2025 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)

Distance-based Laplacian Algebra for Effective Subgraph Filter Learning

Purui Zhang*, Feng Ji*, Yanan Zhao*, Wee Peng Tay*, and Bihan Wen*
* Nanyang Technological University, Singapore
E-mail: PURUI001@e.ntu.edu.sg, jifeng@ntu.edu.sg, YANAN002@e.ntu.edu.sg, wptay@ntu.edu.sg, bihan.wen@ntu.edu.sg

***Abstract*—Graph signal processing tasks that leverage spectral information—such as filter learning—typically assume access to the complete graph topology or its graph shift operator. However, in many practical scenarios, only a subset of graph vertices is accessible. We propose a novel framework for subgraph signal processing that constructs suitable graph shift operators and filters for subgraphs, while preserving their relationship to the ambient graph. Our distance-based Laplacian algebra systematically generates subgraph operators by incorporating node distances, enabling a rich and interpretable filter space. Experimental results on both synthetic and real-world traffic flow datasets demonstrate that our approach outperforms standard polynomial subgraph filters in subgraph filter learning tasks.**

I. INTRODUCTION

Graph signal processing (GSP) covers a wide array of topics since its emergence [1], including structured data analysis [2], graph neural networks [3, 4], graph spectral analysis [5], and topology inference [6, 7]. The graph shift operator (GSO), as the core component of GSP, is directly associated with the topology of the graph and is necessary for filtering [8, 9], recovery [10], and graph Fourier transform (GFT) [11].

Traditional signal processing tasks such as filtering and spectral analysis in GSP typically assume access to the complete graph structure and its associated GSO. However, in many practical scenarios, the full graph may be partially observed: nodes, edges, or parts of node signals can be missing due to factors such as communication delays, data corruption, or privacy constraints. In these cases, only a partial graph signal $\mathbf{x}_0$ supported on a node subset $V_{\mathrm{sub}} \subset V$ is available. Directly restricting filters to the subgraph often leads to information loss, and the GFT computed on subgraphs yields spectra that differ from those of the ambient graph. Therefore, a dedicated subgraph signal processing framework is required to process partial signals $\mathbf{x}_0$ using filters or spectral tools defined solely on V_{sub}. We illustrate with two examples in the following.

Example 1. *The concept of graph shift invariant signals finds many applications, e.g., in stationary statistical signal processing over a graph [12, 13]. Graph shift invariance is defined with respect to (w.r.t.) the GSO of the ambient graph, which we assume is available to a stakeholder. However, in many cases, a third party may only be provided access to a subgraph of the ambient graph, and the usual graph Laplacian of the subgraph is not shift invariant with respect to the ambient graph's GSO. For example, consider a sensor network where each sensor node is connected to its neighbors. If we only have access to a subset of sensors, the shift behavior of the subgraph may not match that of the ambient graph. In this case, we want to design a suitable GSO for node subset V_{sub} so that its node signals $\mathbf{x}_0$ are approximately shift invariant w.r.t. this GSO.*

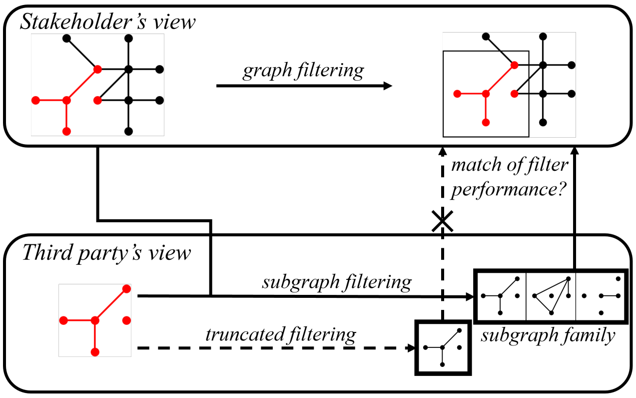

Fig. 1. Illustration of subgraph filtering. Note that the third party has no knowledge of nodes and edges beyond the subgraph structure.

Example 2. *Consider a scenario where a stakeholder owns a sensor network and applies a proprietary filter to process the data for downstream tasks. Due to constraints such as privacy or intellectual property, the stakeholder grants a third party access only to a subset of sensors. The third party wishes to perform the same downstream tasks using this partial data. The objective is for the filtered outputs on the accessible sensors to closely approximate the stakeholder's filtered outputs restricted to the same subset. To achieve this, the stakeholder must design a subgraph filter—supported only on the accessible sensors—that can be provided to the third party. The effectiveness of this subgraph filter is evaluated by comparing two outputs: (i) applying the full filter to the complete data and then restricting the result to the subset, and (ii) applying the subgraph filter directly to the partial data. If the outputs are sufficiently similar, the third party can confidently use the subgraph filter for its processing tasks.*

A straightforward approach is to construct a truncated filter by extracting the principal submatrix of the proprietary filter corresponding to the subgraph nodes. This truncated filter preserves message passing within the subgraph but discards interactions between the subgraph and the rest of the graph. As a result, the truncated filter often suffers from information *loss and suboptimal performance. Therefore, it is necessary to develop a more expressive subgraph filter space that better approximates the desired restricted filter.*

In this paper, we focus on designing GSOs and filters for node subsets. Given access to an ambient graph filter and corresponding full graph signals as training data, our goal is to develop a family of subgraph filters that optimize the filtering performance on a specified subset of nodes. This subgraph filter learning task differs from traditional graph learning approaches [14], which aim to infer the entire graph structure from input-output signal pairs. In contrast, the learned subgraph filter is supported only on the subgraph and does not directly reveal the true ambient graph filter or the structure outside the observed subset.

The rest of this paper is organized as follows. In Section II, we first introduce basic concepts of graph signal processing and matrix algebra, and we formally introduce the subgraph filter learning problem. To solve the problem, we propose a scheme to find a family of subgraphs and construct its corresponding filter space in Section III. We conduct subgraph learning tasks on both synthetic and real-world dataset in Section IV and conclude in Section V.

Throughout this paper, we use boldfaced symbols for vectors and matrices. The space $\mathbb{R}^N$ is the N-dimensional real Euclidean space, $\mathbb{R}_+$ the set of non-negative real values, and $\mathbb{N}$ the set of non-negative integers.

II. PRELIMINARIES

A. Graph Signal Processing

Let $G = (V, E)$ be a graph, and $\mathbf{x} \in \mathbb{R}^{|V|}$ a graph signal on G. Its adjacency matrix is $\mathbf{A} \in \{0,1\}^{|V|\times|V|}$, where $\mathbf{A}_{ij} = 1$ iff $(v_i, v_j) \in E$, and 0 otherwise. Its degree matrix is $\mathbf{D} = \operatorname{diag}\{\deg(v_i)\}_{i=1}^{|V|}$, where $\deg(v_i)$ is the degree of v_i in G. The Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, while the symmetric normalized Laplacian is $\widetilde{\mathbf{L}} = (\mathbf{D}^\dagger)^{1/2}\mathbf{L}(\mathbf{D}^\dagger)^{1/2}$, where $\mathbf{D}^\dagger$ is the Moore-Penrose inverse of $\mathbf{D}$, i.e., $\mathbf{D}^\dagger_{ii} = \deg(v_i)^{-1}$ if $\deg(v_i) \neq 0$, and all other elements are 0. Let $d_G(u, v)$ be the shortest-path distance between u and v in G, and $\operatorname{diam}(G) = \max\{d_G(u, v) : u, v \in G\}$ be the diameter of G.

For a subset $V_{\text{sub}} \subset V$, a V_{sub}-induced subgraph $G_{\text{sub}} := (V_{\text{sub}}, \{(u, v) : u, v \in V_{\text{sub}}, (u, v) \in E\})$. In this paper, all subgraphs are induced subgraphs unless otherwise specified. The subgraph signal $\mathbf{x}_1$ is $\mathbf{x}$ restricted to V_{sub} and can be written as $\mathbf{x}_1 = \mathbf{P}_{\text{sub}}\mathbf{x}$, where $\mathbf{P}_{\text{sub}} \in \{0,1\}^{|V_{\text{sub}}|\times|V|}$ is the *selection matrix*. It satisfies $\mathbf{P}_{\text{sub}}\mathbf{P}_{\text{sub}}^\intercal = \mathbf{I}_{|V_{\text{sub}}|}$, the $|V_{\text{sub}}| \times |V_{\text{sub}}|$ identity matrix.

B. Matrix Algebra

Since the graph Laplacian is symmetric and positive semi-definite, it has orthonormal decomposition $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\intercal$, where $\mathbf{\Lambda} = \operatorname{diag}\{\lambda_i\}_{i=1}^{|V|}$ is the diagonal matrix storing all eigenvalues in non-decreasing order $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{|V|}$. $\widetilde{\mathbf{L}}$ is also symmetric and has eigenvalues $0 = \widetilde{\lambda}_1 \leq \widetilde{\lambda}_2 \leq \ldots \leq \widetilde{\lambda}_{|V|} \leq 2$.

For $p \in \mathbb{R}_+$, we define $\mathbf{L}^p = \mathbf{U}\mathbf{\Lambda}^p\mathbf{U}^\intercal$. Since $\mathbf{L}$ and $\mathbf{L}^p$ can be simultaneously diagonalized and $x \mapsto x^p$ is an injection from $\mathbb{R}_+ \to \mathbb{R}_+$, there exists a polynomial f_p with degree at most $|V| - 1$ such that $f_p(\lambda_i) = \lambda_i^p$. Therefore, we have $f_p(\mathbf{L}) = \mathbf{L}^p$. We define $\widetilde{\mathbf{L}}^p$ in a similar way and its polynomial expression follows as well.

Denote the matrix polynomial algebra as

$$\mathbb{R}[\widetilde{\mathbf{L}}] = \left\{\mathbf{M} = \sum_{i=0}^{r_0} a_i \widetilde{\mathbf{L}}^i \,:\, a_i \in \mathbb{R}, r_0 \in \mathbb{N}\right\},$$

and let $\mathbb{R}\langle\widetilde{\mathbf{L}}_1, \widetilde{\mathbf{L}}_2, \ldots, \widetilde{\mathbf{L}}_k\rangle$ (assuming all matrices are of the same size) be the matrix free algebra:

$$\operatorname{span}_{\mathbb{R}}\left\{\prod_{i=0}^{r_0} \widetilde{\mathbf{L}}_{l_i} \,:\, \widetilde{\mathbf{L}}_{l_i} \in \{\widetilde{\mathbf{L}}_1, \widetilde{\mathbf{L}}_2, \ldots, \widetilde{\mathbf{L}}_k\}, r_0 \in \mathbb{N}\right\}.$$

We use $\mathbb{R}_{\leq r}[\widetilde{\mathbf{L}}]$ or $\mathbb{R}_{\leq r}\langle\widetilde{\mathbf{L}}_1, \widetilde{\mathbf{L}}_2, \ldots, \widetilde{\mathbf{L}}_k\rangle$ to denote the subalgebra with at most r matrices multiplied in a monomial.

C. Subgraph Filter Learning

Suppose a stakeholder has an ambient graph $G = (V, E)$, with $|V| = N$, Laplacian $\mathbf{L}_G$, normalized Laplacian $\widetilde{\mathbf{L}}_G$. Consider a graph filter

$$\mathbf{F} = \sum_{i=0}^{k} a_i \widetilde{\mathbf{L}}_G^i,$$

which is a polynomial of the normalized Laplacian $\widetilde{\mathbf{L}}_G$. The stakeholder applies this filter to a signal $\mathbf{y} \in \mathbb{R}^N$ on G, producing the filtered signal $\mathbf{z} = \mathbf{F}\mathbf{y} \in \mathbb{R}^N$. We assume that the filtered output $\mathbf{z}$ or the filter $\mathbf{F}$ is the input for some downstream tasks. The stakeholder provides access to only a subset of nodes V_{sub} to a third party, which has no knowledge of the full ambient graph G or the filter $\mathbf{F}$. The third party only has access to the V_{sub}-induced subgraph G_{sub}.

To allow the third party the capability to perform the same downstream tasks, the stakeholder aims to derive a subgraph filter $\mathbf{F}_{\text{sub}} \in \mathbb{R}^{|V_{\text{sub}}|\times|V_{\text{sub}}|}$ on the subgraph G_{sub}, which approximates the performance of $\mathbf{F}$ with output restricted to V_{sub}. The subgraph filter $\mathbf{F}_{\text{sub}}$ should be able to process signals restricted to V_{sub}, i.e., $\mathbf{y}' = \mathbf{P}_{\text{sub}}\mathbf{y}$, where $\mathbf{P}_{\text{sub}} \in \{0,1\}^{|V_{\text{sub}}|\times N}$ is the selection matrix for the node subset V_{sub}. The filtered output on the subgraph is then given by $\mathbf{z}' = \mathbf{F}_{\text{sub}}\mathbf{y}$.

To do this, we assume that the stakeholder has T signal realizations $(\mathbf{y}_t)_{t=1}^T$, which it uses to find $\mathbf{F}_{\text{sub}}$ that minimizes the mean squared error (MSE) between the filtered outputs on the ambient graph restricted to V_{sub} and the subgraph:

$$\frac{1}{T \cdot |V_{\text{sub}}|} \sum_{t=1}^{T} \|\mathbf{P}_{\text{sub}}\mathbf{F}\mathbf{y}_t - \mathbf{F}_{\text{sub}}\mathbf{P}_{\text{sub}}\mathbf{y}_t\|_2^2. \tag{1}$$

In this case, the client can use $\mathbf{F}_{\text{sub}}$ to filter signals directly on V_{sub} with the performance close to using $\mathbf{F}$ on the ambient graph graph G and restricting it to V_{sub}.

A simple solution to this problem, as stated in Example 2, is the *truncated filter*, defined as $\mathbf{F}_{\text{trunc}} = \mathbf{P}_{\text{sub}}\mathbf{F}\mathbf{P}_{\text{sub}}^\intercal$. Given that $\mathbf{P}_{\text{sub}}$ is the selection matrix, $\mathbf{F}_{\text{trunc}}$ is the principal submatrix whose rows and columns are selected from the

$$\begin{tikzcd}
 & (G, \mathbf{z}) \arrow[r, "\mathbf{P}_{\text{sub}}"] & (G_{\text{sub}}, \mathbf{P}_{\text{sub}}\mathbf{z}) \arrow[dd, dash, dashed, "\overset{?}{\approx}"] \\
(G, \mathbf{y}) \arrow[ur, "\mathbf{F}"] \arrow[dr, "\mathbf{P}_{\text{sub}}"'] & & \\
 & (G_{\text{sub}}, \mathbf{P}_{\text{sub}}\mathbf{y}) \arrow[r, "\mathbf{F}_{\text{sub}}"] & (G_{\text{sub}}, \mathbf{F}_{\text{sub}}\mathbf{P}_{\text{sub}}\mathbf{y})
\end{tikzcd}$$

Fig. 2. Diagram for subgraph filter learning.

rows and columns in $\mathbf{F}$, according to the nodes in V_{sub}. When $|V_{\text{sub}}| = N$, $\mathbf{F}_{\text{trunc}}$ is just the rearrangement of rows and columns of $\mathbf{F}$.

When $|V_{\text{sub}}| \neq N$, we have $\mathbf{P}_{\text{sub}}{}^{\mathsf{T}}\mathbf{P}_{\text{sub}} \neq \mathbf{I}_N$, and there does not exist any $\mathbf{F}_{\text{sub}}$ such that $\mathbf{F}_{\text{sub}}\mathbf{P}_{\text{sub}} = \mathbf{P}_{\text{sub}}\mathbf{F}$. Consequently, no filter defined solely on G_{sub} can exactly reproduce the output of the ambient graph filter $\mathbf{F}$ when restricted to V_{sub}. Moreover, the truncated filter $\mathbf{F}_{\text{trunc}}$ ignores all interactions between G_{sub} and the remainder of the graph, resulting in information loss and degraded performance.

To address this limitation, we seek to compensate for the missing information by constructing a richer family of subgraph filters that incorporate additional structural information from the ambient graph G. While the space of all polynomial filters of $\widetilde{\mathbf{L}}_G$ is generally smaller than the full space of possible subgraph filters, using the entire matrix space $\mathbb{R}^{|V_{\text{sub}}|\times|V_{\text{sub}}|}$ is impractical due to its high dimensionality and lack of interpretability. Therefore, we propose to augment G_{sub} with additional operators derived from G, enabling the construction of expressive and interpretable subgraph filter spaces tailored to the available information.

III. DISTANCE-BASED LAPLACIAN ALGEBRA

Consider a graph $G = (V, E)$ and its subgraph $G_{\text{sub}} = (V_{\text{sub}}, E_{\text{sub}})$. To construct filters for G_{sub} that leverage information from the original graph G, we begin by examining the message passing properties of common GSOs, such as the adjacency matrix, Laplacian matrix, and their normalized variants. When applied to a signal $\mathbf{x}$, these operators aggregate information from neighboring nodes, and a k-th order polynomial of a GSO captures information within k-hop neighborhoods.

However, in the context of a subgraph G_{sub}, the shortest path between two nodes may be longer, or may not even exist, compared to the original graph G. Specifically, for $u, v \in V_{\text{sub}}$, we have $d_{G_{\text{sub}}}(u, v) \geq d_G(u, v)$, and it is possible for u and v to be disconnected in G_{sub} while remaining connected in G. Relying solely on the subgraph adjacency matrix or Laplacian may therefore require higher-order polynomials to propagate information between nodes, or may fail entirely if the nodes are disconnected.

To address this limitation and better approximate the message passing behavior of the original graph, we introduce auxiliary subgraphs constructed from both G and G_{sub}.

A. Distance-based Subgraphs

Definition 1. *For $1 \leq k \leq \operatorname{diam}(G)$, let the distance-$k$ subgraph of G_{sub} be a graph such that*

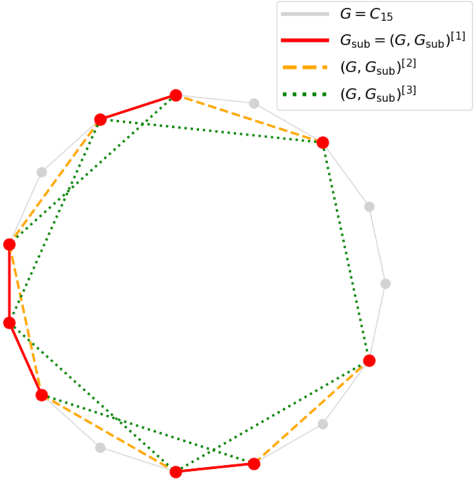

Fig. 3. Example for $(G, G_{\text{sub}})^{[k]}$ on cycle graph $G = C_{15}$ and given G_{sub} for $k = 1, 2, 3$.

- *its vertex set is V_{sub};*
- *and its edge set is $\{(u, v) \in V_{\text{sub}} \times V_{\text{sub}} : d_G(u, v) = k\}$, i.e., the edges are those pairs of vertices in V_{sub} that are k hops apart in the ambient graph G.*

We denote this subgraph as $(G, G_{\text{sub}})^{[k]}$. We use the simplified notation G_k if no confusion arises.

Note that $(G, G_{\text{sub}})^{[1]} = G_{\text{sub}}$ and $(G, G)^{[k]}$ is just the distance-k graph of G [15].

Example 3. *Fig. 3 illustrates how $(G, G_{\text{sub}})^{[k]}$ is created for a cyclic graph $G = C_{15}$ when $k = 1, 2, 3$. In this example, G_{sub} has 5 connected components (2 of which are isolated nodes). If we construct a subgraph filter based on GSO of G_{sub}, information can not be passed between components; but if the GSO based on all distance-k subgraphs are used, information can pass through different components.*

B. Laplacian Algebra

For (G, G_{sub}), assume G_k has normalized Laplacian $\widetilde{\mathbf{L}}_k$. A straightforward way of constructing filters called (k, r)*-distance-based Laplacian polynomials (DLP)* has the following form:

$$\mathbf{H} = \sum_{j=1}^{k}\sum_{i=0}^{r} a_{i,j}\widetilde{\mathbf{L}}_j^i.$$

Each polynomial $\sum_{i=0}^{r} a_{i,j}\widetilde{\mathbf{L}}_j^i$ encompasses message passing schemes inside G_j no more than r hops.

To further optimize the filter space design, we consider two aspects on the propagation of G_k.

1) Propagation "speed" of different subgraphs: The propagation "speed" of different Laplacians varies. For the standard GSO on the ambient graph G, the operator $\widetilde{\mathbf{L}}^r$ aggregates information from nodes up to r hops away in G. For a distance-k subgraph G_k, the operator $\widetilde{\mathbf{L}}_k^r$ collects information from nodes that are r hops away within G_k. However, when interpreted in terms of the original graph G, $\widetilde{\mathbf{L}}_k^r$ effectively aggregates information from nodes up to $k \cdot r$ hops away in G.

To ensure that all order-r polynomials process information within r-hop neighborhoods in G, we introduce fractional powers of the Laplacians:

$$\mathbf{H}_{\text{frac}} = \sum_{j=1}^{k}\sum_{i=0}^{r} a_{i,j}\widetilde{\mathbf{L}}_j^{i/j}.$$

Here, we use $\widetilde{\mathbf{L}}_j^{1/j}$ so that an order-r polynomial in $\mathbf{H}_{\text{frac}}$ covers information within r hops in G. This adjustment aligns the effective propagation range of each term with the desired neighborhood size in the ambient graph.

Note that $\mathbf{H}_{\text{frac}}$ involves fractional powers of Laplacians, such as terms like $\widetilde{\mathbf{L}}_k^{1/k}$. As discussed in Section II-A, there exists a polynomial f_k such that $\widetilde{\mathbf{L}}_k^{1/k} = f_k(\widetilde{\mathbf{L}}_k)$. The actual neighborhood size aggregated by $\widetilde{\mathbf{L}}_k^{1/k}$ depends on the degree of f_k, with the influence typically decaying as the number of hops increases [16, Theorem 4.1]. While the full polynomial spaces $\mathbb{R}[\widetilde{\mathbf{L}}_k]$ and $\mathbb{R}[\widetilde{\mathbf{L}}_k^{1/k}]$ are equivalent, restricting the maximum degree (e.g., considering only $\operatorname{span}_{\mathbb{R}}\{\mathbf{I}, \widetilde{\mathbf{L}}_k, \widetilde{\mathbf{L}}_k^2\}$ versus $\operatorname{span}_{\mathbb{R}}\{\mathbf{I}, \widetilde{\mathbf{L}}_k^{1/k}, \widetilde{\mathbf{L}}_k^{2/k}\}$) leads to different subspaces. Using fractional powers thus provides additional flexibility in shaping the filter spectra within a fixed polynomial order.

2) Mixed propagation scheme between subgraphs: For (G, G_{sub}), it is common for G_{sub} to be disconnected, potentially consisting of multiple connected components, and this disconnectivity can persist even in the distance-k subgraphs for $k > 1$. As illustrated in Example 3, G_{sub} contains 5 components, while G_2 and G_3 each have 3 connected components. When constructing a filter using $\mathbf{H}_{\text{frac}}$, message passing is limited to within each subgraph, even when fractional powers are used. To overcome this limitation and enable information flow across different components, we can combine multiple operators related to different G_ks within a single monomial. For example, applying $\widetilde{\mathbf{L}}_2\widetilde{\mathbf{L}}_1 x$ means the signal $\mathbf{x}$ is first propagated according to G_1, and then further propagated via G_2, thereby facilitating communication between otherwise disconnected components. We introduce the non-commutative algebra structure as follows.

Definition 2. *The subgraph filter free algebra (SFFA) of G_{sub} over G is given by*

$$\mathscr{A}_{(G,G_{\text{sub}})} = \mathbb{R}\langle \widetilde{\mathbf{L}}_1, \widetilde{\mathbf{L}}_2, \dots, \widetilde{\mathbf{L}}_{\operatorname{diam}(G)}\rangle.$$

While the SFFA provides a highly expressive filter space by including all possible monomials generated from $\widetilde{\mathbf{L}}_k$ for $k = 1, \dots, \operatorname{diam}(G)$ of arbitrary length and order, this generality comes at the cost of a prohibitively large number of parameters. For practical applications, it is essential to control the complexity of the filter space. This can be achieved by limiting (i) the number of distance-k subgraphs included, (ii) the maximum propagation order r (i.e., the number of matrices multiplied in each monomial), and (iii) enforcing symmetry to reduce the number of parameters and facilitate spectral analysis. Incorporating fractional-order Laplacians further enhances flexibility. We formalize this more manageable algebraic structure as follows.

Definition 3. *For $k \geq 1$ and $r \geq 0$, a (k, r)-symmetric hop Laplacian algebra (SHLA) is given by*

$$\begin{aligned}&\mathscr{A}^*_{(G,G_{\text{sub}})}(k, r)\\ &= \Big\{\mathbf{M} : \mathbf{M} \in \mathbb{R}_{\leq r}\langle \widetilde{\mathbf{L}}_1, \widetilde{\mathbf{L}}_2^{1/2}, \dots, \widetilde{\mathbf{L}}_k^{1/k}\rangle,\ \mathbf{M} = \mathbf{M}^{\intercal}\Big\}\end{aligned}$$

For $M \in \mathscr{A}^*_{(G,G_{\text{sub}})}(k, r)$, symmetry is ensured by assigning equal coefficients to each monomial and its reverse. Specifically, the coefficient for $\prod_{i=1}^{j}\widetilde{\mathbf{L}}_{l_i}^{1/l_i}$ must match that of its reversed counterpart $\prod_{i=1}^{j}\widetilde{\mathbf{L}}_{l_{j+1-i}}^{1/l_{j+1-i}}$. This guarantees that M remains symmetric, as all normalized Laplacians (and their fractional powers) are themselves symmetric matrices.

C. Expressive Power of Laplacian Algebra

When performing filter learning on a subgraph $G_{\text{sub}} \subset G$, using the subgraph GSO $\mathbf{L}_{\text{sub}}$ directly restricts all possible filters to the commutative algebra $\mathbb{R}[\mathbf{L}_{\text{sub}}]$, whose elements share the same eigenspace. This significantly limits the expressive power of $\mathbb{R}[\mathbf{L}_{\text{sub}}]$, as its dimension is at most the number of distinct eigenvalues of $\mathbf{L}_{\text{sub}}$, which cannot exceed $|V_{\text{sub}}|$. In contrast, the space of all symmetric filters on the subgraph has dimension $|V_{\text{sub}}|(|V_{\text{sub}}| + 1)/2$. Therefore, to achieve greater expressiveness, our filter space should have a dimension substantially larger than $|V_{\text{sub}}|$. We are thus interested in the maximum expressive power of $\mathscr{A}^*_{(G,G_{\text{sub}})}(k, r)$.

Proposition 1. *The maximum possible dimension of a (k, r)-SHLA is*

$$\frac{1}{2}\sum_{l=0}^{r}(k^l + k^{\lfloor (l+1)/2 \rfloor}),$$

achieved when all monomials in the (k, r)-SHLA are linearly independent.

Proof: The calculation can be split into three steps. For a fixed l:

- Counting all monomials: There are k^l monomials with order l.
- Counting all symmetric monomials: For monomial $\prod_{j=1}^{l}\widetilde{\mathbf{L}}_{i_j}^{1/i_j}$, its transpose should be the reverse order $\prod_{j=1}^{l}\widetilde{\mathbf{L}}_{i_{l+1-j}}^{1/i_{l+1-j}}$. If it is symmetric, then the sequence $(i_1, i_2, \dots, i_l)$ should be palindromic. If l is even, there are $k^{l/2}$ palindromic monomials; if l is odd, there are $k^{(l+1)/2}$ monomials. We can combine them into $k^{\lfloor (l+1)/2 \rfloor}$ for both cases.
- Calculate the maximum dimension: Since in SHLA, non-symmetric monomials are paired and share the same coefficient, so $k^l - k^{\lfloor (l+1)/2 \rfloor}$ monomials only contribute half number of dimensions. Symmetric monomials each contribute a dimension. So in total there are

  $$\frac{1}{2}(k^l - k^{\lfloor (l+1)/2 \rfloor}) + k^{\lfloor (l+1)/2 \rfloor} = \frac{1}{2}(k^l + k^{\lfloor (l+1)/2 \rfloor})$$

  dimensions. Summing over $l = 0, \dots, r$ gives the desired result and the proof is complete.

■

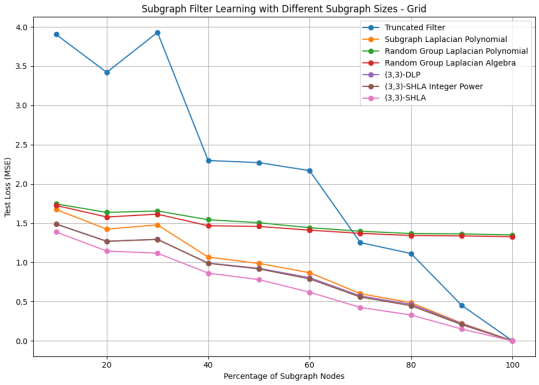

Fig. 4. Subgraph filter learning performance of different filters on grid dataset.

Proposition 1 suggests that to fully span the symmetric filter space $\mathbb{R}^{|V_{\text{sub}}|(|V_{\text{sub}}|+1)/2}$, the parameters k and r should satisfy $\frac{1}{2}\sum_{l=0}^{r}(k^l + k^{\lfloor (l+1)/2 \rfloor}) \geq |V_{\text{sub}}|(|V_{\text{sub}}|+1)/2$. To empirically verify this, we conduct experiments on Erdős–Rényi random graphs $G_{\text{ER}}(N = 25, p = 0.4)$. For each of 200 independently sampled graphs G, we computed the dimension of $\mathscr{A}^*_{(G,G_{\text{sub}}=G)}(k = 2, r = 10)$. Since $\frac{1}{2}\sum_{l=0}^{10}(2^l + 2^{\lfloor (l+1)/2 \rfloor}) = 1086 > 325 = N(N + 1)/2$, the algebra is sufficiently expressive. Indeed, in all 200 samples, $\mathscr{A}^*_{(G,G)}(k = 2, r = 10)$ achieved the full dimension of 325, confirming that the constructed algebra can cover the entire symmetric filter space in these cases.

While it is theoretically possible for the (k, r)-SHLA to span the entire symmetric filter space, in practice, full expressiveness is often unnecessary and may be computationally inefficient. For practical subgraph filter learning, it is advisable to tune the parameters k and r according to the structure of the underlying graph. For example, in sparse graphs such as trees, increasing k (i.e., incorporating more distance-based Laplacians) while keeping r relatively small is typically effective. In contrast, for denser graphs such as Erdős–Rényi graphs with $p > 0.3$, a smaller k (e.g., $k = 2$) combined with a higher polynomial order r is usually sufficient. Careful selection of these parameters balances model expressiveness and computational efficiency.

IV. EXPERIMENTS

A. Subraph Filter Learning on Grid Networks

To test the performance of the subgraph filters, we simulate on a 10 by 10 grid graph G (nodes $N = 100$), with vertex set V, subset V_{sub} (nodes $|V_{\text{sub}}| = pN$, p is the ratio), selection matrix $\mathbf{P}_{\text{sub}} \in \{0, 1\}^{|V_{\text{sub}}|\times N}$, normalized ambient graph Laplacian $\widetilde{\mathbf{L}} \in \mathbb{R}^{N\times N}$, and normalized subgraph laplacian $\widetilde{\mathbf{L}}_0 \in \mathbb{R}^{|V_{\text{sub}}|\times|V_{\text{sub}}|}$. We generate a polynomial filter $\mathbf{F} = a_0\mathbf{I} + a_1\widetilde{\mathbf{L}} + a_2\widetilde{\mathbf{L}}^2 + a_3\widetilde{\mathbf{L}}^3$, and we set $a_0 = 0.5, a_1 = -1, a_2 = 2, a_3 = 1$. For each trial, we randomly pick p portion of nodes and create the induced subgraph G_{sub}, which has normalized subgraph Laplacian $\widetilde{\mathbf{L}}_0$ and distance-k subgraph normalized Laplacians $\widetilde{\mathbf{L}}_k$. The original signal $\mathbf{y}$ is drawn $T = 800$ times independently in Unif $(0, 1)^{|V_{\text{sub}}|}$ for training, yielding $(\mathbf{y}_t, \mathbf{z}_t = \mathbf{F}\mathbf{y}_t)_{i=1}^{T}$; and $T_0 = 200$ times for testing, yielding $(\mathbf{y}_t, \mathbf{z}_t = \mathbf{F}\mathbf{y}_t)_{i=T+1}^{T+T_0}$. Our target is to train a filter $\mathbf{F}_{\text{sub}}$ in a given filter space with training data such that the average recovery error on V_{sub} in (1) is minimized.

For performance evaluation, we compute the MSE on test data:

$$e_{\text{test}} = \frac{1}{T_0 \cdot |V_{\text{sub}}|}\sum_{t=T+1}^{T+T_0} \|\mathbf{P}_{\text{sub}}\mathbf{F}\mathbf{y}_t - \mathbf{F}_{\text{sub}}\mathbf{P}_{\text{sub}}\mathbf{y}_t\|_2^2.$$

We conduct performance comparison of different filter spaces, including the following:

- Truncated filter $\mathbf{F}_{\text{trunc}} = \mathbf{P}_{\text{sub}}\mathbf{F}\mathbf{P}_{\text{sub}}^{\mathsf{T}}$.
- Subgraph Laplacian polynomial filter space $\mathbb{R}_{\leq 3}[\widetilde{\mathbf{L}}_0]$.
- Random group Laplacian polynomials: We randomly sample 90% of nodes of V_{sub} to obtain an induced subgraph and its corresponding normalized Laplacian. We do this independently thrice to obtain the normalized Laplacians $\widetilde{\mathbf{L}}_{r,1}, \widetilde{\mathbf{L}}_{r,2}, \widetilde{\mathbf{L}}_{r,3}$ and construct the filter space $\mathbb{R}_{\leq 3}[\widetilde{\mathbf{L}}_{r,1}] + \mathbb{R}_{\leq 3}[\widetilde{\mathbf{L}}_{r,2}] + \mathbb{R}_{\leq 3}[\widetilde{\mathbf{L}}_{r,3}]$.
- Random group Laplacian algebra: Same as the random group Laplacian polynomial filter space construction, we construct the following filter space from randomly sampled normalized Laplacians:

$$\left\{\mathbf{M} : \mathbf{M} \in \mathbb{R}_{\leq 3}\langle \widetilde{\mathbf{L}}_{r,1}, \widetilde{\mathbf{L}}_{r,2}, \widetilde{\mathbf{L}}_{r,3} \rangle, \mathbf{M} = \mathbf{M}^{\mathsf{T}}\right\}.$$

- $(3, 3)$-DLP, the distance-based Laplacian polynomials.
- Subspace of $(3, 3)$-SHLA that only includes monomials with integer powers (e.g., $\widetilde{\mathbf{L}}_2^{1/2}\widetilde{\mathbf{L}}_1\widetilde{\mathbf{L}}_2^{1/2}$ is not included, but $\widetilde{\mathbf{L}}_1\widetilde{\mathbf{L}}_2^{1/2}\widetilde{\mathbf{L}}_2^{1/2} = \widetilde{\mathbf{L}}_1\widetilde{\mathbf{L}}_2$ is included). This simplifies computations as there is no fractional power, but it reduces the number of trainable coefficients.
- The proposed SHLA $\mathscr{A}^*_{(G,G_{\text{sub}})}(3, 3)$.

For all filter settings except for the truncated filter the coefficients require training. We use Adam optimizer, learning rate 0.01, 20000 epochs to ensure convergence. The whole experiment is repeated 5 times and the average of test MSE is recorded.

From the experiment result shown in Fig. 4 we can see that the proposed SHLA has the best performance, especially when $p \in (0.2, 0.7)$, while SHLA with integer order only has similar performance compared to distance-k Laplacian polynomials. Enabling fractional power ensures the interpretability of matrix order, enlarges the space of trainable coefficients, and in turn, improves the performance.

B. Subgraph Filter Learning on Traffic Flow Data

We conduct experiments on the METR-LA dataset [17]. After removing edge directions, the resulting graph G contains $N = 207$ nodes. We extract all time stamps with complete sensor readings, using the first $T = 800$ samples for training and the subsequent $T_0 = 200$ samples for testing. The ambient filter is set as $\mathbf{F} = a_0\mathbf{I}+a_1\widetilde{\mathbf{L}}_G+a_2\widetilde{\mathbf{L}}_G^2+a_3\widetilde{\mathbf{L}}_G^3$, with coefficients $a_0 = 0.5$, $a_1 = -1$, $a_2 = 2$, and $a_3 = 1$. All subgraph filter types from the previous experiment are evaluated under the same training setup, optimizer, learning rate, and number of

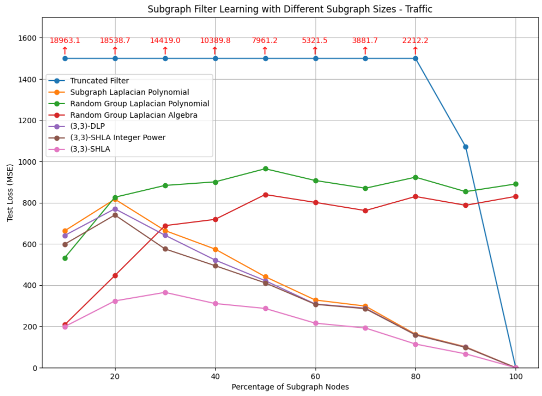

Fig. 5. Subgraph filter learning performance of different filters on METR-LA dataset.

epochs. Each experiment is repeated 5 times, and the average test MSE is reported. The results in Fig. 5 show that the proposed $(3,3)$-SHLA approach consistently outperforms other subgraph filter methods, particularly for $p \in (0.2, 0.7)$.

V. CONCLUSION

We have introduced a novel framework for subgraph filter learning based on distance-based Laplacian algebra. By leveraging distance-k subgraphs, our approach systematically captures multi-scale neighborhood relationships within the ambient graph. The resulting Laplacian algebra provides an interpretable, flexible, and highly expressive filter space for subgraph signal processing. Experimental results on both synthetic and real-world traffic datasets demonstrate that our method consistently outperforms standard polynomial subgraph filters, validating its effectiveness for practical subgraph filter learning tasks.

REFERENCES

[1] A. Ortega, P. Frossard, J. Kovačević, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proc. IEEE*, vol. 106, no. 5, pp. 808–828, May 2018, ISSN: 0018-9219. DOI: 10.1109/JPROC.2018.2820126.

[2] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Mag.*, vol. 30, no. 3, pp. 83–98, Apr. 2013.

[3] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009, ISSN: 1941-0093. DOI: 10.1109/TNN.2008.2005605.

[4] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[5] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in Neural Inform. Process. Syst.*, Barcelona, Spain, 2016, pp. 3844–3852, ISBN: 978-1-5108-3881-9.

[6] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning Laplacian matrix in smooth graph signal representations," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6160–6173, Dec. 2016, ISSN: 1053-587X. DOI: 10.1109/TSP.2016.2602809.

[7] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from data under Laplacian and structural constraints," *IEEE J. Sel. Top. Signal Process.*, vol. 11, no. 6, pp. 825–841, Sep. 2017, ISSN: 1932-4553. DOI: 10.1109/JSTSP.2017.2726975.

[8] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Dec. 2014, pp. 872–876. DOI: 10.1109/GlobalSIP.2014.7032244.

[9] F. Ji, W. P. Tay, and A. Ortega, "Graph signal processing over a probability space of shift operators," *IEEE Transactions on Signal Processing*, vol. 71, pp. 1159–1174, 2023. DOI: 10.1109/TSP.2023.3263675.

[10] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovačević, "Signal recovery on graphs: Variation minimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 17, pp. 4609–4624, Sep. 2015, ISSN: 1941-0476.

[11] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Graph fourier transform," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6167–6170. DOI: 10.1109/ICASSP.2013.6638850.

[12] F. Ji and W. P. Tay, "A Hilbert space theory of generalized graph signal processing," *IEEE Transactions on Signal Processing*, vol. 67, no. 24, pp. 6188–6203, Dec. 2019. DOI: 10.1109/TSP.2019.2952055.

[13] X. Jian and W. P. Tay, "Wide-sense stationarity in generalized graph signal processing," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3414–3428, 2022. DOI: 10.1109/TSP.2022.3184455.

[14] H. E. Egilmez, E. Pavez, and A. Ortega, "Graph learning from filtered signals: Graph system and diffusion kernel identification," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 2, pp. 360–374, Jun. 2019, ISSN: 2373-776X. DOI: 10.1109/TSIPN.2018.2872157.

[15] W. Czech, "Invariants of distance k-graphs for graph embedding," *Pattern Recognition Letters*, vol. 33, no. 15, pp. 1968–1979, 2012, Graph-Based Representations in Pattern Recognition, ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2012.03.024.

[16] S. Maskey, R. Paolino, A. Bacho, and G. Kutyniok, "A fractional graph laplacian approach to oversmoothing," ser. NIPS '23, New Orleans, LA, USA: Curran Associates Inc., 2023.

[17] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *International Conference on Learning Representations (ICLR '18)*, 2018.