analyze performance of the randomized greedy algorithm. First, Theorem 1 below states that if $f(S)$ is characterized by a bounded maximum element-wise curvature, Algorithm 2 returns a sampling subset yielding an MSE that is on average within a multiplicative factor of the MSE associated with the optimal sampling set.

**Theorem 1.** *Let $\mathcal{C}_{max}$ be the maximum element-wise curvature of $f(S) = \mathrm{Tr}(\mathbf{P} - \bar{\mathbf{\Sigma}}_S)$, the objective function in problem (13). Let $\alpha = (1 - e^{-\frac{1}{c}} - \frac{\epsilon\beta}{c})$, where $c = \max\{1, \mathcal{C}_{\max}\}$, $e^{-k} \leq \epsilon < 1$, and $\beta = 1 + \max\{0, \frac{s}{2N} - \frac{1}{2(N-s)}\}$. Let $S_{rg}$ be the sampling set returned by the randomized greedy algorithm and let $O$ denote the optimal solution of (5). Then,*

$$\mathbb{E}\left[\mathrm{Tr}(\bar{\mathbf{\Sigma}}_{S_{rg}})\right] \leq \alpha \mathrm{Tr}(\bar{\mathbf{\Sigma}}_O) + (1-\alpha)\mathrm{Tr}(\mathbf{P}). \tag{16}$$

The proof of Theorem 1 relies on the argument that if $s = \frac{N}{k} \log \frac{1}{\epsilon}$, then with high probability the random subset $R$ in each iteration of Algorithm 2 contains at least one node from $O$.

**Algorithm 2** Randomized Greedy Algorithm for Graph Sampling

1: **Input:** $\mathbf{P}$, $\mathbf{U}$, $k$, $\epsilon$.
2: **Output:** Subset $S \subseteq \mathcal{N}$ with $|S| = k$.
3: Initialize $S = \emptyset$, $\bar{\mathbf{\Sigma}}_S = \mathbf{P}$.
4: **while** $|S| < k$
5: Choose $R$ by sampling $s = \frac{N}{k} \log(1/\epsilon)$ indices uniformly at random from $\mathcal{N} \backslash S$
6: $j_s = \mathrm{argmax}_{j \in R} \frac{\mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S^2 \mathbf{u}_j}{\sigma^2 + \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S \mathbf{u}_j}$
7: $\bar{\mathbf{\Sigma}}_{S\cup\{j_s\}} = \bar{\mathbf{\Sigma}}_S - \frac{\bar{\mathbf{\Sigma}}_S \mathbf{u}_j \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S}{\sigma^2 + \mathbf{u}_j^\top \bar{\mathbf{\Sigma}}_S \mathbf{u}_j}$
8: Set $S \leftarrow S \cup \{j_s\}$
9: **end while**
10: **return** $S$.

Next, we study the performance of the randomized greedy algorithm using the tools of probably approximately correct (PAC) learning theory [30, 31]. The randomization of Algorithm 2 can be interpreted as approximating the marginal gains of the nodes selected by the greedy scheme proposed in [10]. More specifically, for the $i^{th}$ iteration it holds that $f_{j_{rg}}(S_{rg}) = \eta_i f_{j_g}(S_g)$, where subscripts $rg$ and $g$ refer to the sampling sets and nodes selected by the randomized greedy (Algorithm 2) and greedy algorithm in [10], respectively, and $0 < \eta_i \leq 1$ for all $i \in [k]$ are random variables. In view of this argument and by employing the Bernstein inequality [32], we obtain Theorem 2 which states that the randomized greedy algorithm selects a near-optimal sampling set with high probability.

**Theorem 2.** *Instate the notation and hypotheses of Theorem 1. Assume $\{\eta_i\}_{i=1}^k$ are independent such that $\mathbb{E}[\eta_i] \geq \mu$, for all $i \in [k]$. Then, for all $0 < q < 1$ and for some $C > 0$, with probability at least $1 - e^{-Ck}$ it holds that*

$$\mathrm{Tr}(\bar{\mathbf{\Sigma}}_{S_{rg}}) \leq (1 - e^{-\frac{(1-q)\mu}{c}})\mathrm{Tr}(\bar{\mathbf{\Sigma}}_O) + e^{-\frac{(1-q)\mu}{c}}\mathrm{Tr}(\mathbf{P}). \tag{17}$$

Indeed, in simulation studies (see Section 4) we empirically verify the results of Theorems 1 and 2 and illustrate that Algorithm 2 performs favorably compared to the competing schemes both on average and for each individual sampling task. Before moving on to these numerical studies, in Theorem 3 we show that the maximum element-wise curvature of $f(S) = \mathrm{Tr}(\mathbf{P} - \bar{\mathbf{\Sigma}}_S)$ is bounded, even for non-stationary graph signals.

**Theorem 3.** *Let $\mathcal{C}_{max}$ be the maximum element-wise curvature of $f(S) = \mathrm{Tr}(\mathbf{P} - \bar{\mathbf{\Sigma}}_S)$. Then, it holds that*

$$\mathcal{C}_{\max} \leq \frac{\lambda_{\max}^2(\mathbf{P})}{\lambda_{\min}^2(\mathbf{P})} \left(1 + \frac{\lambda_{\max}(\mathbf{P})}{\sigma^2}\right)^3. \tag{18}$$

An implication of Theorem 3 is a generalization of a result in [10] for stationary signals. There, it has been shown that if $\mathbf{x}$ is stationary and $\mathbf{P} = \sigma_{\mathbf{x}}^2 \mathbf{I}_N$ for some $\sigma_{\mathbf{x}}^2 > 0$, then the curvature of the MSE objective is bounded. However, Theorem 3 holds even in the scenarios where the signal is non-stationary and $\mathbf{P}$ is non-diagonal.

## 4. NUMERICAL SIMULATIONS

We study the recovery of simulated noisy signals supported on synthetic and real-world graphs to assess performance of the proposed sampling algorithms in terms of MSE and running time. To this end, we first consider an undirected Erdős-Rényi random graph $\mathcal{G}$ of size $N = 100$ and edge probability 0.2 [33]. Bandlimited graph

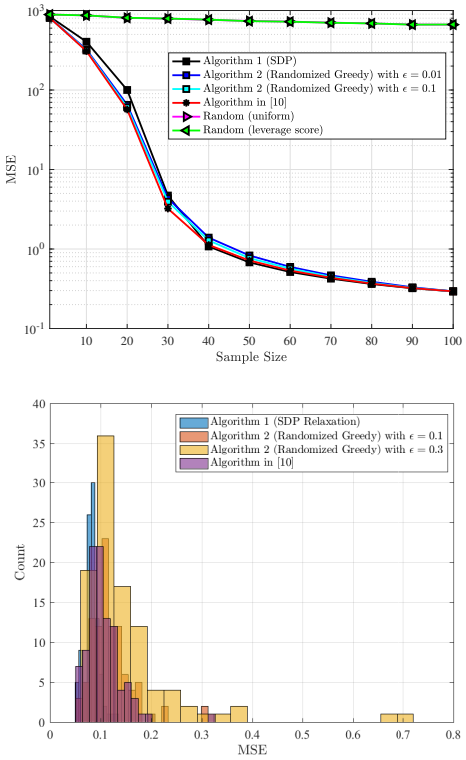


Fig. 1: Erdős-Rényi graph. Comparison of different schemes in terms of (top) MSE as a function of the size of the sampling set; and (bottom) histogram of MSE values for 100 realizations and fixed sampling set size.

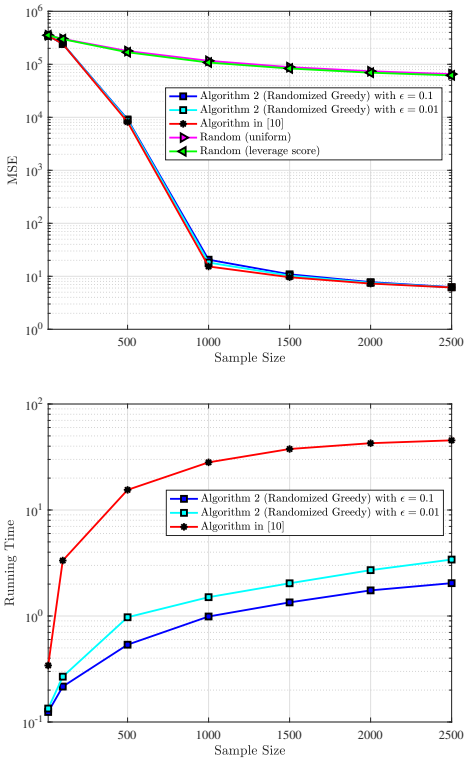


Fig. 2: Minnesota road network. (top) MSE and (bottom) running time comparison of different sampling schemes.

signals $\mathbf{x} = \mathbf{U}\bar{\mathbf{x}}_K$ are generated by taking $\mathbf{U}$ as the first $k = 30$ eigenvectors of the graph adjacency matrix. The non-zero frequency components $\bar{\mathbf{x}}_K$ are drawn from a zero-mean, multivariate Gaussian distribution with covariance matrix $\mathbf{P}$ which is selected uniformly at random from the set of positive semi-definite (PSD) matrices. Zero-mean Gaussian noise $\mathbf{n}$ with covariance $\sigma^2 = 10^{-2}\mathbf{I}_N$ is added to $\mathbf{x}$. Algorithms 1 and 2 are run to recover the signal for different sampling set sizes. We compare the MSE performance of the proposed schemes with the state-of-the-art greedy algorithm [10] and the random sampling approaches in [11]. For the randomized greedy algorithm we use $\epsilon = 0.1$ and $\epsilon = 0.01$. Fig. 1 (top) depicts the MSE versus $k$ (sample size), where the results are obtained by averaging over 100 Monte-Carlo simulations. As the figure indicates, Algorithms 1 and 2 outperform the random sampling schemes of [11] and perform nearly as well as the greedy sampling algorithm [10]. While not shown here for the sake of clarity of the presentation, similar patterns were also observed for other workhorse random graphs, e.g., preferential attachment and Barbási–Albert models [33].

Next, we study the performance of the proposed schemes for each individual sampling tasks (each Monte-Carlo realizations), for the setting where $N = 10$ and $k = 4$. Bandlimited graph signals are generated as before except that this time we take $\mathbf{U}$ as the first 4 eigenvectors of the adjacency matrix. Fig. 1 (bottom) depicts superimposed MSE histograms of Algorithms 1 and 2 as well as the greedy sampling scheme [10] for 100 realizations per method and fixed $|S| = 4$. As the figure illustrates, the proposed SDP relaxation and randomized greedy schemes perform well and are comparable with the greedy approach.

Finally, we test Algorithm 2 on the Minnesota road network[2] with $N = 2642$ nodes in order to showcase scalability of the proposed graph sampling method. To that end, Bandlimited graph signals are generated by taking the first $k = 600$ eigenvectors of the graph Laplacian matrix, where the non-zero frequency components are drawn from a zero-mean, multivariate Gaussian distribution with randomly chosen PSD covariance matrix $\mathbf{P}$. The signals are corrupted with additive white Gaussian noise with $\sigma^2 = 10^{-2}\mathbf{I}_N$. As expected, Figs. 2 (top) and (bottom) depict trends of decreasing MSE and increasing running time versus $|S|$, respectively. The results are averaged over 1000 Monte-Carlo simulations run on a commercial laptop with an Intel Core i7 processor at 3.1 GHz. Remarkably, the proposed randomized greedy procedure achieves an order-of-magnitude speedup over the state-of-the-art algorithm in [10] while showing only a marginal degradation in the MSE performance.

[2]https://sparse.tamu.edu/Gleich/minnesota

## 5. CONCLUSION

We considered the problem of sampling a bandlimited graph signal in the presence of noise, where the goal is to select a subset of graph nodes of prescribed cardinality to minimize the mean square signal reconstruction error. First, we developed an SDP relaxation method to find an approximate solution to the NP-hard sample-set selection task. Then the problem was reformulated as the maximization of a monotone weak submodular function, and a novel randomized-greedy algorithm was proposed to find a near-optimal sample subset. In addition, we analyzed the performance of the randomized greedy algorithm, and showed that the resulting MSE is a constant factor away from the optimal MSE. Unlike prior work, our guarantees do not require stationarity of the graph signal. Simulations studies showed that the proposed sampling algorithms compare favorably to competing alternatives in terms of accuracy and runtime.

## 6. REFERENCES


[1] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *Signal Processing Magazine, IEEE*, vol. 30, no. 3, pp. 83–98, 2013.

[2] Aliaksei Sandryhaila and José M. F. Moura, "Discrete signal processing on graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, 2013.

[3] Han Shomorony and A Salman Avestimehr, "Sampling large data on graphs," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*. IEEE, 2014, pp. 933–936.

[4] Mikhail Tsitsvero, Sergio Barbarossa, and Paolo Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[5] Aamir Anis, Akshay Gadde, and Antonio Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3775–3789, 2016.

[6] Siheng Chen, Rohan Varma, Aliaksei Sandryhaila, and Jelena Kovacevic, "Discrete signal processing on graphs: Sampling theory," *IEEE Transactions on Signal Processing*, vol. 24, no. 63, pp. 6510–6523, 2015.

[7] Sundeep Prabhakar Chepuri and Geert Leus, "Subsampling for graph power spectrum estimation," in *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2016 IEEE*. IEEE, 2016, pp. 1–5.

[8] Antonio G Marques, Santiago Segarra, Geert Leus, and Alejandro Ribeiro, "Sampling of graph signals with successive local aggregations," *IEEE Transactions on Signal Processing*, vol. 64, no. 7, pp. 1832–1843, 2016.

[9] Fernando Gama, Antonio G Marques, Gonzalo Mateos, and Alejandro Ribeiro, "Rethinking sketching as sampling: Linear transforms of graph signals," in *Signals, Systems and Computers, 2016 50th Asilomar Conference on*. IEEE, 2016, pp. 522–526.

[10] Luiz FO Chamon and Alejandro Ribeiro, "Greedy sampling of graph signals," *IEEE Transactions on Signal Processing*, 2017.

[11] Siheng Chen, Rohan Varma, Aarti Singh, and Jelena Kovacevic, "Signal recovery on graphs: Fundamental limits of sampling strategies," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 539–554, 2016.

[12] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro, "Stationary graph processes and spectral estimation," vol. 65, no. 22, pp. 5911–5926, Aug. 2017.

[13] N. Perraudin and P. Vandergheynst, "Stationary signal processing on graphs," vol. 65, no. 13, pp. 3462–3477, July 2017.

[14] B. Girault, "Stationary graph signals using an isometric graph translation," Aug 2015, pp. 1516–1520.

[15] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrak, and Andreas Krause, "Lazier than lazy greedy," in *AAAI Conference on Artificial Intelligence*. AAAI, 2015.

[16] Aliaksei Sandryhaila and José M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, June 2014.

[17] Joya A. Deri and José M. F. Moura, "Spectral projector-based graph fourier transforms," *J. Sel. Topics Signal Processing*, vol. 11, no. 6, pp. 785–795, 2017.

[18] Rasoul Shafipour, Ali Khodabakhsh, Gonzalo Mateos, and Evdokia Nikolova, "A digraph fourier transform with spread frequency components," in *Proc. of IEEE Global Conf. on Signal and Information Processing*, Nov 2017.

[19] Stefania Sardellitti, Sergio Barbarossa, and Paolo Di Lorenzo, "On the graph fourier transform for directed graphs," *J. Sel. Topics Signal Processing*, vol. 11, no. 6, pp. 796–811, 2017.

[20] Steven M Kay, *Fundamentals of statistical signal processing*, Prentice Hall PTR, 1993.

[21] Siddharth Joshi and Stephen Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.

[22] Manohar Shamaiah, Siddhartha Banerjee, and Haris Vikalo, "Greedy sensor selection under channel uncertainty," *IEEE Wireless Communications Letters*, vol. 1, no. 4, pp. 376–379, 2012.

[23] Abolfazl Hashemi, Mahsa Ghasemi, Haris Vikalo, and Ufuk Topcu, "A randomized greedy algorithm for near-optimal sensor scheduling in large-scale sensor networks," *arXiv preprint arXiv:1709.08823*, 2017.

[24] Abhimanyu Das and David Kempe, "Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*. Omnipress, 2011, pp. 1057–1064.

[25] Abolfazl Hashemi and Haris Vikalo, "Sparse linear regression via generalized orthogonal least-squares," in *Signal and Information Processing (GlobalSIP), 2016 IEEE Global Conference on*. IEEE, 2016, pp. 1305–1309.

[26] Roger A Horn and Charles R Johnson, *Matrix analysis*, Cambridge university press, 2012.

[27] Sanjeev Arora, Elad Hazan, and Satyen Kale, "Fast algorithms for approximate semidefinite programming using the multiplicative weights update method," in *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*. IEEE, 2005, pp. 339–348.

[28] Michael Grant, Stephen Boyd, and Yinyu Ye, "CVX: Matlab software for disciplined convex programming," 2008.

[29] Richard Bellman, *Introduction to matrix analysis*, SIAM, 1997.

[30] Leslie G Valiant, "A theory of the learnable," *Communications of the ACM*, vol. 27, no. 11, pp. 1134–1142, 1984.

[31] Leslie Valiant, "Probably approximately correct: Nature's algorithms for learning and prospering in a complex world," 2013.

[32] Joel A Tropp et al., "An introduction to matrix concentration inequalities," *Foundations and Trends® in Machine Learning*, vol. 8, no. 1-2, pp. 1–230, 2015.

[33] Mark Newman, *Networks: an introduction*, Oxford university press, 2010.