MSE-targeted Sampling of Bandlimited Graph Signals via Low-pass Graph Filtering Atoms

Fen Wang, Taihao Li, Minxiang Ye
Institute of Artificial Intelligence, Zhejiang Lab
Hangzhou, China
fenwang@zhejianglab.com

Gene Cheung
Department of EECS, York University
Toronto, Canada
genec@yorku.ca

***Abstract*—Sampling is a fundamental problem in graph signal processing, which selects a node subset to collect samples, so that data in remaining nodes can be well recovered. In this paper, we propose a lightweight sampling algorithm to minimize MSE approximately for bandlimited graph signals via low-pass graph filtering dictionary atoms. Specifically, first we derive a greedy sampling objective without matrix inverse, based on a proxy of the MMSE criterion. We then define dictionary atoms of low-pass graph filters to accelerate computation in the reformulated problem. To further reduce complexity, we reuse partial results computed in the previous greedy step, so that candidate solutions can be evaluated via simple vector-vector multiplications. Extensive experiments show that our proposed method required the least sampling time compared to other MSE-based methods and achieved the best MSE performance overall.**

***Index Terms*—Graph sampling, graph signal processing, greedy approach**

I. Introduction

In the last decade, *graph signal processing* (GSP) [1]–[3] becomes a hotspot research field that that studies signals residing on graphs, such as sensed data in wireless communication networks [4] and movie ratings on social networks [5]. Signal samples on connected node pairs of a similarity graph tend to be alike; this translates to graph signals being (approximately) *bandlimited* in the graph Fourier domain. Specifically, the bandlimited (BL) graph signal has energies only in low graph frequencies, which are the first few eigenvectors of a variation operator like graph adjacency matrix, combinatorial Laplacian matrix, or their variants [2].

When the observation resources are limited, *sampling* of an assumed BL graph signal—select a node subset to collect samples such that the sought BL signal at unsampled nodes can be well estimated[1]—is a fundamental problem in GSP [8]. In the noiseless case, BL graph signals can be recovered from partial samples without error by the *least square* (LS) method, if the corresponding partial eigenvector matrix is full column rank. Assuming that the signal is corrupted by additive independent and identically distributed (i.i.d.) noise, many graph sampling methods were proposed to *minimize the mean squared error* (MMSE) of the LS estimator.

Among MSE-based graph sampling schemes, [9] minimized the original MSE value directly, which is expensive computationally due to full eigen-decomposition (ED) and matrix inverse [10]. [11] leveraged graph *spectral proxies* (SP) to choose samples greedily using the first eigenvector[2] of a sub-matrix of a graph Laplacian raised to some power. [12] and [13] leveraged the Neumann series theorem to minimize a proxy to the MMSE criterion to mitigate matrix inverse, but still suffered from high complexity due to multiple matrix-matrix multiplications. None of the aforementioned graph sampling schemes are scalable to large graphs due to their heavy computation costs. Beyond MSE-based methods, there exist alternative low-complexity graph sampling approaches [14]–[16], but they tend to have sub-par MSE performance. Thus, a fast MSE-minimizing graph sampling scheme is still missing in the literature.

In this paper, we propose a MSE-targeted graph sampling method with low complexity leveraging *low-pass graph filter dictionary atoms* [17]—the ideal low-pass filtered outputs of impulse signals. Specifically, first, we define an equivalent sampling objective of a spectrum-shifted MMSE criterion for a sub-matrix of an ideal low-pass filter operator. Then, we define dictionary atoms of low-pass graph filters to accelerate computation of the reformulated problem. To reduce complexity, we reuse partial results computed in the previous greedy step to evaluate candidate solutions, resulting in simple vector-vector multiplications. Extensive experiments show that our proposed method had the least sampling time compared to existing MSE-based sampling schemes and achieved the best MSE performance.

II. Preliminaries

A combinatorial graph can be expressed as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ with N nodes in set $\mathcal{V} = \{1, \ldots, N\}$. The connected edge set is denoted by $\mathcal{E}$, *i.e.*, $\exists (i,j) \in \mathcal{E}$ iff nodes i and j are connected. Entry $w_{ij} > 0$ in *weight matrix* $\mathbf{W}$ is the edge weight of connected node pair (i,j) if $(i,j) \in \mathcal{E}$, and $w_{ij} = 0$ otherwise. Given weight matrix $\mathbf{W}$, we define a diagonal *degree matrix* $\mathbf{D}$, where $d_{ii} = \sum_j w_{ij}$. In this paper, we focus

This work was supported by the China Postdoctoral Science Foundation under Grant No.2021TQ0309, and by the Major Scientific Project of Zhejiang Lab under Grant No.2020KB0AC01.

Corresponding Author: Taihao Li (lith@zhejianglab.com)

[1]Sampling of graph signals can be divided into three categories: aggregation sampling [6], local measurement [7] and subset sampling [8]. In this paper, we focus on solving the subset sampling problem.

[2]We call the eigenvector corresponding to the smallest eigenvalue by *first eigenvector*.

on connected, undirected positive graphs with no self-loops, and we adopt the symmetric *combinatorial graph Laplacian matrix* $\mathbf{L} = \mathbf{D} - \mathbf{W}$ as the variation operator[3]. Because $\mathbf{L}$ is real, symmetric and *positive semi-definite* (PSD) for a positive graph [18], one can write the eigen-decomposition of $\mathbf{L}$ as $\mathbf{L} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^\top$, where $\mathbf{V} = [\mathbf{v}_1, ..., \mathbf{v}_N]$ is an orthonormal eigenvector matrix, with corresponding non-decreasing eigenvalues $0 = \lambda_1 < \lambda_2 \leq \lambda_3 \leq \ldots \leq \lambda_N$ along the diagonal entries of diagonal matrix $\mathbf{\Sigma}$.

The *graph Fourier transform* (GFT) [1] of a graph signal $\mathbf{x} \in \mathbb{R}^N$ is defined as its projected vector onto the eigenvector space of $\mathbf{L}$, *i.e.*, $\tilde{\mathbf{x}} = \mathbf{V}^\top \mathbf{x}$, and the inverse GFT is $\mathbf{x} = \mathbf{V}\tilde{\mathbf{x}}$. A K-BL graph signal $\mathbf{x}$ means that its GFT coefficients $\tilde{\mathbf{x}}$ are non-zero only for indices smaller than or equal to K, with bandwidth λ_K. A K-BL graph signal can be expressed as $\mathbf{x} = \mathbf{V}_K \tilde{\mathbf{x}}_K$, where $\mathbf{V}_K$ is the first K columns of $\mathbf{V}$, and $\tilde{\mathbf{x}}_K$ is the first K elements of $\tilde{\mathbf{x}}$. To formulate the graph sampling problem, we first define a sampling operator as follows [11].

***Definition** 1:* To sample M elements from $\mathbf{x}$, where $M \leq N$, and produce $\mathbf{x}_\mathcal{S} = \mathbf{C}\mathbf{x} \in \mathbb{R}^M$ with a sample set $\mathcal{S} \subseteq \mathcal{V}$ and $|\mathcal{S}| = M$, we define a binary *sampling matrix* $\mathbf{C} \in \{0,1\}^{M \times N}$ as

$$c_{ij} = \begin{cases} 1, & j = \mathcal{S}(i); \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

For noiseless BL graph signals, the sampled graph signal is $\mathbf{x}_\mathcal{S} = \mathbf{C}\mathbf{V}_K \tilde{\mathbf{x}}_K$. Here, the LS reconstruction can interpolate original signal without error, assuming that matrix $\mathbf{C}\mathbf{V}_K$ is full column rank. However, if the sampled signal is corrupted by additive noise, then different sample sets of the same size would result in different reconstruction MSE. Thus, we can minimize the resulting MSE by carefully choosing sampling matrix $\mathbf{C}$, as done in [12]:

$$\mathbf{C}^* = \underset{\mathbf{C} \in \mathcal{F}}{\arg\min}\, \mathrm{Tr}\left(\left[(\mathbf{C}\mathbf{V}_K)^\top \mathbf{C}\mathbf{V}_K \right]^{-1} \right), \tag{2}$$

where $\mathcal{F}$ is the set of sampling matrices defined in (1). The MMSE criterion (2) is also called the *A-optimality criterion* [19].

III. Lightweight MSE-based Graph Sampling

A. Shifted MMSE Criterion and Its Equivalence

Evaluating the objective in (2) for a candidate solution $\mathbf{C}$ requires matrix inverse. To reduce the evaluation complexity, in [13] the authors augmented the MMSE criterion using a constant spectrum shift:

$$\mathbf{C}^* = \underset{\mathbf{C} \in \mathcal{F}}{\arg\min}\, \mathrm{Tr}\left(\left[(\mathbf{C}\mathbf{V}_K)^\top \mathbf{C}\mathbf{V}_K + \mu\mathbf{I} \right]^{-1} \right), \tag{3}$$

where μ is a small weight (shift) parameter with $0 < \mu < 1$.

The authors then proved the following theorem:

***Proposition** 1:* The augmented objective (3) is equivalent to

$$\mathcal{S}^* = \underset{\mathcal{S}:|\mathcal{S}|=M}{\arg\min}\, \mathrm{Tr}\left[(\mathbf{V}_K \mathbf{V}_K^\top)_\mathcal{S} + \mu\mathbf{I} \right]^{-1}, \tag{4}$$

if the selected matrix $\mathbf{C}\mathbf{V}_K$ is full column rank.
See [13] for a proof.

Given that the sampling problem is combinatorial in nature, a greedy approach to optimally choose one sample at a time was adopted. Specifically, assuming that we have obtained set $\mathcal{S}_t$ after t iterations, to choose the $(t+1)$-th sample, we solve the following local optimization problem:

$$\min_{i \in \mathcal{S}_t^c} \quad \underbrace{\mathrm{Tr}\left[(\mathbf{V}_K \mathbf{V}_K^\top + \mu\mathbf{I})_{\mathcal{S}_t \cup \{i\}} \right]^{-1}}_{f(\mathcal{S}_t \cup \{i\})}. \tag{5}$$

Define $\mathbf{G} \triangleq \mathbf{V}_K \mathbf{V}_K^\top + \mu\mathbf{I}$ for notation simplicity. Equation (5) needs to compute inverse of $\mathbf{G}_{\mathcal{S}_t \cup \{i\}}$ for every possible candidate i in set $\mathcal{S}_t^c$. To reduce the computation cost of matrix inverse, the authors introduced the next *matrix inverse lemma* to compute value $\mathbf{G}_{\mathcal{S}_t \cup \{i\}}^{-1}$ for each i efficiently.

***Lemma** 1:* The inverse of matrix $\mathbf{M}$ can be computed using the inverse of sub-matrix $\mathbf{A}$ and the inverse of the Schur complement $\mathbf{H} \triangleq \mathbf{C} - \mathbf{V}\mathbf{A}^{-1}\mathbf{U}$ of sub-matrix $\mathbf{A}$ of matrix $\mathbf{M}$, *i.e.*, $\mathbf{M}/\mathbf{A}$, as

$$\begin{aligned} \mathbf{M}^{-1} &= \begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{V} & \mathbf{C} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{U}\mathbf{H}^{-1}\mathbf{V}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{U}\mathbf{H}^{-1} \\ -\mathbf{H}^{-1}\mathbf{V}\mathbf{A}^{-1} & \mathbf{H}^{-1} \end{bmatrix}. \end{aligned} \tag{6}$$

See [20] for a proof.

Note that matrix $\mathbf{G}$ is symmetric, and under some permutation, its sub-matrix $\mathbf{G}_{\mathcal{S}_t \cup \{i\}}$ can be expressed as

$$\mathbf{G}_{\mathcal{S}_t \cup \{i\}} = \begin{bmatrix} \mathbf{G}_{\mathcal{S}_t} & \mathbf{G}_{\mathcal{S}_t, \{i\}} \\ \mathbf{G}_{\{i\}, \mathcal{S}_t} & G_{ii} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{\mathcal{S}_t} & \mathbf{g}_{t,i} \\ \mathbf{g}_{t,i}^\top & G_{ii} \end{bmatrix}, \tag{7}$$

where $\mathbf{g}_{t,i}$ denotes the partial vector of i-th column of $\mathbf{G}$ indexed by $\mathcal{S}_t$, and G_{ii} is the i-th diagonal entry in $\mathbf{G}$.

From Lemma 1, for $t \geq 1$, the inverse of $\mathbf{G}_{\mathcal{S}_t \cup \{i\}}$ is

$$\mathbf{G}_{\mathcal{S}_t \cup \{i\}}^{-1} = \begin{bmatrix} \mathbf{G}_{\mathcal{S}_t}^{-1} + h_i^{-1}\mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{g}_{t,i}\mathbf{g}_{t,i}^\top\mathbf{G}_{\mathcal{S}_t}^{-1} & -h_i^{-1}\mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{g}_{t,i} \\ -h_i^{-1}\mathbf{g}_{t,i}^\top\mathbf{G}_{\mathcal{S}_t}^{-1} & h_i^{-1} \end{bmatrix} \tag{8}$$

where $h_i = G_{ii} - \mathbf{g}_{t,i}^\top \mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i}$ is a scalar.

Thus,

$$\begin{aligned} f(\mathcal{S}_t \cup \{i\}) &= \mathrm{Tr}\left(\mathbf{G}_{\mathcal{S}_t \cup \{i\}}^{-1} \right) \\ &= \mathrm{Tr}\left(\mathbf{G}_{\mathcal{S}_t}^{-1} \right) + h_i^{-1} \mathrm{Tr}\left(\mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i} \mathbf{g}_{t,i}^\top \mathbf{G}_{\mathcal{S}_t}^{-1} \right) + h_i^{-1} \\ &= f(\mathcal{S}_t) + h_i^{-1} \|\mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i}\|_2^2 + h_i^{-1}. \end{aligned} \tag{9}$$

Because $f(\mathcal{S}_t)$ is a constant not affected by the selection of candidate i, during the $(t+1)$-th greedy step, given input $\mathcal{S}_t$, the sampling problem (5) can be simplified to

$$\begin{aligned} \min_{i \in \mathcal{S}_t^c} \quad & g(i) = h_i^{-1} \|\mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i}\|_2^2 + h_i^{-1} \\ \text{s.t.} \quad & h_i = G_{ii} - \mathbf{g}_{t,i}^\top \mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i};\ \mathbf{g}_{t,i} = \mathbf{G}_{\mathcal{S}_t, \{i\}} \end{aligned}. \tag{10}$$

Compared to problem (5), instead of computing matrix inverse $\mathbf{G}_{\mathcal{S}_t \cup \{i\}}^{-1}$ for each candidate $i \in \mathcal{S}^c$, (10) just requires computing matrix-vector product $\mathbf{G}_{\mathcal{S}_t}^{-1} \mathbf{g}_{t,i}$ using known $\mathbf{G}_{\mathcal{S}_t}^{-1}$ with complexity $\mathcal{O}(M^2)$ to evaluate one candidate. In this

[3]Some existing methods [12], [14] are designed based on symmetric normalized graph Laplacian defined by $\mathbf{L}_n = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$. Our proposed methods are applicable for different graph variation operators.

paper, we circumvent the explicit computation of the low-pass filter matrix $\mathbf{G}$ to obtain $\mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{g}_{t,i}$, G_{ii} and $\mathbf{g}_{t,i}$ using low-pass graph filtering dictionary atoms.

B. Low-pass Graph Filtering Atoms (LPGFA) and Greedy Problem Reformulation

Denote by $\mathbf{d}_i$ by

$$\mathbf{d}_i = \mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i \tag{11}$$

where $\boldsymbol{\delta}_i$ is the i-th column of identity matrix $\mathbf{I}$, *i.e.*, the *impulse signal* with $\boldsymbol{\delta}_i(i) = 1$ and $\boldsymbol{\delta}_i(j) = 0, \forall j \neq i$. The atoms $\{\mathbf{d}_i\}_{i\in\mathcal{V}}$ are called LPGFA in this paper, which is a special case of atoms defined in [17]. The *local / squared graph coherence* at node i is defined by quantity $\|\mathbf{d}_i\|_2^2$ [15], [21]. Next, we propose a graph sampling algorithm to solve (10) using $\mathbf{d}_i$'s.

Given $\mathbf{d}_i$ defined in (11), we compute G_{ii} in (10) as

$$G_{ii} = \boldsymbol{\delta}_i^\top \mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i + \mu = \mathbf{d}_i(i) + \mu = \|\mathbf{d}_i\|_2^2 + \mu, \tag{12}$$

where $\mathbf{d}_i(i)$ is the i-th element in vector $\mathbf{d}_i$. The last equation holds since $\|\mathbf{d}_i\|_2^2 = (\mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i)^\top (\mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i) = \boldsymbol{\delta}_i^\top \mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i = \mathbf{d}_i(i)$.

Similarly, $\mathbf{g}_{t,i}$ can be computed as

$$\begin{aligned} \mathbf{g}_{t,i} &= \mathbf{G}_{\mathcal{S}_t,\{i\}} = \mathbf{C}_{\mathcal{S}_t}(\mathbf{V}_K \mathbf{V}_K^\top + \mu\mathbf{I})\boldsymbol{\delta}_i \\ &= \mathbf{d}_i(\mathcal{S}_t) + \mu \mathbf{C}_{\mathcal{S}_t}\boldsymbol{\delta}_i \end{aligned} \tag{13}$$

where $\mathbf{C}_{\mathcal{S}_t}$ is the sampling matrix corresponding to set $\mathcal{S}_t$, and $\mathbf{d}_i(\mathcal{S}_t)$ is a sub-vector of $\mathbf{d}_i$ specified by set $\mathcal{S}_t$.

Given $i \in \mathcal{S}_t^c$, we know that $\boldsymbol{\delta}_i(\mathcal{S}_t) = \mathbf{0}$. Hence,

$$\mu \mathbf{C}_{\mathcal{S}_t}\boldsymbol{\delta}_i = \mathbf{0}. \tag{14}$$

Thus, we can write

$$\mathbf{g}_{t,i} = \mathbf{d}_i(\mathcal{S}_t). \tag{15}$$

Using (12) to (15), we simplify the greedy optimization problem (10) to the following

$$\begin{aligned} \min_{i\in\mathcal{S}_t^c} \quad & h_i^{-1}\|\mathbf{r}_{t,i}\|_2^2 + h_i^{-1} \\ s.t. \quad & h_i = \|\mathbf{d}_i\|_2^2 + \mu - \mathbf{r}_{t,i}^\top \mathbf{d}_i(\mathcal{S}_t) \\ & \mathbf{r}_{t,i} = \mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{d}_i(\mathcal{S}_t) \\ & \mathbf{d}_i = \mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i \end{aligned} \tag{16}$$

To obtain $\mathbf{r}_{t,i}$, we need to compute **matrix-vector** multiplication. Next, we seek to obtain $\mathbf{r}_{t,i} = \mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{d}_i(\mathcal{S}_t)$ via simple **vector-vector** multiplications instead, reusing computed results in last greedy step.

C. Evaluation Complexity Reduction via Solution Reuse

Suppose that the optimal sample in step t is j_t^*. Then $\mathcal{S}_t = \mathcal{S}_{t-1} \cup \{j_t^*\}$. For candidate node in unsampled set $i \in \mathcal{S}_t^c$, we can write

$$\mathbf{d}_i(\mathcal{S}_t) = \begin{bmatrix} \mathbf{d}_i^\top(\mathcal{S}_{t-1}) & \mathbf{d}_i(j_t^*) \end{bmatrix}^\top \in \mathbb{R}^t. \tag{17}$$

Then, for $i \in \mathcal{S}_t^c$,

$$\begin{aligned} \mathbf{r}_{t,i} &= \mathbf{G}_{\mathcal{S}_t}^{-1}\mathbf{d}_i(\mathcal{S}_t) = \mathbf{G}_{\mathcal{S}_{t-1}\cup\{j_t^*\}}^{-1}\mathbf{d}_i(\mathcal{S}_t) \\ &= \begin{bmatrix} \mathbf{G}_{\mathcal{S}_{t-1}}^{-1} + h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*}\mathbf{r}_{t-1,j_t^*}^\top & -h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*} \\ -h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*}^\top & h_{j_t^*}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{d}_i(\mathcal{S}_{t-1}) \\ \mathbf{d}_i(j_t^*) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{r}_{t-1,i} + h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*}\mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_i(\mathcal{S}_{t-1}) - h_{j_t^*}^{-1}\mathbf{d}_i(j_t^*)\mathbf{r}_{t-1,j_t^*} \\ -h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_i(\mathcal{S}_{t-1}) + h_{j_t^*}^{-1}\mathbf{d}_i(j_t^*) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{r}_{t-1,i} + \alpha\mathbf{r}_{t-1,j_t^*} - \beta\mathbf{r}_{t-1,j_t^*} \\ -\alpha + \beta \end{bmatrix} \end{aligned} \tag{18}$$

where

$$h_{j_t^*} = G_{j_t^*,j_t^*} - \mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_{j_t^*}(\mathcal{S}_{t-1}) \tag{19}$$

$$\alpha = h_{j_t^*}^{-1}\mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_i(\mathcal{S}_{t-1}) \tag{20}$$

$$\beta = h_{j_t^*}^{-1}\mathbf{d}_i(j_t^*). \tag{21}$$

Note that $\mathbf{r}_{t-1,i}$ was computed in the last greedy step for all $i \in \mathcal{S}_{t-1}^c$. Hence, we can reuse them to obtain value $\mathbf{r}_{t,i}$. Specifically, given computed $\mathbf{r}_{t-1,i}$ and $\mathbf{r}_{t-1,j_t^*}$, we obtain $\mathbf{r}_{t,i}$ by computing vector-vector products $\mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_{j_t^*}(\mathcal{S}_{t-1})$ and $\mathbf{r}_{t-1,j_t^*}^\top \mathbf{d}_i(\mathcal{S}_{t-1})$. The greedy evaluation complexity is thus reduced from $\mathcal{O}(M^2)$ to $\mathcal{O}(M)$ for each candidate i.

However, to evaluate all candidates in set $\mathcal{S}_t^c$, we must compute $\mathbf{d}_i$'s for all $i \in \mathcal{V}$. Next, we leverage a strategy in [15] to avoid computing all $\mathbf{d}_i$'s.

It is known that

$$\mathbf{d}_i(j) = \boldsymbol{\delta}_j^\top (\mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_i) = \boldsymbol{\delta}_i^\top (\mathbf{V}_K \mathbf{V}_K^\top \boldsymbol{\delta}_j) = \mathbf{d}_j(i). \tag{22}$$

Thus, to obtain $\mathbf{d}_i(\mathcal{S}_t)$ in equation (16), we compute only

$$\begin{aligned} \mathbf{d}_i(\mathcal{S}_t) &= \begin{bmatrix} \mathbf{d}_i(\mathcal{S}_t(1)) & \mathbf{d}_i(\mathcal{S}_t(2)) & \cdots & \mathbf{d}_i(\mathcal{S}_t(t)) \end{bmatrix}^\top \\ &= \begin{bmatrix} \mathbf{d}_{\mathcal{S}_t(1)}(i) & \mathbf{d}_{\mathcal{S}_t(2)}(i) & \cdots & \mathbf{d}_{\mathcal{S}_t(t)}(i) \end{bmatrix}^\top \\ &\triangleq \mathbf{d}_{\mathcal{S}_t}(i) \in \mathbb{R}^t \end{aligned} \tag{23}$$

which consists of the i-th value of vectors $\mathbf{d}_{\mathcal{S}_t(k)}, k = 1, \cdots, t$. $\mathbf{d}_{\mathcal{S}_t(k)}$'s are LPGFAs on nodes in sampled set $\mathcal{S}_t$. Thus, we compute at most M LPGFAs rather than N to obtain $\mathbf{d}_i(\mathcal{S}_t)$, where in practice sample size $M \ll N$. At step $t+1$, we only need to compute one new LPGFA $\mathbf{d}_{j^*}$. Combined results in (18) and (23) with problem (16), we obtain the optimal greedy solution via five vector-vector multiplications and at most M LPGFAs.

D. Algorithm and Complexity Analysis

The procedure for problem (10) is illustrated in Algorithm 1, called LPGFA-based graph sampling. The subscript t is abbreviated for clarity. When $\mathcal{S} = \emptyset$, to find the first node, we need to solve the next problem based on (5):

$$\min_{i\in\mathcal{V}} \quad \mathrm{Tr}\left[(\mathbf{V}_K \mathbf{V}_K^\top + \mu\mathbf{I})_{ii}\right]^{-1} = \frac{1}{G_{ii}} \tag{24}$$

which has the same optimal solution as

$$\max_{i\in\mathcal{V}} \quad G_{ii} = \|\mathbf{d}_i\|_2^2 + \mu \tag{25}$$

based on equation (12).

Algorithm 1 Low-pass graph filtering atoms (LPGFA)-based sampling of bandlimited graph signals

Input: $\mathbf{L}$, $\mathcal{S} = \emptyset$ and μ

Compute squared graph coherence $\|\mathbf{d}_i\|_2^2$, $\forall i \in \mathcal{V}$
Select the first node by $i^* = \text{argmax}_{i\in\mathcal{V}} \|\mathbf{d}_i\|_2^2$
Update $\mathcal{S} \leftarrow \mathcal{S} \cup \{i^*\}$
While $|\mathcal{S}| < M$ $\quad \triangleright \mathcal{O}(M)$
    Compute one LPGFA $\mathbf{d}_{i^*}$
    $\forall i \in \mathcal{S}^c$, compute $\quad \triangleright \mathcal{O}(N)$
    If $|\mathcal{S}| = 1$
        $\mathbf{r}_i = \frac{1}{\|\mathbf{d}_{i^*}\|_2^2 + \mu} \mathbf{d}_{i^*}(i)$
    else
        $\alpha = h_{i^*}^{-1} \mathbf{r}_{i^*}^\top \mathbf{d}_{\mathcal{S}\setminus\{i^*\}}(i)$ $\quad \triangleright \mathcal{O}(M)$
        $\beta = h_{i^*}^{-1} \mathbf{d}_{i^*}(i)$
        $\mathbf{r}_i = \begin{bmatrix} \mathbf{r}_i + \alpha \mathbf{r}_{i^*} - \beta \mathbf{r}_{i^*} \\ -\alpha + \beta \end{bmatrix}$
    end If
    Compute $h_i = \|\mathbf{d}_i\|_2^2 + \mu - \mathbf{r}_i^\top \mathbf{d}_{\mathcal{S}}(i)$
    Select $i^* = \arg\min_{i\in\mathcal{S}^c} h_i^{-1} \|\mathbf{r}_i\|_2^2 + h_i^{-1}$
    Update $\mathcal{S} \leftarrow \mathcal{S} \cup \{i^*\}$
end While
Return $\mathcal{S}$

TABLE I
IMPACT OF μ ON RECONSTRUCTION MSE OF THE PROPOSED METHOD

μ	0.02	0.06	0.10	0.14	0.18	0.22	0.26
$\mathcal{G}_1$	**19.1**	19.4	19.9	21.4	21.1	22.6	129
$\mathcal{G}_2$	**20.3**	20.7	21.3	21.9	23.0	51.2	1.1e3
$\mathcal{G}_3$	**22.0**	22.1	22.7	23.2	24.1	24.9	57.6

After one node is sampled, *i.e.*, $\mathcal{S} = \{i^*\}$, there is no memorized $\mathbf{r}_i$ for greedy update using (18). Based on equation (16), for $i \in \mathcal{S}^c$,

$$\mathbf{r}_i = G_{i^*i^*}^{-1} \mathbf{d}_i(i^*) = \frac{\mathbf{d}_{i^*}(i)}{\|\mathbf{d}_{i^*}\|_2^2 + \mu} \tag{26}$$

which is easy to compute. When the sample size $|\mathcal{S}| \geq 2$, $\mathbf{r}_i$ can be greedily updated using equation (18).

Algorithm 1 has three parts of complexities: (1) squared graph coherence $\|\mathbf{d}_i\|_2^2$ for all nodes $i \in \mathcal{V}$; (2) LPGFA $\mathbf{d}_i$ on sampled nodes $i \in \mathcal{S}$, and (3) sample selection. For sample selection, from Algorithm 1, we evaluate all candidates in set $\mathcal{S}^c$, whose size is at most N. For each evaluation, we need to compute vector-vector multiplication $\mathbf{r}_i^\top \mathbf{d}_{\mathcal{S}}(i)$ with complexity $\mathcal{O}(M)$. Combined with sample budget M, the sample selection complexity is $\mathcal{O}(NM^2)$. This is lower than the method proposed in [13], which is $\mathcal{O}(NM^3)$ during sampling.

IV. EXPERIMENTAL RESULTS

We evaluate the efficacy of our proposed LPGFA sampling scheme for BL graph signals via extensive simulations. All experiments were performed in MATLAB R2019a, running on a desktop with Intel Core i7-9700K CPU with 3.6 GHz CPU and 64 GB RAM. First, we generated three graphs using code in [22], with parameters listed as follows:
$\mathcal{G}_1$: community graph with $N = 1000$ nodes and 10 communities. Edge weights are computed based on generated coordinates $\mathbf{c}_i$ with function $w_{ij} = \exp\left(-(\mathbf{c}_i - \mathbf{c}_j)^2\right)$;
$\mathcal{G}_2$: random sensor graph with 1000 nodes;
$\mathcal{G}_3$: hyper-cube graph with 1002 nodes in three dimensions.

The graph signals were assumed to be BL with bandwidth $K = 50$ on the eigen-space of those graphs. Their first K GFT coefficients were generated randomly using distribution $\mathcal{N}(1, 0.5^2)$, and the remaining GFT coefficients were set to be zeros. Graph signals were then computed via inverse GFT of the generated spectral coefficients. Gaussian noise was added into graph signals in the vertex domain based on different signal to noise ratios (SNRs).

First, we compared our proposed methods with five other deterministic MSE-targeted graph sampling algorithms on $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}_3$: A-optimal [9], E-optimal [23], spectral proxies [11], GFS [13] and MIA [12]. We did not simulate other low-complexity methods since their goal is not targeting on MSE performance. We used combinatorial graph Laplacian matrix ($\mathbf{L} = \mathbf{D} - \mathbf{W}$) as the variation operator for all simulated methods. The parameters of those approaches were the same as that used in [13]. The shift parameter μ was set to 0.01 for the proposed LPGFA sampling. In this paper, we choose explicit ED to obtain the exact $\mathbf{d}_i$'s.

Average reconstruction MSE and execution time as functions of sample size are shown in Fig. 1, where SNR is 0dB. In this figure, the proposed LPGFA had almost the same performance as the A-optimal sampling, which optimized the original A-optimality problem using greedy approach directly, considered as the performance benchmark. Further, compared with all simulated MSE-targeted methods, our method LPGFA had the least sampling time. Based on the above analysis, we conclude that among MSE-based methods, the speed superiority of our proposed method was significant, and it also achieves the best performance. We conducted experiments on $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}_3$ with different μ to investigate its impact on the MSE value of LPGFA method, where the sample size is 100 and SNR is 0dB. The simulation results are shown in Table. I, which indicates that the proposed LPGFA has the best performance when μ is small.

V. CONCLUSION

In this paper, we proposed a low-pass graph filtering atoms (LPGFA)-based sampling method for bandlimited graph signals targeting the MMSE criterion. We first introduced a spectrum shifted MMSE criterion and its equivalent version, which is a function of a submatrix of an ideal low-pass filter. Then, we avoided inverse computation of a greedily enlarged submatrix by using a matrix inverse lemma. We defined graph filtering atoms—the low-pass filtered impulse signal—to reformulate the problem for acceleration. For fast sampling, we reused results from the previous greedy step to speed up computation of candidates' scores. Experiments had

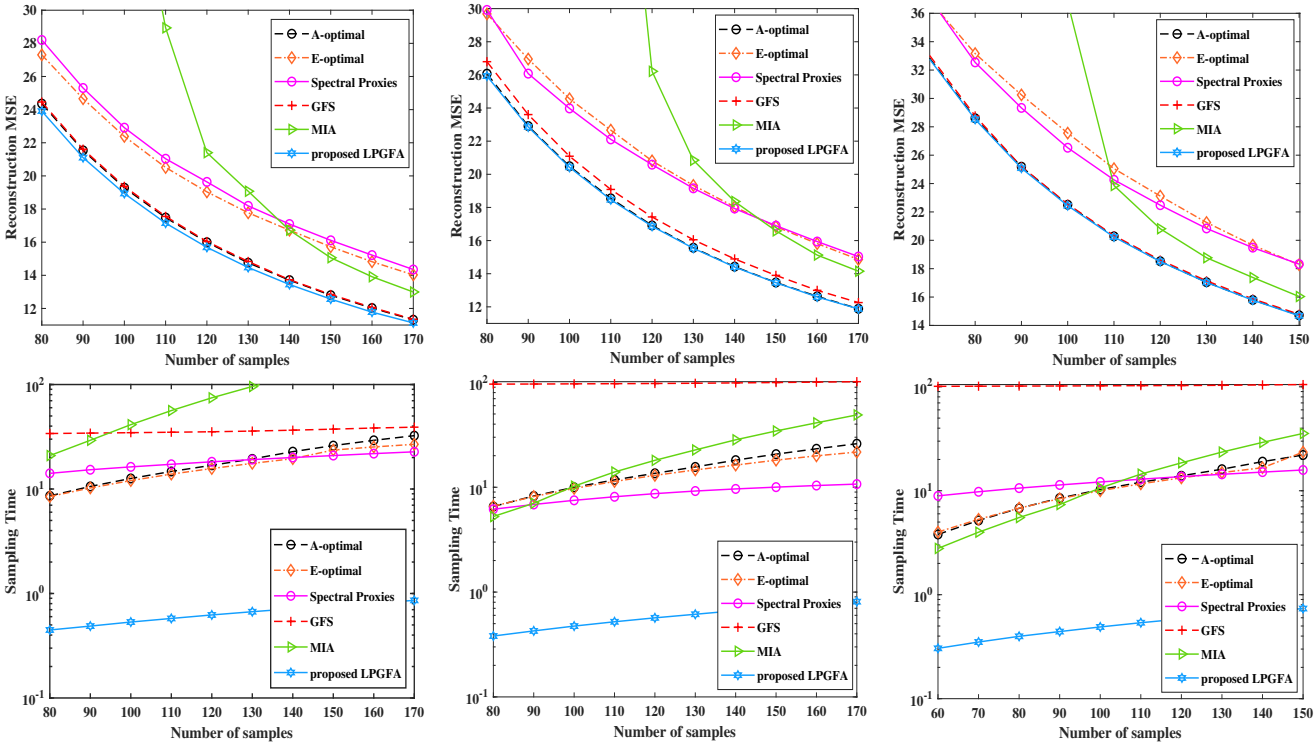

Fig. 1. Experimental results of different MMSE-based graph sampling algorithms in terms of sample size. From left to right: $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}_3$.

validated the efficiency of our proposed method in both speed and MSE performance.

REFERENCES

[1] A. Ortega, P. Frossard, J. Kovacević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[2] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, 2014.

[3] G. Cheung, E. Magli, Y. Tanaka, and M. K. Ng, "Graph spectral image processing," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 907–930, 2018.

[4] A. Sakiyama, Y. Tanaka, T. Tanaka, and A. Ortega, "Efficient sensor position selection using graph signal sampling theory," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 6225–6229.

[5] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 3697–3707.

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

[7] X. Wang, J. Chen, and Y. Gu, "Local measurement and reconstruction for noisy bandlimited graph signals," *Signal Processing*, vol. 129, pp. 119–129, 2016.

[8] Y. Tanaka, Y. C. Eldar, A. Ortega, and G. Cheung, "Sampling signals on graphs: From theory to applications," *IEEE Signal Processing Magazine*, vol. 37, no. 6, pp. 14–30, 2020.

[9] M. Tsitsvero, S. Barbarossa, and P. Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[10] L. F. Chamon and A. Ribeiro, "Greedy sampling of graph signals," *IEEE Transactions on Signal Processing*, vol. 66, no. 1, pp. 34–47, 2018.

[11] A. Anis, A. Gadde, and A. 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.

[12] F. Wang, Y. Wang, and G. Cheung, "A-optimal sampling and robust reconstruction for graph signals via truncated Neumann series," *IEEE Signal Processing Letters*, vol. 25, no. 5, pp. 680–684, 2018.

[13] F. Wang, G. Cheung, and Y. Wang, "Low-complexity graph sampling with noise and signal reconstruction via Neumann series," *IEEE Transactions on Signal Processing*, vol. 67, no. 21, pp. 5511–5526, 2019.

[14] A. Sakiyama, Y. Tanaka, T. Tanaka, and A. Ortega, "Eigendecomposition-free sampling set selection for graph signals," *IEEE Transactions on Signal Processing*, 2019.

[15] A. Jayawant and A. Ortega, "Practical graph signal sampling with log-linear size scaling," *Signal Processing*, vol. 194, p. 108436, 2022.

[16] Y. Bai, F. Wang, G. Cheung, Y. Nakatsukasa, and W. Gao, "Fast graph sampling set selection using gershgorin disc alignment," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2419–2434, 2020.

[17] D. I. Shuman, "Localized spectral graph filter frames: A unifying framework, survey of design considerations, and numerical comparison," *IEEE Signal Processing Magazine*, vol. 37, no. 6, pp. 43–63, 2020.

[18] G. Cheung, E. Magli, Y. Tanaka, and M. K. Ng, "Graph spectral image processing," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 907–930, 2018.

[19] F. Pukelsheim, *Optimal design of experiments*. siam, 1993, vol. 50.

[20] W. Hager, "Updating the inverse of a matrix," *SIAM Review*, vol. 31, no. 2, pp. 221–239, 1989.

[21] G. Puy, N. Tremblay, R. Gribonval, and P. Vandergheynst, "Random sampling of bandlimited signals on graphs," *Applied and Computational Harmonic Analysis*, vol. 44, no. 2, pp. 446–475, 2018.

[22] N. Perraudin, J. Paratte, D. Shuman, L. Martin, V. Kalofolias, P. Vandergheynst, and D. K. Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv preprint arXiv:1408.5781*, 2014.

[23] S. Chen, R. Varma, A. Sandryhaila, and J. Kovacević, "Discrete signal processing on graphs: Sampling theory," *IEEE Transactions on Signal Processing*, vol. 63, no. 24, pp. 6510–6523, 2015.