LINEAR-TIME SAMPLING ON SIGNED GRAPHS VIA GERSHGORIN DISC PERFECT ALIGNMENT

Chinthaka Dinesh*,† *Saghar Bagheri** *Gene Cheung** *Ivan V. Bajić*†

* York University, Toronto, Canada † Simon Fraser University, Burnaby, Canada

ABSTRACT

In graph signal processing (GSP), an appropriate underlying graph encodes pairwise (anti-)correlations of targeted discrete signals as edge weights. However, existing fast graph sampling schemes are designed and tested for positive graphs describing only positive correlations. In this paper, we show that for datasets with inherent strong anti-correlations, a suitable graph structure is instead a signed graph with both positive and negative edge weights, and in response, we propose a linear-time signed graph sampling method. Specifically, given an empirical covariance data matrix $\bar{\mathbf{C}}$, we first employ graphical lasso to learn a sparse inverse matrix $\mathcal{L}$, interpreted as a generalized graph Laplacian for signed graph $\mathcal{G}$. We then propose a fast signed graph sampling scheme containing three steps: i) augment $\mathcal{G}$ to a balanced graph $\mathcal{G}_B$, ii) align all Gershgorin disc left-ends of corresponding Laplacian $\mathcal{L}_B$ at smallest eigenvalue $\lambda_{\min}(\mathcal{L}_B)$ via similarity transform $\mathcal{L}_p = \mathbf{S}\mathcal{L}_B\mathbf{S}^{-1}$, leveraging a recent linear algebra theorem called Gershgorin disc perfect alignment (GDPA), and iii) perform sampling on $\mathcal{L}_p$ using a previous fast Gershgorin disc alignment sampling scheme (GDAS). Experimental results show that our signed graph sampling method outperformed existing fast sampling schemes noticeably on two political voting datasets.

Index Terms— Graph sampling, Gershgorin circle theorem

1. INTRODUCTION

A central premise in *graph signal processing* (GSP) [1, 2]—a growing field studying discrete signals on combinatorial graphs—is that an underlying graph captures inter-node (anti-)correlations or (dis)similarities in the data as edge weights. Given a meaningful graph encoded with pairwise signal statistics, a myriad of graph spectral processing can thus take place: compression, denoising, dequantization, etc. [3–5]. In particular, *graph sampling*[1] [6] addresses the problem of choosing a subset of nodes to collect samples, so that the entire signal can be reconstructed in high fidelity under a signal smoothness / bandlimited assumption. Among the numerous sampling methods are fast *eigen-decomposition-free* (EDF) schemes that mitigate explicit computation of frequency-defining eigenvectors of large graph Laplacian matrices [7–10]. Various mathematical tools were employed to realize the EDF property: spectral proxies (SP) [7], Neumann series (NS) [8], localization operator (LP) [9], and Gershgorin disc alignment sampling (GDAS) [10]. Common among these EDF methods is that they are designed and tested for *positive* graphs capturing positive inter-node correlations only.

However, there exist practical real-world datasets with inherent strong *anti-correlations* in the signals. For example, voting records in the Canadian Parliament often show opposite positions between the left-leaning Liberal and right-leaning Conservative parties[2]. For such datasets, a most suitable graph structure is instead a *signed* graph with both positive and negative edge weights [11, 12]. In response, in this paper[3] we propose a linear-time signed graph sampling method, the first to do so in the graph sampling literature.

Specifically, given an empirical covariance matrix $\bar{\mathbf{C}}$ computed from data, we first employ *graphical lasso* (GLASSO) [15] to learn a sparse inverse matrix $\mathcal{L}$, interpreted as a generalized graph Laplacian to a signed graph $\mathcal{G}$. We then propose a fast signed graph sampling scheme composed of three steps. First, we approximate $\mathcal{G}$ with a *balanced* graph[4] $\mathcal{G}_B$ [16], with corresponding generalized graph Laplacian $\mathcal{L}_B$, via edge weight augmentation in linear time. Second, we leverage a recent linear algebra theorem called *Gershgorin Disc Perfect Alignment* (GDPA) [17], stating that Gershgorin disc left-ends [18] of a generalized graph Laplacian matrix $\mathbf{M}$ for a balanced graph can be exactly aligned at the smallest eigenvalue $\lambda_{\min}(\mathbf{M})$ via a similarity transform $\mathbf{S}\mathbf{M}\mathbf{S}^{-1}$, where $\mathbf{S} = \text{diag}(v_1^{-1}, \ldots, v_N^{-1})$ and $\mathbf{v} = [v_1 \ldots v_N]^{\top}$ is the first eigenvector of $\mathbf{M}$. Given GDPA, in step two we compute $\mathcal{L}_p = \mathbf{S}\mathcal{L}_B\mathbf{S}^{-1}$, so that Gershgorin disc left-ends of $\mathcal{L}_p$ are perfectly aligned at $\lambda_{\min}(\mathcal{L}_B)$. Finally, in step three we perform sampling on $\mathcal{L}_p$ using a previous fast GDAS sampling scheme [10] that has roughly linear-time complexity. Experimental results show that our signed graph sampling method outperformed existing fast EDF sampling schemes [7–10] on two political voting datasets.

2. PRELIMINARIES

Graph Definitions: A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ is defined by a set of N nodes $\mathcal{V} = \{1, \ldots, N\}$, edges $\mathcal{E} = \{(i, j)\}$, and an *adjacency matrix* $\mathbf{W}$. $W_{i,j} \in \mathbb{R}$ is the edge weight if $(i, j) \in \mathcal{E}$, and $W_{i,j} = 0$ otherwise. Self-loops may exist, in which case $W_{i,i} \in \mathbb{R}$ is the weight of the self-loop for node i. *Degree matrix* $\mathbf{D}$ has diagonal entries $D_{i,i} = \sum_j W_{i,j}, \forall i$. A *combinatorial graph Laplacian matrix* $\mathbf{L}$ is defined as $\mathbf{L} \triangleq \mathbf{D} - \mathbf{W}$ [2]. If self-loops exist, then the *generalized graph Laplacian matrix* $\mathcal{L}$ is $\mathcal{L} \triangleq \mathbf{D} - \mathbf{W} + \text{diag}(\mathbf{W})$.

A graph signal $\mathbf{x} \in \mathbb{R}^N$ is smooth with respect to (w.r.t.) graph $\mathcal{G}$ if its *graph Laplacian regularizer* (GLR), $\mathbf{x}^{\top}\mathcal{L}\mathbf{x}$, is small [4]:

$$\mathbf{x}^{\top}\mathcal{L}\mathbf{x} = \sum_{(i,j)\in\mathcal{E}} W_{i,j}(x_i - x_j)^2 + \sum_{i\in\mathcal{V}} W_{i,i}x_i^2. \tag{1}$$

Gene Cheung acknowledges the support of the NSERC grants RGPIN-2019-06271, RGPAS-2019-00110. Ivan V. Bajić acknowledges the support of the NSERC grants RGPIN-2021-02485, RGPAS-2021-00038.

[1] Graph sampling can be divided into aggregation sampling, local measurements, and node subset selection. We focus on the last category here.

[2] https://www.ourcommons.ca/members/en/votes

[3] An earlier variant of our sampling scheme was proposed in [13] specifically for 3D point clouds. We compare our new proposal against [13] in Section 5. A longer version of this paper specifically for 3D point cloud sampling is recently accepted in IEEE TPAMI [14].

[4] A balanced graph has no cycles of odd number of negative edges.

GLR is commonly used to regularize ill-posed inverse problems such as denoising, dequantization and interpolation [4,5,19].

Gershgorin Circle Theorem: Given a real symmetric matrix $\mathbf{M}$, corresponding to each row i is a *Gershgorin disc* i with center $c_i \triangleq M_{i,i}$ and radius $r_i \triangleq \sum_{j \neq i} |M_{i,j}|$. A corollary of *Gershgorin Circle Theorem* (GCT) [18] is that the smallest Gershgorin disc left-end $\lambda^-_{\min}(\mathbf{M})$ is a lower bound of the smallest eigenvalue $\lambda_{\min}(\mathbf{M})$ of $\mathbf{M}$, *i.e.*,

$$\lambda^-_{\min}(\mathbf{M}) \triangleq \min_i c_i - r_i \leq \lambda_{\min}(\mathbf{M}). \tag{2}$$

Note that a *similarity transform* $\mathbf{SMS}^{-1}$ for an invertible matrix $\mathbf{S}$ has the same set of eigenvalues as $\mathbf{M}$. Thus, a GCT lower bound for $\mathbf{SMS}^{-1}$ is also a lower bound for $\mathbf{M}$, *i.e.*, for any invertible $\mathbf{S}$,

$$\lambda^-_{\min}(\mathbf{SMS}^{-1}) \leq \lambda_{\min}(\mathbf{M}). \tag{3}$$

3. SIGNED GRAPH LEARNING

We use two frameworks where graph learning is formulated as the estimation of different types of graph Laplacian matrices from data.

To estimate a *signed* (generalized) graph Laplacian matrix, given an empirical covariance matrix $\bar{\mathbf{C}}$ computed from data, we use the following GLASSO formulation [15,20]:

$$\min_{\mathcal{L}} \;\; \text{Tr}(\mathcal{L}\bar{\mathbf{C}}) - \log\det\mathcal{L} + \rho \, \|\mathcal{L}\|_1 \tag{4}$$

where $\rho > 0$ is a shrinkage parameter for the l_1-norm term. We solve (4) using a variant of the *block Coordinate descent* (BCD) algorithm in [21]. Specifically, we solve its dual iteratively by updating one row / column of dual variable $\mathbf{C} = \mathcal{L}^{-1}$ till convergence. Here, by construction, $\mathcal{L}$ is a *positive definite* (PD) matrix.

To estimate a combinatorial graph Laplacian matrix corresponding to a *positive* graph, we use the formulation in [22]. Specifically, we formulate the following optimization

$$\min_{\mathbf{L}} \;\; \text{Tr}(\mathbf{LK}) - \log|\mathbf{L}| \;\; \text{subject to} \;\; \mathbf{L} \in \mathcal{L}_c(\mathbf{A}) \tag{5}$$

where $\mathbf{K} = \bar{\mathbf{C}} + \mathbf{F}$, $\mathbf{F}$ is the regularization matrix defined as $\mathbf{F} = \rho(2\mathbf{I} - \mathbf{1}\mathbf{1}^\top)$, $\mathbf{A}$ is the connectivity matrix, and the set of constraints $\mathcal{L}_c(\mathbf{A})$ restricts $\mathbf{L}$ to a combinatorial graph Laplacian matrix. Since $\mathbf{L}$ is a *positive semi-definite* (PSD) matrix with eigenvalue 0, $|\mathbf{L}|$ is the pseudo-determinant. Given that the graph connectivity is unknown, $\mathbf{A}$ is set to represent a fully connected graph, *i.e.*, $\mathbf{A} = \mathbf{A}_{\text{full}} = \mathbf{1}\mathbf{1}^\top - \mathbf{I}$. See [22] for details on how (5) is computed.

4. SIGNED GRAPH SAMPLING

4.1. Signal Reconstruction

We first define a sampling objective to choose a set of samples minimizing a worst-case reconstruction error. Assume first that we reconstruct signal $\mathbf{x}^* \in \mathbb{R}^N$ from samples $\mathbf{y} \in \mathbb{R}^M$, $M < N$, by solving the following optimization regularized using GLR in (1):

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{Hx} - \mathbf{y}\|_2^2 + \mu\mathbf{x}^\top\mathcal{L}\mathbf{x} \tag{6}$$

where $\mathbf{H} \in \{0,1\}^{M \times N}$ is the sampling matrix [8] that selects M samples $\mathbf{y}$ from length-N signal $\mathbf{x}$, and $\mu > 0$ is a weight parameter. Solution $\mathbf{x}^*$ can be obtained by solving the following linear system:

$$\underbrace{\left(\mathbf{H}^\top\mathbf{H} + \mu\mathcal{L}\right)}_{\mathbf{B}} \mathbf{x}^* = \mathbf{H}^\top\mathbf{y} \tag{7}$$

where $\mathbf{B}$ is provably PD since $\mathbf{H}^\top\mathbf{H}$ is PSD and $\mathcal{L}$ is PD.

4.2. Objective Definition

Our sampling objective is to minimize the condition number—ratio of largest to smallest eigenvalues $\rho = \lambda_{\max}(\mathbf{B})/\lambda_{\min}(\mathbf{B})$—via selection of $\mathbf{H}$ to optimize the stability of the linear system (7). Since $\lambda_{\max}(\mathbf{B})$ can be easily upper-bounded [10], we focus on maximizing $\lambda_{\min}(\mathbf{B})$ instead:

$$\max_{\mathbf{H} \,|\, \text{Tr}(\mathbf{H}^\top\mathbf{H}) \leq M} \lambda_{\min}(\mathbf{B}). \tag{8}$$

Maximizing $\lambda_{\min}(\mathbf{B})$—also called *E-optimality* in optimal design [23–25]—is equivalent to minimizing the worst-case signal reconstruction error between a solution of (7) and the ground-truth signal [10].

4.3. GDAS Sampling for Positive Graphs

We first overview GDAS sampling for positive graph sampling [10]. Assuming $\mathcal{L}$ is a combinatorial graph Laplacian for a positive graph $\mathcal{G}$ without self-loops, all Gershgorin disc left-ends of $\mathcal{L}$ are at 0 exactly, *i.e.*, $c_i - r_i = D_{i,i} - \sum_{j \neq i} W_{i,j} = 0, \forall i$. In a nutshell, GDAS maximizes GCT lower bound $\lambda^-_{\min}(\mathbf{C})$ of a similar-transformed matrix $\mathbf{C} = \mathbf{SBS}^{-1}$ of $\mathbf{B}$, *i.e.*, $\lambda^-_{\min}(\mathbf{C}) \leq \lambda_{\min}(\mathbf{C}) = \lambda_{\min}(\mathbf{B})$. $\mathbf{S} = \text{diag}(s_1, \ldots, s_N)$ is a diagonal matrix with scalars $s_i > 0$ on its diagonal. GDAS maximizes the smallest disc left-end of $\mathbf{C}$ via two operations:

1. *Disc Shifting*: sample a node i in the graph means that the (i,i)-th entry of $\mathbf{H}^\top\mathbf{H}$ in $\mathbf{B}$ becomes 1 and shifts disc center c_i of row i of $\mathbf{C}$ to the right by 1.
2. *Disc Scaling*: select scalar $s_i > 1$ to increase disc radius r_i of row i corresponding to sampled node i, thus decreasing disc radii r_j of neighboring nodes j due to $\mathbf{S}^{-1}$, and moving disc left-ends $c_j - r_j$ of nodes j to the right.

GDAS describes a procedure that systematically performs these operations to maximally move the smallest disc left-end of $\mathbf{C}$ to the right. *We note that the procedure works only if Gershgorin disc left-ends are initially aligned at the same value.* See [10] for details.

4.4. GDA Sampling for Signed Graphs

However, $\mathcal{L}$ computed from GLASSO is a generalized graph Laplacian corresponding to a signed graph $\mathcal{G}$ possibly with self-loops. This means that the Gershgorin disc left-ends of $\mathcal{L}$ are not initially at the same exact value, and GDAS cannot be used directly.

Fortunately, a recent linear algebraic theorem GDPA [17] provides the mathematical machinery to align disc left-ends. Specifically, GDPA proves that Gershgorin disc left-ends of a generalized graph Laplacian matrix $\mathcal{L}_B$ for an irreducible[5], *balanced*, signed graph $\mathcal{G}_B$ can be aligned *exactly* at $\lambda_{\min}(\mathcal{L}_B)$ via similarity transform $\mathbf{S}\mathcal{L}_B\mathbf{S}^{-1}$, where $\mathbf{S} = \text{diag}(v_1^{-1}, \ldots, v_N^{-1})$, and $\mathbf{v} = [v_1 \ldots v_N]^\top$ is the first eigenvector of $\mathcal{L}_B$ corresponding to $\lambda_{\min}(\mathcal{L}_B)$. First eigenvector $\mathbf{v}$ of $\mathcal{L}_B$ can be computed efficiently using *Locally Optimal Block Preconditioned Conjugate Gradient* (LOBPCG) [27], in roughly linear time, given $\mathcal{L}_B$ is sparse.

Leveraging GDPA, we can employ the following recipe to solve the sampling problem (8) for $\mathcal{L}$ approximately.

[5] An irreducible graph $\mathcal{G}_B$ means that there exists a path from any node to any other node in $\mathcal{G}_B$ [26].

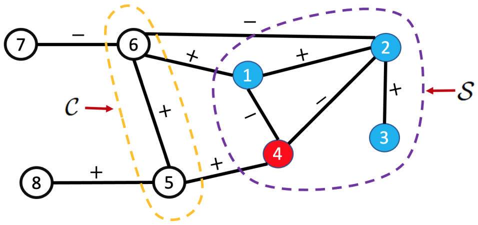


Fig. 1. An example of an 8-node graph $\mathcal{G}$ with sets $\mathcal{S} = \{1, 2, 3, 4\}$ and $\mathcal{C} = \{5, 6\}$.

Step 1: Approximate $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ with a balanced graph $\mathcal{G}_B = (\mathcal{V}, \mathcal{E}_B, \mathbf{W}_B)$ while satisfying the following condition:

$$\lambda_{\min}(\mathbf{H}^\top\mathbf{H} + \mu\mathcal{L}_B) \leq \lambda_{\min}(\mathbf{H}^\top\mathbf{H} + \mu\mathcal{L}). \tag{9}$$

Step 2: Given $\mathcal{L}_B$, perform similarity transform $\mathcal{L}_p = \mathbf{S}\mathcal{L}_B\mathbf{S}^{-1}$, so that disc left-ends of matrix $\mathcal{L}_p$ are aligned exactly at $\lambda_{\min}(\mathcal{L}_p) = \lambda_{\min}(\mathcal{L}_B)$.
Step 3: Employ GDAS [10] for sampling on $\mathcal{L}_p$ to maximize $\lambda^-_{\min}(\mathbf{H}^\top\mathbf{H} + \mu\mathcal{L}_p)$.

Clearly, one should obtain a balanced graph $\mathcal{G}_B$ that induces a *tight* lower bound $\lambda_{\min}(\mathbf{H}^\top\mathbf{H}+\mu\mathcal{L}_B)$ for $\lambda_{\min}(\mathbf{H}^\top\mathbf{H}+\mu\mathcal{L})$ in (9). Thus, we next define an optimization objective for graph balancing and propose a fast balancing algorithm.

4.5. Graph Balancing Formulation

We first show that if we obtain a balanced graph $\mathcal{G}_B$ such that $\mathbf{L}-\mathbf{L}_B$ is a PSD matrix (*i.e.*, $\mathbf{L} - \mathbf{L}_B \succeq 0$), then (9) will be satisfied, where $\mathbf{L}$ and $\mathbf{L}_B$ are combinatorial graph Laplacian matrices for graphs $\mathcal{G}$ and $\mathcal{G}_B$, respectively.

Proposition 1. *Given two undirected graphs* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ *and* $\mathcal{G}_B = (\mathcal{V}, \mathcal{E}_B, \mathbf{W}_B)$*, if* $\mathbf{L} - \mathbf{L}_B \succeq 0$*, then* (9) *is satisfied.*

See [14] for a proof. From the proof, one can see that $\lambda_{\min}(\mathbf{H}^\top\mathbf{H}+\mu\mathcal{L}_B)$ forms a tight lower bound for $\lambda_{\min}(\mathbf{H}^\top\mathbf{H} + \mu\mathcal{L})$ when $\mathbf{x}^\top\mathbf{L}_B\mathbf{x}$ is a tight lower bound for $\mathbf{x}^\top\mathbf{L}\mathbf{x}$, $\forall\mathbf{x} \in \mathbb{R}^N$. Thus, we define our objective for a balanced graph $\mathcal{G}_B$ to maximize $\mathbf{x}^\top\mathbf{L}_B\mathbf{x}$ subject to $\mathbf{L} - \mathbf{L}_B \succeq 0$. Specifically, we assume $\mathbf{x}$ is modeled as a zero-mean *Gaussian Markov random field* (GMRF) [28] over $\mathcal{G}$, *i.e.*, $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is the *covariance matrix* and $\mathbf{\Sigma}^{-1} = \mathcal{L}$ [29] is the *precision matrix*. $\mathbf{\Sigma}^{-1}$ is so defined since $\mathcal{L}$ is a PD matrix.

Next, we choose a balanced graph $\mathcal{G}_B$ to maximize the expectation of $\mathbf{x}^\top\mathbf{L}_B\mathbf{x}$ subject to $\mathbf{L} - \mathbf{L}_B \succeq 0$:

$$\begin{aligned}\mathbb{E}[\mathbf{x}^\top\mathbf{L}_B\mathbf{x}] &= \mathbb{E}[\text{Tr}(\mathbf{x}^\top\mathbf{L}_B\mathbf{x})] = \mathbb{E}[\text{Tr}(\mathbf{L}_B\mathbf{x}\mathbf{x}^\top)] \\ &= \text{Tr}(\mathbf{L}_B\,\mathbb{E}[\mathbf{x}\mathbf{x}^\top]) = \text{Tr}(\mathbf{L}_B\mathbf{\Sigma}).\end{aligned} \tag{10}$$

Maximizing (10) will thus promote tightness of bound (9) given $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$. We now formalize our optimization objective for graph balancing as follows:

$$\max_{\mathbf{L}_B} \text{Tr}(\mathbf{L}_B\mathbf{\Sigma}) \;\;;\;\; \text{s.t.} \begin{cases}\mathbf{L} - \mathbf{L}_B \succeq 0, \\ \mathbf{L}_B \in \mathcal{B} \subset \mathbb{R}^{N\times N}\end{cases} \tag{11}$$

where $\mathcal{B}$ is the set of $N \times N$ combinatorial graph Laplacian matrices corresponding to balanced signed graphs. We next propose a greedy algorithm to add one "most beneficial" node at a time to construct a balanced graph as a solution to (11).

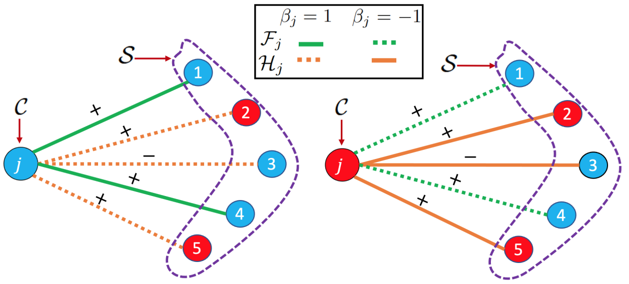


Fig. 2. An example of consistent edges $\mathcal{F}_j$ and inconsistent edges $\mathcal{H}_j$ connecting node $j \in \mathcal{C}$ to $\mathcal{S}$ when $\beta_j = 1$ (left), and consistent edges $\mathcal{H}_j$ and inconsistent edges $\mathcal{F}_j$ when $\beta_j - 1$ (right). For simplicity, we omit edges connecting nodes within set $\mathcal{S}$.

4.6. Graph Balancing Algorithm

According to the *Cartwright-Harary Theorem* (CHT) [16], a graph is balanced iff graph nodes can be colored into red and blue, so that a positive edge always connects nodes of the same color, and a negative edge always connects nodes of opposite colors. Leveraging CHT, we present our iterative greedy algorithm to approximately solve (11) as follows.

To facilitate understanding of our graph balancing algorithm, we first introduce the following notations and definitions.

1. Given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, we define a *bi-colored* node set $\mathcal{S} \subseteq \mathcal{V}$, where nodes in $\mathcal{S}$ are colored into red and blue, and edges connecting nodes in $\mathcal{S}$ are *consistent*. A consistent edge is a positive edge connecting two nodes of the same color, or a negative edge connecting two nodes of opposite colors. Further, denote by $\mathcal{C} \subseteq \mathcal{V} \setminus \mathcal{S}$ the set of nodes within one hop from $\mathcal{S}$. An example of graph $\mathcal{G}$ with sets $\mathcal{C}$ and $\mathcal{S}$ is shown in Fig. 1, where $\mathcal{S} = \{1, 2, 3, 4\}$ and $\mathcal{C} = \{5, 6\}$. If all graph nodes are in set $\mathcal{S}$, *i.e.*, $\mathcal{S} = \mathcal{V}$ and $\mathcal{C} = \emptyset$, then $\mathcal{G}$ is balanced by CHT.

2. Denote by β_i the color assignment of node i; *i.e.*, $\beta_i = 1$ if node i is blue, and $\beta_i = -1$ if node i is red.

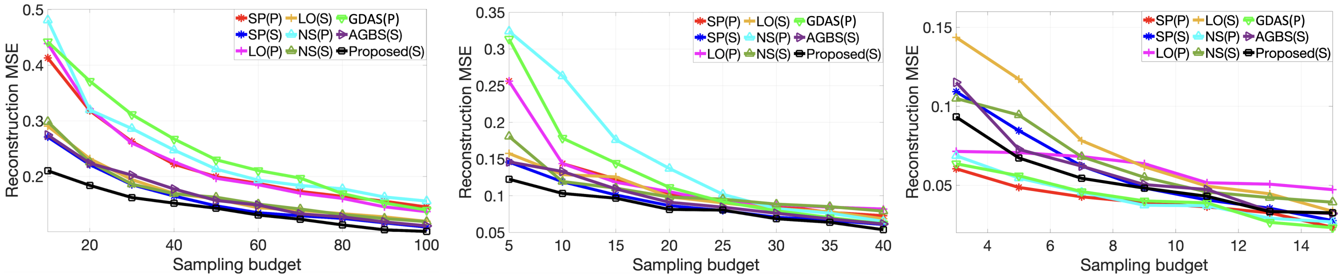
3. For each node $j \in \mathcal{C}$, we partition edges connecting j to $\mathcal{S}$ into two disjoint edge sets: i) set $\mathcal{F}_j$ contains consistent edges from j to $\mathcal{S}$ when $\beta_j = 1$, and ii) set $\mathcal{H}_j$ contains consistent edges from j to $\mathcal{S}$ when $\beta_j = -1$. An example of edge sets $\mathcal{F}_j$ and $\mathcal{H}_j$ connecting $j \in \mathcal{C}$ to $\mathcal{S}$ is shown in Fig. 2, where consistent and inconsistent edges are drawn in different colors for a given value β_j. If we remove inconsistent edges from $j \in \mathcal{C}$ to $\mathcal{S}$ for a given β_j value, node j can be added to $\mathcal{S}$.

4. Denote by $f_j(\beta_j)$ the value of the objective in (11) when node j is assigned color β_j and the corresponding inconsistent edges from $j \in \mathcal{C}$ to $\mathcal{S}$ are removed. This means removing $\mathcal{H}_j$ if $\beta_j = 1$, and removing $\mathcal{F}_j$ if $\beta_j = -1$.

Given an unbalanced graph $\mathcal{G}$, we construct a corresponding balanced graph $\mathcal{G}_B$ by adding one node at a time to set $\mathcal{S}$. At each iteration, we select a "most beneficial" node $j \in \mathcal{C}$ with color β_j to add to $\mathcal{S}$, so that the objective in (11) is locally maximized. Specifically, first, we initialize bi-colored set $\mathcal{S}$ with a random node i and color it blue, *i.e.*, $\beta_i = 1$. Then, at a given iteration, we select a node $j \in \mathcal{C}$ and corresponding color β_j that maximizes objective $f_j(\beta_j)$, *i.e.*,

$$(j^*, \beta^*_{j^*}) = \arg \max_{j\in\mathcal{C}, \beta_j\in\{-1,1\}} f_j(\beta_j). \tag{12}$$

We add j^* of solution $(j^*, \beta^*_{j^*})$ in (12) to $\mathcal{S}$, assign it color $\beta^*_{j^*}$, and remove inconsistent edges from j^* to $\mathcal{S}$ to conclude the iteration.



(a) Canadian parliament voting dataset (b) US senate voting dataset (c) Animal dataset

Fig. 3. Reconstruction MSE for different sampling algorithms under different sampling budgets using GLR-reconstruction. "P" and "S" within each bracket represent the positive graph and signed graph, respectively.

When removing inconsistent edges, the constraint $\mathbf{L} - \mathbf{L}_B \succeq 0$ in (11) must be preserved. We first show that when removing a positive edge, the constraint $\mathbf{L} - \mathbf{L}_B \succeq 0$ is always maintained, as formally stated in Proposition 2.

Proposition 2. *Given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, removing a positive edge $(q, r) \in \mathcal{E}$ from $\mathcal{G}$—resulting in graph $\mathcal{G}_B = (\mathcal{V}, \mathcal{E} \setminus (q, r), \mathbf{W}_B)$—entails $\mathbf{L} - \mathbf{L}_B \succeq 0$.*

See [14] for a proof. Given the ease of removing inconsistent positive edges, we remove them before inconsistent negative edges.

In contrast, removal of a negative edge may violate $\mathbf{L} - \mathbf{L}_B \succeq 0$. Thus, when removing a negative inconsistent edge (j, i), where $j \in \mathcal{C}$ and $i \in \mathcal{S}$, we update weights of *two* additional edges that together with (j, i) form a triangle, in order to ensure constraint $\mathbf{L} - \mathbf{L}_B \succeq 0$ is satisfied, while keeping updated edges consistent. We state this formally in Proposition 3.

Proposition 3. *Given an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, assume that there is an inconsistent negative edge $(j, i) \in \mathcal{E}$ connecting nodes $j \in \mathcal{C}$ and $i \in \mathcal{S}$ of the same color. Let $k \in \mathcal{S}$ be a node of opposite color to nodes j and i. If edge (j, i) is removed—resulting in graph $\mathcal{G}_B = (\mathcal{V}, \mathcal{E} \setminus (j, i), \mathbf{W}_B)$—and weights for edges (k, j) and (k, i) are updated as*

$$\tilde{W}_{p,q} = W_{p,q} + 2W_{j,i} \quad \text{for } (p, q) \in \{(k, j), (k, i)\} \tag{13}$$

then a) $\mathbf{L} - \mathbf{L}_B \succeq 0$; and b) (k, j) and (k, i) are consistent edges.

See [14] for a proof of Proposition 3 and a complexity analysis of the linear-time graph balancing algorithm.

5. EXPERIMENTS

Experimental Setup: We conducted comparison experiments using three datasets. The first dataset consists of Canadian Parliament voting records data from the 38th parliament to the 43rd parliament (2005 - 2021)[6]. This dataset contains voting records of 340 constituencies voted in 3154 elections. The votes are recorded as -1 for "no" and 1 for "yes" and 0 for "abstain / absent". Thus, we define a signal for a given vote as $\mathbf{x} \in \{1, 0, -1\}^{340}$. The second dataset consists of US Senate voting records data from the 115th and 116th congress (2017 - 2020)[7]. This dataset contains voting records of 100 senators in 1320 elections. The third dataset contains (non-)existence information of 102 features (e.g., "has lungs?", "is warm-blooded?", "lives in groups?", etc.) for 33 different animals [30]. The information is recorded as 1 (0) for "existence" ("non-existence"). Thus, we define a signal for a given feature as $\mathbf{x} \in \{1, 0\}^{33}$. For the experiments, we randomly selected 90% of signals from each dataset to learn a signed graph and a positive graph as discussed in Section 3. The remaining 10% were used to test different graph sampling algorithms.

We compare our proposed method with several existing EDF graph sampling methods—SP [7], LO [9], NS [8], GDAS [10], and AGBS [13]—under positive and signed graphs constructed from each dataset. Though SP, LO, NS, and GDAS were designed and tested for positive graphs, during our experiments, we found that SP, LO, and NS methods can operate for signed graphs also. Thus, we tested those methods using both positive and signed graphs constructed from each dataset.

Experimental Results: Performance of different graph sampling methods are shown in Fig 3, in terms of signal reconstruction MSE. For a given chosen sampling matrix $\mathbf{H}$, we solved (7) to reconstruct the complete signals. For Canadian and US voting datasets (Fig. 3(a) and (b)), we observe that the performance of the proposed method is at least as good as the competing schemes at all sampling budgets, and noticeably better when the sampling budgets were small. As an example, for the Canadian dataset, our scheme reduced the lowest MSE among competitor schemes by 22.2%, 18.2%, 13.5%, 10.4%, for sampling budget 10, 20, 30, 40.

There exist strong anti-correlations between constituencies / senators who are in opposite political positions during Canadian and US elections, and hence a signed graph with both positive and negative edge weights is most appropriate. Thus, as shown in Fig. 3(a) and (b), graph sampling methods under a signed graph (*i.e.*, SP(S), NS(S), LO(S), AGBS(S), Proposed) provided better results than under positive graphs (*i.e.*SP(P), NS(P), LP(P), GDAS(P)). In contrast, there is no obvious anti-correlation between two animals in the animal dataset, and hence a positive graph is the most appropriate. Thus, as shown in Fig. 3(c), graph sampling methods under a positive graph give better results than methods under a signed graph.

6. CONCLUSION

To model a dataset with strong anti-correlations, a signed graph with both positive and negative edges is the most appropriate. In this paper, we proposed a fast graph sampling method for signed graphs, leveraging a previous method based on Gershgorin disc alignment sampling (GDAS) [10]. For political voting in Canadian and US elections, we show that our method noticeably outperformed existing eigen-decomposition-free (EDF) sampling schemes that were designed and tested for positive graphs only.

[6] https://www.ourcommons.ca/members/en/votes

[7] https://www.congress.gov/roll-call-votes

7. REFERENCES

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

[2] G. Cheung, E. Magli, Y. Tanaka, and M. Ng, "Graph spectral image processing," *Proc. IEEE*, vol. 106, no. 5.

[3] W. Hu, G. Cheung, A. Ortega, and O. Au, "Multi-resolution graph Fourier transform for compression of piecewise smooth images," *IEEE Trans. Image Process.*, vol. 24, no. 1.

[4] J. Pang and G. Cheung, "Graph Laplacian regularization for image denoising: Analysis in the continuous domain," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1770–1785, 2017.

[5] X. Liu, G. Cheung, X. Wu, and D. Zhao, "Random walk graph Laplacian based smoothness prior for soft decoding of JPEG images," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 509–524, February 2017.

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

[7] A. Anis, A. Gadde, and A. Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Trans. Signal Process.*, vol. 64, no. 14, pp. 3775–3789, 2016.

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

[9] A. Sakiyama, Y. Tanaka, T. Tanaka, and A. Ortega, "Eigendecomposition-free sampling set selection for graph signals," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2679–2692, 2019.

[10] Y. Bai, F. Wang, G. Cheung, Y. Nakatsukasa, and W. Gao, "Fast graph sampling set selection using Gershgorin disc alignment," *IEEE Trans. Signal Process.*, vol. 68, pp. 2419–2434, 2020.

[11] W.-T. Su, G. Cheung, and C.-W. Lin, "Graph Fourier transform with negative edges for depth image coding," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 1682–1686.

[12] G. Cheung, W. T. Su, Y. Mao, and C. W. Lin, "Robust semisupervised graph classifier learning with negative edge weights," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 4, no. 4, pp. 712–726, 2018.

[13] C. Dinesh, Gene G. Cheung, Fen F. Wang, and I. V. Bajic, "Sampling of 3D point cloud via Gershgorin disc alignment," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 2736–2740.

[14] C. Dinesh, G. Cheung, and I. V. Bajic, "Point cloud sampling via graph balancing and Gershgorin disc alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2022.

[15] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics (Oxford, England)*, vol. 9, pp. 432–41, 08 2008.

[16] D. Easley and J. Kleinberg, *Networks, crowds, and markets: Reasoning about a Highly Connected World*, vol. 8, Cambridge university press Cambridge, 2010.

[17] C. Yang, G. Cheung, and W. Hu, "Signed graph metric learning via Gershgorin disc perfect alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021.

[18] R. S. Varga, *Gershgorin and his circles*, Springer, 2004.

[19] F. Chen, G. Cheung, and X. Zhang, "Fast & robust image interpolation using gradient graph Laplacian regularizer," in *2021 IEEE International Conference on Image Processing (ICIP)*, 2021, pp. 1964–1968.

[20] R. Mazumder and T. Hastie, "The graphical lasso: New insights and alternatives," *Electron. J. Statist.*, vol. 6, pp. 2125–2149, 2012.

[21] S. J. Wright, "Coordinate descent algorithms," *Math. Program.*, vol. 151, no. 1, pp. 3–34, 2015.

[22] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under Laplacian and structural constraints," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 6, pp. 825–841, Sep. 2017.

[23] F. Pukelsheim, *Optimal design of experiments*, SIAM, 2006.

[24] H. T. Banks, K. H. Kathleen, and F. Kappel, "Comparison of optimal design methods in inverse problems," *Inverse problems*, vol. 27, no. 7, pp. 075002, 2011.

[25] S. Ehrenfeld, "On the efficiency of experimental designs," *The annals of mathematical statistics*, vol. 26, no. 2, pp. 247–255, 1955.

[26] M. Milgram, "Irreducible graphs," *Journal of Combinatorial Theory, Series B*, vol. 12, no. 1, pp. 6–31, 1972.

[27] A. V. Knyazev, "Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method," *SIAM journal on scientific computing*, vol. 23, no. 2, pp. 517–541, 2001.

[28] H. Rue and L. Held, *Gaussian Markov random fields: theory and applications*, CRC press, 2005.

[29] A. Gadde and A. Ortega, "A probabilistic interpretation of sampling theory of graph signals," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 3257–3261.

[30] C. Kemp and J. B. Tenenbaum, "The discovery of structural form," *Proceedings of the National Academy of Sciences*, vol. 105, no. 31, pp. 10687–10692, 2008.