Decimated Framelet System on Graphs and Fast $\mathcal{G}$-Framelet Transforms

Xuebin Zheng   xuebin.zheng@sydney.edu.au
University of Sydney Business School
University of Sydney

Bingxin Zhou[*]   bzho3923@uni.sydney.edu.au
University of Sydney Business School
University of Sydney

Yu Guang Wang[†]   yuguang.wang@sjtu.edu.cn
Institute of Natural Sciences and School of Mathematical Sciences
Shanghai Jiao Tong University
& Max Planck Institute for Mathematics in Sciences
& School of Mathematics and Statistics
University of New South Wales

Xiaosheng Zhuang   xzhuang7@cityu.edu.hk
Department of Mathematics
City University of Hong Kong

Editor: Joan Bruna

Abstract

Graph representation learning has many real-world applications, from self-driving LiDAR, 3D computer vision to drug repurposing, protein classification, social networks analysis. An adequate representation of graph data is vital to the learning performance of a statistical or machine learning model for graph-structured data. This paper proposes a novel multiscale representation system for graph data, called decimated framelets, which form a localized tight frame on the graph. The decimated framelet system allows storage of the graph data representation on a coarse-grained chain and processes the graph data at multi scales where at each scale, the data is stored on a subgraph. Based on this, we establish decimated $\mathcal{G}$-framelet transforms for the decomposition and reconstruction of the graph data at multi resolutions via a constructive data-driven filter bank. The graph framelets are built on a chain-based orthonormal basis that supports fast graph Fourier transforms. From this, we give a fast algorithm for the decimated $\mathcal{G}$-framelet transforms, or F$\mathcal{G}$T, that has linear computational complexity $\mathcal{O}(N)$ for a graph of size N. The effectiveness for constructing the decimated framelet system and the F$\mathcal{G}$T is demonstrated by a simulated example of random graphs and real-world applications, including multiresolution analysis for traffic network and representation learning of graph neural networks for graph classification tasks.

Keywords: Graphs, Decimated tight framelets, Tree, SPOC, Undecimated tight framelets, Filter bank, Fast $\mathcal{G}$-framelet transforms, Fast Fourier transforms, Coarse-grained chain, Graph Laplacian, Haar basis, Graph convolution, Graph neural networks, Multiresolution analysis

---

[*]. First corresponding author
[†]. Second corresponding author

1. Introduction

Geometric structure and feature of data are in the center of many commonly seen systems, which plays a pivotal role in understanding and guiding modern data science and machine learning avenues. In physics, particles interact with each other following physics laws to form matters at different scales, such as atoms, molecules, planets, stars, solar systems, galaxies, and the whole universe Shlomi et al. (2020); Perraudin et al. (2019). In biology, genes as sequences of DNA or RNA encode molecules' functions to form cells, tissues, organs, organ systems, plants, animals, and the entire ecosystem. In human societies, individuals linked by the social rules interact to shape the new communities, societies, governments, nations, and countries. In computer science, data specified by the Internet protocols provide end-to-end data communications from the lowest link layers, through packeting, addressing, transmitting, routing, and receiving, to the highest application layers between PCs, smartphones, digital sensors. All these systems, and many others in the fields of neuroscience, cognitive science, sociology, have an underlying data structure that can be represented by graphs, where vertices are distinct elements or actors, and edges indicate connections or interactions between the elements or actors. Moreover, a multiscale structure appears in all these systems, which is determined by the intensity of 'force' of the interactions which form clusters or subsystems in different scales. A large amount of information might be generated and attached to the complex system, where we can view a system and its information as a graph and graph data. This paper constructs a tight decimated framelet system to represent graph data and develops an efficient algorithm for framelet transforms.

On Euclidean domains, harmonic analysis (Stein, 1970; Stein and Weiss, 1971; Stein, 1993) has been an active research branch of mathematics since the seminal work of Fourier (Fourier, 1822). In the past two centuries, it has become a well developed subject with application in areas as diverse as signal processing, representation theory, number theory, quantum mechanics, tidal analysis, and neuroscience. In the last four decades, one of its sub-branches – wavelet analysis has been intensively studied by many pioneers (Meyer, 1990; Chui, 1992; Daubechies, 1992; Mallat, 2009; Han, 2017). In recent years, there has been a great interest in developing wavelet-like representation systems for data on non-Euclidean domains, including manifolds and graphs. One motivation comes from the interdisciplinary research demand of computer science and mathematics, when the data are not only big but also have intrinsic geometric structure, from such as social networks, biology, chemistry, physics, finance to image processing. The big data are usually regarded as samples from a smooth manifold, where the graph Laplacian approximates the underlying manifold Laplacian (Singer, 2006). The underlying manifold encodes the geometric information of the data, which approach has been widely used in machine learning and statistical models (Roweis and Saul, 2000; Tenenbaum et al., 2000; Belkin et al., 2004; Zhang and Zha, 2004; Belkin and Niyogi, 2003; Belkin and Niyogi, 2008; Bruna et al., 2014; Defferrard et al., 2016; Cheung et al., 2018, 2020). Moreover, people have combined deep learning and graph signal processing (Sandryhaila and Moura, 2013, 2014; Shuman et al., 2013; Zhou et al., 2021a), which fosters emerging fields of *geometric deep learning* (Bronstein et al., 2017, 2021; Bodnar et al., 2021b,a) and *graph neural networks* (Wu et al., 2020; Ma et al., 2021; Zheng et al., 2020, 2021; Zhou et al., 2021b).

Motivated by the importance of data processing, the sparse representation of graph data, and the increasing interest in harmonic analysis for graph signal processing, in this paper, we investigate the characterization, construction, and computation for tight framelets on graphs. We lay out the framework of this paper, as follows. We focus on an undirected connected graph $\mathcal{G} = (V, E)$ with the vertex set V and edge set E. Our goal is to provide a systematic method for multiscale decomposition and reconstruction of a graph signal $\boldsymbol{f} : V \to \mathbb{C}$ defined on the graph $\mathcal{G}$.
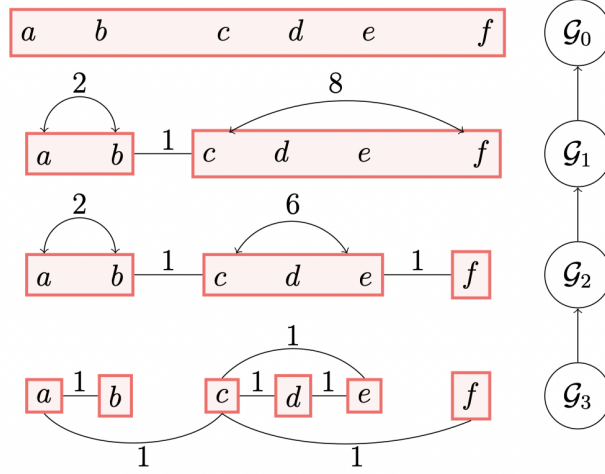

Figure 1: Coarse-grained chain $\mathcal{G}_{3 \to 0} := \mathcal{G}_3 \to \mathcal{G}_2 \to \mathcal{G}_1 \to \mathcal{G}_0$ of $\mathcal{G} \equiv \mathcal{G}_3$. Here the bottom graph $\mathcal{G}_3 \equiv \mathcal{G}$ is the underlying graph. Each box in $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$ is a vertex of $\mathcal{G}_j$, the line between two vertices is a edge, the arc on a same vertex indicates a self-loop edge, and the number on a line/curve is the corresponding edge weight.

(1) Graph clustering generates a coarse-grained chain $\mathcal{G}_{J \to J_0}$ with $\mathcal{G}_J \equiv \mathcal{G}$ which can be built with a *nested* property (finest to coarsest). That is, each vertex in $\mathcal{G}_j = (V_j, E_j)$ is a cluster of the previous $\mathcal{G}_{j-1}$ and hence a cluster of the original graph $\mathcal{G}$. The *bottom-up* construction provides a multiscale structure on $\mathcal{G}$. It assigns a level of the chain to *scale* values and defines *translation* by cluster centroids. The resulting chain plays an important role in construction of our *decimated tight framelets*. Figure 1 illustrates clustering of a graph $\mathcal{G}$ to a coarse-grained chain $\mathcal{G}_{3 \to 0}$.

(2) With the constructed coarse-grained chain, we can define a sequence of orthonormal bases $\{\boldsymbol{u}_\ell^{\mathcal{G}_j}\}_{\ell=1}^{|V_j|}$, $j = J_0, \dots, J$ in a *top-down* manner. Eventually, we would obtain a *global* orthonormal basis $\{\boldsymbol{u}_\ell : V \to \mathbb{C}\}_{\ell=1}^{|V|}$ for the underlying graph $\mathcal{G}$, which can be used to construct tight framelets on $\mathcal{G}$. The strategy for framelet construction is independent of the orthogonal basis: one can apply either spectral methods (such as eigendecomposition of the graph Laplacian) or non-spectral methods (such as interval decomposition). The top-down approach utilizes a chain, which provides an orthogonal basis mimicking the classical Fourier bases, and defines a notion of *frequencies* that can be used to distinguish *low-pass* and *high-pass* information for a graph signal.

Any signal $\boldsymbol{f} : V \to \mathbb{C}$ can be precisely represented in the graph framelet domain by the framelet transforms.

(3) Motivated by the localization property of the filtered kernel

$$K_{j,\alpha}(p, v) := \sum_{\ell} \widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right) \overline{\boldsymbol{u}_\ell(p)} \boldsymbol{u}_\ell(v), \text{ for } p, v \in V.$$

and the framelet filter bank on $\mathbb{R}$, we define *tight framelets* $\boldsymbol{\psi}_{j,[p]} = K_{j,\alpha}(p, \cdot)$ on a graph. Framelet $\boldsymbol{\psi}_{j,[p]} = K_{j,\alpha}(p, \cdot)$ at scale j is localized around the centroid $[p]$, and the set of framelets is a tight frame with multiscale property. Here, $\widehat{\alpha}$ is the filter which localizes and smooths the kernel, and λ_ℓ is the eigenvalue of the underlying graph Laplacian for degree ℓ. The 2^j is the scaling factor for level j in the multiscale system. In this work, we investigate two types of tight frames on the graph: *undecimated* and *decimated tight framelets*. Undecimated framelets take the form of $\boldsymbol{\psi}_{j,p}^n : V \to C$ with node $p \in V$ running through vertices on V (at scale j), while decimated framelets are of the form $\boldsymbol{\psi}_{j,[p]}^n : V \to C$ with node $[p] \in V_{j+1}$ through all vertices on V_{j+1}. Both framelets are defined on the underlying graph $\mathcal{G}$, while for the decimated version, the 'centers' p and $[p]$ locate at different layers of the chain. Tightness of the framelet system provides an exact representation of the graph data in the framelet domain. It also generates the perfect reconstruction for graph signals under framelet transforms. By the characterization theorem, designing a tight framelet system is equivalent to finding a proper filter bank. For decimated framelets, the filter bank depends on the chain structure.

(4) The chain-based orthogonal basis gives rise to fast algorithmic transforms for the framelet system. With the chain-based orthogonal basis $\{\boldsymbol{u}_\ell\}$, we have the *fast graph Fourier transforms* to compute the graph Fourier coefficients $\widehat{\boldsymbol{f}}_\ell$ from $\boldsymbol{f}$. With the above tight framelet filter banks for the decimated framelets on $\mathcal{G}$, we can implement fast $\mathcal{G}$-framelet transforms for signals defined on $\mathcal{G}$. Such fast framelet transforms provide a way for sparse representation, efficient computation, and effective processing of signals on a graph. See Figure 2 for an illustration for the one-level discrete $\mathcal{G}$-framelet transforms.

The contributions of the paper lie in the following aspects.

(i) We introduce the decimated framelet system on a graph, as a special affine system. Wang and Zhuang (2018) constructed continuous framelets and semi-discrete framelets on a continuous (or smooth) manifold. Dong (2017) constructed undecimated framelets on graphs, where the framelets $\psi_{j,p}$ are defined for all $p \in V$ and all j. These correspond to a continuous wavelet tight frame on a manifold. However, the undecimated framelet system may result in a high redundancy rate when the number of levels is big, as there is no *coarsening* in the framelet system (see also Hammond et al. (2011)). One needs to use Chebyshev approximation for fast framelet transforms in this case. In contrast, by using the graph's clustering feature, we embed the edge relation and clustering features into a chain-based orthonormal system to establish a *decimated framelet system*. The resulting tight framelet system that corresponds to a
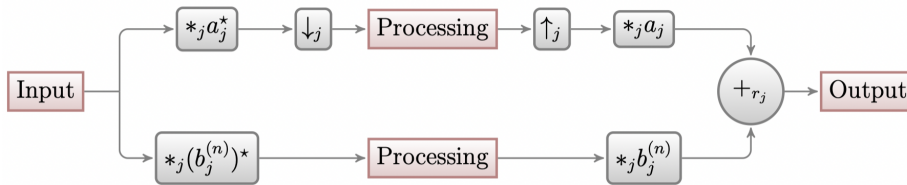
Figure 2: One-level $\mathcal{G}$-framelet transforms including framelet decomposition and reconstruction based on the filter bank $\{a_j; b_j^{(1)}, \ldots, b_j^{(r_j)}\}$. Input graph signal is decomposed using convolutions ($*$) with filters and possible downsampling ($\downarrow$) to output framelet coefficient sequences, which are processed before reconstruction that uses the same filter bank. Here the node $*_j b_j^{(n)}$ of the convolution operation ranges over $n = 1, \ldots, r_j$ and the $+_{r_j}$ is the summation over the low-pass filtered coefficient sequence and all r_j high-pass filtered coefficient sequences.

discrete framelet system on a manifold of Wang and Zhuang (2018) would achieve a low redundancy rate.

(ii) We present a framework for the characterization and construction of decimated and undecimated tight framelets on graphs. In particular, we offer a complete characterization for the tightness of decimated framelet systems on graphs. The characterization implies an equivalent condition of the framelet tightness that significantly simplify the design of tight graph framelets, due to the excellent property of the constructed chain-based orthonormal system.

(iii) We provide a fast algorithm in $\mathcal{O}(|V|)$ for the discrete Fourier transforms under the established chain-based orthonormal system. The fast algorithm overcomes the general difficulty of applying FFT for graph-structured signals on irregular nodes. Building on a chain-based orthogonal basis and a sequence of tight framelet filter banks, the discrete decimated framelets allow fast computation of decomposition and reconstruction for a graph signal in the multiscale framelet representation — *fast $\mathcal{G}$-framelet transforms* or F$\mathcal{G}$Ts.

(iv) The chain-based orthonormal basis and tight framelets provide a spectral method in defining graph convolution and pooling, which are key ingredients of graph neural networks. See preliminary empirical studies in Wang et al. (2020); Li et al. (2020); Zheng et al. (2020) and Section 8.4 provide a multiscale analysis tool with potential applications in areas such as superresolution analysis (Deng et al., 2019; Candès and Fernandez-Granda, 2014; Schiebinger et al., 2017) and geometric matrix/tensor completion (Jiang et al., 2020; Berg et al., 2018; Monti et al., 2017). We provide framelet multiresolution analysis on the Minnesota traffic network. We would develop an F$\mathcal{G}$T-based spectral graph convolution for graph neural networks on various graph classification tasks.

We organize the rest of the paper as follows. In Section 3, we introduce basic notions used in the paper, including graph, chain, orthonormal basis, frame, framelet and tightness. In Section 4, we define both undecimated and decimated framelet systems on graphs

and provide a complete characterization for the tightness of both types of framelet systems. In Section 5, we construct a chain-based orthonormal basis, including examples of graph Laplacian-based basis and Haar-like basis. We then define decimated tight framelets on the graph using the above chain-based orthogonal basis. In Section 6.2, we develop a fast computational strategy for the transforms in the orthogonal basis representation and framelet representation. We develop algorithms for discrete Fourier transforms under the chain-based orthonormal basis, and give a sufficient condition that guarantees linear computational complexity of the algorithm. Decimated $\mathcal{G}$-framelet transforms can then be implemented based on the underlying filter bank, and the computational complexity is in the same order as the corresponding DFT. In Section 7, we show a simple example to illustrate the full construction of a chain-based orthonormal basis and decimate tight framelets. In Section 8, we show examples of filter banks for decimated framelets, and provide a test of computational complexity which shows the consistent result to the algorithmic analysis. We finally apply decimated framelets for MRA of Minnesota traffic network, and graph convolution of GNNs on graph classification tasks. Discussion and remarks are given in Section 9.

2. Related Work

Clustering Our construction of decimated tight framelets on the graph utilizes graph clustering. There are many existing clustering algorithms (Schaeffer, 2007; Nascimento and De Carvalho, 2011; Filippone et al., 2008; Aggarwal and Wang, 2010; Rui Xu and Wunsch, 2005; Xu and Tian, 2015). While we do not focus on clustering methods, which themselves are an active research area, we mention a few examples. The coarse grained chain $\mathcal{G}_{J \to J_0} = \mathcal{G}_J \to \mathcal{G}_{J-1} \to \cdots \to \mathcal{G}_{J_0}$ has a tree structure in nature (see Figure 1 for an example. One can simply take V at the root node as a single cluster if $\mathcal{G}_{J_0}$ is not with a single cluster). To obtain such a chain, one can use a partition approach, which recursively partitions a graph into subgraphs, or, reversely, a merging approach, which groups various vertices to generate clusters. The partition approach starts from the root of a tree (the full graph) and continues to its branches (clusters) up to a certain level, for example, the graph cut method (Karp, 1972; Shi and Malik, 2000). In contrast, the merging approach starts from the leaves (vertices of a graph) of a tree and climbs up to a certain level (clusters), for example, the k-means methods (MacQueen et al., 1967; Lloyd, 1982; Karypis and Kumar, 1999). One can also categorize graph clustering algorithms based on *spatial-frequency* criteria, that is, by whether the clustering method is spectral-based. The spectral-based graph clustering algorithms usually use graph Laplacian, eigendecomposition or wavelet decomposition (Merriman et al., 1992; Bertozzi and Flenner, 2012; Garcia-Cardona et al., 2014), while the non-spectral clustering employs information from an appropriately defined weight function $\boldsymbol{w}$ (Coifman et al., 2005; Dongen, 2000; Gavish and Coifman, 2012; Gavish et al., 2010; Chaudhuri and Dasgupta, 2010; Lafon and Lee, 2006; Boykov et al., 2001; Chen et al., 2014).

Wavelets and framelets Once a chain of nested graphs is obtained, we can build the multiscale structure on a graph $\mathcal{G}$ by framelet filter banks, which bridges with the classical wavelet/framelet systems on $\mathbb{R}$. We use non-homogeneous affine systems to describe our framelet systems on $\mathcal{G}$, where a non-homogeneous affine system on a domain $\mathcal{D}$ typically

takes the form
$$\{\varphi_{J_0,k} : k \in \Lambda_{J_0}\} \cup \{\psi_{j,k}^{(n)} : k \in \Lambda_j, n = 1, \ldots, r\}_{j=J_0}^{J}$$
for $J_0, J \in \mathbb{Z}$, $r \in \mathbb{N}$. Here Λ_j are the index sets of the *translation points* and *centers* for the framelets on the domain $\mathcal{D}$. The function $\varphi_{j,k}$ is from a scaling function φ which captures the *low frequency* information of the graph signal and $\psi_{j,k}^{(n)}$ is associated with framelet functions $\psi^{(n)}$ which captures *high frequency* information at a specific scale. The functions of $\Psi := \{\varphi; \psi^{(1)}, \ldots, \psi^{(r)}\}$ are associated with a filter bank $\boldsymbol{\eta} := \{a; b^{(1)}, \ldots, b^{(r)}\}$ by the *Unitary Extension Principle* (UEP). On $\mathbb{R}^d$, affine systems and the construction of wavelet/framelet systems have been well studied (Ron and Shen, 1997a,b; Chui et al., 1998, 2002; Daubechies et al., 2003). Non-homogeneous affine systems on $\mathbb{R}^d$ for a more general setting are studied in Han (2010, 2012). The construction of directional affine systems was further exploited in Guo et al. (2004); Han and Zhuang (2015); Zhuang (2016); Che and Zhuang (2018); Atreas et al. (2019). In particular, Wang and Zhuang (2018) constructed a tight framelet system on a compact, smooth manifold $\mathcal{M}$. A framelet system on a more abstract locally compact abelian group is proposed by Christensen and Goh (2019).

A variety of wavelets/framelets on manifolds have been proposed. Coifman and Maggioni (2006) introduced orthogonal diffusion wavelets on a smooth manifold by diffusion operators. Mhaskar et al. (Mhaskar and Prestin, 2004; Maggioni and Mhaskar, 2008; Mhaskar, 2010) extended the construction from diffusion wavelets to diffusion polynomial frames on a manifold. Dong (2017) constructed continuous framelets on a compact Riemannian manifold, where they made the first connection of filter bank with continuous framelets on manifolds. Wang and Zhu (2018) developed semi-discrete and fully discrete tight framelets on a simplex of any dimension by using a quadrature rule on the simplex for integral discretization. Wang et al. (2017); Le Gia et al. (2017) developed discrete spherical needlets (a highly localized tight frame on the unit sphere) by filtered spherical harmonic expansion and quadrature rule, and applied to the decomposition of spherical random fields. Li et al. (2019) constructed tight tensor framelets for vector fields on the sphere and developed a fast algorithm for the tensor framelet transforms. The impressive advance of data science and deep learning, such as graph neural networks, has driven the rapid development of the field of wavelets and framelets on graphs (Shuman et al., 2015).

Wavelets on graphs were first proposed in Crovella and Kolaczyk (2003). The diffusion polynomial approach was then introduced to construct diffusion wavelets on graphs (Coifman and Maggioni, 2006; Maggioni and Mhaskar, 2008). With the spectral graph theory, Hammond et al. (2011) established the wavelet analysis, and Dong (2017) constructed undecimated tight wavelet frames on graphs. Haar transform-based approach on graphs were studied in Gavish et al. (2010); Gavish and Coifman (2012); Li et al. (2020). Chui et al. (2015) constructed an orthogonal polynomial system on a weighted tree, which gave a chain-based Haar orthonormal basis. On another direction, Bruna and Mallat (2013); Mallat (2012, 2016) developed wavelet scattering that is a wavelet system combined with neural activation that mimics the architecture of a deep convolutional neural network. Gao et al. (2019); Gama et al. (2019); Zou and Lerman (2020) developed the wavelet scattering methods for graphs. We propose to use the graph framelet transforms to provide a multiscale representation for graph data and feature extraction and thus to define a new graph

convolution as a core computational block for the graph neural network rather than using wavelet scattering to replace the whole GNN architecture.

Fast transform and graph convolution The fast transform algorithms are a fundamental tool for efficient computation with wavelets or framelets. One applies the forward transform to decompose a signal representation from the spatial domain to a set of wavelet basis. The adjoint transform, in contrast, reconstructs the signal from its wavelet representation back to the spatial domain. A fast implementation benefits, in particular, for large-scale graphs. The fast algorithm of wavelet transforms was first proposed by Mallat (1989) with multi-resolution analysis, which later inspires a number of applications in signal processing. The wavelet coefficients through the fast algorithm are evaluated level by level using discrete convolution and decimated operators rather than estimating inner products for wavelet coefficients. The domain of fast wavelet transforms has been well developed for discrete signals, see Candès et al. (2006); Han (2013b); Han et al. (2016b), to mention but a few.

Graph convolution was first introduced in Bruna et al. (2014) based on the convolution theorem and spectral graph theory (Chung, 1997). Since then, a good number of graph convolution and graph neural network models have been proposed (Defferrard et al., 2016; Kipf and Welling, 2017; Veličković et al., 2018; Fawzi et al., 2018; Cheung et al., 2018; Li et al., 2020; Ma et al., 2019; Wang et al., 2020; Ma et al., 2020; Zhou et al., 2021a). The graph convolution is conducted in the spectral domain as diagonal multiplications. However, due to the non-Euclidean nature, discrete Fourier transforms, suffer from a high cost for computing the orthogonal basis from the graph Laplacian. To overcome the difficulty, Defferrard et al. (2016) approximates smooth filters with Chebyshev polynomials, which then circumvents eigendecomposition for the graph Laplacian. Kipf and Welling (2017); Gilmer et al. (2017); Veličković et al. (2018); Xu et al. (2019b) further simplified the convolutional layer with first-order approximation and proposed spatial-based graph convolution. Such methods rely on node feature aggregation among neighbour vertices. Meanwhile, fast Fourier or wavelet transform algorithms were developed which can be exploited to accelerate spectral-based graph convolution (Dong, 2017; Li et al., 2020; Zheng et al., 2020).

Most related works We discuss important papers that are close to and motivate our work. Chui et al. (2015) (see also its extension to digraph in Chui et al. (2018)) established an orthogonal system with localization properties on the tree and its approximation theory based on the notion of *filtration* for a tree. Such an orthogonal system is indeed a Haar-type wavelet system on a tree. The idea of Haar (1910) has been 'invented', 'reinvented' and 'rejuvenated' in different domains. By defining 'h-hop' neighborhoods on a graph, Crovella and Kolaczyk (2003) reduced the construction of wavelets on a graph to the classical Haar wavelets on the interval $[0, 1)$. Using the concept of 'folders' and 'subfolders', Haar-like bases are constructed in Gavish et al. (2010) for high-dimensional data. Directional Haar tight framelets exist in an arbitrary dimension $\mathbb{R}^d$, and their projection to lower dimensions induce the directional box splines (Han et al., 2019; Li and Zhuang, 2019). The application of Haar transforms on graphs for deep learning tasks can be seen in Li et al. (2020); Wang et al. (2020); Zheng et al. (2020).

In Hammond et al. (2011), spectral graph theory is used to define scaling operator $T_g^t = g(t\mathcal{L})$ for functions on a graph, where g is a function, t is a scaling parameter,

and $\mathcal{L}$ is the graph Laplacian. A wavelet $\psi_{t,v}$ at scale t and 'centered' at a vertex v can then be defined as such operator applied to a delta impulse signal, that is, $\psi_{t,v} = T_g^t \delta_v$. Spectral wavelet graph transforms (SGWT) are then implemented based on the Chebyshev polynomial approximation. Dong (2017) constructed tight wavelet frames on both manifolds and graphs, where $\psi_{j,v}$ is defined from a scaling function ϕ and hence associated the construction of wavelet frames on a graph with a filter bank. Fast wavelet frame transforms on the graph (WFTG) can then be implemented based on the Chebyshev polynomial approximation of filters.

In Wang and Zhuang (2018), tight framelets, including both continuous and semi-discrete tight framelets on a Riemannian manifold, are constructed based on orthogonal eigenpairs on the manifold, where localized filtered kernel functions were used to define *scaling* and *translation* on the manifold. Framelets on the manifold could then be constructed from the filtered kernels. More importantly, the framelets are associated with a filter bank that enables a simple characterization of tight framelets. In particular, quadrature rules (weighted sampling point sets) with polynomial exactness property naturally induced the construction of semi-discrete tight framelets on the manifold, which are significantly different from the construction of Hammond et al. (2011); Dong (2017). The quadrature-based framelets have fast tight framelet filter bank transforms (F$\mathcal{M}$T), which is useful for practical applications in manifold-data multiresolution analysis. This paper follows the line of Wang and Zhuang (2018) but is mainly focused on the multiscale data analysis on graphs and its fast algorithms.

3. Preliminaries

In this section, we introduce some basic notation and properties on graphs, frames and filter banks, which are based on the works of Chung (1997); Dong (2017); Hammond et al. (2011); Wang and Zhuang (2018, 2019).

3.1 Graph and Chains

An undirected and weighted graph $\mathcal{G}$ is an ordered triple $\mathcal{G} = (V, E, \boldsymbol{w})$ with a non-empty finite set V of vertices, a set $E \subseteq V \times V$ of edges between vertices in V, and a non-negative weight function $\boldsymbol{w} : E \to \mathbb{R}$. For an undirected graph $\mathcal{G}$, we denote $|V|$ and $|E|$ the numbers of vertices and edges. An edge $e \in E$ with vertices $p, v \in V$ is an unordered pair denoted by (p, v) or (v, p). In this paper, we assume the self-loops (v, v), $v \in V$ in the edge set E, and we extend $\boldsymbol{w}$ from E to $V \times V$ by letting $\boldsymbol{w}(p, v) := 0$ for $(p, v) \notin E$. The extended weight function $\boldsymbol{w}$ is also called the *adjacency (representation) matrix* of $\mathcal{G}$. Note that for an undirected graph, the weight $\boldsymbol{w}$ is *symmetric* in the sense that $\boldsymbol{w}(p, v) = \boldsymbol{w}(v, p)$ for all $p, v \in V$. We denote the *degree* of a vertex $v \in V$ by $\boldsymbol{d}(v) := \sum_{p \in V} \boldsymbol{w}(v, p)$. The *volume* of the graph $\mathrm{vol}(\mathcal{G}) := \mathrm{vol}(V) = \sum_{v \in V} \boldsymbol{d}(v)$, which is the sum of degrees of all vertices of $\mathcal{G}$. Given a subset V_0 of V, the *volume* of V_0 is the sum of degrees of all nodes in V_0, that is, $\mathrm{vol}(V_0) := \sum_{v \in V_0} \boldsymbol{d}(v)$.

Let $(e_1, \ldots, e_n)$ be a sequence of edges in $\mathcal{G}$. If there exist distinct vertices $v_0, \ldots, v_n$ in V such that any pair of consecutive nodes is connected by an edge of $\mathcal{G}$, that is, $e_j = (v_{j-1}, v_j)$ for $j = 1, \ldots, n$, then the sequence $(e_1, \ldots, e_n)$ is called a *path* of $\mathcal{G}$ between v_0 and v_n, and the *length* of the path is defined to be $\sum_{j=1}^{n} \boldsymbol{w}(v_{j-1}, v_j)$. Two vertices p and v are

connected if there exists a path between p and v. If p and v are connected, the *distance* $\boldsymbol{\rho}(p, v)$ between two vertices p and v is defined as the length of the shortest possible path between them. We define $\boldsymbol{\rho}(p, v) = \infty$ if there is no path between p and v. A graph itself is called *connected* if any two distinct vertices of $\mathcal{G}$ are connected. In the paper, we only consider connected graphs.

Let $\mathcal{G} = (V, E, \boldsymbol{w})$ and $\mathcal{G}_c = (V_c, E_c, \boldsymbol{w}_c)$ be two graphs. We say that $\mathcal{G}_c$ is a *coarse-grained graph* of $\mathcal{G}$ if V_c is a *partition* of V; that is, there exist k $(k > 1)$ subsets $V_1, \ldots, V_k$ of V such that

$$V_c = \{V_1, V_2, \ldots, V_k\}, \quad V_1 \cup \cdots \cup V_k = V, \quad V_i \cap V_j = \emptyset,\ 1 \le i < j \le k.$$

In this case, each vertex V_j of $\mathcal{G}_c$ is called a *cluster* for $\mathcal{G}$. The edges of $\mathcal{G}_c$ are the links between clusters of $\mathcal{G}$. The nodes in $\mathcal{G}$ that are in the same cluster (node) of $\mathcal{G}_c$ define an equivalence relation on $\mathcal{G}$: two vertices p and v are equivalent, denoted by $p \sim v$, if p and v are in the same cluster. An equivalent class (cluster), as a vertex in $\mathcal{G}_c$, associated with a vertex $v \in V$ can then be denoted as $[v]_{\mathcal{G}_c} := \{p \in \mathcal{G} : p \sim v\}$, and we let the set of clusters $V_c = V_\sim = \{[v]_{\mathcal{G}_c} : v \in V\}$. If no confusion arises, we will drop the subscript $\mathcal{G}_c$ and simply use $[v]$ to denote a cluster in $\mathcal{G}$ with respect to the coarse-grained graph $\mathcal{G}_c$. We denote by $\|[v]\|$ or $\#[v]$ the number of vertices in the cluster $[v]$.

Let J, J_0, $J \ge J_0$ be two integers. A *coarse-grained chain* $\mathcal{G}_{J \to J_0} := (\mathcal{G}_J, \mathcal{G}_{J-1}, \ldots, \mathcal{G}_{J_0})$ of $\mathcal{G}$ is a sequence of graphs with $\mathcal{G}_J \equiv \mathcal{G}$ such that each $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$ is a coarse-grained graph of $\mathcal{G}$ for all $J_0 \le j \le J$, and $[v]_{\mathcal{G}_j} \subseteq [v]_{\mathcal{G}_{j-1}}$ for all $j = J_0 + 1, \ldots, J$ and all $v \in \mathcal{G}$. Here $[v]_{\mathcal{G}_j}$ is the set of nodes of $\mathcal{G}$ at level j that contains node v, and can also be viewed as an (equivalence class) cluster of any finer level. Note that in the coarse-grained chain $\mathcal{G}_{J \to J_0}$, for any $p, v \in \mathcal{G}$, $[p]_{\mathcal{G}_j}$ is either belonging to $[v]_{\mathcal{G}_j}$ or has no intersection with $[v]_{\mathcal{G}_j}$. So the children at a finer level of any cluster $[v]_{\mathcal{G}_j}$ form a partition of the node set $[v]_{\mathcal{G}_j}$.

We call $\mathcal{G}_j$ *the level-j graph* of the chain $\mathcal{G}_{J \to J_0}$. The $\mathcal{G}_{j-1}$ can be viewed as a *coarse-grained graph* of $\mathcal{G}_j$ for $j = J_0 + 1, \ldots, J$. For convenience of discussion, we treat each vertex v of the finest level graph $\mathcal{G}_J \equiv \mathcal{G}$ as a cluster of singleton, that is, $[v]_{\mathcal{G}_J} = \{v\}$. We say the vertices of the graph on each level of the chain are *nodes* of the chain. When $\mathcal{G}_{J_0}, \ldots, \mathcal{G}_J$ are not all equal, we call $\mathcal{G}_{J \to J_0}$ a *decimated chain*. See Figure 1 for an illustration of a coarse-grained chain.

When $\mathcal{G}_j \equiv \mathcal{G}$ for all $j = J_0, \ldots, J$, we call $\mathcal{G}_{J \to J_0}$ an *undecimated chain* of $\mathcal{G}$. When there is only one vertex in the coarsest graph $\mathcal{G}_{J_0}$, that is, $V_{J_0} = \{[v]_{\mathcal{G}_{J_0}}\} = \{V\}$, we call $\mathcal{G}_{J \to J_0}$ a *tree* and denote it by $\mathcal{T}$. The $\{V\}$ is the root of $\mathcal{T}$ at the top level J_0, and all vertices of $\mathcal{G}$ are the *leaves* at the bottom level J. The vertices of $\mathcal{G}_j$ (clusters of $\mathcal{G}$) are the *nodes* of the tree at level j. The node $[v]$ is called the parent of v.

3.2 Orthonormal Bases on Graphs and Chains

Let $l_2(\mathcal{G}) := l_2(\mathcal{G}, \langle \cdot, \cdot \rangle_{\mathcal{G}})$ be the Hilbert space of *vectors* $\boldsymbol{f} : V \to \mathbb{C}$ on the graph $\mathcal{G}$ equipped with the inner product

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\mathcal{G}} := \sum_{v \in V} \boldsymbol{f}(v)\overline{\boldsymbol{g}(v)}, \quad \boldsymbol{f}, \boldsymbol{g} \in l_2(\mathcal{G}),$$

where $\overline{\boldsymbol{g}}$ is the complex conjugate to $\boldsymbol{g}$. The induced norm $\|\cdot\|_{\mathcal{G}}$ is then given by $\|\boldsymbol{f}\|_{\mathcal{G}} := \sqrt{\langle \boldsymbol{f}, \boldsymbol{f} \rangle_{\mathcal{G}}}$ for $\boldsymbol{f} \in l_2(\mathcal{G})$. For simplicity, we shall drop the subscript $\mathcal{G}$, and use $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$ if no confusion arises.

Let $\delta_{\ell,\ell'}$ be the *Kronecker delta* satisfying $\delta_{\ell,\ell'} = 1$ if $\ell = \ell'$ and $\delta_{\ell,\ell'} = 0$ if $\ell \neq \ell'$, and $N := |V|$ the total number of vertices of the graph $\mathcal{G}$. A finite subset $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ of $l_2(\mathcal{G})$ is said an *orthonormal basis* for $l_2(\mathcal{G})$ if

$$\langle \boldsymbol{u}_\ell, \boldsymbol{u}_{\ell'} \rangle = \delta_{\ell,\ell'}, \quad 1 \leq \ell, \ell' \leq N.$$

Let $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ be an orthonormal basis for $l_2(\mathcal{G})$. For $\ell = 1, \ldots, N$, let

$$\widehat{\boldsymbol{f}}_\ell := \langle \boldsymbol{f}, \boldsymbol{u}_\ell \rangle$$

be the (generalized) *Fourier coefficient* of degree ℓ for $\boldsymbol{f} \in l_2(\mathcal{G})$ with respect to $\boldsymbol{u}_\ell$. Let $\widehat{\boldsymbol{f}}_\ell := (\widehat{\boldsymbol{f}}_1, \ldots, \widehat{\boldsymbol{f}}_N) \in \mathbb{C}^N$ be the sequence of the Fourier coefficients for $\boldsymbol{f}$. Then, $\boldsymbol{f} = \sum_{\ell=1}^N \widehat{\boldsymbol{f}}_\ell \boldsymbol{u}_\ell$ for all $f \in l_2(\mathcal{G})$, and Parseval's identity holds: $\|\boldsymbol{f}\|^2 = \sum_{\ell=1}^N |\widehat{\boldsymbol{f}}_\ell|^2$ for all $\boldsymbol{f} \in l_2(\mathcal{G})$. We say $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ is an *orthonormal eigen-pair* for $l_2(\mathcal{G})$ if $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ is an orthonormal basis for $l_2(\mathcal{G})$ with $\boldsymbol{u}_1 \equiv 1/\sqrt{N}$ and $\{\lambda_\ell\}_{\ell=1}^N \subseteq \mathbb{R}$ is a nondecreasing sequence of nonnegative numbers satisfying $0 = \lambda_1 \leq \cdots \leq \lambda_N$. A typical example is the *eigen-pairs*, that is, the set of all pairs of the eigenvectors and eigenvalues of the *graph Laplacian* on $\mathcal{G}$. The (combinatorial or unnormalized) *graph Laplacian* operator $\mathcal{L} : l_2(\mathcal{G}) \to l_2(\mathcal{G})$ is

$$[\mathcal{L}\boldsymbol{f}](p) := \boldsymbol{d}(p)\boldsymbol{f}(p) - \sum_{v \in V} \boldsymbol{w}(p,v)\boldsymbol{f}(v), \quad p \in V, \ \boldsymbol{f} \in l_2(\mathcal{G}). \tag{1}$$

One can verify that $\langle \boldsymbol{f}, \mathcal{L}\boldsymbol{f} \rangle = \frac{1}{2}\sum_{p,v} \boldsymbol{w}(p,v)|\boldsymbol{f}(p) - \boldsymbol{f}(v)|^2 \geq 0$. The eigenvalues $\widetilde{\lambda}_\ell$ of $\mathcal{L}$ are then nonnegative, and associated with eigenvectors $\boldsymbol{u}_\ell : \mathcal{L}\boldsymbol{u}_\ell = \widetilde{\lambda}_\ell \boldsymbol{u}_\ell$, $\ell = 1, \ldots, N$. For simplicity, we need to take the square root of eigenvalue to define $\lambda_\ell := \sqrt{\widetilde{\lambda}_\ell}$, which is to meet the construction of framelets (see below). To be precise, the set of eigenvectors $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ then forms an orthonormal basis for $l_2(\mathcal{G})$ satisfying $0 = \lambda_1 \leq \ldots \leq \lambda_N$ and $\boldsymbol{u}_1 \equiv 1/\sqrt{N}$. An orthonormal eigen-pair can be deduced from other positive semi-definite operators on $l_2(\mathcal{G})$, for example, diffusion operators (Maggioni and Mhaskar, 2008), or by interval decomposition method, such as Haar orthonormal basis (Chui et al., 2015).

Let $\mathcal{G}_{J \to J_0} := (G_J, \ldots, G_{J_0})$ be a chain of the graph $\mathcal{G}$ with N vertices. Let $l(\mathcal{G}_{J \to J_0})$ be the set of all vectors $\boldsymbol{f}$ defined on the union of vertices at all levels $V_J \cup \cdots \cup V_{J_0}$.

Definition 1 *A set of pairs of vectors and complex numbers $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ in $l(\mathcal{G}_{J \to J_0})$ is called an orthonormal basis for the chain $\mathcal{G}_{J \to J_0}$ if the restriction $\{(\boldsymbol{u}_\ell|_{\mathcal{G}_j}, \lambda_\ell)\}_{\ell=1}^N$ on the jth level graph $\mathcal{G}_j$ is an orthonormal basis for $l_2(\mathcal{G}_j)$ at each level $j = J_0, \ldots, J$.*

In Sections 5.2 and 5.3, we will show the construction of an orthonormal basis for chain $\mathcal{G}_{J \to J_0}$ by either graph Laplacian or interval decomposition.

3.3 Tight Frames and Filter Banks

Orthonormal bases as we mentioned above are non-redundant systems for $l_2(\mathcal{G})$. In this paper, we would mainly focus on *redundant systems* for $l_2(\mathcal{G})$ that are frames. Let $\{\boldsymbol{g}_\ell\}_{\ell=1}^M$

be a set of elements from $l_2(\mathcal{G})$. We say that $\{\boldsymbol{g}_\ell\}_{\ell=1}^M$ is a *frame* for $l_2(\mathcal{G})$ if there exist constants A and B, $0 < A \le B < \infty$ such that

$$A\,\|\boldsymbol{f}\|^2 \le \sum_{\ell=1}^M |\,\langle \boldsymbol{f}, \boldsymbol{g}_\ell \rangle\,|^2 \le B\,\|\boldsymbol{f}\|^2 \quad \forall \boldsymbol{f} \in l_2(\mathcal{G}). \tag{2}$$

Here, we call A, B the *frame bounds*. When the frame bounds $A = B = 1$, we say $\{\boldsymbol{g}_\ell\}_{\ell=1}^M$ a *tight frame* for $l_2(\mathcal{G})$, and by the polarization identity, (2) is then equivalent to

$$\boldsymbol{f} = \sum_{\ell=1}^M \langle \boldsymbol{f}, \boldsymbol{g}_\ell \rangle\,\boldsymbol{g}_\ell. \tag{3}$$

When $\{\boldsymbol{g}_\ell\}_{\ell=1}^M$ is a tight frame and $\|\boldsymbol{g}_\ell\| = 1$ for $\ell = 1, \ldots, M$, we must have $M = N$ and $\{\boldsymbol{g}_\ell\}_{\ell=1}^N$ an orthonormal basis for $l_2(\mathcal{G})$. Tight frames are of significance as we can use coefficients $\langle \boldsymbol{f}, \boldsymbol{g}_\ell \rangle$ to represent the vector $\boldsymbol{f}$. See Daubechies (1992).

Let $\Psi := \{\alpha; \beta^{(1)}, \ldots, \beta^{(r)}\}$ be a set of functions in $L_1(\mathbb{R})$, which is the space of absolutely integrable functions on $\mathbb{R}$ with respect to the Lebesgure measure. The *Fourier transform* $\widehat{\gamma}$ of a function $\gamma \in L_1(\mathbb{R})$ is defined by $\widehat{\gamma}(\xi) := \int_{\mathbb{R}} \gamma(t)e^{-2\pi i t\xi}\,\mathrm{d}t$, $\xi \in \mathbb{R}$. The Fourier transform on $L_1(\mathbb{R})$ can be naturally extended to the space $L_2(\mathbb{R})$ of square integrable functions on $\mathbb{R}$. See, for example, Stein and Shakarchi (2011).

A *filter (or mask)* $h := \{h_k\}_{k\in\mathbb{Z}} \subseteq \mathbb{C}$ is a complex-valued sequence in $l_1(\mathbb{Z}) := \{h = \{h_k\}_{k\in\mathbb{Z}} \subseteq \mathbb{C} : \sum_{k\in\mathbb{Z}} |h_k| < \infty\}$. A *filter bank* is a set of filters. The *Fourier series* of a sequence $\{h_k\}_{k\in\mathbb{Z}}$ is the 1-periodic function $\widehat{h}(\xi) := \sum_{k\in\mathbb{Z}} h_k e^{-2\pi i k\xi}$, $\xi \in \mathbb{R}$. Let $\Psi_j = \{\alpha; \beta^{(1)}, \ldots, \beta^{(r)}\}$ be a set of *framelet generators* associated with a filter bank $\boldsymbol{\eta} := \{a; b^{(1)}, \ldots, b^{(r)}\}$, where the Fourier transforms of the functions in Ψ and the Fourier series of the filters in $\boldsymbol{\eta}$ satisfy

$$\widehat{\alpha}(2\xi) = \widehat{a}(\xi)\widehat{\alpha}(\xi), \quad \widehat{\beta^{(n)}}(2\xi) = \widehat{b^{(n)}}(\xi)\widehat{\alpha}(\xi), \quad n = 1, \ldots, r,\ \xi \in \mathbb{R}. \tag{4}$$

The first equation in (4) is called the *refinement equation*. The a is called *low-pass filter* or *refinement mask* and $b^{(n)}$, $n = 1, \ldots, r$ are called *high-pass filters* or *framelet masks*. One can then consider (stationary non-homogeneous) *affine system* of the form

$$\mathsf{AS}_J(\Psi) = \{\alpha(2^J \cdot -k) : k \in \mathbb{Z}\} \cup \{\beta^{(n)}(2^j \cdot -k) : k \in \mathbb{Z}, n = 1, \ldots, r, j \ge J\}. \tag{5}$$

Under certain extension principles such as the unitary extension principle (UEP) (Ron and Shen, 1997a; Daubechies et al., 2003), one can build the affine system $\mathsf{AS}_J(\Psi)$ that is a tight frame for $L_2(\mathbb{R})$. We call the elements $\mathsf{AS}_J(\Psi)$ *tight framelets* for $L_2(\mathbb{R})$ in this case. More generally, one can consider the (non-stationary, non-homogeneous) affine system

$$\mathsf{AS}_J(\{\Psi_j\}_{j=J}^\infty) = \{\alpha_j(2^J \cdot -k) : k \in \mathbb{Z}\} \cup \{\beta_j^{(n)}(2^j \cdot -k) : k \in \mathbb{Z}, n = 1, \ldots, r_j, j \ge J\}, \tag{6}$$

where $\Psi_j := \{\alpha_j; \beta_j^{(1)}, \ldots, \beta_j^{(r_j)}\} \subset L_2(\mathbb{R})$ are framelet generators at scale j. Examples of such (non-stationary, non-homogeneous) affine systems include framelets on $\mathbb{R}^d$ (Han, 2012; Zhuang, 2016), and framelets on compact Riemannian manifolds (Wang and Zhuang, 2018).

In the paper, we use the same "$\widehat{\cdot}$" notation for Fourier coefficients, Fourier transforms and Fourier series for simplicity. For a set Ω, let $l_2(\Omega)$ represent the set of all l_2-summable sequences on Ω and $l(\Omega)$ be the set of all complex-valued sequences supported on Ω. For a finite set Ω, a complex-valued sequence supported on Ω is a l_2-summable sequence. Thus, $l(\Omega) \equiv l_2(\Omega)$. Then, we only use the notation $l_2(\Omega)$ when we discuss a space of sequences on a finite set Ω.

4. Characterization of Tight Framelets on $\mathcal{G}$

In this section, we construct undecimated and decimated framelet systems on a graph and give equivalence characterizations for the tightness of them.

Given a set of orthonormal eigen-pairs for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of the graph $\mathcal{G}$, we can define a (stationary) *undecimated framelet system* on $\mathcal{G}$ by a set of framelet generators $\Psi := \{\alpha; \beta^{(1)}, \ldots, \beta^{(r)}\}$ and a filter bank $\boldsymbol{\eta} := \{a; b^{(1)}, \ldots, b^{(r)}\}$. Alternatively, we can construct a *decimated framelet system* with a sequence of framelet generator sets $\Psi_J, \ldots, \Psi_{J_0}$ and a sequence of filter banks $\boldsymbol{\eta}_J, \ldots, \boldsymbol{\eta}_{J_0}$. Here, we use "undecimated" when the framelet system is invariant on all levels of the coarse-grained chain, and "decimated" when a sequence of framelets has different scales at different levels of a chain. Each framelet generator and the corresponding filter bank are associated with the graph of a particular level.

4.1 Undecimated Tight Framelets on $\mathcal{G}$

The undecimated framelets on a graph can be seen as framelets on the discretised manifold, which are important counterpart to compact Riemannian manifolds (Wang and Zhuang, 2018). Here, we give detailed construction of undecimated tight framelets on a graph.

The construction of framelets uses the graph spectrum and framelet generators. Let $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ be orthonormal eigen-pairs for $l_2(\mathcal{G})$ and $\Psi = \{\alpha; \beta^{(1)}, \ldots, \beta^{(r)}\}$ be a set of functions in $L_1(\mathbb{R})$ associated with a filter bank $\boldsymbol{\eta} = \{a; b^{(1)}, \ldots, b^{(r)}\}$ satisfying (4). For $j \in \mathbb{Z}$ and $p \in V$, the *undecimated framelets* $\boldsymbol{\varphi}_{j,p}(v)$ and $\boldsymbol{\psi}_{j,p}^n(v)$, $v \in V$ at scale j are *filtered Bessel kernels* (or summability kernels)

$$
\begin{aligned}
\boldsymbol{\varphi}_{j,p}(v) &:= \sum_{\ell=1}^N \widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right) \overline{\boldsymbol{u}_\ell(p)} \boldsymbol{u}_\ell(v), \\
\boldsymbol{\psi}_{j,p}^n(v) &:= \sum_{\ell=1}^N \widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right) \overline{\boldsymbol{u}_\ell(p)} \boldsymbol{u}_\ell(v), \quad n = 1, \ldots, r.
\end{aligned}
\tag{7}
$$

See for example, Brauchart et al. (2015); Maggioni and Mhaskar (2008). Here, j and p in $\boldsymbol{\varphi}_{j,p}(v)$ and $\boldsymbol{\psi}_{j,p}^n(v)$ indicate the "dilation" at scale j and the "translation" at a vertex $p \in V$. They are analogues of those of wavelets in $\mathbb{R}^d$. The functions α, β^n of Ψ are called *framelet generators* or *scaling functions* for the undecimated framelet system.

Let $J, J_1, J > J_1$ be two integers. An *undecimated framelet system* $\mathsf{UFS}(\Psi, \boldsymbol{\eta}; \mathcal{G})$ (starting from a scale J_1) is a (non-homogeneous, stationary) affine system:

$$
\begin{aligned}
\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}) &:= \mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}; \mathcal{G}) \\
&:= \{\boldsymbol{\varphi}_{J_1,p} : p \in V\} \cup \{\boldsymbol{\psi}_{j,p}^n : p \in V, n = 1, \ldots, r, j = J_1, \ldots, J\}.
\end{aligned}
\tag{8}
$$

The system $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta})$ is then called an *undecimated tight frame* for $l_2(\mathcal{G})$ and the elements in $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta})$ are called *undecimated tight framelets* on $\mathcal{G}$.

The sequence $\{\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}) : J_1 = J_0, J_0 + 1, \ldots, J\}$ of undecimated framelet systems provides a tool for the multiresolution analysis on $l_2(\mathcal{G})$, which passes messages between finer and coarser scales of a chain on $\mathcal{G}$. The precision of *transforms* is ensured by the tightness of the framelet system. The following theorem gives important equivalence conditions for the tightness of a sequence of undecimated framelet systems.

Theorem 2 *Let $\mathcal{G} = (V, E, \boldsymbol{w})$ be a graph and $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ a set of orthonormal eigenpairs for $l_2(\mathcal{G})$. Let $\Psi = \{\alpha; \beta^{(1)}, \ldots, \beta^{(r)}\}$ be a set of functions in $L_1(\mathbb{R})$ associated with a filter bank $\boldsymbol{\eta} = \{a; b^{(1)}, \ldots, b^{(r)}\}$ satisfying (4). Let J_0, J, $J_0 \leq J$, be two integers, which indicate the coarsest scale and the finest scale of a chain. Let $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}; \mathcal{G}), J_1 = J_0, \ldots, J$, be an undecimated framelet system given in (8) with framelets $\boldsymbol{\varphi}_{j,p}$ and $\boldsymbol{\psi}_{j,p}^n$ given in (7). Then, the following statements are equivalent.*

(i) *The undecimated framelet system $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}; \mathcal{G})$ is a tight frame for $l_2(\mathcal{G})$ for each $J_1 = J_0, \ldots, J$. That is,*

$$\|\boldsymbol{f}\|^2 = \sum_{p \in V} \left|\langle \boldsymbol{f}, \boldsymbol{\varphi}_{J_1,p}\rangle\right|^2 + \sum_{j=J_1}^J \sum_{n=1}^r \sum_{p \in V} \left|\langle \boldsymbol{f}, \boldsymbol{\psi}_{j,p}^n\rangle\right|^2 \quad \forall \boldsymbol{f} \in l_2(\mathcal{G}), J_1 = J_0, \ldots, J. \quad (9)$$

(ii) *For all $\boldsymbol{f} \in l_2(\mathcal{G})$ and for $j = J_0, \ldots, J-1$, the following identities hold:*

$$\boldsymbol{f} = \sum_{p \in V} \langle \boldsymbol{f}, \boldsymbol{\varphi}_{J,p}\rangle \boldsymbol{\varphi}_{J,p} + \sum_{n=1}^r \sum_{p \in V} \langle \boldsymbol{f}, \boldsymbol{\psi}_{J,p}^n\rangle \boldsymbol{\psi}_{J,p}^n, \quad (10)$$

$$\sum_{p \in V} \langle \boldsymbol{f}, \boldsymbol{\varphi}_{j+1,p}\rangle \boldsymbol{\varphi}_{j+1,p} = \sum_{p \in V} \langle \boldsymbol{f}, \boldsymbol{\varphi}_{j,p}\rangle \boldsymbol{\varphi}_{j,p} + \sum_{n=1}^r \sum_{p \in V} \langle \boldsymbol{f}, \boldsymbol{\psi}_{j,p}^n\rangle \boldsymbol{\psi}_{j,p}^n. \quad (11)$$

(iii) *For all $\boldsymbol{f} \in l_2(\mathcal{G})$ and for $j = J_0, \ldots, J-1$, the following identities hold:*

$$\|\boldsymbol{f}\|^2 = \sum_{p \in V} |\langle \boldsymbol{f}, \boldsymbol{\varphi}_{J,p}\rangle|^2 + \sum_{n=1}^r \sum_{p \in V} |\langle \boldsymbol{f}, \boldsymbol{\psi}_{J,p}^n\rangle|^2, \quad (12)$$

$$\sum_{p \in V} |\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j+1,p}\rangle|^2 = \sum_{p \in V} |\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j,p}\rangle|^2 + \sum_{n=1}^r \sum_{p \in V} |\langle \boldsymbol{f}, \boldsymbol{\psi}_{j,p}^n\rangle|^2. \quad (13)$$

(iv) *The functions in Ψ satisfy*

$$1 = \left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^J}\right)\right|^2 + \sum_{n=1}^r \left|\widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^J}\right)\right|^2 \quad \forall \ell = 1, \ldots, N, \quad (14)$$

$$\left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^{j+1}}\right)\right|^2 = \left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 + \sum_{n=1}^r \left|\widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 \quad \forall \begin{array}{l} \ell = 1, \ldots, N, \\ j = J_0, \ldots, J-1. \end{array} \quad (15)$$

14

(v) *The identities in* (14) *hold and the filters in the filter bank* $\boldsymbol{\eta}$ *satisfy*

$$\left|\widehat{a}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 + \sum_{n=1}^{r}\left|\widehat{b^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 = 1 \quad \forall \ell \in \sigma_\alpha^{(j)}, \ j = J_0 + 1, \dots, J, \qquad (16)$$

where

$$\sigma_\alpha^{(j)} := \left\{\ell \in \{1, \dots, N\} : \widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right) \neq 0\right\}.$$

Proof (i)$\Longleftrightarrow$(ii). Let $\mathcal{V}_j := \mathrm{span}\{\varphi_{j,p} : p \in V\}$ and $\mathcal{W}_j^n := \mathrm{span}\{\psi_{j,p}^n : p \in V\}$. Define projections $\mathbf{P}_{\mathcal{V}_j}, \mathbf{P}_{\mathcal{W}_j^n}, \ n = 1, \dots, r$ as

$$\mathbf{P}_{\mathcal{V}_j}(\boldsymbol{f}) := \sum_{p \in V}\langle \boldsymbol{f}, \varphi_{j,p}\rangle\, \varphi_{j,p}, \quad \mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f}) := \sum_{p \in V}\langle \boldsymbol{f}, \psi_{j,p}^n\rangle\, \psi_{j,p}^n, \quad \boldsymbol{f} \in l_2(\mathcal{G}). \qquad (17)$$

Since $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta})$ is a tight frame for $l_2(\mathcal{G})$ for $J_1 = J_0, \dots, J$, by polarization identity,

$$\boldsymbol{f} = \mathbf{P}_{\mathcal{V}_{J_1}}(\boldsymbol{f}) + \sum_{j=J_1}^{J}\sum_{n=1}^{r}\mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f}) = \mathbf{P}_{\mathcal{V}_{J_1+1}}(\boldsymbol{f}) + \sum_{j=J_1+1}^{J}\sum_{n=1}^{r}\mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f}) \qquad (18)$$

for all $\boldsymbol{f} \in l_2(\mathcal{G})$ and for all $J_1 = J_0, \dots, J$. Thus, for $J_1 = J_0, \dots, J-1$, we have

$$\mathbf{P}_{\mathcal{V}_{J_1+1}}(\boldsymbol{f}) = \mathbf{P}_{\mathcal{V}_{J_1}}(\boldsymbol{f}) + \sum_{n=1}^{r}\mathbf{P}_{\mathcal{W}_{J_1}^n}(\boldsymbol{f}), \qquad (19)$$

which is (11). Moreover, when $J_1 = J$, (18) gives (10). Consequently, (i)$\Longrightarrow$(ii). Conversely, recursively using (19) gives

$$\mathbf{P}_{\mathcal{V}_{m+1}}(\boldsymbol{f}) = \mathbf{P}_{\mathcal{V}_{J_1}}(\boldsymbol{f}) + \sum_{j=J_1}^{m}\sum_{n=1}^{r}\mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f}) \qquad (20)$$

for all $J_1 \leq m \leq J-1$. Taking $m = J-1$ together with (10), we deduce (18), which is equivalent to (9). Thus, (ii)$\Longrightarrow$(i).

(ii)$\Longleftrightarrow$(iii). The equivalence between (ii) and (iii) simply follows from the polarization identity.

(ii)$\Longleftrightarrow$(iv). By the orthonormality of $\boldsymbol{u}_\ell$,

$$\langle \boldsymbol{f}, \varphi_{j,p}\rangle = \sum_{\ell=1}^{N}\overline{\widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right)}\widehat{\boldsymbol{f}}_\ell\, \boldsymbol{u}_\ell(p), \quad \langle \boldsymbol{f}, \psi_{j,p}^n\rangle = \sum_{\ell=1}^{N}\overline{\widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right)}\widehat{\boldsymbol{f}}_\ell\, \boldsymbol{u}_\ell(p),$$

where $\widehat{\boldsymbol{f}}_\ell = \langle \boldsymbol{f}, \boldsymbol{u}_\ell\rangle$ is the Fourier coefficient of $\boldsymbol{f}$ with respect to $\boldsymbol{u}_\ell$. This together with (17) and (7) gives, for $j \geq J_0$ and $n = 1, \dots, r$, the Fourier coefficients for the projections $\mathbf{P}_{\mathcal{V}_j}(\boldsymbol{f})$ and $\mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f})$:

$$\left(\widehat{\mathbf{P}_{\mathcal{V}_j}(\boldsymbol{f})}\right)_\ell = \left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 \widehat{\boldsymbol{f}}_\ell, \quad \left(\widehat{\mathbf{P}_{\mathcal{W}_j^n}(\boldsymbol{f})}\right)_\ell = \left|\widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 \widehat{\boldsymbol{f}}_\ell, \quad \ell = 1, \dots, N, \qquad (21)$$

15

which implies that (10) and (11) are equivalent to (14) and (15) respectively. Thus, (ii)$\Longleftrightarrow$(iv).

(iv)$\Longleftrightarrow$(v). By the relation in (4), it can be deduced that for $\ell \geq 0$ and $j \geq J_0$,

$$\left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 + \sum_{n=1}^r \left|\widehat{\beta^{(n)}}\left(\frac{\lambda_\ell}{2^j}\right)\right|^2 = \left(\left|\widehat{a}\left(\frac{\lambda_\ell}{2^{j+1}}\right)\right|^2 + \sum_{n=1}^r \left|\widehat{b^{(n)}}\left(\frac{\lambda_\ell}{2^{j+1}}\right)\right|^2\right)\left|\widehat{\alpha}\left(\frac{\lambda_\ell}{2^{j+1}}\right)\right|^2.$$

This shows that (15) is equivalent to (16). Therefore, (iv)$\Longleftrightarrow$(v). ∎

When $\widehat{\alpha}(\xi) = 1$ in a neighborhood of the origin, the system in (8) is simplified by the following property. Let $\boldsymbol{\delta}_p \in l_2(\mathcal{G})$ be the *Dirac sequence* defined by that $\boldsymbol{\delta}_p(p) = 1$ and $\boldsymbol{\delta}_p(v) = 0$ if $p \neq v$

Proposition 3 *Suppose there exists a constant $C > 0$ such that*

$$\widehat{\alpha}(\xi) \equiv 1 \quad \forall|\xi| \leq C. \tag{22}$$

Then, for sufficiently large j, we have $\boldsymbol{\varphi}_{j,p} = \boldsymbol{\delta}_p$.

Proof In view of (22), we have for sufficient large j, $\widehat{\alpha}(\lambda_\ell/2^j) \equiv 1$ for all $\ell = 1, \ldots, N$. Then, by the orthonormality of $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$, we have

$$\boldsymbol{\varphi}_{j,p}(v) = \sum_{\ell=1}^N \overline{\boldsymbol{u}_\ell(p)}\boldsymbol{u}_\ell(v) = \boldsymbol{\delta}_p(v), \quad v \in V.$$

∎

Hence, when (22) holds and J_1 is sufficiently large, the undecimated framelet system $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta})$ defined in (8) becomes

$$\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta}) \equiv \{\boldsymbol{\delta}_p : p \in V\},$$

which is a trivial orthonormal basis for $l_2(\mathcal{G})$. We observe from this case that the framelet systems become more and more localized as the level J_1 increases.

4.2 Decimated Tight Framelets on $\mathcal{G}$

In this section, we introduce a *decimated tight framelet system* on a chain of a graph $\mathcal{G}$. The scaling of a framelet is associated with the level of the associated graph in the chain. The framelets at level j ($j = J_0, \ldots, J$) take the vertices of the jth-level graph of the chain as their transition points. By doing this, the number of framelets equals to the nodes of the chain. The redundancy of the representation system is then determined by the clustering that constructed the chain of the graph. The decimated framelet system, compared with the undecimated framelet system, compresses the graph size at a coarser scale and saves the storage while no information in the framelet representation is lost.

Let $\mathcal{G} = (V, E, \boldsymbol{w})$ be a graph and $\mathcal{G}_{J \to J_0} := (\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$ be a coarse-grained chain of $\mathcal{G}$. For each vertex $[p]$ in $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$, we assign a real number $\omega_{j,[p]} \in \mathbb{R}$, which we call the *(associated) weight*. At the bottom level with $j = J$, we let $\omega_{J,[p]_{\mathcal{G}_J}} \equiv 1$ for all $[p]_{\mathcal{G}_J} = \{p\}$ in V_J. Let $\mathcal{Q}_j := \{\omega_{j,[p]} : [p] \in V_j\}$ be the set of weights on $\mathcal{G}_j$ and $\mathcal{Q}_{J \to J_0} := (\mathcal{Q}_J, \ldots, \mathcal{Q}_{J_0})$ be the sequence of sets of weights for the coarse-grained chain $\mathcal{G}_{J \to J_0}$.

Definition 4 (Scaling functions) *Let $\Psi_j = \{\alpha_j; \beta_j^{(1)}, \ldots, \beta_j^{(r_j)}\}$ be a set of functions in $L_1(\mathbb{R})$ at scale j for $j = J_0, \ldots, J$. We link the framelet generators in Ψ_j and Ψ_{j-1} by a filter bank $\boldsymbol{\eta}_j := \{a_j; b_j^{(1)}, \ldots, b_j^{(r_{j-1})}\}$ in that, for $\xi \in \mathbb{R}$ and $0 < \Lambda_{J_0} \leq \Lambda_{J_0+1} \leq \cdots \leq \Lambda_J < \infty$,*

$$
\begin{aligned}
\widehat{\alpha_{j-1}}(\xi/\Lambda_{j-1}) &= \widehat{a_j}(\xi/\Lambda_j)\widehat{\alpha_j}(\xi/\Lambda_j), \\
\widehat{\beta_{j-1}^{(n)}}(\xi/\Lambda_{j-1}) &= \widehat{b_j^{(n)}}(\xi/\Lambda_j)\widehat{\alpha_j}(\xi/\Lambda_j), \quad n = 1, \ldots, r_{j-1},
\end{aligned}
\tag{23}
$$

where $\Lambda_{J_0}, \Lambda_{J_0+1}, \ldots, \Lambda_J$ are called scaling factors. *We call $\alpha_j; \beta_j^{(1)}, \ldots, \beta_j^{(r_j)}$ are the framelet generators or scaling functions of level j.*

Definition 5 (Decimated framelets) *The decimated framelets $\boldsymbol{\varphi}_{j,[p]}(v)$ and $\boldsymbol{\psi}_{j,[p]}^n(v)$, $p, v \in V$, at scale $j = J_0, \ldots, J$ for the chain $\mathcal{G}_{J \to J_0}$ of the graph $\mathcal{G}$, framelet generators in (23) and a weights sequence $\mathcal{Q}_{J \to J_0}$ are defined by*

$$
\boldsymbol{\varphi}_{j,[p]}(v) := \sqrt{\omega_{j,[p]}} \sum_{\ell=1}^{N} \widehat{\alpha_j}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \overline{\boldsymbol{u}_\ell([p])}\boldsymbol{u}_\ell(v), \quad [p] \in V_j,
$$

$$
\boldsymbol{\psi}_{j,[p]}^n(v) := \sqrt{\omega_{j+1,[p]}} \sum_{\ell=1}^{N} \widehat{\beta_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \overline{\boldsymbol{u}_\ell([p])}\boldsymbol{u}_\ell(v), \quad [p] \in V_{j+1}, \ n = 1, \ldots, r_j,
\tag{24}
$$

where for $j = J$, we let $V_{J+1} := V_J$ and $\omega_{J+1,[p]} := \omega_{J,[p]}$. We call $\boldsymbol{\varphi}_{j,[p]}$ and $\boldsymbol{\psi}_{j,[p]}^n$ low-pass and high-pass (decimated) framelets at scale j.

The decimated framelets in Definition 5 are constructed based on the orthonormal basis associated with the chain $\mathcal{G}_{J \to J_0}$. Here, the function $\boldsymbol{u}_\ell([p]_{\mathcal{G}_j})$ can either be defined as $\boldsymbol{u}_\ell([p]_{\mathcal{G}_j}) := \frac{1}{\#[p]_{\mathcal{G}_j}} \sum_{v \in [p]_{\mathcal{G}_j}} \boldsymbol{u}_\ell(v)$, or $\boldsymbol{u}_\ell([p]_{\mathcal{G}_j}) := \max_{v \in [p]_{\mathcal{G}_j}} \boldsymbol{u}_\ell(v)$. In (24), the vertices $[p] = [p]_{\mathcal{G}_{j+1}} \in V_{j+1}$ for the high-pass framelet $\boldsymbol{\psi}_{j,[p]}^n$ of level j are at the $(j+1)$th level while the vertices $[p]_{\mathcal{G}_j}$ in V_j for $\boldsymbol{\varphi}_{j,[p]}$ of level j are at the jth level. These can be interpreted from the view of multiresolution analysis that $\boldsymbol{\psi}_{j,[p]}^n$ in fact lies in the scale $j+1$ while $\boldsymbol{\varphi}_{j,[p]}$ is in the scale j.

Definition 6 (Decimated framelet system) *The (decimated) framelet system $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$ on $\mathcal{G}$ (starting from a scale J_1) is a non-homogeneous, non-stationary affine system which is a set of low-pass and high-pass framelets given by*

$$
\begin{aligned}
\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J) &:= \mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J; \mathcal{G}_{J \to J_1}, \mathcal{Q}_{J \to J_1}) \\
&:= \{\boldsymbol{\varphi}_{J_1,[p]} : [p] \in V_{J_1}\} \cup \{\boldsymbol{\psi}_{j,[p]}^n : [p] \in V_{j+1}, \ j = J_1, \ldots, J\}.
\end{aligned}
\tag{25}
$$

For $\mathcal{Q}_j = \{\omega_{j,[p]} : [p] \in V_j\}$ on $\mathcal{G}_j$, we define

$$
\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) := \sum_{[p] \in V_j} \omega_{j,[p]} \, \boldsymbol{u}_\ell([p])\overline{\boldsymbol{u}_{\ell'}([p])}.
\tag{26}
$$

Note that $\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_J) = \delta_{\ell,\ell'}$ since $\omega_{J,[p]} \equiv 1$ and $[p]_{\mathcal{G}_J} = \{p\}$ is a singleton.

Similar to the undecimated setting, the following then gives equivalence characterizations for the tightness of a decimated framelet system $\{\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J) : J_1 = J_0, J_0 + 1, \ldots, J\}$.

Theorem 7 *Let* $\Psi_j := \{\alpha_j; \beta_j^{(1)}, \ldots, \beta_j^{(r_j)}\}$, $j = J_0, \ldots, J$ *be a sequence of framelet generators sets in* $L_1(\mathbb{R})$ *associated with a sequence of filter banks* $\boldsymbol{\eta}_j = \{a_j; b_j^{(1)}, \ldots, b_j^{(r_j-1)}\}$, $j = J_0 + 1, \ldots, J$, *see* (23). *Let* $\mathcal{G}_{J \to J_0}$ *be a coarse-grained chain of a graph* $\mathcal{G}$ *with a weight sequence* $\mathcal{Q}_{J \to J_0}$. *Let* $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$, $J_1 = J_0, \ldots, J$ *be a sequence of decimated framelet systems for the chain* $\mathcal{G}_{J \to J_0}$ *in Definition 6 with framelets in Definition 5 with framelet generators* Ψ_j. *Then, the following statements are equivalent.*

(i) *The decimated framelet system* $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$ *is a tight frame for* $l_2(\mathcal{G})$ *for all* $J_1 = J_0, \ldots, J$, *that is,*

$$\|\boldsymbol{f}\|^2 = \sum_{[p] \in V_{J_1}} \left| \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{J_1,[p]} \right\rangle \right|^2 + \sum_{j=J_1}^J \sum_{n=1}^{r_j} \sum_{[p] \in V_{j+1}} \left| \left\langle \boldsymbol{f}, \boldsymbol{\psi}_{j,[p]}^n \right\rangle \right|^2 \quad \forall \boldsymbol{f} \in l_2(\mathcal{G}), J_1 = J_0, \ldots, J.$$
(27)

(ii) *For* $\boldsymbol{f} \in l_2(\mathcal{G})$ *and* $j = J_0, \ldots, J - 1,$

$$\boldsymbol{f} = \sum_{[p] \in V_J} \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{J,[p]} \right\rangle \boldsymbol{\varphi}_{J,[p]} + \sum_{n=1}^{r_J} \sum_{[p] \in V_J} \left\langle \boldsymbol{f}, \boldsymbol{\psi}_{J,[p]}^n \right\rangle \boldsymbol{\psi}_{J,[p]}^n,$$
(28)

$$\sum_{[p] \in V_{j+1}} \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j+1,[p]} \right\rangle \boldsymbol{\varphi}_{j+1,[p]} = \sum_{[p] \in V_j} \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j,[p]} \right\rangle \boldsymbol{\varphi}_{j,[p]} + \sum_{n=1}^{r_j} \sum_{[p] \in V_{j+1}} \left\langle \boldsymbol{f}, \boldsymbol{\psi}_{j,[p]}^n \right\rangle \boldsymbol{\psi}_{j,[p]}^n.$$
(29)

(iii) *For* $\boldsymbol{f} \in l_2(\mathcal{G})$ *and* $j = J_0, \ldots, J - 1,$

$$\|\boldsymbol{f}\|^2 = \sum_{[p] \in V_J} \left| \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{J,[p]} \right\rangle \right|^2 + \sum_{n=1}^{r_J} \sum_{[p] \in V_J} \left| \left\langle \boldsymbol{f}, \boldsymbol{\psi}_{J,[p]}^n \right\rangle \right|^2,$$
(30)

$$\sum_{[p] \in V_{j+1}} \left| \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j+1,[p]} \right\rangle \right|^2 = \sum_{[p] \in V_j} \left| \left\langle \boldsymbol{f}, \boldsymbol{\varphi}_{j,[p]} \right\rangle \right|^2 + \sum_{n=1}^{r_j} \sum_{[p] \in V_{j+1}} \left| \left\langle \boldsymbol{f}, \boldsymbol{\psi}_{j,[p]}^n \right\rangle \right|^2.$$
(31)

(iv) *The framelet generators in* Ψ_j *and the weights in* $\mathcal{Q}_j$ *satisfy, for* $1 \le \ell, \ell' \le N,$

$$1 = \left| \widehat{\alpha_j} \left(\frac{\lambda_\ell}{\Lambda_J} \right) \right|^2 + \sum_{n=1}^{r_J} \left| \widehat{\beta_j^{(n)}} \left(\frac{\lambda_\ell}{\Lambda_J} \right) \right|^2,$$
(32)

$$\overline{\widehat{\alpha_{j+1}} \left(\frac{\lambda_\ell}{\Lambda_{j+1}} \right)} \widehat{\alpha_{j+1}} \left(\frac{\lambda_{\ell'}}{\Lambda_{j+1}} \right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j+1}) - \overline{\widehat{\alpha_j} \left(\frac{\lambda_\ell}{\Lambda_j} \right)} \widehat{\alpha_j} \left(\frac{\lambda_{\ell'}}{\Lambda_j} \right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$$

$$= \sum_{n=1}^{r_j} \overline{\widehat{\beta_j^{(n)}} \left(\frac{\lambda_\ell}{\Lambda_j} \right)} \widehat{\beta_j^{(n)}} \left(\frac{\lambda_{\ell'}}{\Lambda_j} \right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j+1}) \quad \forall j = J_0, \ldots, J - 1,$$
(33)

where $\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$ *is given by* (26).

18

(v) *The identities in* (32) *hold, and for all* $(\ell, \ell') \in \sigma_{\alpha,\overline{\alpha}}^{(j)}$ *and* $j = J_0 + 1, \ldots, J$,

$$\overline{\widehat{a}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{a}_j\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j-1}) + \sum_{n=1}^{r_{j-1}}\overline{\widehat{b}_j^{(n)}\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{b}_j^{(n)}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) = \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j), \quad (34)$$

where

$$\sigma_{\alpha,\overline{\alpha}}^{(j)} := \left\{(\ell, \ell') \in \mathbb{N} \times \mathbb{N} : \overline{\widehat{\alpha}\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\alpha}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right) \neq 0\right\}. \tag{35}$$

Proof The proofs of (i)$\Longleftrightarrow$(ii)$\Longleftrightarrow$(iii) are similar to those in Theorem 2. We next prove (iii)$\Longleftrightarrow$(iv)$\Longleftrightarrow$(v).

(iii) $\Longleftrightarrow$ (iv). For $\boldsymbol{f} \in l_2(\mathcal{G})$, by the the orthonormality of $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$, we have

$$\left\langle \boldsymbol{f}, \varphi_{j,[p]} \right\rangle = \sqrt{\omega_{j,[p]}} \sum_{\ell=1}^N \overline{\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\boldsymbol{f}}_\ell\, \boldsymbol{u}_\ell([p]), \quad [p] \in V_j.$$

$$\left\langle \boldsymbol{f}, \psi_{j,[p]}^n \right\rangle = \sqrt{\omega_{j+1,[p]}} \sum_{\ell=1}^N \overline{\widehat{\beta}_j^{(n)}\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\boldsymbol{f}}_\ell\, \boldsymbol{u}_\ell([p]), \quad [p] \in V_{j+1}.$$

$$(36)$$

Then,

$$\sum_{[p]\in V_j}\left|\left\langle \boldsymbol{f}, \varphi_{j,[p]} \right\rangle\right|^2 = \sum_{[p]\in V_j}\omega_{j,[p]}\left|\sum_{\ell=1}^N \overline{\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\boldsymbol{f}}_\ell\, \boldsymbol{u}_\ell([p])\right|^2$$

$$= \sum_{\ell=1}^N\sum_{\ell'=1}^N \widehat{\boldsymbol{f}}_\ell\overline{\widehat{\boldsymbol{f}}_{\ell'}}\,\overline{\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\alpha}_j\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\sum_{[p]\in V_j}\omega_{j,[p]}\boldsymbol{u}_\ell([p])\overline{\boldsymbol{u}_{\ell'}([p])}$$

$$= \sum_{\ell=1}^N\sum_{\ell'=1}^N \widehat{\boldsymbol{f}}_\ell\overline{\widehat{\boldsymbol{f}}_{\ell'}}\,\overline{\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\alpha}_j\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$$

Since $\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_J) = \delta_{\ell,\ell'}$,

$$\sum_{[p]\in V_J}\left|\left\langle \boldsymbol{f}, \varphi_{J,[p]} \right\rangle\right|^2 + \sum_{n=1}^{r_J}\sum_{[p]\in V_J}\left|\left\langle \boldsymbol{f}, \psi_{J,[p]}^n \right\rangle\right|^2 = \sum_{\ell=1}^N |\boldsymbol{f}_\ell|^2\left(\left|\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_J}\right)\right|^2 + \sum_{n=1}^{r_J}\left|\widehat{\beta}_j^{(n)}\left(\frac{\lambda_\ell}{\Lambda_J}\right)\right|^2\right),$$

which shows the equivalence between (30) and (32). Moreover, (31) can be rewritten as

$$\sum_{\ell=1}^N\sum_{\ell'=1}^N \widehat{\boldsymbol{f}}_\ell\overline{\widehat{\boldsymbol{f}}_{\ell'}}\,\overline{\widehat{\alpha}_{j+1}\left(\frac{\lambda_\ell}{\Lambda_{j+1}}\right)}\widehat{\alpha}_{j+1}\left(\frac{\lambda_{\ell'}}{\Lambda_{j+1}}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j+1})$$

$$= \sum_{\ell=1}^N\sum_{\ell'=1}^N \widehat{\boldsymbol{f}}_\ell\overline{\widehat{\boldsymbol{f}}_{\ell'}}\left[\overline{\widehat{\alpha}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\alpha}_j\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) + \sum_{n=1}^{r_j}\overline{\widehat{\beta}_j^{(n)}\left(\frac{\lambda_\ell}{\Lambda_j}\right)}\widehat{\beta}_j^{(n)}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right)\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j+1})\right],$$

which is equivalent to (33).

(iv) $\Longleftrightarrow$ (v). By (23), we have

$$
\left[\overline{\widehat{\alpha_{j-1}}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)} \widehat{\alpha_{j-1}}\left(\frac{\lambda_{\ell'}}{\Lambda_{j-1}}\right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j-1}) + \sum_{n=1}^{r_{j-1}} \overline{\widehat{\beta_{j-1}^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)} \widehat{\beta_{j-1}^{(n)}}\left(\frac{\lambda_{\ell'}}{\Lambda_{j-1}}\right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) \right]
$$

$$
= \left[\overline{\widehat{a_j}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \widehat{a_j}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_{j-1}) + \sum_{n=1}^{r_{j-1}} \overline{\widehat{b_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \widehat{b_j^{(n)}}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right) \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) \right] \overline{\widehat{\alpha_j}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \widehat{\alpha_j}\left(\frac{\lambda_{\ell'}}{\Lambda_j}\right),
$$

which implies (33)$\Longleftrightarrow$(34) and thus proves the equivalence between (iv) and (v). ∎

The next corollary shows a condition on $\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$ such that the conditions (iv) and (v) of Theorem 7 take a simplified form.

Corollary 8 *Retain all assumptions in Theorem 7, and in addition suppose that*

$$
\sigma_{\alpha,\overline{\alpha}}^{(j)} \subseteq \sigma_{\alpha,\overline{\alpha}}^{(j+1)} \quad and \quad \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) = \delta_{\ell,\ell'} \quad \forall (\ell,\ell') \in \sigma_{\alpha,\overline{\alpha}}^{(j)}, \; j = J_0, \ldots, J-1. \tag{37}
$$

Then, the following statements are equivalent.

(i) $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$ *is a tight frame for $l_2(\mathcal{G})$ for all $J_1 = J_0, \ldots, J$.*

(ii) *The identities in (32), and for $j = J_0, \ldots, J-1$ and $\ell = 1, \ldots, N$,*

$$
\left| \widehat{\alpha_{j+1}}\left(\frac{\lambda_\ell}{\Lambda_{j+1}}\right) \right|^2 = \left| \widehat{\alpha_j}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2 + \sum_{n=1}^{r_j} \left| \widehat{\beta_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2. \tag{38}
$$

(iii) *The identities in (32), and for $j = J_0+1, \ldots, J$ and $\ell = 1, \ldots, N$,*

$$
\left| \widehat{a_j}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2 + \sum_{n=1}^{r_{j-1}} \left| \widehat{b_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2 = 1. \tag{39}
$$

Proof If (37) holds, (33) and (34) are reduced to (38) and (39) respectively. The equivalence between (i), (ii) and (iii) follow from Theorem 7. ∎

Remark 9 *The characterization conditions in Theorems 2 and 7 are simplified when $\boldsymbol{\varphi}_{J+1,u}$ is the Dirac delta function δ_u.*

Undecimated and decimated framelets are both defined on a graph $\mathcal{G}$, but their construction utilises orthonormal eigenpairs for the l_2 spaces on the graph and the chain: $l(\mathcal{G})$ and $l(\mathcal{G}_{J \to J_0})$. One can view an undecimated framelet system as a decimated system on a special coarse-grained chain, when each level is identical to $\mathcal{G}$. Decimated framelets depend on the structure of the chain, that is, the connection between graphs at different levels of the chain. In the next section, we will show how to construct a coarse-grained chain of a graph, and study the impact of the chain structure on the generating set of the decimated framelet system.

5. Construction of Decimated Tight Framelets on $\mathcal{G}$

In the last section, we provide the characterization of undecimated and decimated framelet systems to be tight frames for $l_2(\mathcal{G})$. Based on the characterization, in this section, we detail the construction of tight decimated framelets on a given graph. We would first discuss the construction of coarse-grained chains of $\mathcal{G}$ and orthonormal eigen-pairs on $\mathcal{G}$ with desired properties. Based on the coarse-grained chains and orthonormal eigen-pairs, together with a careful design of generating set Ψ_j on $\mathbb{R}$, we then provide explicit construction of decimated tight framelets on $\mathcal{G}$.

5.1 Graph Clustering and Coarse-graining on $\mathcal{G}$

Graph partitioning and clustering are among central topics in the structured data analysis. The partitioned graph requires to contain various clusters for appropriate applications such as unsupervised or semi-supervised learning and data mining in various types of networks. One can use a top-down approach, which recursively splits a graph to subgraphs. This approach starts from the root (usually with one node) of a tree and proceeds to its branches (clusters) down to the bottom level. In contrast, a bottom-up approach groups the leaves of a tree of the bottom level and then clusters the nodes level by level up to the coarsest. Another way of categorization for graph clustering algorithms takes account of whether the spectral method is used. For example, the clustering algorithms by using the graph Laplacian is a spectral method, see Merriman et al. (1992); Bertozzi and Flenner (2012); Garcia-Cardona et al. (2014). The clustering algorithm that utilises the weight function $\boldsymbol{w}$ on a graph is a non-spectral method, see Coifman et al. (2005); Dongen (2000); Gavish and Coifman (2012); Gavish et al. (2010); Chaudhuri and Dasgupta (2010); Lafon and Lee (2006); Boykov et al. (2001); Chen et al. (2014).

For a graph of triplet $\mathcal{G} = (V, E, \boldsymbol{w})$, a clustering algorithm is usually based on a partition for $\mathcal{G}$. Suppose $V_c := \{[p] : p \in V\}$ the resulting partition with each $[p] \subseteq V$ a cluster on $\mathcal{G}$. To obtain a coarse-grained chain of $\mathcal{G}$, we can use the clustering algorithm hierarchically. In the first place, a connection relation between clusters needs to be defined on V_c. Following Lafon and Lee (2006), we define

$$\boldsymbol{w}_c([p], [v]) := \sum_{p \in [p]} \sum_{v \in [v]} \frac{\boldsymbol{w}(p, v)}{\text{vol}(\mathcal{G})} \quad \text{for } [p], [v] \in V_c. \tag{40}$$

Then, $\boldsymbol{w}_c$ becomes a weight function on $V_c \times V_c$. The weight $\boldsymbol{w}_c$ is symmetric on V_c and hence determines an (undirected) edge set E_c by $E_c := \{([p], [v]) : \boldsymbol{w}_c([p, v]) > 0\}$. We obtain $\mathcal{G}_c := (V_c, E_c, \boldsymbol{w}_c)$, which is called a *coarse-grained graph* of $\mathcal{G}$. Using the clustering algorithm for $\mathcal{G}_c$ then gives a coarse-grained graph $(\mathcal{G}_c)_c$ of $\mathcal{G}_c$. Recursively doing this step, we would obtain a *coarse-grained chain* of the original graph $\mathcal{G}$. We call the process of constructing a coarse-grained chain by a clustering algorithm the *Coarse-Grained Chain (CGC)* algorithm. The clustering algorithm that one uses determines the coarse-grained chain. Algorithm 1 shows an implementation of the CGC algorithm, which utilises a modified version of the non-spectral hierarchical clustering (NHC) algorithm given in Chui et al. (2018).

Algorithm 1: Coarse-Grained Chain Algorithm (CGC)

Input : Graph $\mathcal{G} = (V, E, \boldsymbol{w})$. The number of clusters on each level $(N_{J-1}, \ldots, N_{J_0})$ with $1 \leq N_{J_0} < N_{J_1} < \cdots < N_{J-1} < N$, where N is the number of vertices in $\mathcal{G}$.

Output: A coarse-grained chain $(\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$ of $\mathcal{G}$ with $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$ such that $|V_j| = N_j$ for $j = J - 1, \ldots, J_0$.

1   Initialization: $\mathcal{G}_J \leftarrow \mathcal{G}$ and $j \leftarrow J$.

2   **while** $j > J_0$ **do**

3   Compute graph distance $\boldsymbol{\rho}_{\mathcal{G}_j}$ of $\mathcal{G}_j$.

4   Randomly choose N_{j-1} vertices $u_1, \ldots, u_{k_{j-1}}$ from V_j as centers. In the semi-supervised case, these centers are given in advance.

5   **while** *true* **do**

6   Construct cluster C_ℓ for $\ell = 1, \ldots, N_{j-1}$ such that $[v] \in V_j$ belongs to C_ℓ and the distance between $[v]$ and the cluster center u_ℓ is the minimal among all clusters:

$$\ell = \mathrm{argmin}_{1 \leq \ell \leq N_{j-1}} \boldsymbol{\rho}_{\mathcal{G}_j}(u_\ell, [v]).$$

7   Update the centers: for each C_ℓ, find a new center $u_\ell \in C_\ell$ such that $\sum_{v \in C_\ell} \boldsymbol{\rho}_{\mathcal{G}_j}(u_\ell, v)$ is minimized.

8   Break if all centers remain the same.

9   **end**

10   Construct a coarse-grained graph $\mathcal{G}_{j-1} = (V_{j-1}, E_{j-1}, \boldsymbol{w}_{j-1})$ of $\mathcal{G}$, where

$$V_j = \{\cup_{[v] \in C_\ell} [v] : \ell = 1, \ldots, N_{j-1}\},$$

is the vertex set with k_{j-1} vertices, and the weight function $\boldsymbol{w}_{j-1}$ is defined by (40).

11     $j \leftarrow j - 1$.

12   **end**

5.2 Orthonormal Bases for a Coarse-grained Chain via Graph Laplacian

Let $\mathcal{G}_c = (V_c, E_c, \boldsymbol{w}_c)$ be a coarse-grained graph of $\mathcal{G} = (V, E, \boldsymbol{w})$, and $n_1 := |V|$, $n_0 := |V_c|$. Suppose $\{(\boldsymbol{u}_\ell^{(0)}, \lambda_\ell^{(0)})\}_{\ell=1}^{n_0}$ and $\{(\boldsymbol{u}_\ell^{(1)}, \lambda_\ell^{(1)})\}_{\ell=1}^{n_1}$ are orthonormal eigen-pairs for $l_2(\mathcal{G}_c)$ and $l_2(\mathcal{G})$ respectively. The orthonormal eigen-pair $\{(\boldsymbol{u}_\ell^{(0)}, \lambda_\ell^{(0)})\}_{\ell=1}^{n_0}$ can be extended to an orthonormal system on $\mathcal{G}$ using the following Gram-Schmidt process.

Define vectors $\boldsymbol{u}_\ell, \ell = 1, \ldots, n_0$ on $\mathcal{G}$ by

$$\boldsymbol{u}_\ell(v) := \frac{\boldsymbol{u}_\ell^{(0)}([v])}{\sqrt{\#[v]}}, \quad v \in V, \ \ell = 1, \ldots, n_0.$$

Then $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^{n_0}$ are orthonormal in $l_2(\mathcal{G})$. Since $\{(\boldsymbol{u}_\ell^{(1)}, \lambda_\ell^{(1)})\}_{\ell=1}^{n_1}$ is a set of orthonormal eigen-pairs for $l_2(\mathcal{G})$, there must exist a subsequence $\boldsymbol{u}_{\ell_{n_0+1}}^{(1)}, \ldots, \boldsymbol{u}_{\ell_{n_1}}^{(1)}$ of $\boldsymbol{u}_1^{(1)}, \ldots, \boldsymbol{u}_{n_1}^{(1)}$ such

that

$$\{\boldsymbol{u}_\ell : \ell = 1, \ldots, n_0\} \cup \{\boldsymbol{u}_{\ell_k}^{(1)} : k = n_0 + 1, \ldots, n_1\} \tag{41}$$

is linearly independent in $l_2(\mathcal{G})$. Applying the Gram-Schmidt process to the system of linearly independent vectors in (41), we would obtain the following orthonormal basis for $l_2(\mathcal{G})$:

$$\{\boldsymbol{u}_\ell : \ell = 1, \ldots, n_0\} \cup \{\boldsymbol{u}_\ell : \ell = n_0 + 1, \ldots, n_1\}. \tag{42}$$

Now letting $\lambda_\ell = \lambda_\ell^{(1)}$, $\ell = 1, \ldots, n_1$, we then have a set of orthonormal eigen-pairs $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^{n_1}$ for $l_2(\mathcal{G})$ satisfying $\boldsymbol{u}_\ell(p) \equiv const$ for $p \in [v] \in V_c$ and $\ell = 1, \ldots, n_0$. Here, the first n_0 vectors can be regarded as eigenvectors on the coarse-grained graph $\mathcal{G}_c$.

Let $(\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$ be a coarse-grained chain associated with a sequence $\{(\boldsymbol{u}_\ell^{(j)}, \lambda_\ell^{(j)}) : \ell = 1, \ldots, |V_j| =: N_j\}_{j=J_0}^{J}$ of orthonormal eigen-pairs of $\mathcal{G}_j$, which can be constructed from positive semi-definite operators on the chain, such as graph Laplacians and diffusion operators. For example, if we use the graph Laplacian $\mathcal{L}_j$ on $\mathcal{G}_j$ in (1), the sequence $\{\mathcal{L}_j : j = J_0, \ldots, J\}$ for a sequence of orthonormal eigen-pairs is given by

$$\mathcal{L}_j \boldsymbol{u}_\ell^{(j)} = \lambda_\ell^{(j)} \boldsymbol{u}_\ell^{(j)}, \quad \ell = 1, \ldots, N_j, \; j = J_0, \ldots, J.$$

Recursively using the Gram-Schmidt process for orthonormal eigen-pairs

$$\left(\left\{ (\boldsymbol{u}_\ell^{(j)}, \lambda_\ell^{(j)}) \right\}_{\ell=1}^{N_j}, \left\{ (\boldsymbol{u}_\ell^{(j+1)}, \lambda_\ell^{(j+1)}) \right\}_{\ell=1}^{N_{j+1}} \right)$$

for $j = J_0, \ldots, J-1$, we obtain orthonormal eigen-pairs $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ for $l_2(\mathcal{G})$, which satisfy for each $j = J_0, \ldots, J$,

$$\boldsymbol{u}_\ell(v) \equiv const \quad \forall v \in [v]_{\mathcal{G}_j} \text{ and } \ell \leq N_j. \tag{43}$$

We group the $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ as $\cup_{j=J_0}^{J}\{\boldsymbol{u}_\ell : \ell = N_{j-1} + 1, \ldots, N_j\}$ where we have let $N_{J_0-1} := 0$. Specifically, the jth group $\{\boldsymbol{u}_\ell : \ell = N_{j-1}+1, \ldots, N_j\}$ is an orthogonal basis on the graph $\mathcal{G}_j$ for $j = J_0, \ldots, J$. We call $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ *global orthonormal eigen-pair* for the coarse-grained chain $\mathcal{G}_{J \to J_0}$. We present the detailed implementation of CGC eigen-pairs in Algorithm 2, which we call *orthonormalization on the coarse-grained chain (ONBC)* algorithm.

5.3 Orthonormal Bases on $\mathcal{G}$ with Low Spoc: Haar Basis on Graphs

We define a notion to measure the sparsity and locality for a vector on the graph.

Definition 10 (SPOC) *For a vector $\boldsymbol{u}_\ell$ on $\mathcal{G}$, we define the spoc as the number of distinct non-zero entries in the vector $\boldsymbol{u}_\ell$, denoted by*

$$\mathrm{spoc}(\boldsymbol{u}_\ell) := \#\{\boldsymbol{u}_\ell(v) \neq 0 : v \in V\}. \tag{45}$$

Recall that the support of a vector $\boldsymbol{u}_\ell$ is given by

$$\mathrm{supp}(\boldsymbol{u}_\ell) := \{v \in V : \boldsymbol{u}_\ell(v) \neq 0\}. \tag{46}$$

Algorithm 2: Chain-based Graph Laplacian Basis

    Input : A coarse-grained chain $(\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$ of $\mathcal{G}$ with $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$
               associating with a sequence $\{(\boldsymbol{u}_\ell^{(j)}, \lambda_\ell^{(j)}) : \ell = 1, \ldots, |V_j|\}_{j=J_0}^{J}$ of
               orthonormal eigen-pairs.
    Output: An orthonormal basis $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^{|V|}$ for $l_2(\mathcal{G})$ satisfying (43).

1 Initialization: $j \leftarrow J_0$.
2 **while** $j < J$ **do**
3     $n_0 \leftarrow |V_j|, n_1 \leftarrow |V_{j+1}|$.
4   **Extension**: Define $\boldsymbol{u}_\ell \in l_2(\mathcal{G}_{j+1}), \ell = 1, \ldots, n_0$ by

$$\boldsymbol{u}_\ell([v]) := \frac{\boldsymbol{u}_\ell^{(j)}([v]_{\mathcal{G}_j})}{\sqrt{N_{[v]_{\mathcal{G}_j}}}}, \quad [v] \in V_{j+1}, \ \ell = 1, \ldots, n_0, \tag{44}$$

    where $N_{[v]_{\mathcal{G}_j}} := \#\{[p]_{\mathcal{G}_{j+1}} : p \in [v]_{\mathcal{G}_j}\}$ is the number of vertices in $\mathcal{G}_{j+1}$ which
    are also in the cluster $[v]_{\mathcal{G}_j}$.
5   **Gram-Schmidt process**: Choose $\boldsymbol{u}_{\ell_k}^{(j+1)}, k = n_0 + 1, \ldots, n_1$ from
    $\boldsymbol{u}_\ell^{(j+1)} : \ell = 1, \ldots, n_1$ so that

$$\{\boldsymbol{u}_\ell : \ell = 1, \ldots, n_0\} \cup \{\boldsymbol{u}_{\ell_k}^{(j+1)} : k = n_0 + 1, \ldots, n_1\}$$

    is linearly independent for $l_2(\mathcal{G}_{j+1})$. Apply the Gram-Schmidt process to the
    above vectors to obtain the full orthonormal basis for $l_2(\mathcal{G}_{j+1})$,

$$\{\boldsymbol{u}_\ell : \ell = 1, \ldots, n_1\}.$$

6   **Update**: Replace $\{(\boldsymbol{u}_\ell^{(j+1)}, \lambda_\ell^{(j+1)})\}_{\ell=1}^{n_1}$ by $\{(\boldsymbol{u}_\ell, \lambda_\ell^{(j+1)})\}_{\ell=1}^{n_1}$.
7     $j \leftarrow j + 1$.
8 **end**

The spoc is different from support. For example, a vector $\boldsymbol{u}_\ell \equiv const$ has spoc 1 yet it has full support. The computational complexity for the transforms for a basis on the graph is usually dependent on the spoc of the basis but not on its support.

For Algorithm 2, the computational complexity might go above linear, as the spoc of the global orthonormal basis by ONBC is not always bounded. Here, we construct a global orthonormal basis for the coarse-grained chain of a graph $\mathcal{G}$ based on the interval decomposition method (Chui et al., 2015). The resulting basis is Haar-like and satisfies condition (43), and its spoc is at most 2.

We first give the construction of a basis for a chain with two levels. To construct a basis on the chain with more levels, we can use this method recursively. Let $\mathcal{G}_c = (V_c, E_c, \boldsymbol{w}_c)$ be a coarse-grained graph of $\mathcal{G} = (V, E, \boldsymbol{w})$, and $n_1 := |V|$ and $n_0 := |V_c|$. To construct an orthogonal basis on $\mathcal{G}_c$, we sequence the vertices of $\mathcal{G}_c$ by their degrees as

$$V_c := \{[p_j]_{\mathcal{G}_c} : j = 1, \ldots, n_0\} \text{ and } \boldsymbol{d}([p_j]_{\mathcal{G}_c}) \geq \boldsymbol{d}([p_{j+1}]_{\mathcal{G}_c}).$$

For each vertex $[p_j]_{\mathcal{G}_c}$, we associate with it a characteristic function $\chi_{[p_j]}$ on $\mathcal{G}_c$:

$$\chi_j^c([v]) := \begin{cases} 1, & [v] = [p_j]_{\mathcal{G}_c}, \\ 0, & \text{otherwise.} \end{cases}$$

The system $\{\chi_j^c : j = 1, \ldots, n_0\}$ is then an orthogonal system on $\mathcal{G}_c$. We need to include the constant vector $\boldsymbol{u}_1^c = \frac{1}{\sqrt{n_0}}\boldsymbol{1}$ in the system. To achieve that, we replace by $\boldsymbol{u}_1^c = \frac{1}{\sqrt{n_0}}\boldsymbol{1}$ the characteristic function $\chi_{[p_{n_0}]_{\mathcal{G}_c}}$ on the set $\{\chi_{[p_j]_{\mathcal{G}_c}} : j = 1, \ldots, n_0\}$ and make the set $\{\boldsymbol{u}_1, \chi_1^c, \ldots, \chi_{n_0-1}^c\}$ orthonormal by the Gram-Schmidt process. It then gives a new orthonormal system, as follows.

$$\begin{aligned} \boldsymbol{u}_1^c &= \frac{1}{\sqrt{n_0}}\boldsymbol{1}, \\ \boldsymbol{u}_\ell^c &= \sqrt{\frac{n_0 - \ell + 1}{n_0 - \ell + 2}} \left(\chi_{\ell-1}^c - \frac{1}{n_0 - \ell + 1} \sum_{j=\ell}^{n_0} \chi_j^c \right), \quad \ell = 2, \ldots, n_0. \end{aligned} \tag{47}$$

Proposition 11 *The system $\{\boldsymbol{u}_\ell^c : \ell = 1, \ldots, n_0\}$ given in (47) is an orthonormal basis for $\mathcal{G}_c$ with $\mathrm{spoc}(\boldsymbol{u}_\ell^c) \leq 2$ for all $\ell = 1, \ldots, n_0$.*

Proof By (47), it follows that $\mathrm{spoc}(\boldsymbol{u}_\ell^c) \leq 2$ and $\|\boldsymbol{u}_\ell^c\| = 1$ for all $\ell = 1, \ldots, n_0$. Next, we prove $\boldsymbol{u}_\ell^c$, $\ell = 1, \ldots, n_0$, are orthogonal. For $\ell = 2, \ldots, n_0$,

$$\langle \boldsymbol{u}_1^c, \boldsymbol{u}_\ell^c \rangle = \sqrt{\frac{n_0 - \ell + 1}{n_0(n_0 - \ell + 2)}} \left\langle \boldsymbol{1}, \chi_{\ell-1}^c - \frac{1}{n_0 - \ell + 1} \sum_{j=\ell}^{n_0} \chi_j^c \right\rangle = 0.$$

For $2 \leq k < \ell \leq n_0$,

$$\begin{aligned} \langle \boldsymbol{u}_\ell^c, \boldsymbol{u}_k^c \rangle &= c_{k,\ell} \left\langle \chi_{\ell-1}^c - \frac{1}{n_0 - \ell + 1} \sum_{j=\ell}^{n_0} \chi_j^c, \chi_{k-1}^c - \frac{1}{n_0 - k + 1} \sum_{j=k}^{n_0} \chi_j^c \right\rangle \\ &= c_{k,\ell} \left(-\frac{\sum_{j=k}^{n_0} \langle \chi_{\ell-1}^c, \chi_j^c \rangle}{n_0 - k + 1} + \frac{\langle \sum_{j=\ell}^{n_0} \chi_j^c, \sum_{j=k}^{n_0} \chi_j^c \rangle}{(n_0 - k + 1)(n_0 - \ell + 1)} \right) \\ &= c_{k,\ell} \left(-\frac{1}{n_0 - k + 1} + \frac{n_0 - \ell + 1}{(n_0 - k + 1)(n_0 - \ell + 1)} \right) = 0, \end{aligned}$$

where $c_{k,\ell} = \sqrt{\frac{n_0 - \ell + 1}{n_0 - \ell + 2}} \times \sqrt{\frac{n_0 - k + 1}{n_0 - k + 2}}$. Thus, $\{\boldsymbol{u}_\ell^c : \ell = 1, \ldots, n_0\}$ is an orthonormal basis for $\mathcal{G}_c$. ∎

We associate each element of $\{\boldsymbol{u}_\ell^c : \ell = 1, \ldots, n_0\}$ with the vertex $[p_\ell]_{\mathcal{G}_c}$ in $\mathcal{G}_c$, and define a sequence of vectors by extending $\boldsymbol{u}_\ell^c$, $\ell = 1, \ldots, n_0$ onto $\mathcal{G}$

$$\boldsymbol{u}_{\ell,1}(v) := \frac{\boldsymbol{u}_\ell^c([v])}{\sqrt{\#[v]}}, \quad v \in V, \ \ell = 1, \ldots, n_0, \tag{48}$$

25

which satisfy

$$\boldsymbol{u}_{\ell,1}(v) \equiv \text{const} \quad \forall v \in [p_j]_{\mathcal{G}_c} \text{ and } j = 1, \ldots, n_0. \tag{49}$$

The resulting $\boldsymbol{u}_\ell$, $\ell = 1, \ldots, n_0$, are orthogonal for $l_2(\mathcal{G})$.

For a cluster $[p_\ell]_{\mathcal{G}_c}$, let $k_\ell := \#[p_\ell]_{\mathcal{G}_c}$. We sequence the vertices in $[p_\ell]_{\mathcal{G}_c}$ by their degrees to obtain

$$[p_\ell]_{\mathcal{G}_c} = \{v_{\ell,1}, \ldots, v_{\ell,k_\ell}\} \text{ and } \boldsymbol{d}(v_{\ell,j}) \geq \boldsymbol{d}(v_{\ell,j+1}).$$

For each vertex in the cluster $[p_\ell]_{\mathcal{G}_c}$, we define a characteristic function $\chi_{\ell,j}$ on $\mathcal{G}$ by

$$\chi_{\ell,j}(v) := \begin{cases} 1 & v = v_{\ell,j}, \\ 0 & \text{otherwise.} \end{cases}$$

Using the Gram-Schmidt process for the system $\{\boldsymbol{u}_{\ell,1}, \chi_{\ell,1}, \ldots, \chi_{\ell,k_\ell-1}\}$, we then obtain

$$\boldsymbol{u}_{\ell,k} = \sqrt{\frac{k_\ell - k + 1}{k_\ell - k + 2}} \left(\chi_{\ell,k-1} - \frac{1}{k_\ell - k + 1} \sum_{j=k}^{k_\ell} \chi_{\ell,j} \right), \quad k = 2, \ldots, k_\ell. \tag{50}$$

Proposition 12 *Let*

$$\{\boldsymbol{u}_{\ell,k} : k = 1, \ldots, k_\ell, \ell = 1, \ldots, n_0\} \tag{51}$$

be the set of vectors in (48) and (50). Then it is an orthonormal basis on $\mathcal{G}$ with $\text{spoc}(\boldsymbol{u}_{\ell,k}) \leq 2$ for all $k = 1, \ldots, k_\ell$ and $\ell = 1, \ldots, n_0$.

Proof Fixed a level ℓ, in a similar way to the proof in Proposition 11, one can show that $\{\boldsymbol{u}_{\ell,k} : k = 1, \ldots, k_\ell\}$ is an orthonormal system for $l_2(\mathcal{G})$. Now for two different ℓ, ℓ', by definition, $\text{supp}(\boldsymbol{u}_{\ell,k}) = [p_\ell]_{\mathcal{G}_c}$ and $\text{supp}(\boldsymbol{u}_{\ell',k'}) = [p_{\ell'}]_{\mathcal{G}_c}$, and hence $\text{supp}(\boldsymbol{u}_{\ell,k}) \cap \text{supp}(\boldsymbol{u}_{\ell',k'}) = \emptyset$ for all $k = 2, \ldots, k_\ell$ and $k' = 2, \ldots, k_{\ell'}$. Moreover, by (49), $\boldsymbol{u}_{\ell,1}(v)$ is constant for all $v \in [p_{\ell'}]_{\mathcal{G}_c}$, and $\boldsymbol{u}_{\ell',1}(v')$ is constant for all $v' \in [p_\ell]_{\mathcal{G}_c}$. Thus, $\langle \boldsymbol{u}_{\ell,k}, \boldsymbol{u}_{\ell',k'} \rangle = \delta_{\ell,\ell'} \delta_{k,k'}$, which shows that (51) is an orthonormal basis for $l_2(\mathcal{G})$. $\blacksquare$

To construct a global orthonormal basis for the coarse-grained chain $(\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$, we continue the above construction for $\mathcal{G}_c$. Starting from $\mathcal{G}_{J_0}$, we obtain an orthonormal basis on $\mathcal{G}_{J_0}$ as $\{\boldsymbol{u}_\ell^{(J_0)} : \ell = 1, \ldots, N_{J_0}\}$. By the chain relation of $\mathcal{G}_{J_0}$ and $\mathcal{G}_{J_1}$, we can extend this basis to an orthonormal basis $\{\boldsymbol{u}_\ell^{(J_1)} : \ell = 1, \ldots, n_{J_1}\}$ on $\mathcal{G}_{J_1}$. Continuing carrying on this process, we then obtain an orthonormal basis for each $\mathcal{G}_j$,

$$\{\boldsymbol{u}_\ell^{(j)} : \ell = 1, \ldots, N_j\}$$

with $\text{spoc}(\boldsymbol{u}_\ell^{(j)}) \leq 2$ for all ℓ and $j = J_0, \ldots, J$.

In particular, when $j = J$, $\{\boldsymbol{u}_\ell := \boldsymbol{u}_\ell^{(J)} : \ell = 1, \ldots, N_J =: N\}$ becomes an orthonormal basis for $l_2(\mathcal{G})$, which satisfies $\text{spoc}(\boldsymbol{u}_\ell) \leq 2$ for all $\ell = 1, \ldots, N$. Moreover, we can group $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ by their associated levels in the chain as $\cup_{j=J_0}^J \{\boldsymbol{u}_\ell : \ell = N_{j-1} + 1, \ldots, N_j\}$. Here, we have let $N_{J_0-1} := 0$, and the jth group $\{\boldsymbol{u}_\ell : \ell = N_{j-1} + 1, \ldots, N_j\}$ is the extension with respect to the orthogonal basis on the graph $\mathcal{G}_j$ for $j = J_0, \ldots, J$. By this rearrangement, $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ then satisfies the condition in (43). We call the resulting orthonormal basis $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ *Haar global orthonormal basis* for the chain.

We show the detailed steps of constructing the global orthonormal basis in Algorithm 3, and call the algorithm *Haar orthonormalization on the coarse-grained chain*, or HONBC.

5.4 Construction of Decimated Tight Framelets on $\mathcal{G}$

In this section, we construct decimated tight framelets using band-limited filters on $\mathcal{G}$. To this end, we introduce some auxiliary functions for the construction of generating sets Ψ_j in Theorem 7 and then derive a filter bank $\boldsymbol{\eta}_j$. Together with the global orthonormal basis satisfying (43), we can define framelets $\boldsymbol{\varphi}_{j,[p]}$ and $\boldsymbol{\psi}_{j,[p]}^n$ on $\mathcal{G}$.

Here we define the filter bank based on polynomial splines in Han (1997, 2013a); Han and Zhuang (2010); Han et al. (2016a). Let $P_m(x)$ be a polynomial given by

$$P_m(x) := \left(\frac{1+x}{2}\right)^m \sum_{k=0}^{m-1} \binom{m-1+k}{k} \left(\frac{1-x}{2}\right)^k,$$

which satisfies $P_m(x) + P_m(-x) = 1$ for any $m \in \mathbb{N}$ (Daubechies, 1992). We define a bump function $\nu_{[c_L,c_R];\varepsilon_L,\varepsilon_R}(\xi)$ for $c_L < c_R$ and positive numbers $\varepsilon_L, \varepsilon_R$ satisfying $\varepsilon_L + \varepsilon_R \leq c_R - c_L$ by

$$\nu_{[c_L,c_R];\varepsilon_L,\varepsilon_R}(\xi) := \begin{cases} 0, & \xi \leq c_L - \varepsilon_L \text{ or } \xi \geq c_R + \varepsilon_R, \\ \sin(\frac{\pi}{2}P_m(\frac{\xi-c_L+\varepsilon_L}{2\varepsilon_L})), & c_L - \varepsilon_L < \xi < c_L + \varepsilon_L, \\ 1, & c_L + \varepsilon_L \leq \xi \leq c_R - \varepsilon_R, \\ \cos(\frac{\pi}{2}P_m(\frac{\xi-c_R+\varepsilon_R}{2\varepsilon_R})), & c_R - \varepsilon_R < \xi \leq c_R + \varepsilon_R. \end{cases} \tag{52}$$

Algorithm 3: Haar-Like Orthonormal Basis on the Coarse-Grained Chain (HONBC)

Input : A coarse-grained chain $(\mathcal{G}_J, \ldots, \mathcal{G}_{J_0})$ of $\mathcal{G}$ with $\mathcal{G}_j = (V_j, E_j, \boldsymbol{w}_j)$.

Output: Haar-like orthonormal basis $\{\boldsymbol{u}_\ell^{(j)} : \ell = 1, \ldots, N_j\}$ for $l_2(\mathcal{G}_j)$ with low spoc, $j = J_0, \ldots, J$.

1 Initialization: $j \leftarrow J_0 - 1$. $V_j \leftarrow [p_1]_{\mathcal{G}_j} := [V]$. $\boldsymbol{u}_1^{(j)} \equiv 1$. ($\boldsymbol{u}_1^{(j)}$ is associated with the vertex $[p_1]_{\mathcal{G}_j}$.)

2 **while** $j < J$ **do**

3    $n_0 \leftarrow |V_j|$, $n_1 \leftarrow |V_{j+1}|$. (Note that $\boldsymbol{u}_\ell^{(j)}$ is associated with $[p_\ell]_{\mathcal{G}_j} = \{[p]_{\mathcal{G}_{j+1}} : p \in [p_\ell]_{\mathcal{G}_j}\}$.)

4   **Extension**: Define $\boldsymbol{u}_{\ell,1} \in l_2(\mathcal{G}_{j+1})$, $\ell = 1, \ldots, n_0$ by

$$\boldsymbol{u}_{\ell,1}([v]) := \frac{\boldsymbol{u}_\ell^{(j)}([v]_{\mathcal{G}_j})}{\sqrt{N_{[v]_{\mathcal{G}_j}}}}, \quad [v] \in V_{j+1}, \ell = 1, \ldots, n_0, \tag{53}$$

   where $N_{[v]_{\mathcal{G}_j}} := \#\{[p]_{\mathcal{G}_{j+1}} : p \in [v]_{\mathcal{G}_j}\}$ is the number of vertices in $\mathcal{G}_{j+1}$ which lie in the cluster $[v]_{\mathcal{G}_j}$.

5   **Gram-Schmidt process**:

6   **for** $\ell = 1, \ldots, n_0$ **do**

7   Sort the vertices in $[p_\ell]_{\mathcal{G}_j}$ by their degrees as $[p_\ell]_{\mathcal{G}_j} = \{[v_{\ell,1}]_{\mathcal{G}_{j+1}}, \ldots, [v_{\ell,k_\ell}]_{\mathcal{G}_{j+1}}\}$ and define $\chi_{\ell,k}$ as characteristic function with respect to the vertex $[v_{\ell,k}]_{\mathcal{G}_{j+1}}$, where k_ℓ is the number of children in $\mathcal{G}_{j+1}$ of $[p_\ell]_{\mathcal{G}_j}$. Compute

$$\boldsymbol{u}_{\ell,k} = \sqrt{\frac{k_\ell - k + 1}{k_\ell - k + 2}} \left(\chi_{\ell,k-1} - \frac{1}{k_\ell - k + 1} \sum_{j=k}^{k_\ell} \chi_{\ell,j} \right), \quad k = 2, \ldots, k_\ell,$$

      and associate it with the node $[v_{\ell,k}]_{\mathcal{G}_{j+1}}$.

8   **end**

9   **Update**: Rearrange $\{(\boldsymbol{u}_{\ell,k}, [v_{\ell,k}]_{\mathcal{G}_{j+1}}) : k = 1, \ldots, k_\ell, \ell = 1, \ldots, n_0\}$ in the following order

$$\{\boldsymbol{u}_{\ell,1} : \ell \in \{1, \ldots, n_0\}\} \cup \{\boldsymbol{u}_{\ell,2} : \ell \in \{1, \ldots, n_0\}\} \cup \cdots \cup \{\boldsymbol{u}_{\ell,k_{max}} : \ell \in \{1, \ldots, n_0\}\}$$

   to form an orthonormal basis $\{(\boldsymbol{u}_\ell^{(j+1)}, [v_\ell]_{\mathcal{G}_{j+1}}) : \ell = 1, \ldots, n_1\}$ for $l_2(\mathcal{G}_{j+1})$ such that the first n_0 vectors are orthogonal for $l_2(\mathcal{G}_j)$.

10    $j \leftarrow j + 1$.

11 **end**

The scaling functions and filters for the decimated framelets are defined by $\nu_{[c_L, c_R]; \varepsilon_L, \varepsilon_R}$ and the coarse-grained chain of the graph. Let $0 < \varepsilon < 1$ and define α_ε by

$$\widehat{\alpha_\varepsilon}(\xi) := \nu_{[-\frac{1+\varepsilon}{2}, \frac{1+\varepsilon}{2}]; \frac{1-\varepsilon}{2}, \frac{1-\varepsilon}{2}}(\xi), \quad \xi \in \mathbb{R}.$$

Its support is supp $\widehat{\alpha_\varepsilon} \subseteq [-1,1]$, and $\widehat{\alpha_\varepsilon}(\xi) \equiv 1$ for all $\xi \in [-\varepsilon, \varepsilon]$. This is our low-pass scaling functions.

We now construct the high-pass scaling functions by using the low-pass $\widehat{\alpha_\varepsilon}$. Let J, J_0, $J \geq J_0$ be two fixed integers corresponding to the finest and coarsest scales. We choose $J - J_0 + 1$ *scaling factors* $\Lambda_j, j = J_0, \ldots, J$ such that

$$\begin{aligned} \Lambda_j > 1, \qquad & j = J_0, \ldots, J, \\ \Lambda_{j+1} > \Lambda_j / \varepsilon, \quad & j = J_0, \ldots, J-1. \end{aligned}$$

For each level $j = J_0, \ldots, J$, let r_j be a positive integer, we then construct r_j high-passes functions $\beta_j^{(n)} \in L_2(\mathbb{R})$, $n = 1, \ldots, r_j$, such that

$$|\widehat{\alpha_\varepsilon}(\xi/\Lambda_{j+1})|^2 - |\widehat{\alpha_\varepsilon}(\xi/\Lambda_j)|^2 = \sum_{n=1}^{r_j} \left| \widehat{\beta_j^{(n)}}(\xi/\Lambda_j) \right|^2. \tag{54}$$

To construct the above function $\beta_j^{(n)}$ for scale j, we choose positive numbers $\varepsilon_1, \ldots, \varepsilon_{r_j}$ and $c_1, \ldots c_{r_j}$ such that

$$c_1 < \cdots < c_{r_j-1} < 1 < c_{r_j}, \ c_{r_j} = 1 + \varepsilon_{r_j} \text{ and } \varepsilon_{n-1} + \varepsilon_n \leq c_n - c_{n-1}, \ n = 2, \ldots, r_j.$$

We need to define partition functions $\gamma^{(1)}, \ldots, \gamma^{(r_j)}$ by

$$\gamma^{(1)} := \nu_{[-c_1, c_1], \varepsilon_1, \varepsilon_1} \text{ and } \gamma^{(n)} := \nu_{[c_{n-1}, c_n], \varepsilon_{n-1}, \varepsilon_n}, \ n = 2, \ldots, r_j.$$

It can be verified that the $\gamma^{(n)}$ satisfies the partition of unity:

$$\sum_{n=1}^{r_j} |\gamma^{(n)}(\xi)|^2 = 1 \ \ \forall \xi \in [0,1].$$

We then define the *high-pass scaling functions* for scale j by, for $n = 1, \ldots, r_j$,

$$\widehat{\beta_j^{(n)}}(\xi/\Lambda_j) := \sqrt{|\widehat{\alpha_\varepsilon}(\xi/\Lambda_{j+1})|^2 - |\widehat{\alpha_\varepsilon}(\xi/\Lambda_j)|^2} \cdot \gamma^{(n)}(\xi/\Lambda_{j+1}),$$

which satisfy (54). The associated *framelet generating set* is then

$$\Psi_j := \{\alpha_\varepsilon; \beta_j^{(1)}, \ldots, \beta_j^{(r_j)}\}, \quad j = J_0, \ldots, J. \tag{55}$$

From this, we define the filter bank, as follows.

Definition 13 (Filter bank for chain) *For $j = J_0 + 1, \ldots, J$, the filter bank for the chain $\mathcal{G}_{J \to J_0}$ of graph $\mathcal{G}$ is*

$$\boldsymbol{\eta}_j := \{a_j; b_j^{(1)}, \ldots, b_j^{(r_{j-1})}\} \tag{56}$$

with

$$\widehat{a_j}(\xi/\Lambda_j) := \frac{\widehat{\alpha_\varepsilon}(\xi/\Lambda_{j-1})}{\widehat{\alpha_\varepsilon}(\xi/\Lambda_j)}, \quad \widehat{b_j^{(n)}}(\xi/\Lambda_j) := \frac{\widehat{\beta_j^{(n)}}(\xi/\Lambda_{j-1})}{\widehat{\alpha_\varepsilon}(\xi/\Lambda_j)}, \quad \xi \in \mathbb{R}, \ n = 1, \ldots, r_{j-1}. \tag{57}$$

Then, (23) holds with a_j and $b_j^{(n)}$ in (57).

Remark 14 *Different from the undecimated case, the scaling factor Λ_j should be adaptive to the chain structure. In practice, we can simply choose $\Lambda_j = N_j$.*

Example of filter bank

We show an example of filter bank with 1, 2 or 3 high passes. The filters in the filter bank can be obtained by the function in (52), using the parameters in the following table. Here, besides the parameters $c_R, c_L, \varepsilon_R, \varepsilon_L$ used in ν in (52), there are extra free parameters $\zeta_c^a, \zeta_c^{b^{(1)}}, \zeta_c^{b^{(2)}}$ which can adjust the intersection points of the low-pass and high-pass filter curves. The $\zeta_c^a, \zeta_c^{b^{(1)}}, \zeta_c^{b^{(2)}}$ are needed to lie in $(0, 0.5)$. For simplicity in the case with 3 high passes, we define $N^{b^{(1)}}$ and $N^{b^{(2)}}$ by

$$N^{b^{(1)}} = N_{j-1} + 0.3(N_j - N_{j-1}), \quad N^{b^{(2)}} = N_{j-1} + 0.8(N_j - N_{j-1}).$$

| Filter bank with 1 high pass | | | Filter bank with 2 high passes | | |
	a	b	a	$b^{(1)}$	$b^{(2)}$
c_R	$0.5(1+N_{j-1})\zeta_c^a$	$2N_j$	$0.5(1+N_{j-1})\zeta_c^a$	$2(N_j+N_{j-1})\zeta_c^b$	$2N_j$
ε_R	$N_{j-1} - c_R^a$	$N_j/4$	$N_j - c_R^a$	$N_j - c_R^b$	1
c_L	$-c_R^a$	c_R^a	$-c_R^a$	c_R^a	$c_R^{b^{(1)}}$
ε_L	ε_R^a	ε_R^a	ε_R^a	ε_R^a	$\varepsilon_R^{b^{(1)}}$

Filter bank with 3 high passes

	c_R	ε_R	c_L	ε_L
a	$0.5N_{j-1}(1+\zeta_c^a)$	$N_{j-1} - c_R^a$	$-c_R^a$	ε_R^a
$b^{(1)}$	$0.5(N^{b^{(1)}} + N_{j-1}) + 0.5\zeta_c^{b^{(1)}}(N^{b^{(1)}} - N_{j-1})$	$N^{b^{(1)}} - c_R^{b^{(1)}}$	c_R^a	ε_R^a
$b^{(2)}$	$0.5(N^{b^{(2)}} + N^{b^{(1)}}) + 0.5\zeta_c^{b^{(2)}}(N^{b^{(2)}} - N^{b^{(1)}})$	$N^{b^{(2)}} - c_R^{b^{(2)}}$	$c_R^{b^{(1)}}$	$\varepsilon_R^{b^{(1)}}$
$b^{(3)}$	$2N_j$	1	$c_R^{b^{(2)}}$	$\varepsilon_R^{b^{(2)}}$

Figure 3 shows a filter bank with three high passes based on the above construction for the chain $\mathcal{G}_{4 \to 0}$ with 500, 250, 100, 40 and 8 nodes for levels from the finest to coarsest. The supports of low-pass and high-pass filters are determined by the chain. For the jth level, the low-pass filter is supported on $[0, N_j]$; the high passes are supported on subintervals of $(0, N_j)$, where the lower end of the support interval is greater than 0. The intersection points between the low pass and high passes are tunable parameters determined by the parameters ζ_c^a, $\zeta_c^{b^{(1)}}$ and $\zeta_c^{b^{(2)}}$.

With the filter bank in Definition 13, we can construct a tight decimated framelet system $\mathsf{DFS}\big(\{\Psi_j\}_{j=J_1}^J, \{\eta_j\}_{j=J_1+1}^J\big)$, $J_1 = J_0, \ldots, J$, for $l_2(\mathcal{G}_{J \to J_0})$ with the framelets given by, for $j = J_0, \ldots, J$,

$$\varphi_{j,[p]}(v) := \sqrt{\omega_{j,[p]}} \sum_{\ell=1}^N \widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \overline{u_\ell([p])} u_\ell(v), \quad [p] \in V_j,$$

$$\psi_{j,[p]}^n(v) := \sqrt{\omega_{j+1,[p]}} \sum_{\ell=1}^N \widehat{\beta_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \overline{u_\ell([p])} u_\ell(v), \quad [p] \in V_{j+1}, \ n = 1, \ldots, r_j,$$

(58)

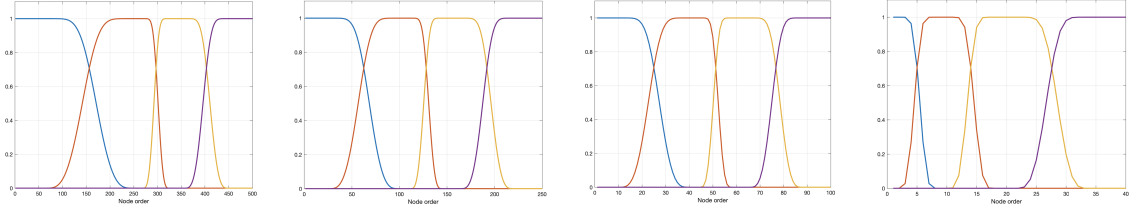where $\omega_{j,[p]} := \#[p]_{\mathcal{G}_j}$.

Figure 3: Filter banks with three high passes. The chain has 5 levels and the graphs of the chain have 500, 250, 100, 40 and 8 nodes from the finest to coarsest. Note that we do not need a filter bank for the coarsest level. From left to right are the pictures of the filters for the chain level from the finest to the second last coarsest. The top and bottom use the intersection point parameters $\zeta_c^a = 0.25$, $\zeta_c^{b^{(1)}} = 0.25$ and $\zeta_c^{b^{(2)}} = 0.25$. The support of low-pass and high-passes are exactly controlled by the graph size at each level of the chain.

Theorem 15 *Let $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ be a global orthonormal eigen-pair for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of $\mathcal{G}$. Let*

$$\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J), \quad J_1 = J_0, \ldots, J$$

be a sequence of the decimated framelet systems associated with the generating sets and filter banks in (55) and (56) and with $\boldsymbol{\varphi}_{j,[p]}$ and $\boldsymbol{\psi}_{j,[p]}^n$ in (58). Then, the decimated framelet system $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$ is a tight frame for $l_2(\mathcal{G})$ for each $J_1 = J_0, \ldots, J$.

Proof For the framelets $\boldsymbol{\varphi}_{j,[p]}$ and $\boldsymbol{\psi}_{j,[p]}^n$ given in (58), the weighted sum of the product of two eigenvectors by the quadrature weights $\mathcal{Q}_j$ is

$$
\begin{aligned}
\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j) &= \sum_{[p] \in V_j} \omega_{j,[p]} \, \boldsymbol{u}_\ell([p]) \overline{\boldsymbol{u}_{\ell'}([p])} \\
&= \sum_{[p] \in V} \sum_{v \in [p]_{\mathcal{G}_j}} \boldsymbol{u}_\ell(v) \overline{\boldsymbol{u}_{\ell'}(v)} \\
&= \sum_{v \in V} \boldsymbol{u}_\ell(v) \overline{\boldsymbol{u}_{\ell'}(v)} = \delta_{\ell,\ell'}
\end{aligned}
\tag{59}
$$

for all pairs in $\{(\ell, \ell') : 1 \leq \ell, \ell' \leq N_j\} = \sigma_{\alpha,\overline{\alpha}}^{(j)}$, where the second equality has used $\omega_{j,[p]} := \#[p]_{\mathcal{G}_j}$. By our construction follows $\sigma_{\alpha,\overline{\alpha}}^{(j)} \subseteq \sigma_{\alpha,\overline{\alpha}}^{(j+1)}$. Then, the tightness of $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$ for each $J_1 = J_0, \ldots, J$ follows by Corollary 8. ∎

6. Fast Decimated $\mathcal{G}$-Framelet Transforms

In this section, we describe the multi-level decimated $\mathcal{G}$-framelet transforms for a coarse-grained chain of the graph $\mathcal{G}$ based on the decimated tight framelet system constructed in Section 4.2. We would introduce the discrete Fourier transforms for a global orthonormal basis based on a decimated coarse-grained chain, and their fast algorithm. With decimated filter banks, it then gives fast $\mathcal{G}$-framelet transforms.

6.1 Discrete Fourier Transforms on $\mathcal{G}$

Let $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ be a global orthonormal basis for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of a graph $\mathcal{G}$. The *discrete Fourier transform* (DFT) of a vector $\boldsymbol{f} \in l_2(\mathcal{G})$ is to compute the Fourier coefficient vector $\widehat{\boldsymbol{f}} := (\widehat{\boldsymbol{f}}_\ell)_{\ell=1}^N$:

$$\widehat{\boldsymbol{f}}_\ell := \langle \boldsymbol{f}, \boldsymbol{u}_\ell \rangle = \sum_{v \in V} \boldsymbol{f}(v)\overline{\boldsymbol{u}_\ell(v)}, \quad \ell = 1, \ldots, N. \tag{60}$$

The *adjoint discrete Fourier transform* (ADFT) of a coefficient vector $\boldsymbol{c} := (c_\ell)_{\ell=1}^N$ with respect to $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ is to compute the vector $\boldsymbol{f} \in l_2(\mathcal{G})$:

$$\boldsymbol{f}(v) := \sum_{\ell=1}^N c_\ell \, \boldsymbol{u}_\ell(v), \quad v \in V. \tag{61}$$

Computational cost for DFT and ADFT We denote $\mathrm{flop}^*(\mathrm{DFT}_\mathcal{G})$ and $\mathrm{flop}^*(\mathrm{ADFT}_\mathcal{G})$ the minimal total number of summations and multiplications for the DFT and ADFT in (60) and (61) for a given orthonormal basis over all possible algorithmic realizations. Direct evaluation of (60) or (61) requires total number of N^2 summations and N^2 multiplications. Thus, $\mathrm{flop}^*(\mathrm{DFT}_\mathcal{G}) = \mathcal{O}(N^2)$ and $\mathrm{flop}^*(\mathrm{ADFT}_\mathcal{G}) = \mathcal{O}(N^2)$. The property (43) suggests that for specific bases, the computational complexity for DFT's can be significantly reduced, as we discuss now.

Let $\{\boldsymbol{u}_\ell\}_{\ell=1}^{|V|}$ be a global orthonormal basis for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of the graph $\mathcal{G}$. Let $N_j := |V_j|$ for $j = J_0, \ldots, J$ and $N_{J_0-1} := 0$, and the indicator function $\mathbf{1}_{[v]} : V \to \mathbb{R}$ by $\mathbf{1}_{[v]}(p) = 1$ for all $p \in [v]$ and 0 for $p \in V \backslash [v]$, and $\mathbf{1}_\mathcal{G} := \mathbf{I}_V$ the identity function on $\mathcal{G}$. One can then show that $\mathbf{1}_\mathcal{G} := \sum_{[v] \in V_j} \mathbf{1}_{[v]} \equiv 1$ for any coarse-grained graph $\mathcal{G}_j$ of $\mathcal{G}$.

We first consider the computational steps of the discrete Fourier transform in (60). Suppose $N_{j-1} < \ell \le N_j$ for some $J_0 \le j \le J$. Then, by (43),

$$\begin{aligned}
\sum_{v \in V} \boldsymbol{f}(v)\boldsymbol{u}_\ell(v) &= \sum_{[v] \in V_j} \sum_{p \in [v]} \boldsymbol{f}(p)\boldsymbol{u}_\ell(p) \\
&= \sum_{[v] \in V_j} \boldsymbol{u}_\ell([v]) \sum_{p \in [v]} \boldsymbol{f}(p) \\
&= \sum_{[v] \in V_j} \boldsymbol{u}_\ell([v])s(\boldsymbol{f}, [v]) \\
&= \sum_{[v] \in V_j} t_\ell(\boldsymbol{f}, [v]),
\end{aligned}$$

where we have let

$$s(\boldsymbol{f}, [v]) := \sum_{v \in [v]} \boldsymbol{f}(v), \quad t_\ell(\boldsymbol{f}, [v]) := \boldsymbol{u}_\ell([v])s(\boldsymbol{f}, [v]).$$

We can evaluate $\widehat{\boldsymbol{f}}_\ell$, $\ell = 1, \ldots, N$ in (60) in two steps, for which we state a pseudocode in Algorithm 4.

(1) Evaluate for $j = J_0, \ldots, J-1$,

$$s(\boldsymbol{f}, [v]) \quad \forall [v] \in V_j. \tag{62}$$

With the hierarchical structure of the chain, we can evaluate the $s(\boldsymbol{f}, [v]_{\mathcal{G}_j})$ by using the $s(\boldsymbol{f})$ over nodes at the finer $(j+1)$th level, that is,

$$s(\boldsymbol{f}, [v]_{\mathcal{G}_j}) = \sum_{[p]_{\mathcal{G}_{j+1}} \subseteq [v]_{\mathcal{G}_j}} s(\boldsymbol{f}, [p]_{\mathcal{G}_{j+1}}).$$

Here the summation over all the children of $[v]_{\mathcal{G}_j}$. Consequently, the total number of summations required to compute $\{s(\boldsymbol{f}, [v]) : [v] \in V_j, \; j = J_0, \ldots, J-1\}$ is no more than $\sum_{j=J_0}^{J-1} N_{j+1}$.

(2) For each ℓ, let j be the integer such that $N_{j-1} < \ell \le N_j$. The evaluation of $\sum_{[v] \in V_j} \boldsymbol{u}_\ell([v]) s(\boldsymbol{f}, [v])$ includes the following steps.

   (a) Compute the product $t_\ell(\boldsymbol{f}, [v]) = \boldsymbol{u}_\ell([v]) s(\boldsymbol{f}, [v])$ for all $[v] \in V_j$, which requires no more than $\mathrm{spoc}(\boldsymbol{u}_\ell)$ multiplications.

   (b) Evaluate the sum $\sum_{[v] \in V_j} t_\ell(\boldsymbol{f}, [v])$, which needs no more than $\mathrm{spoc}(\boldsymbol{u}_\ell)$ summations.

The total number of multiplication and summation operations is at most $\sum_\ell \mathrm{spoc}(\boldsymbol{u}_\ell)$.

Algorithm 4: Discrete Fourier Transform on $\mathcal{G}$ (DFT)

    Input : A global orthonormal basis $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^{|V|}$ on $\mathcal{G}_{J \to J_0}$ satisfying (43), and a vector $\boldsymbol{f}$ on $\mathcal{G}$.

    Output: Fourier coefficients $\widehat{\boldsymbol{f}}$ given in (60).

1 Initialization: $N_j \leftarrow |V|$. $s(\boldsymbol{f}, [v]_{\mathcal{G}_{J+1}}) \leftarrow \boldsymbol{f}(v)$, $v \in V$.

2 **for** $j = J$ **to** J_0 **do**

3    $N_j \leftarrow |V_{j-1}|$.

4    $s(\boldsymbol{f}, [v]_{\mathcal{G}_j}) \leftarrow \sum_{[p]_{\mathcal{G}_{j+1}} \subseteq [v]_{\mathcal{G}_j}} s(\boldsymbol{f}, [p]_{\mathcal{G}_{j+1}})$, $[v]_{\mathcal{G}_j} \in V_j$.

5   **for** $\ell = N_{j-1} + 1$ **to** N_j **do**

6      $\widehat{f}_\ell \leftarrow \sum_{[v] \in V_j} \boldsymbol{u}_\ell([v]) s(\boldsymbol{f}, [v])$, $[v] \in V_j$.

7   **end**

8 **end**

For the computation of the ADFT in (61), by using the clustering feature of the chain, we can rewrite

$$\boldsymbol{f} = \sum_{\ell} c_{\ell} \boldsymbol{u}_{\ell} = \sum_{j=J_0}^{J} \sum_{\ell=N_{j-1}+1}^{N_j} c_{\ell} \boldsymbol{u}_{\ell} = \sum_{j=J_0}^{J} \sum_{\ell=N_{j-1}+1}^{N_j} \sum_{[v] \in V_j} c_{\ell} \boldsymbol{u}_{\ell}([v]) \mathbf{1}_{[v]}$$

$$= \sum_{j=J_0}^{J} \sum_{\ell=N_{j-1}+1}^{N_j} \sum_{[v] \in V_j} t_{\ell}([v], \boldsymbol{c}) \mathbf{1}_{[v]}$$

$$= \sum_{j=J_0}^{J} \sum_{[v] \in V_j} \left(\sum_{\ell=N_{j-1}+1}^{N_j} t_{\ell}([v], \boldsymbol{c}) \right) \mathbf{1}_{[v]}$$

$$= \sum_{j=J_0}^{J} \sum_{[v] \in V_j} s([v], \boldsymbol{c}) \mathbf{1}_{[v]},$$

where we have let

$$t_{\ell}([v], \boldsymbol{c}) := c_{\ell} \boldsymbol{u}_{\ell}([v]), \quad s([v]_{\mathcal{G}_j}, \boldsymbol{c}) := \sum_{\ell=N_{j-1}+1}^{N_j} t_{\ell}(\boldsymbol{c}, [v]_{\mathcal{G}_j}).$$

It suggests that the evaluation of $\boldsymbol{f}(v)$ for $v \in V$ in (61) can be split into the following three steps, for which we show a pseudocode in Algorithm 5.

(1) For each j, let ℓ be the integer such that $N_{j-1} < \ell \le N_j$. Compute the product

$$t_{\ell}([v], \boldsymbol{c}) = c_{\ell} \boldsymbol{u}_{\ell}([v])$$

for all $[v] \in V_j$, which requires multiplications no more than $\mathrm{spoc}(\boldsymbol{u}_{\ell})$ steps. The total number of multiplications is thus no greater than $\sum_{\ell} \mathrm{spoc}(\boldsymbol{u}_{\ell})$.

(2) For each $j = J_0, \ldots, J$, evaluate the sums

$$s([v], \boldsymbol{c}) = \sum_{\ell=N_{j-1}+1}^{N_j} t_{\ell}(\boldsymbol{c}, [v]), \quad [v] \in V_j.$$

Again, the total number of summations is no more than $\sum_{\ell} \mathrm{spoc}(\boldsymbol{u}_{\ell})$.

(3) Evaluate $\boldsymbol{f} := \sum_{j=J_0}^{J_1} \sum_{[v] \in V_j} s([v], \boldsymbol{c}) \mathbf{1}_{[v]}$. We exploit the hierarchical chain structure to compute it efficiently. Let

$$\boldsymbol{f}_{J_1} := \sum_{j=J_0}^{J_1} \sum_{[v] \in V_j} s([v], \boldsymbol{c}) \mathbf{1}_{[v]}, \quad J_1 = J_0, \ldots, J,$$

which can be regarded as a function on $\mathcal{G}_{J_1}$. We observe that

$$\boldsymbol{f}_{J_1+1} = \boldsymbol{f}_{J_1} + \sum_{[v] \in V_{J_1+1}} s([v], \boldsymbol{c}) \mathbf{1}_{[v]},$$

for which the total number of summations is no more than $\sum_{j=J_0}^{J-1} N_{j+1}$.

Algorithm 5: Adjoint Discrete Fourier Transform on $\mathcal{G}$ (ADFT)

   Input : A global orthonormal basis $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^{|V|}$ on $\mathcal{G}_{J \to J_0}$ satisfying (43) and
       $\boldsymbol{c} = (c_\ell)_{\ell=1}^N$.
   Output: $\boldsymbol{f}$ given in (61).

1   Initialization: $N_{J_0-1} \leftarrow 0$. $\boldsymbol{f}_{J_0-1}([v]) \leftarrow 0$, $[v] \in \mathcal{G}_{J_0}$.
2   **for** $j=J_0$ **to** J **do**
3 $N_j \leftarrow |V_j|$.
4 $s([v], \boldsymbol{c}) \leftarrow 0$ for all $[v] \in V_j$.
5   **for** $\ell=N_{j-1}+1$ **to** N_j **do**
6 $s([v], \boldsymbol{c}) \leftarrow s([v], \boldsymbol{c}) + c_\ell \boldsymbol{u}_\ell([v])$, $[v] \in V_j$.
7   **end**
8 $\boldsymbol{f}_j([v]) \leftarrow \boldsymbol{f}_{j-1}([v]) + s([v], \boldsymbol{c})$, $[v] \in V_j$.
9   **end**
10 $\boldsymbol{f} \leftarrow \boldsymbol{f}_J$

From the above analysis, we obtain the upper bounds for the computational complexity of the discrete Fourier transform and its adjoint for a coarse-grained chain.

Theorem 16 *Let $\{\boldsymbol{u}_\ell\}_{\ell=1}^N$ be a global orthonormal basis for the coarse-grained chain $\mathcal{G}_{J \to J_0}$ of a graph $\mathcal{G}$. Then,*

$$\mathrm{flop}^*(\mathrm{DFT}_\mathcal{G}) = \mathcal{O}\left(\sum_{\ell=1}^N \mathrm{spoc}(\boldsymbol{u}_\ell)\right) \quad and \quad \mathrm{flop}^*(\mathrm{ADFT}_\mathcal{G}) = \mathcal{O}\left(\sum_{\ell=1}^N \mathrm{spoc}(\boldsymbol{u}_\ell)\right),$$

where the constants in the big $\mathcal{O}$ is independent of the N and $\mathrm{spoc}(\boldsymbol{u}_\ell)$ for any $\ell = 1, \ldots, N$.

Thus, when the $\mathrm{spoc}(\boldsymbol{u}_\ell) = \mathcal{O}(1)$. The DFT and ADFT have linear computational complexity.

6.2 Decimated $\mathcal{G}$-Framelet Transforms

In this section, we study the multi-level decimated framelet transforms on $\mathcal{G}$, which include the *framelet decomposition* and *reconstruction* algorithms. For a signal $\boldsymbol{f}$ on a graph $\mathcal{G}$ and a sequence of decimated tight framelets $\mathsf{DFS}\left(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J; \mathcal{G}_{J \to J_0}, \mathcal{Q}_{J \to J_0}\right)$, $J_1 = J_0, \ldots, J$ in (25), the *framelet decomposition* algorithm produces a sequence of the vectors as the *framelet approximation* and *detail coefficients*

$$\{\mathbf{v}_{J_0}\} \cup \{\mathbf{w}_j^n : n = 1, \ldots, r_j, \ j = J_0, \ldots, J-1\} \tag{63}$$

where for level $j = J_0, \ldots, J$, $\mathbf{v}_j$ is the vector of the approximation framelet coefficients for $\mathcal{G}_j$, and $\mathbf{w}_j^n$, $n = 1, \ldots, r_j$, is the vector of the detail framelet coefficients for $\mathcal{G}_{j+1}$:

$$\begin{aligned}
\mathbf{v}_j([p]) &:= \left\langle \boldsymbol{f}, \varphi_{j,[p]} \right\rangle, \quad [p] \in V_j, \\
\mathbf{w}_j^n([p]) &:= \left\langle \boldsymbol{f}, \psi_{j,[p]}^n \right\rangle, \quad [p] \in V_{j+1}, \ n = 1, \ldots, r_j.
\end{aligned} \tag{64}$$

35

The *framelet reconstruction* algorithm is to reconstruct $\boldsymbol{f}$ with the framelet coefficients in (63). We give a constructive implementation for the framelet decomposition and reconstruction by using generalized Fourier transforms on the graph, as follows.

Let $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ be a global orthonormal basis for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of the graph $\mathcal{G}$. For $j = J_0, \ldots, J$, let $\mathcal{Q}_j := \{\omega_{j,[p]} : [p] \in V_j\}$ be the set of weights on $\mathcal{G}_j$ and $\mathcal{Q}_{J \to J_0} := (\mathcal{Q}_J, \ldots, \mathcal{Q}_{J_0})$ for the coarse-grained chain $\mathcal{G}_{J \to J_0}$ which satisfies (37). For a finite index set Ω, we denote by $l_2(\Omega) := \{\boldsymbol{c} : \Omega \to \mathbb{C}\}$ all complex-valued sequences supported on Ω. For $j = J_0, \ldots, J$, let

$$\Omega_j := \{\ell : 1 \le \ell \le N_j\},$$

where $N_j := |V_j|$, and $l_2(\Omega_j)$ and $l_2(V_j)$ are the sequences supported on Ω_j and V_j respectively.

Definition 17 (Discrete Fourier transform) *The (generalized) discrete Fourier transform (DFT) $\mathbf{F}_j^* : l_2(V_j) \to l_2(\Omega_j)$ on $\mathcal{G}_j$ is*

$$(\mathbf{F}_j^* \mathbf{v})_\ell := \sum_{[p] \in V_j} \mathbf{v}([p]) \, \sqrt{\omega_{j,[p]}} \, \overline{\boldsymbol{u}_\ell([p])}, \quad \ell \in \Omega_j. \tag{65}$$

We say the sequence $\mathbf{F}_j^* \mathbf{v}$ a (V_j, Ω_j)-sequence and let $l_2(V_j, \Omega_j)$ be the set of all (V_j, Ω_j)-sequences.

Definition 18 (Adjoint discrete Fourier transform) *Define $\mathbf{F}_j : l_2(\Lambda_j) \to l_2(V_j)$ the (generalized) adjoint discrete Fourier transform (ADFT) (operator) on $\mathcal{G}_j$ by*

$$[\mathbf{F}_j \boldsymbol{c}]([p]) := \sum_{\ell \in \Omega_j} c_\ell \, \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]), \quad [p] \in V_j, \; \boldsymbol{c} = (c_\ell)_{\ell=1}^{N_j} \in l_2(\Omega_j). \tag{66}$$

We say the sequence $(\mathbf{F}_j \boldsymbol{c})$ a (Ω_j, V_j)-sequence. Let $l_2(\Omega_j, V_j)$ be the set of all (Ω_j, V_j)-sequences.

The following proposition shows that $\mathbf{F}_j$ and $\mathbf{F}_j^*$ are invertible when the weight sequence for the chain satisfies (37).

Proposition 19 *Let $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ be a global orthonormal basis for a coarse-grained chain $\mathcal{G}_{J \to J_0}$ of the graph $\mathcal{G}$. Let $\mathcal{Q}_{J \to J_0}$ be a weight sequence for $\mathcal{G}_{J \to J_0}$ which satisfies (37). Let $\mathbf{F}_j^*$ and $\mathbf{F}_j$ be the DFT and ADFT for $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^N$ given in (65) and (66). Then, $\mathbf{F}_j^*$ and $\mathbf{F}_j$ satisfy*

$$\mathbf{F}_j^* \mathbf{F}_j = \mathbf{I}_{V_j}, \qquad \mathbf{F}_j \mathbf{F}_j^* = \mathbf{I}_{\Omega_j},$$

where $\mathbf{I}_{V_j}$ and $\mathbf{I}_{\Omega_j}$ are the identity operators on $l_2(V_j)$ and $l_2(\Omega_j)$.

Proof Let $\boldsymbol{c} = (c_\ell)_{\ell=1}^{N_j}$ be a sequence in $l_2(\Omega_j)$. Then, for $\ell = 1, \ldots, N_j$,

$$[\mathbf{F}_j^*(\mathbf{F}_j \boldsymbol{c})]_\ell = \sum_{[p] \in V_j} [\mathbf{F}_j \boldsymbol{c}]([p]) \, \sqrt{\omega_{j,[p]}} \, \overline{\boldsymbol{u}_\ell([p])}$$

$$= \sum_{[p] \in V_j} \left(\sum_{\ell'=1}^{N_j} c_{\ell'} \, \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_{\ell'}([p]) \right) \sqrt{\omega_{j,[p]}} \, \overline{\boldsymbol{u}_\ell([p])}$$

$$= \sum_{\ell'=1}^{N_j} c_{\ell'} \left(\sum_{[p] \in V_j} \omega_{j,[p]} \, \boldsymbol{u}_{\ell'}([p]) \, \overline{\boldsymbol{u}_\ell([p])} \right)$$

$$= \sum_{\ell'=1}^{N_j} c_{\ell'} \, \mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$$

$$= c_\ell.$$

Hence, $[\mathbf{F}_j^*(\mathbf{F}_j \boldsymbol{c})] = \boldsymbol{c}$ for all $\boldsymbol{c} \in l_2(\Omega_j)$. In a similar way, we can prove $\mathbf{F}_j \mathbf{F}_j^* = \mathbf{I}_{\Omega_j}$. ∎

For every (Ω_j, V_j)-sequence $\mathbf{v}$, there exists a *unique* sequence $\boldsymbol{c} \in l_2(\Omega_j)$ such that $\mathbf{F}_j \boldsymbol{c} = \mathbf{v}$. This implies that we can well define *Fourier coefficients* by DFT as $\widehat{\mathbf{v}} := \boldsymbol{c} = \mathbf{F}_j^* \mathbf{v}$ for the graph signal $\mathbf{v}$.

Based on the discrete Fourier transform operators, we can define convolution, down-sampling, and upsampling operators. Let $h \in l_1(\mathbb{Z})$ be a mask and $\mathbf{v} \in l_2(\Omega_j, V_j)$ be a (Ω_j, V_j)-sequence. Let $\widehat{\mathbf{v}} := (\widehat{v}_\ell)_{\ell \in \Omega_j}$ be its discrete Fourier coefficient sequence.

Definition 20 (Discrete convolution) *The discrete convolution* $\mathbf{v} *_j h$ *is defined as the following sequence in* $l_2(\Omega_j, V_j)$:

$$[\mathbf{v} *_j h]([p]) := \sum_{\ell \in \Omega_j} \widehat{v}_\ell \, \widehat{h}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]), \quad [p] \in V_j. \tag{67}$$

That is, in the Fourier domain $(\widehat{\mathbf{v} *_j h})_\ell = \widehat{v}_\ell \, \widehat{h}\left(\frac{\lambda_\ell}{2^j}\right)$ for $\ell \in \Omega_j$.

Definition 21 (Downsampling) *We define the downsampling operator* $\downarrow_j: l_2(\Omega_j, V_j) \to l_2(\Omega_{j-1}, V_{j-1})$ *for a* (Ω_j, V_j)*-sequence* $\mathbf{v}$ *by*

$$[\mathbf{v} \downarrow_j]([p]) := \sum_{\ell \in \Omega_j} \widehat{v}_\ell \, \sqrt{\omega_{j-1,[p]}} \, \boldsymbol{u}_\ell([p]), \quad [p] \in V_{j-1}. \tag{68}$$

Definition 22 (Upsampling) *The upsampling operator* $\uparrow_j: l_2(\Omega_{j-1}, V_{j-1}) \to l_2(\Omega_j, V_j)$ *for a sequence* $\mathbf{v} \in l_2(\Omega_{j-1}, V_{j-1})$ *is defined by*

$$[\mathbf{v} \uparrow_j]([p]) := \sum_{\ell \in \Omega_{j-1}} \widehat{v}_\ell \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]), \quad [p] \in V_j. \tag{69}$$

For a mask $h \in l_1(\mathbb{Z})$, we denote $h^\star$ the filter such that its Fourier series $\widehat{h^\star}(\xi) = \overline{\widehat{h}(\xi)}$, $\xi \in \mathbb{R}$. With the above notions, we have the following theorem which shows the framelet decomposition and reconstruction for the framelet approximation and detail coefficients at different scales.

Theorem 23 *Let* $\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$, $J_1 = J_0, \ldots, J$ *be the sequence of the decimated tight framelet systems given in* (25), *and the framelet approximation and detail coefficients given in* (63). *Then,*

(i) $\mathbf{v}_j$ *is a* (Ω_j, V_j)-*sequence for all* $j = J_0, \ldots, J$, *and* $\mathbf{w}_j^n, n = 1, \ldots, r_j$ *are* (Ω_{j+1}, V_{j+1})-*sequences for all* $j = J_0 - 1, \ldots, J$.

(ii) *For any* $J_0 + 1 \leq j \leq J$, *the decomposition is given by*

$$\mathbf{v}_{j-1} = (\mathbf{v}_j *_j a_j^\star)\!\downarrow_j, \quad \mathbf{w}_{j-1}^n = (\mathbf{v}_j *_j (b_j^{(n)})^\star), \quad n = 1, \ldots, r_{j-1}. \tag{70}$$

(iii) *For any* $J_0 + 1 \leq j \leq J$, *the reconstruction is given by*

$$\mathbf{v}_j = (\mathbf{v}_{j-1}\!\uparrow_j) *_j a_j + \sum_{n=1}^{r_{j-1}} \mathbf{w}_{j-1}^n *_j b_j^{(n)}. \tag{71}$$

Remark 24 *For scale (or level)* j, *the low-pass sequence* $\mathbf{v}_j$ *is in* $l_2(\Omega_j, V_j)$ *while the high-pass sequences* $\mathbf{w}_j^n$ *are in* $l_2(\Omega_{j+1}, V_{j+1})$. *Figure 2 shows the framelet decomposition and reconstruction for the one-level* $\mathcal{G}$-*framelet transforms based on a filter bank* $\{a_j; b_j^{(1)}, \ldots, b_j^{(r_j)}\}$, *which consists of discrete convolution, upsampling and downsampling on* $\mathcal{G}$, *as stated in Theorem 23. Here the operator* $*_j b_j^{(n)}$ *ranges over* $n = 1, \ldots, r$ *and the operation* $+_r$ *is the summation over the low-pass filtered coefficient sequence and all* r *high-pass filtered coefficient sequences.*

Proof [Proof of Theorem 23] For $\mathbf{v}_j$ and $\mathbf{w}_j^n$ in (63), by (36) and $\operatorname{supp}\widehat{\alpha_\varepsilon} \subseteq [-1, 1]$, we have

$$\mathbf{v}_j([p]) = \sum_{\ell \in \Omega_j} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p])$$

and

$$\mathbf{w}_{j-1}^n([p]) = \sum_{\ell \in \Omega_j} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\beta^n}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)} \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]).$$

Hence, the statement in (i) holds. Moreover, the discrete Fourier coefficient sequences $\widehat{\mathbf{v}_j} = (\widehat{\mathbf{v}}_{j,\ell})_{\ell \in \Omega_j}$ and $\widehat{\mathbf{w}_{j-1}^n} = (\widehat{\mathbf{w}}_{j-1,\ell}^n)_{\ell \in \Omega_j}$ are given by $\widehat{\mathbf{v}}_\ell = \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_j}\right)}$ and $\widehat{\mathbf{w}}_{j-1,\ell}^n = \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\beta_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)}$, $\ell \in \Omega_j$.

Since $\mathbf{v}_{j-1}$ is a (Λ_{j-1}, V_{j-1})-sequence, (23) implies that for $[p] \in V_{j-1}$, the approximate framelet coefficients

$$
\begin{aligned}
\mathbf{v}_{j-1}([p]) &= \sum_{\ell \in \Omega_{j-1}} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)} \sqrt{\omega_{j-1,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= \sum_{\ell \in \Omega_{j-1}} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \, \overline{\widehat{a}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \sqrt{\omega_{j-1,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= \sum_{\ell \in \Omega_{j-1}} \widehat{\mathbf{v}}_{j,\ell} \, \overline{\widehat{a}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \sqrt{\omega_{j-1,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= [(\mathbf{v}_j *_j a_j^\star) \!\downarrow_j]([p]).
\end{aligned}
$$

And for the detail framelet coefficients, for $[p] \in V_j$ and $n = 1, \ldots, r_{j-1}$,

$$
\begin{aligned}
\mathbf{w}_{j-1}^n([p]) &= \sum_{\ell \in \Omega_j} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\beta_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_{j-1}}\right)} \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= \sum_{\ell \in \Omega_j} \widehat{\boldsymbol{f}}_\ell \, \overline{\widehat{\alpha_\varepsilon}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \, \overline{\widehat{b_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right)} \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= [(\mathbf{v}_j *_j (b_j^{(n)})^\star)]([p]).
\end{aligned}
$$

This gives (70).

Using the equations $\mathbf{v}_{j-1} = (\mathbf{v}_j *_j (a_j)^\star) \!\downarrow_j$ and $\mathbf{w}_{j-1}^n = \mathbf{v}_j *_j (b_j^{(n)})^\star$, we write

$$
\begin{aligned}
\widetilde{\mathbf{v}} &:= (\mathbf{v}_{j-1} \!\uparrow_j) *_j a_j + \sum_{n=1}^{r_{j-1}} \mathbf{w}_{j-1}^n *_j b_j^{(n)} \\
&= \left(((\mathbf{v}_j *_j (a_j)^\star) \!\downarrow_j) \!\uparrow_j \right) *_j a_j + \sum_{n=1}^{r_{j-1}} (\mathbf{v}_j *_j (b_j^{(n)})^\star) *_j b_j^{(n)}
\end{aligned}
$$

where r_{j-1} is the number of high-pass filters at level $j - 1$. This together with (67), (68), (69), and (39) gives, for $[p] \in V_j$,

$$
\begin{aligned}
\widetilde{\mathbf{v}}([p]) &= \sum_{\ell \in \Omega_j} \widehat{\mathbf{v}}_{j,\ell} \left(\left| \widehat{a}_j\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2 + \sum_{n=1}^{r_{j-1}} \left| \widehat{b_j^{(n)}}\left(\frac{\lambda_\ell}{\Lambda_j}\right) \right|^2 \right) \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= \sum_{\ell \in \Omega_j} \widehat{\mathbf{v}}_{j,\ell} \sqrt{\omega_{j,[p]}} \, \boldsymbol{u}_\ell([p]) \\
&= \mathbf{v}_j([p]).
\end{aligned}
$$

Thus, (71) holds. ∎

Recursively using the decimated $\mathcal{G}$-framelet transforms in (70) and (71), we would obtain the multi-level decimated $\mathcal{G}$-framelet transforms on the graph, in which the graph signal has zero-loss.

Definition 25 (Decimated $\mathcal{G}$-framelet transforms) *For a sequence of graph data $\mathbf{v}_{J_1} \in l(\Lambda_{J_1}, \Omega_{J_1})$, $J_0 \leq J_1 \leq J$ on $\mathcal{G}$, the multi-level framelet decomposition on $\mathcal{G}$ is to compute for $j = J, \ldots, J_1 + 1$,*

$$\mathbf{v}_{j-1} = (\mathbf{v}_j *_j a_j^\star)\!\downarrow_j, \quad \mathbf{w}_{j-1}^n = (\mathbf{v}_j *_j (b_j^{(n)})^\star), \quad n = 1, \ldots, r_{j-1}. \tag{72}$$

For a sequence $(\mathbf{w}_{J-1}^1, \ldots, \mathbf{w}_{J-1}^{r_{J-1}}, \ldots, \mathbf{w}_{J_0}^1, \ldots, \mathbf{w}_{J_0}^{r_{J_0}}, \mathbf{v}_{J_0})$ of the framelet coefficients derived from a multi-level decomposition, the multi-level $\mathcal{G}$-framelet reconstruction is to evaluate for $j = J_0 + 1, \ldots, J$,

$$\mathbf{v}_j = (\mathbf{v}_{j-1}\!\uparrow_j) *_j a_j + \sum_{n=1}^{r_{j-1}} \mathbf{w}_{j-1}^n *_j b_j^{(n)}. \tag{73}$$

We call (72) and (73) multi-level decimated $\mathcal{G}$-framelet transforms.

Figure 4 below illustrates a flowchart for a two-level decomposition and reconstruction of the $\mathcal{G}$-framelet transforms.
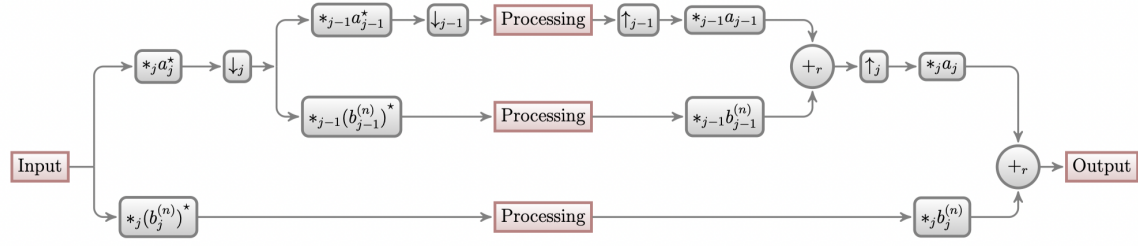

Figure 4: Two-level $\mathcal{G}$-framelet decomposition and reconstruction based on the filter banks $\{a_{j-1}; b_{j-1}^{(1)}, \ldots, b_{j-1}^{(r_{j-1})}\}$ and $\{a_j; b_j^{(1)}, \ldots, b_j^{(r_j)}\}$. Graph signal input is decomposed into the low-pass and high-pass coefficients in the first level decomposition. The low-pass coefficient is then decomposed into low-pass and high-pass coefficients at the next scale. The reconstruction uses the low-pass coefficients at the coarsest level (or the first level) of the chain and all high-pass coefficients.

Definition 26 (Framelet analysis and synthesis operators) *The multi-level decimated $\mathcal{G}$-framelet transforms define the multi-level $\mathcal{G}$-framelet analysis operator*

$$\mathbf{W} : l_2(\Lambda_J, \Omega_J) \to l_2(\Omega_{J-1})^{1 \times r_{J-1}} \times l_2(\Omega_{J-2})^{1 \times r_{J-2}} \times \cdots \times l_2(\Omega_{J_0})^{1 \times r_{J_0}} \times l_2(\Omega_{J_0})$$

with

$$\mathbf{W}\mathbf{v}_J = (\mathbf{w}_{J-1}^1, \ldots, \mathbf{w}_{J-1}^{r_{J-1}}, \ldots, \mathbf{w}_{J_0}^1, \ldots, \mathbf{w}_{J_0}^{r_{J_0}}, \mathbf{v}_{J_0}), \quad \mathbf{v}_J \in l_2(\Lambda_J, \Omega_J); \tag{74}$$

and the multi-level $\mathcal{G}$-framelet synthesis operator

$$\mathbf{V} : l_2(\Omega_{J-1})^{1 \times r_{J-1}} \times l_2(\Omega_{J-2})^{1 \times r_{J-2}} \times \cdots \times l_2(\Omega_{J_0})^{1 \times r_{J_0}} \times l_2(\Omega_{J_0}) \to l_2(\Lambda_J, \Omega_J)$$

with

$$\mathbf{V}(\mathbf{w}_{J-1}^1, \ldots, \mathbf{w}_{J-1}^{r_{J-1}}, \ldots, \mathbf{w}_{J_0}^1, \ldots, \mathbf{w}_{J_0}^{r_{J_0}}, \mathbf{v}_{J_0}) = \mathbf{v}_J.$$

Under the condition of Theorem 23, the analysis and synthesis operators are invertible on $l_2(\Lambda_j, \Omega_j)$ for $J_0 \leq j \leq J$, that is, $\mathbf{V}\mathbf{W} = \mathbf{I}|_{l_2(\Lambda_j, \Omega_j)}$. The type of the filter banks and the number of high passes in each filter bank in the multi-level $\mathcal{G}$-framelet transforms may vary at multi scales.

Algorithm 6: Decomposition for F$\mathcal{G}$T

 Input : $\mathbf{v}_J$, which is a (Λ_J, Ω_J)-sequence; filter bank
 Output: $(\widehat{\mathrm{w}}_{J-1}^1, \ldots, \widehat{\mathrm{w}}_{J-1}^{r_{J-1}}, \ldots, \widehat{\mathrm{w}}_{J_0}^1, \ldots, \widehat{\mathrm{w}}_{J_0}^{r_{J_0}}, \widehat{\mathrm{v}}_{J_0})$

1 $\mathbf{v}_J \longrightarrow \widehat{\mathrm{v}}_J$ `// fast DFT`
2   **for** $j \leftarrow J$ **to** $J_0 + 1$ **do**
3 $\widehat{\mathrm{v}}_{j-1} \longleftarrow \widehat{\mathrm{v}}_{j,\cdot}\, \overline{\widehat{a_j}}\,(\lambda./\Lambda_j)$ `// discrete convolution at level` j
4   `// downsample from level` j `to` $j-1$
5   **for** $n \leftarrow 1$ **to** r_{j-1} **do**
6 $\widehat{\mathrm{w}}_{j-1}^n \longleftarrow \widehat{\mathrm{v}}_{j,\cdot}\, \overline{(\widehat{b_j^{(n)}})}\,(\lambda./\Lambda_j)$ `// discrete convolution at level` j
     $\mathbf{w}_{j-1}^n \longleftarrow \widehat{\mathrm{w}}_{j-1}^n$ `// fast DFT`
7   **end**
8 $\mathbf{v}_{J_0} \longleftarrow \widehat{\mathrm{v}}_{J_0}$ `// fast DFT`
9   **end**

6.3 Fast $\mathcal{G}$-Framelet Transforms

In this section, we show that the decomposition and reconstruction of the decimated $\mathcal{G}$-framelet transforms can be implemented in linear computational complexity, that is, in steps proportional to the size of the graph by fast discrete Fourier transforms for the coarse-grained chain of the graph $\mathcal{G}$. We call this algorithm *fast $\mathcal{G}$-framelet transform*, or F$\mathcal{G}$T. The fast computation is due to the following relation between the decimated $\mathcal{G}$-framelet transforms and discrete Fourier transforms on $\mathcal{G}$.

Proposition 27 *For $j = J_0 + 1, \ldots, J$, the $\mathcal{G}$-framelet decomposition and reconstruction at level j can be written as*

$$\mathbf{v}_{j-1} = \mathbf{F}_{j-1}(\mathbf{v}_j \,\widehat{*_j\,(a_j)}\,)^\star, \quad \mathbf{w}_{j-1}^n = \mathbf{F}_j(\mathbf{v}_j \,\widehat{*_j\,(b_j^{(n)})}\,)^\star, \quad n = 1, \ldots, r_{j-1}$$

and

$$\mathbf{v}_j = \left(\mathbf{F}_j^*(\mathbf{v}_{j-1})\right) *_j a_j + \sum_{n=1}^{r_{j-1}} \left(\mathbf{F}_j^*(\mathbf{w}_{j-1}^n)\right) *_j b_j^{(n)}.$$

By Proposition: 16, the discrete Fourier transforms are implementable fast when the global orthornormal basis is appropriately chosen. Algorithms 6 and 7 below give a pseudocode for the decomposition and reconstruction of F$\mathcal{G}$T based on the formula in Proposition 27. Using the Haar global orthonormal basis for the chain(see Section 5.3), the discrete Fourier transforms for the input data with size N has the computational cost $\mathcal{O}(N)$. This and Algorithms 6 and 7 then show that the computational cost for decimated $\mathcal{G}$-framelet transforms is $\mathcal{O}(N)$.

Algorithm 7: Reconstruction for F$\mathcal{G}$T

    Input : $(\widehat{\mathrm{w}}_{J-1}^1, \ldots, \widehat{\mathrm{w}}_{J-1}^{r_{J-1}}, \ldots, \widehat{\mathrm{w}}_{J_0}^1, \ldots, \widehat{\mathrm{w}}_{J_0}^{r_{J_0}}, \widehat{\mathrm{v}}_{J_0})$
    Output: $\mathbf{v}_J$

1 $\widehat{\mathrm{v}}_{J_0} \longleftarrow \mathbf{v}_{J_0}$ `// fast DFT`
2   **for** $j \leftarrow J_0 + 1$ **to** J **do**
3 | **for** $n \leftarrow 1$ **to** r_{j-1} **do**
4 | | $\widehat{\mathrm{w}}_{j-1}^n \longleftarrow \mathbf{w}_{j-1}^n$ `// fast DFT`
5 | **end**

6 | $\widehat{\mathrm{v}}_j \longleftarrow (\widehat{\mathrm{v}}_{j-1,\cdot}) \, \widehat{a}_j \, (\lambda./\Lambda_j) + \sum_{n=1}^{r_{j-1}} \widehat{\mathrm{w}}_{j-1,\cdot}^n \, \widehat{\left(b_j^{(n)}\right)} (\lambda./\Lambda_j)$
7   **end**
8 $\mathbf{v}_J \longleftarrow \widehat{\mathrm{v}}_J$ `// fast ADFT`
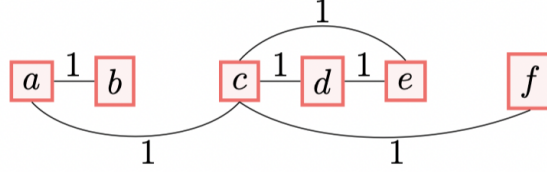

Figure 5: Graph $\mathcal{G}$, where the vertices are represented by the boxes and the edges are by the lines for the pairs of connected vertices, and the weight for each edge is 1.

7. Toy Example to Illustrate Framelet Construction

In this section, we show full construction of the decimated framelet system on a graph using the example in Figure 5. The graph $\mathcal{G} = (V, E, \boldsymbol{w})$ has vertices and edges

$$V := \{a, b, c, d, e, f\} \text{ and } E = \{(a,b), (a,c), (c,d), (c,e), (c,f), (d,e)\}.$$

We apply Algorithm 1 for clustering. As shown by Figure 1, the resulting coarse-grained chain of $\mathcal{G}$ is $\mathcal{G}_{3\to 0}$: the graph $\mathcal{G} =: \mathcal{G}_3$ which is at the bottom level 3 includes all six nodes of the original graph; at level 2 is $\mathcal{G}_2$ which has 3 clusters with three initial centers $\{a\}, \{c\}, \{f\}$; the next level is $\mathcal{G}_1$ with 2 clusters; and eventually the coarsest level $\mathcal{G}_0$ has 1 cluster, which is the root of the chain (or tree). We give the detailed description of the coarse-grained chain $\mathcal{G}_{3\to 0} = (\mathcal{G}_3, \mathcal{G}_2, \mathcal{G}_1, \mathcal{G}_0)$, as follow.

(1) At the finest level 3, $\mathcal{G}_3 := \mathcal{G}$, of which each vertex is a leaf and a cluster of singleton. The graph $\mathcal{G}$ is associated with the adjacency matrix $\boldsymbol{w}$, the degree matrix $\boldsymbol{d}$, and the

graph Laplacian matrix $\mathcal{L} := \boldsymbol{d} - \boldsymbol{w}$:

$$
\boldsymbol{w} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{d} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathcal{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 4 & -1 & -1 & -1 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}.
$$
(75)

Here, the row or column is with respect to the vertices in the order a, b, c, d, e, f.

(2) At level 2, we obtain three clusters $[a]_{\mathcal{G}_2} = \{a, b\}$, $[c]_{\mathcal{G}_2} = \{c, d, e\}$, and $[f]_{\mathcal{G}_2} = \{f\}$ for the coarse-grained graph $\mathcal{G}_2 := (V_2, E_2, \boldsymbol{w}_2)$ of $\mathcal{G}_3$, where $V_2 = \{[a]_{\mathcal{G}_2}, [c]_{\mathcal{G}_2}, [f]_{\mathcal{G}_2}\}$, $E_2 = \{([a]_{\mathcal{G}_2}, [a]_{\mathcal{G}_2}), ([a]_{\mathcal{G}_2}, [c]_{\mathcal{G}_2}), ([c]_{\mathcal{G}_2}, [c]_{\mathcal{G}_2}), ([c]_{\mathcal{G}_2}, [f]_{\mathcal{G}_2})\}$, and

$$
\boldsymbol{w}_2 = \frac{1}{12} \begin{bmatrix} 2 & 1 & 0 \\ 1 & 6 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \boldsymbol{d}_2 = \frac{1}{12} \begin{bmatrix} 3 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathcal{L}_2 = \frac{1}{12} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}.
$$
(76)

(3) At level 1, we obtain two clusters $[a]_{\mathcal{G}_1} = \{a, b\}$ and $[c]_{\mathcal{G}_1} = \{c, d, e, f\}$ for the coarse-grained graph $\mathcal{G}_1 := (V_1, E_1, \boldsymbol{w}_1)$ of $\mathcal{G}_2$, where $V_1 = \{[a]_{\mathcal{G}_1}, [c]_{\mathcal{G}_1}\}$, $E_1 = \{([a]_{\mathcal{G}_1}, [a]_{\mathcal{G}_1}), ([a]_{\mathcal{G}_1}, [c]_{\mathcal{G}_1}), ([c]_{\mathcal{G}_1}, [c]_{\mathcal{G}_1})\}$, and

$$
\boldsymbol{w}_1 = \frac{1}{12} \begin{bmatrix} 2 & 1 \\ 1 & 8 \end{bmatrix} \quad \boldsymbol{d}_1 = \frac{1}{12} \begin{bmatrix} 3 & 0 \\ 0 & 9 \end{bmatrix} \quad \mathcal{L}_1 = \frac{1}{12} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.
$$
(77)

(4) At the coarsest level 0, we reach the root $\mathcal{G}_0 := (V_0, E_0, \boldsymbol{w}_0)$, where $V_0 = \{[a, b, c, d, e, f] =: [a]_{\mathcal{G}_0}\}$ has only one cluster $[a]_{\mathcal{G}_0}$ which contains all vertices from $\mathcal{G}$, $E_0 = \{([a]_{\mathcal{G}_0}, [a]_{\mathcal{G}_0})\}$, and $\boldsymbol{w}_0 = 1/12$.

Next, we build the Haar global orthonormal eigen-pairs $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^6$ for $L_2(\mathcal{G})$. It utilises the hierarchical information of the chain $\mathcal{G}_{3\to0}$, and the resulting Haar basis satisfies for $j = 0, 1, 2, 3$,

$$
\boldsymbol{u}_\ell(v) \equiv \text{const} \quad \forall v \in [v]_{\mathcal{G}_j} \text{ and } \forall \ell \le |V_j|.
$$
(78)

(5) At level 0, $\mathcal{G}_0$ is a graph of singleton. In this case, $\lambda_1^{\mathcal{G}_0} = 0$ and $\boldsymbol{u}_1^{\mathcal{G}_0} = 1$. We then let

$$
\boldsymbol{u}_1 = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^\top.
$$

(6) At level 1, the eigenvalues of $\mathcal{L}_1$ as in (77) are $\lambda_1^{\mathcal{G}_1} = 0$ and $\lambda_2^{\mathcal{G}_1} = \frac{1}{6}$. The eigenvectors of $\mathcal{L}_1$ with respect to $0, \frac{1}{6}$ are

$$
\boldsymbol{u}_1^{\mathcal{G}_1} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \end{bmatrix}^\top, \quad \boldsymbol{u}_2^{\mathcal{G}_1} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \end{bmatrix}^\top.
$$

43

By the discussion in Section 5.3, we extend $\boldsymbol{u}_2^{\mathcal{G}_1}$, with respect to clusters $[a]_{\mathcal{G}_1}$ and $[c]_{\mathcal{G}_1}$, to $\boldsymbol{u}_2^{(1)}$ on $\mathcal{G}$:

$$\boldsymbol{u}_2^{(1)} = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & 1 & -1 & -1 & -1 & -1 \end{bmatrix}^\top.$$

Apply the Gram-Schmidt orthonormalization process to $\{\boldsymbol{u}_1, \boldsymbol{u}_2^{(1)}\}$, we then obtain a new vector

$$\boldsymbol{u}_2 = \frac{1}{2\sqrt{3}} \begin{bmatrix} 2 & 2 & -1 & -1 & -1 & -1 \end{bmatrix}^\top.$$

(7) At level 2, the eigenvalues of $\mathcal{L}_2$ in (76) are $\lambda_1^{\mathcal{G}_2} = 0$, $\lambda_2^{\mathcal{G}_2} = \frac{1}{12}$, $\lambda_3^{\mathcal{G}_2} = \frac{1}{4}$. The eigenvectors of $\mathcal{L}_2$ with respect to 0, $\frac{1}{12}$, $\frac{1}{4}$ are

$$\boldsymbol{u}_1^{\mathcal{G}_2} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^\top \quad \boldsymbol{u}_2^{\mathcal{G}_2} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}^\top \quad \boldsymbol{u}_3^{\mathcal{G}_2} = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}^\top.$$

We extend $\boldsymbol{u}_3^{\mathcal{G}_2}$, with respect to clusters $[a]_{\mathcal{G}_2}$, $[c]_{\mathcal{G}_2}$ and $[f]_{\mathcal{G}_2}$, to $\boldsymbol{u}_3^{(2)}$ on $\mathcal{G}$ as

$$\boldsymbol{u}_3^{(2)} = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & 1 & -1 & -1 & -1 & 1 \end{bmatrix}^\top.$$

Apply the Gram-Schmidt orthonormalization process to $\{\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3^{\mathcal{G}_3}\}$, we then obtain a new vector $\boldsymbol{u}_3$:

$$\boldsymbol{u}_3 = \frac{1}{2\sqrt{3}} \begin{bmatrix} 0 & 0 & -1 & -1 & -1 & 3 \end{bmatrix}^\top.$$

(8) Continue the above similar steps, at level 3, from the graph Laplacian in (75), we obtain an orthonormal basis $\{\boldsymbol{u}_1, \dots, \boldsymbol{u}_6\}$ for $L_2(\mathcal{G})$ satisfying (78) as

$$\boldsymbol{u}_1 = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^\top$$

$$\boldsymbol{u}_2 = \frac{1}{2\sqrt{3}} \begin{bmatrix} 2 & 2 & -1 & -1 & -1 & -1] \end{bmatrix}^\top$$

$$\boldsymbol{u}_3 = \frac{1}{2\sqrt{3}} \begin{bmatrix} 0 & 0 & -1 & -1 & -1 & 3 \end{bmatrix}^\top$$

$$\boldsymbol{u}_4 = \frac{1}{\sqrt{6}} \begin{bmatrix} 0 & 0 & 2 & -1 & -1 & 0 \end{bmatrix}^\top$$

$$\boldsymbol{u}_5 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}^\top$$

$$\boldsymbol{u}_6 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}^\top,$$

and we let $\lambda_\ell = \ell - 1$ for $\ell = 1, \dots, 6$.

Based on the orthonormal eigen-pair $\{(\boldsymbol{u}_\ell, \lambda_\ell)\}_{\ell=1}^6$, we next construct decimated framelet systems as in (25).

(9) At level 3, $\mathcal{G}_3 \equiv \mathcal{G}$ and $[p]_{\mathcal{G}_3} = \{p\} \in V_3$ are singletons. Let $\omega_{j,[p]} = 1$ for all $p \in V$, and $\widehat{\alpha_3}\left(\frac{\lambda_\ell}{\Lambda_3}\right) \equiv 1$ for all ℓ, then by (24),

$$\varphi_{3,[p]}(v) = \varphi_{3,p} = \delta_{[p],v} \quad \forall p, v \in V.$$

There is no framelet $\psi_{j,[p]}^{(n)}(v)$ at this level. The system $\{\varphi_{3,[p]} : [p] \in V_3\} = \{\delta_{p,v} : p, v \in V\}$ is the trivial orthonormal basis.

(10) At level 2, by the discussion in Section 5.4, the scaling functions are given by

$$\widehat{\alpha_2}\left(\frac{\lambda_\ell}{\Lambda_2}\right) = \begin{cases} 1 & \ell = 1,2 \\ \frac{1}{\sqrt{2}} & \ell = 3 \\ 0 & \text{otherwise.} \end{cases}$$

$$\widehat{\beta_2^{(1)}}\left(\frac{\lambda_\ell}{\Lambda_2}\right) = \begin{cases} \frac{1}{\sqrt{2}} & \ell = 3,5 \\ 1 & \ell = 4 \\ 0 & \text{otherwise.} \end{cases} \qquad \widehat{\beta_2^{(2)}}\left(\frac{\lambda_\ell}{\Lambda_2}\right) = \begin{cases} \frac{1}{\sqrt{2}} & \ell = 5 \\ 1 & \ell = 6 \\ 0 & \text{otherwise.} \end{cases}$$

It can be verified that $\left|\widehat{\alpha_2}(\frac{\lambda_\ell}{\Lambda_2})\right|^2 + \left|\widehat{\beta_2^{(1)}}(\frac{\lambda_\ell}{\Lambda_2})\right|^2 + \left|\widehat{\beta_2^{(2)}}(\frac{\lambda_\ell}{\Lambda_2})\right|^2 \equiv 1$ for all ℓ. By (24), we then obtain $\varphi_{2,[p]}$, $\psi_{2,[p]}^{(1)}$ and $\psi_{2,[p]}^{(2)}$ in Table 1, where the corresponding weights on V_2 are

$$\omega_{2,[a]_{\mathcal{G}_2}} = 2 \quad \omega_{2,[c]_{\mathcal{G}_2}} = 3 \quad \omega_{2,[f]_{\mathcal{G}_2}} = 1.$$

(11) At level 1, set

$$\widehat{\alpha_1}\left(\frac{\lambda_\ell}{\Lambda_1}\right) = \begin{cases} 1 & \ell = 1 \\ \frac{1}{\sqrt{2}} & \ell = 2 \\ 0 & \text{otherwise.} \end{cases} \qquad \widehat{\beta_1^{(1)}}\left(\frac{\lambda_\ell}{\Lambda_1}\right) = \begin{cases} \frac{1}{\sqrt{2}} & \ell = 2 \\ \frac{1}{\sqrt{2}} & \ell = 3 \\ 0 & \text{otherwise.} \end{cases}$$

which satisfies $\left|\widehat{\alpha_1}(\frac{\lambda_\ell}{\Lambda_1})\right|^2 + \left|\widehat{\beta_1^{(1)}}(\frac{\lambda_\ell}{\Lambda_1})\right|^2 = \left|\widehat{\alpha_2}(\frac{\lambda_\ell}{\Lambda_2})\right|^2$ for all ℓ. Letting the weights on V_1 as $\omega_{1,[a]_{\mathcal{G}_1}} = 2, \omega_{2,[c]_{\mathcal{G}_1}} = 4$, by (24), we then obtain $\varphi_{1,[p]}$ and $\psi_{1,[p]}^{(1)}$ in Table 2.

(12) At level 0, set

$$\widehat{\alpha_0}\left(\frac{\lambda_\ell}{\Lambda_0}\right) = \begin{cases} 1 & \ell = 1 \\ 0 & \text{otherwise.} \end{cases} \qquad \widehat{\beta_0^{(1)}}\left(\frac{\lambda_\ell}{\Lambda_0}\right) = \begin{cases} \frac{1}{\sqrt{2}} & \ell = 2 \\ 0 & \text{otherwise.} \end{cases}$$

Here $\left|\widehat{\alpha_0}(\frac{\lambda_\ell}{\Lambda_0})\right|^2 + \left|\widehat{\beta_0^{(1)}}(\frac{\lambda_\ell}{\Lambda_0})\right|^2 = \left|\widehat{\alpha_1}(\frac{\lambda_\ell}{\Lambda_1})\right|^2$ for all ℓ. Let the weights on V_0 as $\omega_{1,[a]_{\mathcal{G}_0}} = 6$, then by (24), we obtain $\varphi_{0,[p]}$ and $\psi_{0,[p]}^{(1)}$ in Table 3.

	a	b	c	d	e	f
$\varphi_{2,[a]_{\mathcal{G}_2}}$	$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$	0	0	0	0
$\varphi_{2,[c]_{\mathcal{G}_2}}$	0	0	$\frac{\sqrt{3}}{4}+\frac{\sqrt{6}}{24}$	$\frac{\sqrt{3}}{4}+\frac{\sqrt{6}}{24}$	$\frac{\sqrt{3}}{4}+\frac{\sqrt{6}}{24}$	$\frac{\sqrt{3}}{4}-\frac{\sqrt{6}}{8}$
$\varphi_{2,[f]_{\mathcal{G}_2}}$	0	0	$\frac{1}{4}-\frac{\sqrt{2}}{8}$	$\frac{1}{4}-\frac{\sqrt{2}}{8}$	$\frac{1}{4}-\frac{\sqrt{2}}{8}$	$\frac{1}{4}+\frac{3\sqrt{2}}{8}$
$\psi^{(1)}_{2,[a]_{\mathcal{G}_3}}$	0	0	0	0	0	0
$\psi^{(1)}_{2,[b]_{\mathcal{G}_3}}$	0	0	0	0	0	0
$\psi^{(1)}_{2,[c]_{\mathcal{G}_3}}$	0	0	$\frac{2}{3}+\frac{\sqrt{2}}{24}$	$-\frac{1}{3}+\frac{\sqrt{2}}{24}$	$-\frac{1}{3}+\frac{\sqrt{2}}{24}$	$-\frac{\sqrt{2}}{8}$
$\psi^{(1)}_{2,[d]_{\mathcal{G}_3}}$	0	0	$-\frac{1}{3}+\frac{\sqrt{2}}{24}$	$\frac{1}{6}+\frac{7\sqrt{2}}{24}$	$\frac{1}{6}-\frac{5\sqrt{2}}{24}$	$-\frac{\sqrt{2}}{8}$
$\psi^{(1)}_{2,[e]_{\mathcal{G}_3}}$	0	0	$-\frac{1}{3}+\frac{\sqrt{2}}{24}$	$\frac{1}{6}-\frac{5\sqrt{2}}{24}$	$\frac{1}{6}+\frac{7\sqrt{2}}{24}$	$-\frac{\sqrt{2}}{8}$
$\psi^{(1)}_{2,[f]_{\mathcal{G}_3}}$	0	0	$-\frac{\sqrt{2}}{8}$	$-\frac{\sqrt{2}}{8}$	$-\frac{\sqrt{2}}{8}$	$\frac{3\sqrt{2}}{8}$
$\psi^{(2)}_{2,[a]_{\mathcal{G}_3}}$	$\frac{1}{2}$	$-\frac{1}{2}$	0	0	0	0
$\psi^{(2)}_{2,[b]_{\mathcal{G}_3}}$	$-\frac{1}{2}$	$\frac{1}{2}$	0	0	0	0
$\psi^{(2)}_{2,[c]_{\mathcal{G}_3}}$	0	0	0	0	0	0
$\psi^{(2)}_{2,[d]_{\mathcal{G}_3}}$	0	0	0	$\frac{\sqrt{2}}{4}$	$-\frac{\sqrt{2}}{4}$	0
$\psi^{(2)}_{2,[e]_{\mathcal{G}_3}}$	0	0	0	$-\frac{\sqrt{2}}{4}$	$\frac{\sqrt{2}}{4}$	0
$\psi^{(2)}_{2,[f]_{\mathcal{G}_3}}$	0	0	0	0	0	0

Table 1: Decimated framelets $\varphi_{2,[p]_{\mathcal{G}_2}}, \psi^{(1)}_{2,[p]_{\mathcal{G}_3}}$ and $\psi^{(2)}_{2,[p]_{\mathcal{G}_3}}$ at level $j=2$

	a	b	c	d	e	f
$\varphi_{1,[a]_{\mathcal{G}_1}}$	$\frac{1}{3}+\frac{\sqrt{2}}{6}$	$\frac{1}{3}+\frac{\sqrt{2}}{6}$	$-\frac{1}{6}+\frac{\sqrt{2}}{6}$	$-\frac{1}{6}+\frac{\sqrt{2}}{6}$	$-\frac{1}{6}+\frac{\sqrt{2}}{6}$	$-\frac{1}{6}+\frac{\sqrt{2}}{6}$
$\varphi_{1,[c]_{\mathcal{G}_1}}$	$\frac{1}{3}-\frac{\sqrt{2}}{6}$	$\frac{1}{3}-\frac{\sqrt{2}}{6}$	$\frac{1}{3}+\frac{\sqrt{2}}{12}$	$\frac{1}{3}+\frac{\sqrt{2}}{12}$	$\frac{1}{3}+\frac{\sqrt{2}}{12}$	$\frac{1}{3}+\frac{\sqrt{2}}{12}$
$\psi^{(1)}_{1,[a]_{\mathcal{G}_2}}$	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{1}{6}$	$-\frac{1}{6}$	$-\frac{1}{6}$	$-\frac{1}{6}$
$\psi^{(1)}_{1,[c]_{\mathcal{G}_2}}$	$-\frac{\sqrt{6}}{12}$	$-\frac{\sqrt{6}}{12}$	$\frac{\sqrt{6}}{12}$	$\frac{\sqrt{6}}{12}$	$\frac{\sqrt{6}}{12}$	$-\frac{\sqrt{6}}{12}$
$\psi^{(1)}_{1,[f]_{\mathcal{G}_2}}$	$-\frac{\sqrt{2}}{12}$	$-\frac{\sqrt{6}}{12}$	$-\frac{\sqrt{2}}{12}$	$-\frac{\sqrt{2}}{12}$	$-\frac{\sqrt{2}}{12}$	$-\frac{5\sqrt{2}}{12}$

Table 2: Decimated framelets $\varphi_{1,[p]_{\mathcal{G}_1}}$ and $\psi^{(1)}_{1,[p]_{\mathcal{G}_2}}$ at level $j=1$

It can be verified that conditions in (37) and (38) hold for the above framelets constructed through the above steps (1)–(12). Hence, by Theorem 7 the framelet system for $J_1 = 0,1,2,3$,

$$\{\varphi_{J_1,[p]} : [p] \in V_{J_1}\} \cup \{\psi^{(n)}_{j,[p]} : [p] \in V_{j+1},\ j = J_1,\ldots,J\}$$

is a decimated tight frame for $L_2(\mathcal{G})$.

	a	b	c	d	e	f
$\varphi_{1,[a]_{\mathcal{G}_1}}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$
$\psi^{(1)}_{1,[a]_{\mathcal{G}_2}}$	$\frac{1}{3}$	$\frac{1}{3}$	$-\frac{1}{6}$	$-\frac{1}{6}$	$-\frac{1}{6}$	$-\frac{1}{6}$
$\psi^{(1)}_{1,[c]_{\mathcal{G}_2}}$	$-\frac{\sqrt{2}}{6}$	$-\frac{\sqrt{2}}{6}$	$\frac{\sqrt{2}}{12}$	$\frac{\sqrt{2}}{12}$	$\frac{\sqrt{2}}{12}$	$\frac{\sqrt{2}}{12}$

Table 3: Decimated framelets $\varphi_{0,[p]_{\mathcal{G}_0}}$ and $\psi^{(1)}_{0,[p]_{\mathcal{G}_1}}$ at level $j = 0$

8. Numerical Examples

In this section, we present three experiments for the F$\mathcal{G}$T algorithm. We show the computational complexity analysis of F$\mathcal{G}$T in Section 8.2 and the multiscale analysis by decimated framelets for the real-world traffic network in Section 8.3. Also, we use F$\mathcal{G}$T to define a spectral graph convolution in Section 8.4, which shows good performance in graph-level classification. The Python and Matlab codes for F$\mathcal{G}$T can be downloaded from Github[1].

8.1 Filter Bank for Decimated Framelets

We show examples of the filter bank for decimated framelets on a graph with different numbers of high passes. The constructed filter banks use the parameters in the example shown in Section 5.4. Figure 6 shows three types of filter banks, with the numbers of high passes 1, 2, or 3. For each case, we show two examples of filter bank with a slightly different choice of intersection points (between the curves of low-pass and high-pass filters). The explicit values for the parameters $(\zeta_c^a, \zeta_c^b, \zeta_c^{b(1)}$ and $\zeta_c^{b(2)}$ are given in the caption of the figure. From left to right, we show the filters from the finest level to the coarsest level. Note that the low-pass filter's support is strictly $[0, N_j]$ for the chain's jth level. The lower end of any high-pass filter's support is always positive, and the support is dependent on the chain structure.

8.2 Computational Complexity of F$\mathcal{G}$T on Random Graphs

In the first experiment, we show the computational cost of the Fast Fourier transforms (DFT [Algorithm 4] & ADFT [Algorithm 5]) and the fast framelet transforms (Decomposition [Algorithm 6] & Reconstruction [Algorithm 7] for F$\mathcal{G}$T). We demonstrate the computational complexities by timing the elapsed time for each of these algorithms on a set of randomly generated graphs with different sizes ranging from 500 nodes to 30,000 nodes. We simulate each adjacency matrix $A \sim \mathsf{U}_{(0,1)}$ with $A_{ij} = A_{ji}$ if $i \neq j$ and $A_{ij} = 0$ if $i = j$ for $i, j = 1, 2, \ldots, N$, where N denotes the number of nodes of the graph. This implies that the simulated graph does not contain any isolated node. We use the uninformative feature (random scalar) for each node of all the randomly generated graphs. We build the coarse-grained chains with 6 levels for the graphs larger than 2,500 nodes; 5 levels for the graphs with sizes between 1,000 and 2,500 nodes; 4 levels for the graphs smaller than 1,000 nodes. The choice of the clustering method and the resulting chain will have some impact on the performance of the GNNs with framelet convolution. Here, as an empirical study, we

---
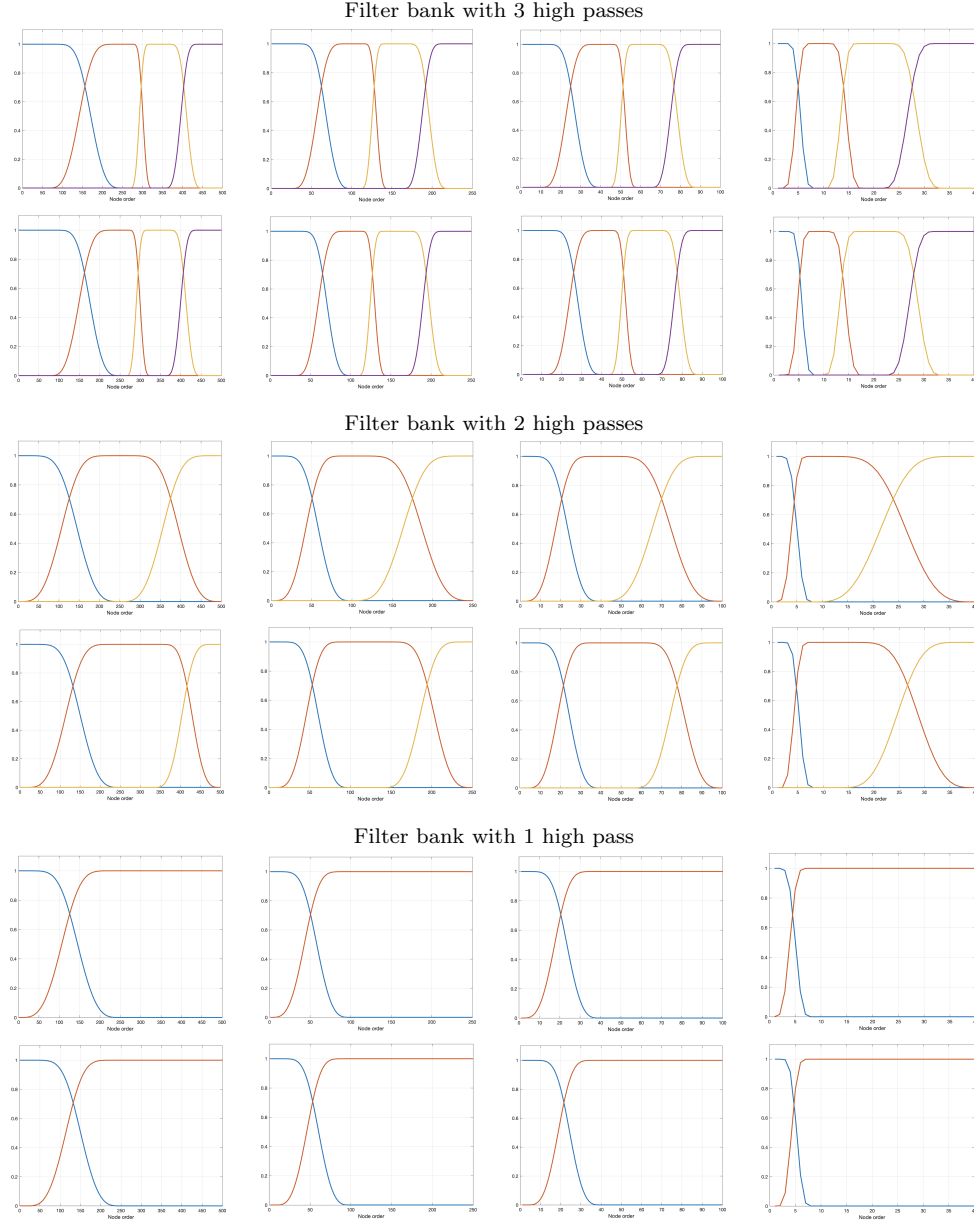
1. `https://github.com/YuGuangWang/FGT`

Figure 6: Filter banks with different number of high passes and different intersection points. For each filter bank case, from left to right are the pictures for the filter banks for the chain level from finest to the second last coarsest. For 3-high pass case, the top and bottom use the intersection point parameters $\zeta_a = 0.25, \zeta_c^{b^{(1)}} = 0.25, \zeta_c^{b^{(2)}} = 0.25$ and $\zeta_a = 0.3, \zeta_c^{b^{(1)}} = 0.2, \zeta_c^{b^{(2)}} = 0.3$ respectively. For the 2-high pass case, the top and bottom use the parameters $\zeta_c^a = 0.25, \zeta_c^b = 0.25$ and $\zeta_a = 0.3, \zeta_c^b = 0.2$. For 1-high pass case, the top and bottom use the parameters $\zeta_c^a = 0.25$ and $\zeta_c^a = 0.3$. The graphs of the chain have 500, 250, 100, 40 and 8 from finest to coarsest. Note that we do not need a filter bank for the coarsest level.

drop roughly 60% of the nodes in every coarsening. All the coarse-grained chains used in Sections 8.2 & 8.3 are generated by METIS (Karypis and Kumar, 1999), which is a computationally efficient graph partitioning method. Algorithm 3 is applied to construct the Haar-like orthonormal basis for each coarse-grained chain for DFTs and F$\mathcal{G}$Ts.

The computing environment of this experiment is MATLAB® R2019b installed on a macOS Catalina machine with 2.3GHz 8-Core Intel Core i9 processor and 16GB RAM. Figure 10 shows for both DFTs and F$\mathcal{G}$Ts, the computational time is approximately proportional to N. This observation verifies our theoretical analysis in Section 6 that F$\mathcal{G}$Ts have a computational cost $\mathcal{O}(N)$ for the graph with size N.

8.3 Decimated Framelets on Road Network

In this section, we present an application of multiscale analysis by F$\mathcal{G}$Ts for a real-world traffic network of Minnesota (Rossi and Ahmed, 2015). The dataset represents the roads of Minnesota by edges of a graph and the intersections and towns by $2,642$ graph vertices. In this experiment, the graph $\mathcal{G}$ is unweighted, which means all the edge weights are equal to unity regardless of the length of the road segment. The spatial coordinates of each node are only used for visualization purpose but do not affect the input graph data of F$\mathcal{G}$Ts. We use the uninformative feature (constant scalar) for the nodes of the graph. The road network is visualized in Figure 7(a).

We first employ Algorithm 6 for F$\mathcal{G}$T to decompose the graph signal into a set of framelet detail and approximation coefficients and visualize the significance and distribution of these coefficients (here the values are in the spectral domain) on the road network. The framelets use the filter banks, each of which has one low pass and two high passes. At the jth level, the lengths of the coefficient vectors for the low-pass and high-pass are equal to the number of nodes for the jth and $(j+1)$th levels of the chain respectively.

We construct a four-level coarse-grained chain $\mathcal{G}_{3\to 0} := (\mathcal{G}_3, \mathcal{G}_2, \mathcal{G}_1, \mathcal{G}_0)$ with $\mathcal{G}_3 = \mathcal{G}$, $|V_2| = 1,000$, $|V_1| = 300$ and $|V_0| = 100$ by using the same graph partitioning method as above. Algorithm 3 is applied to calculate the Haar-like orthonormal basis and the computing environment is same as the previous experiment. Figure 7 visualizes the experimental results.

Remark 28 *In practice, the numerical approximation error of eigendecomposition for the graph Laplacian is dependent on the structure and sparsity of the graph, the clustering for the chain and the size of the graph.*

The framelet approximation coefficients $\widehat{v}_0$ correspond to the low passes of the filter banks, which capture the global information of the input graph. In Figure 7(b), we observe that the elements of $\widehat{v}_0$ with more significant values are located over both high-density area (lower part of the road network) and low-density area (upper part of the road network). Thus, the framelet approximation coefficients $\widehat{v}_0$ can represent the global structure of the input graph data. On the other hand, the framelet detail coefficients $\{\widehat{w}_j^n : n = 1, \ldots, r_j, j = J_0, \ldots, J-1\}$ are related to the high passes of the filter banks, which capture the information from the local regions of the input graph. From Figures 7(c)–(f), we can see that the framelet detail coefficients with more significant values are mostly spread over the specific subregions. For example, the coefficients are concentrated in the lower left and upper right parts of the
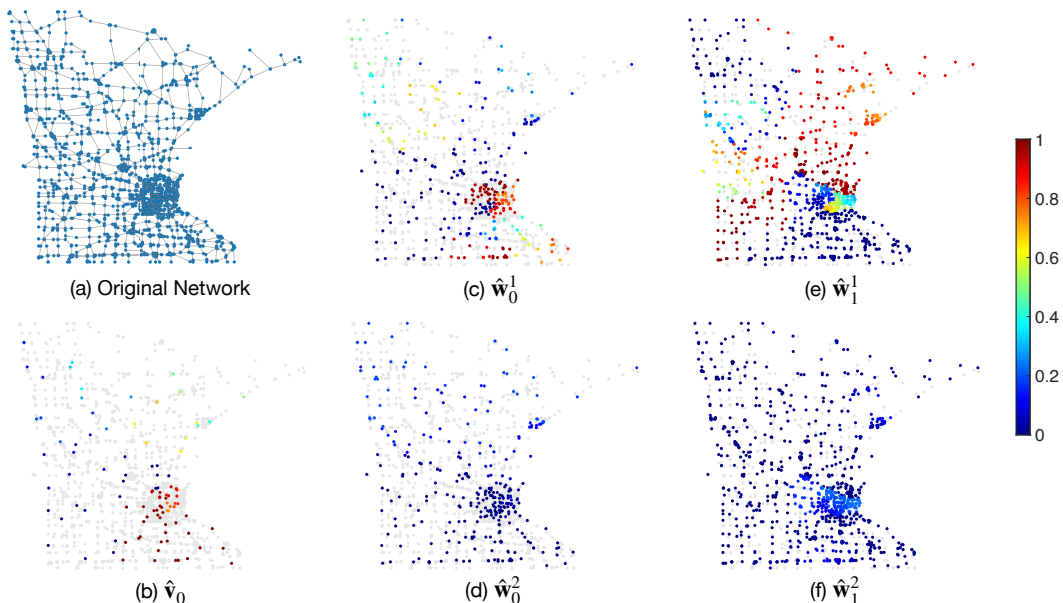
Figure 7: (a) Graph of Minnesota road network $\mathcal{G}_3$. (b) Framelet approximation coefficients $\widehat{v}_0$ on $\mathcal{G}_0$ with 100 clusters. (c) Framelet detail coefficients $\widehat{w}_0^1$ on $\mathcal{G}_1$ with 300 clusters. (d) Framelet detail coefficients $\widehat{w}_0^2$ on $\mathcal{G}_1$ with 300 clusters. (e) Framelet detail coefficients $\widehat{w}_1^1$ on $\mathcal{G}_2$ with 1,000 clusters. (f) Framelet detail coefficients $\widehat{w}_1^2$ on $\mathcal{G}_2$ with 1,000 clusters. For subfigures (b)–(f), the dots in pale grey represent the background (which is the original graph) which outlines the geographical structure of the road network and exhibits the spatial and distributional information of the framelet coefficients (colored dots). Each colored dot corresponds to the first node of the cluster $[p]_{\mathcal{G}_i}$, for $p \in V_i$ and $i = 0, 1, 2$, with different colors depicting the values of the framelet coefficients. Note that all the displayed framelet coefficients are in the spectral domain.

road network in Figure 7(e), and the lower right and upper left parts of the network in Figure 7(c). Note that the subregions with more significant framelet detail coefficients in Figures 7(c) and (e) are complementary, which reflects different parts of details the two high-passes captured from the graph signal. The second sets of the framelet detail coefficients $\widehat{w}_0^2$ and $\widehat{w}_1^2$ visualized in Figures 7(d) and (f) represent even more detailed information of their first high-passes $\widehat{w}_0^1$ and $\widehat{w}_1^1$. Hence, these framelet detail coefficients enable to capture the detail information from all the subregions of the input graph $\mathcal{G}$.

8.4 Graph Classification with F$\mathcal{G}$T-based Graph Convolution

Graph-level classification task relies on the graph representation learning which has a broad range of real-world applications, such as social network analysis (Hamilton et al., 2017; Veličković et al., 2018) and molecule classification (Duvenaud et al., 2015; Gilmer et al., 2017). Graph convolution is one of the key components in a graph convolutional neural network, which has been proved effective and powerful for learning a graph representation (Ying et al., 2018; Ma et al., 2019; Wang et al., 2020). In this section, we explore the feasibility of using F$\mathcal{G}$T with Haar-like orthonormal basis (Algorithm 3) to define a spectral graph convolution layer based on the classic work (Bruna et al., 2014) which is the

first attempt at implementing CNNs on graphs in the spectral domain with graph Fourier transforms.

In $\mathbb{R}^d$, convolution induced by Fourier transforms or wavelet transforms are well known, see for example, Stein and Shakarchi (2011); Mallat (2009). For $g, f \in l_2(\mathcal{G})$, we define **Framelet Graph Convolution** (FGConv) as

$$g \star f = \mathbf{V}\big((\mathbf{W}g) \odot (\mathbf{W}f)\big), \tag{79}$$

where g is a trainable filter; f denotes the graph data with one feature on each node; the symbol $\odot$ is the Hadamard product; and the operators $\mathbf{V}$ and $\mathbf{W}$ are the framelet decomposition (Algorithm 6) and reconstruction (Algorithm 7) of F$\mathcal{G}$T, respectively. By presuming that the trainable filter g lies in the spectral domain and has an identical shape as $\mathbf{W}f$, the convolution (79) can be simplified as

$$g \star f = \mathbf{V}\big(g \odot (\mathbf{W}f)\big).$$

Here we use $\widehat{\mathrm{v}}_j$ and $\widehat{\mathrm{w}}_j^n$ in Algorithms 6 and 7 to simplify computation. Similar graph wavelet convolution was developed by Xu et al. (2019a), where the wavelets Hammond et al. (2011) are undecimated framelets which do not include downsampling and upsampling processes.

Computationally, the FGConv is performed by first decomposing the graph signal f into several sets of framelet detail and approximation coefficients, then taking the Hadamard product between the trainable filter g and the coefficients $\mathbf{W}f$ in the spectral domain, and finally reconstructing the output graph data from the processed framelet coefficients $g \odot (\mathbf{W}f)$. By applying the weight detaching trick (Xu et al., 2019a; Li et al., 2020; Zheng et al., 2020), the FGConv with multiple input features d and reduced parameter complexity reads

$$\mathcal{F}^{\mathrm{out}} = \sigma\big(\mathbf{V}(G(\mathbf{W}(\mathcal{F}^{\mathrm{in}}\mathcal{W})))\big), \tag{80}$$

where $\mathcal{F}^{\mathrm{in}} \in \mathbb{R}^{|V| \times d^{\mathrm{in}}}$ and $\mathcal{F}^{\mathrm{out}} \in \mathbb{R}^{|V| \times d^{\mathrm{out}}}$ denote the input and embedded feature matrices of the graph with the indicated shapes; $\mathcal{W} \in \mathbb{R}^{d^{\mathrm{in}} \times d^{\mathrm{out}}}$ is a trainable weight matrix for affine transformation; and σ is the activation function (e.g., ReLU). For an example graph $\mathcal{G}$ with feature matrix $\mathcal{F}^{\mathrm{in}}$ and coarse-grained chain $\mathcal{G}_{3 \to 0} := (\mathcal{G}_3, \mathcal{G}_2, \mathcal{G}_1, \mathcal{G}_0)$ where $\mathcal{G}_3 = \mathcal{G}$, we concatenate all the framelet detail and approximation coefficients from $\mathbf{W}(\mathcal{F}^{\mathrm{in}}\mathcal{W})$ into a matrix with shape $K \times d^{\mathrm{out}}$ for the ease of computation, where $K = |V_0| + r \sum_{i=1}^{3} |V_i|$ with r representing the number of high-pass filters. Hence, the matrix G in (80) denotes a trainable diagonal matrix with shape $K \times K$ and the parameter complexity of FGConv in (80) is $\mathcal{O}(K + d^{\mathrm{in}} \times d^{\mathrm{out}})$.

Datasets We evaluate the performance of the proposed FGConv on three graph classification benchmarks. The selected datasets are described as follows. **D&D** (Dobson and Doig, 2003; Shervashidze et al., 2011) is a graph dataset consists of 1,113 protein structures, each of which is represented by a graph whose nodes are amino acids and there is an edge if two nodes are less than six angstroms apart. The node features of each graph are formed by the binary encoding of some chemical properties. The task of using this dataset is a binary classification problem, and we aim to classify each protein structure into either enzymes or non-enzymes. **PROTEINS** (Dobson and Doig, 2003; Borgwardt et al., 2005) is another protein structure dataset with the same task, which is treated as a simplified version of

Table 4: Statistical information of the datasets used for graph classification

Datasets	PROTEINS	MUTAG	D&D
Max. #Nodes	620	28	5,748
Min. #Nodes	4	10	30
Avg. #Nodes	39.06	17.93	284.32
Avg. #Edges	72.82	19.79	715.66
#Graphs	1,113	188	1,178
#Classes	2	2	2

D&D in terms of graph size. **MUTAG** (Debnath et al., 1991; Kriege and Mutzel, 2012) is a mutagen dataset which contains 188 chemical compounds. We use graphs to depict the compounds, where the nodes and edges of each graph correspond to the atoms and covalent bonds of each compound, respectively. The task is to predict whether the compounds in the dataset are mutagenic or not. Important statistical information of these datasets is provided in Table 4.

Table 5: Mean test accuracy (in percentage) and standard deviation of FGConv-Sum as compared with existing methods on the benchmark graph classification datasets, over 10 repetitions.

Methods	PROTEINS	MUTAG	D&D
SP	75.07*	85.79*	–
Graphlet	71.67*	81.58*	78.45*
RW	74.22*	83.68*	–
WL	72.92*	80.72*	77.95*
GIN	76.2	**89.4**	–
PatchySan	75.00	91.58	76.27
DGCNN	75.54	85.83	79.37
DiffPool	76.25	–	**80.64**
SAGPool	72.17	–	77.07
EigenPool	**76.6**	–	78.6
g-U-Nets	77.68	–	82.43
FGConv-Sum	**78.3±2.26**	**90.8±2.50**	**82.9±2.55**

'∗' denotes the record retrieved from Niepert et al. (2016).

'–' means that there is no public record for the method on the dataset.

! The records without superscription are retrieved from their corresponding original papers.

! The decimal place is not modified when transferring the results.

! The top three scores are highlighted as: **First**, **Second**, and **Third**.

Model architecture In this experiment, we propose to employ the following network architecture for composing our graph neural network (GNN) FGConv-Sum

$$\text{FGConv} - \text{FGConv} - \text{SumPool} - \text{MLP}.$$

Specifically, we use two FGConv layers followed by a sum pooling to generate a unified vectorial graph representation which is then sent to MLP for classification. Batch normalization (Ioffe and Szegedy, 2015) is employed after each layer of MLP, except for the output layer. We implement a three-layer MLP for **PROTEINS** and **MUTAG**, and a two-layer MLP for **D&D** in order to achieve a better performance on each dataset. Figure 8 depicts the architecture of our composed model FGConv-Sum.

Figure 8: Network architecture of FGConv-Sum. Blocks $g_1, \ldots, g_\zeta$ are concatenated to form the diagonal of the filter matrix G, and $\hat{y}$ denotes the predicted label for the input graph $\mathcal{G}$. All the framelet coefficients are in the spectral domain in order to perform the Hadamard product with the trainable filters $g = [g_0, g_1, \ldots, g_\zeta]$.

Baselines To evaluate the effectiveness of FGConv on the three benchmark graph classification datasets, we include the following representative GNN methods as our baselines. GIN (Xu et al., 2019b), PatchySan (Niepert et al., 2016), DGCNN (Zhang et al., 2018), DiffPool (Ying et al., 2018), SAGPool (Lee et al., 2019), EigenPool (Ma et al., 2019), g-U-Nets (Gao and Ji, 2019). Additionally, we also consider several graph kernel methods for comparison, including Shortest-Path kernel (SP) (Borgwardt and Kriegel, 2005), Graphlet Count kernel (Shervashidze et al., 2009), Random Walk kernel (RW) (Gärtner et al., 2003) and Weisfeiler-Lehman subtree kernel (WL) (Shervashidze et al., 2011).

Table 6: Grid search space for the hyperparameters.

Hyperparameters	Choices
Learning Rate	1e-4, 5e-4, 1e-3, 5e-3, 1e-2
Hidden Size	16, 32, 64, 128
Weight Decay (L2)	1e-4, 5e-4, 1e-3, 5e-3
Batch Size	32, 64, 128, 256

Training scheme We employ spectral clustering (Shi and Malik, 2000; Stella and Shi, 2003) to construct a two-level coarse-grained chain for each graph in the dataset. Spectral

clustering has been proved capable of clustering various data patterns and can handle the graph with isolated nodes. The number of parents/clusters in the coarsened level is set to the half of that in its finer level. We split the dataset into training, validation and test sets with portions 80%, 10% and 10% respectively. Since different data splits might have a great impact on the performance of a GNN model Shchur et al. (2018), we repeat the experiment on each dataset 10 times with random shuffling for the dataset before splitting. We report the mean test accuracy along with the standard deviation of our model FGConv-Sum for each dataset. We use the Adam optimizer (Kingma and Ba, 2015) with an early stopping strategy suggested in Shchur et al. (2018) to train our model. Specifically, we stop the training if the validation loss does not improve for continuous 10 epochs with a maximum of 50 epochs. We use a simple grid search to tune the hyperparameters. We show a list of the hyperparameters in the model along with their search spaces in Table 6. All the programs used in this section are written in PyTorch and the library PyTorch Geometric (Fey and Lenssen, 2019), and we run the experiments on NVIDIA® Tesla V100 GPU with 5,120 CUDA cores and 16GB HBM2 mounted on a high performance computing cluster.

(a) PROTEINS   (b) MUTAG   (c) D&D

Figure 9: Plots of losses against epoch from one repetition on the three benchmark graph classification datasets. The early stopping chechpoint corresponds to the epoch where the model is saved for evaluation on the test set, i.e. when the smallest validation loss was achieved.

Results The experimental results are reported in Table 5. We can observe that our FGConv-Sum achieves comparable performances with the state-of-the-art baseline methods on all the datasets. Particularly, our model obtains the top mean test accuracy on **PROTEINS** and **D&D**, while the performance of our FGConv-Sum is still ranked top-three among all the baseline methods on **MUTAG**. We visualize the convergence of training and validation losses of FGConv-Sum for one running on each dataset in Figure 9. Note that the fluctuations of the loss values near the end of the training are due to the use of moderate learning rate and the absence of learning rate scheduler. However, an obvious convergence demonstrated by each sub-figure further shows evidence of the feasibility and effectiveness of using F$\mathcal{G}$T in a spectral graph convolution.

9. Discussion

We construct a decimated tight framelet system on a graph by filtered spectral expansion with well-designed scaling functions. The scaling function gives a set of the filter bank. A coarse-grained chain for the graph is generated to achieve decimation of the framelet system.
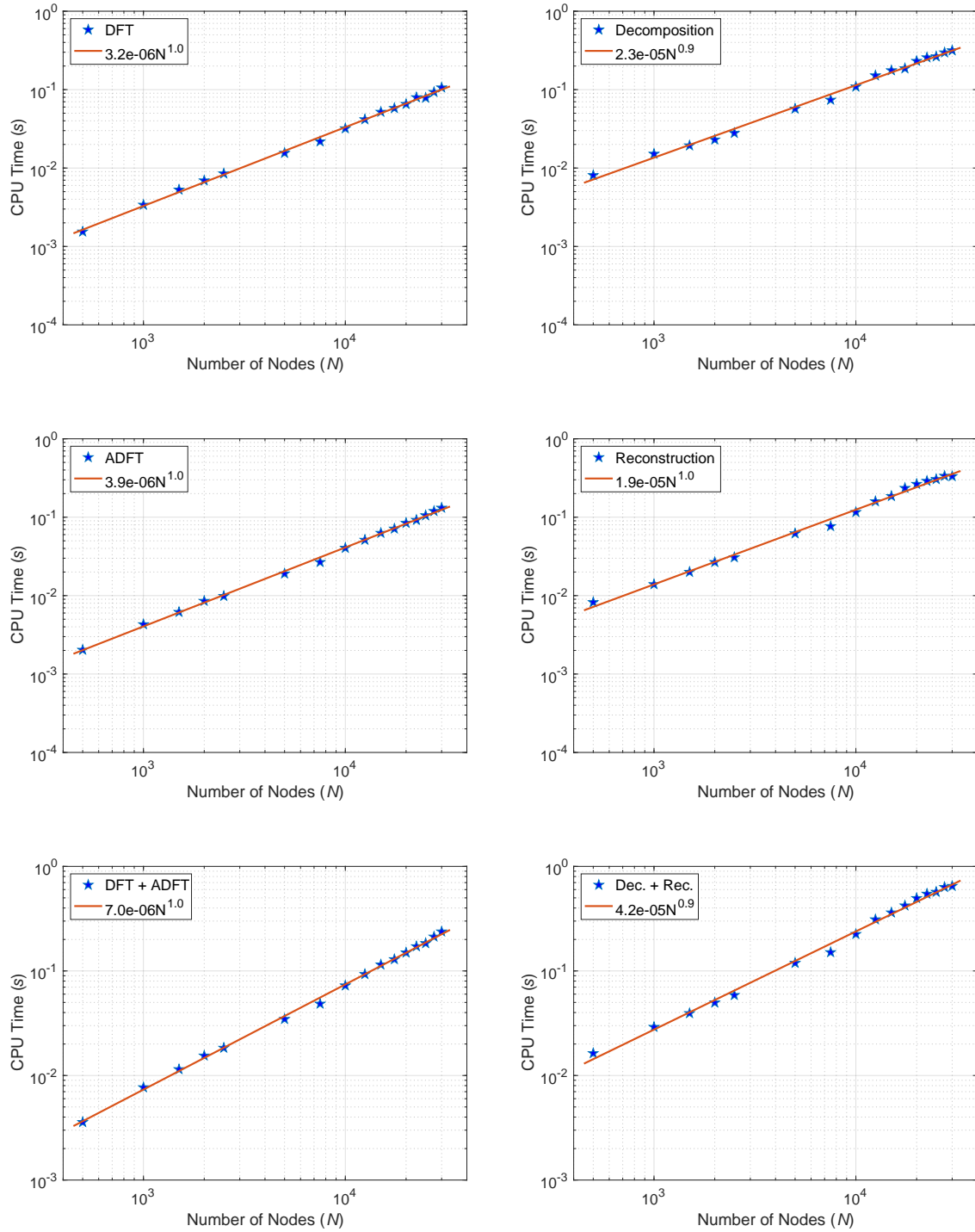
Figure 10: CPU times (in seconds) of DFTs and F$\mathcal{G}$Ts elapsed on 16 random graphs with different sizes ranging from 500 nodes to 30,000 nodes. The least-squares method is used to estimate the order of computational time (in a power function of the form $y = k_1 N^{k_2}$). We visualize the fitted curves with red lines and report the estimated order in the legends.

The associated filter bank allows fast implementation of the decimated $\mathcal{G}$-framelet transforms. The linear computational complexity depends on the speed of the discrete Fourier transforms for the orthogonal basis. To this end, the orthogonal basis is generated based on the chain. When the chain has an appropriate structure, the resulting orthogonal basis has fast evaluation. It distinguishes our method and construction from existing framelet or wavelet methods that are undecimated Hammond et al. (2011); Dong (2017).

Choice of chain By Theorem 7, the choice of chain and its clustering algorithm does not affect the tightness of the decimated framelet system. Nevertheless, the chain structure is highly related to the computation for the decimated $\mathcal{G}$-transforms. For instance, Proposition 12 provides a sufficient condition of the chain for the sparsity of the Haar global orthonormal basis matrix. Theorem 16 and Proposition 27 then guarantee the linear computational complexity of the discrete Fourier transforms and $\mathcal{G}$-framelet transforms under the Haar basis. On the other hand, the subgraphs of the chain carry the clustering information of the original graph. The chain-based orthonormal basis and the subsequent decimated framelet system then have a clustering feature of the graph. Thus, besides the neighbour information, the chain-based graph Laplacian basis also has the clustering property of the graph embedded in the system. As the Haar basis is solely constructed based on the chain, it mainly reflects the clustering property rather than the neighbour linking. For graph convolution based on our decimated $\mathcal{G}$-framelet transforms, the Haar version is more suitable for graph-level property prediction tasks (graph classification and graph regression) while the graph Laplacian version could also work for node property prediction tasks.

Number of high passes The number of high passes does not affect the tightness of the system. However, the representation by two framelet systems with different numbers of high passes contains a distinct extent of details in the framelet domain. Empirically, framelets with more high passes have a better signal-to-noise ratio (Wang and Zhuang, 2018).

Application There are many potential application of the F$\mathcal{G}$T. For example, it can be used to accelerate the diffusion-based Gaussian process when the inversion of the Gaussian covariance matrix based on graph Laplacian needs efficient evaluation (Dunson et al., 2020). F$\mathcal{G}$T can also be used in graph pooling (Wang et al., 2020) where we can filter out the high pass signal for pooling operation in GNN models. When filtering out only the framelet details while keeping the framelet approximation in the pooling output, the main information of the graph signal is preserved while the graph size is compressed.

Appendix A. Table of Notations

Symbol	Meaning
$\mathbb{R}^d$	d-dimensional real coordinate space
$\mathcal{G}$	An undirected connected graph with a non-empty finite vertex set V and edge set E
Λ_j	The index sets of the translation points and centers on the domain $\mathcal{D}$
$|V|$ or N	The number of vertices
p or v	An arbitrary vertex of V
$\boldsymbol{w}$	A non-negative weight function that project $E \to \mathbb{R}$
$\mathrm{vol}(\mathcal{G})$	The volume of the graph
$\boldsymbol{K}(p,v)$	The transition kernel or a Markov random walk on $\mathcal{G}$
$\boldsymbol{\rho}(p,v)$	The distance between two vertices p and v
$\mathcal{G}_c$	A coarse-grained graph of $\mathcal{G}$, where V_c is a partition of V
$[v]_{\mathcal{G}_c}$	An equivalent class (cluster) where a vertex in $\mathcal{G}_c$ is associated with a vertex $v \in V$
$\|[v]\|$ or $\#[v]$	The number of vertices in the cluster $[v]$
$\mathcal{G}_{J \to J_0}$	A coarse-grained chain of $\mathcal{G}$ constituted by a sequence of graphs $\mathcal{G}_J, \mathcal{G}_{J-1}, \dots, \mathcal{G}_{J_0}$
$\mathcal{G}_j$	The level-j graph of the chain $\mathcal{G}_{J \to J_0}$. The $\mathcal{G}_{j-1}$ is the coarse-grained graph of $\mathcal{G}_j$
$\mathcal{T}$	A tree of the coarse-grained chain $(\mathcal{G}_J, \dots, \mathcal{G}_{J_0})$ of $\mathcal{G}$. The $\{V\}$ is the root of $\mathcal{T}$ at the top level J_0, and all vertices of $\mathcal{G}$ are the leaves at the bottom level J.
$l_2(\mathcal{G})$	The Hilbert space of vectors $\boldsymbol{f} : V \to \mathbb{C}$ on $\mathcal{G}$
$\{\boldsymbol{u}_\ell\}_{\ell=1}^N$	An orthonormal basis for $l_2(\mathcal{G})$, usually defined as an eigenvector set with eigenvalues λ_ℓ of $\mathcal{L}$
$\overline{\boldsymbol{g}}$	The complex conjugate to $\boldsymbol{g} \in l_2(\mathcal{G})$
$\|\cdot\|_{\mathcal{G}}$	The induced norm, which is defined as the square root of an inner product on the vectors.
$\delta_{\ell,\ell'}$	The Kronecker delta satisfying $\delta_{\ell,\ell'} = 1$ if $\ell = \ell'$ and $\delta_{\ell,\ell'} = 0$ if $\ell \neq \ell'$
$\widehat{\boldsymbol{f}}_\ell$	The generalized Fourier coefficient of degree ℓ for $\boldsymbol{f} \in l_2(\mathcal{G})$ with respect to $\boldsymbol{u}_\ell$
$\boldsymbol{f}$	The input vector on the time domain
$\mathcal{L}$	The combinatorial or unnormalized graph Laplacian operator
$\{\boldsymbol{g}_\ell\}_{\ell=1}^M$	A frame for $l_2(\mathcal{G})$ if satisfying (2)
L	Lebesgue measure
$\widehat{\gamma}$	The Fourier transform of a function $\gamma \in L_1(\mathbb{R})$
h	A filter (or mask) of a complex-valued sequence $\{h_k\}_{k \in \mathbb{Z}} \subseteq \mathbb{C}$
Ψ	A set of framelet generators (or scaling functions) $\{\alpha; \beta^{(1)}, \dots, \beta^{(r)}\}$ in $L_1(\mathbb{R})$
Ψ_j	A set of framelet generators (or scaling functions) $\{\alpha_j; \beta_j^{(1)}, \dots, \beta_j^{(r)}\}$ in $L_2(\mathbb{R})$ at scale j, see (23)
$\boldsymbol{\eta}$	A filter bank with a set of filters $\{a; b^{(1)}, \dots, b^{(r)}\}$

$\boldsymbol{\eta}_j$	A filter bank at level j connecting Ψ_j and Ψ_{j-1}
α	Scaling function associated with low pass for undecimated framelets
$\beta^{(n)}$	Scaling function associated with nth high pass for undecimated framelets
α_j	Scaling function associated with low pass at level j for decimated framelets
$\beta_j^{(n)}$	Scaling function associated with nth high pass at level j for decimated framelets
$a^\star$	The low-pass filter or refinement mask at decomposition
a	The low-pass filter or refinement mask at reconstruction
$(b^{(n)})^\star$	The high-pass filters or framelet masks at decomposition
$b^{(n)}$	The high-pass filters or framelet masks at reconstruction
ξ	A signal on the time domain
$\mathsf{AS}_J(\Psi)$	A stationary non-homogeneous affine system of the form (5)
$\varphi_{j,p}$	Framelets at scale j and at a vertex $p \in V$ for low-pass signals
$\psi_{j,p}^{(n)}$	The n-th framelets at scale j for high-pass signals
$\mathsf{UFS}(\Psi, \boldsymbol{\eta}; \mathcal{G})$ or $\mathsf{UFS}_{J_1}^J(\Psi, \boldsymbol{\eta})$	An undecimated tight framelet system for $l_2(\mathcal{G})$
$\mathsf{DFS}(\{\Psi_j\}_{j=J_1}^J, \{\boldsymbol{\eta}_j\}_{j=J_1+1}^J)$	The decimated framelet system defined as (25)
$\omega_{j,[p]}$	The associated weight of vertex $[p]$ at level j
$\mathcal{Q}_j$	A set of weights $\{\omega_{j,[p]} : [p] \in V_j\}$ on $\mathcal{G}_j$
$\mathcal{U}_{\ell,\ell'}(\mathcal{Q}_j)$	The weighted sum of the product of $\boldsymbol{u}_\ell$, $\boldsymbol{u}_{\ell'}$ by (26)
$c_{[v]_{\mathcal{G}_j}}$	The number of vertices in $\mathcal{G}_{j+1}$ which are in the cluster $[v]_{\mathcal{G}_j}$
$\mathrm{spoc}(\boldsymbol{u}_\ell)$	Spoc of a vector $\boldsymbol{u}_\ell$
$\mathrm{supp}(\boldsymbol{u}_\ell)$	Support of a vector $\boldsymbol{u}_\ell$
χ_j^c	The characteristic function for the jth vertices on $\mathcal{G}_c$
$\mathbf{v}_{J_0}$	The framelet approximation coefficients
$\mathbf{w}_j^n$	The framelet detail coefficients for $n = 1, \ldots, r_j, \ j = J_0, \ldots, J$
$\boldsymbol{c}$ or $\widehat{\mathbf{v}}$	The discrete Fourier coefficients of the vector $\mathbf{v}$
$\mathbf{F}_j^*$	The discrete Fourier transform (DFT) operator $l_2(V_j) \to l_2(\Omega_j)$ on $\mathcal{G}_j$
$\mathbf{F}_j$	The adjoint discrete Fourier transform (ADFT) operator $l_2(\Lambda_j) \to l_2(V_j)$ on $\mathcal{G}_j$
$*_j$	The discrete convolution operator
$\downarrow_j$	The downsampling operator, defined in (68), from level j to $j-1$
$\uparrow_j$	The upsampling operator, defined in (69), from level $j-1$ to level j
$\mathbf{W}$	The multi-level $\mathcal{G}$-framelet analysis operator
$\mathbf{V}$	The multi-level $\mathcal{G}$-framelet synthesis operator

References

Charu C Aggarwal and Haixun Wang. A survey of clustering algorithms for graph data. In *Managing and Mining Graph Data*, pages 275–301. Springer, 2010.

N. Atreas, N. Karantzas, M. Papadakis, and T. Stavropoulos. On the design of multi-dimensional compactly supported parseval framelets with directional characteristics. *Linear Algebra and its Applications*, 582:1 – 36, 2019. URL `http://www.sciencedirect.com/science/article/pii/S0024379519303155`.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, June 2003.

Mikhail Belkin and Partha Niyogi. Towards a theoretical foundation for laplacian-based manifold methods. *Journal of Computer and System Sciences*, 74(8):1289 – 1308, 2008. URL `http://www.sciencedirect.com/science/article/pii/S0022000007001274`.

Mikhail Belkin, Irina Matveeva, and Partha Niyogi. Regularization and semi-supervised learning on large graphs. In John Shawe-Taylor and Yoram Singer, editors, *Learning Theory*, pages 624–638, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

Rianne van den Berg, Thomas N Kipf, and Max Welling. Graph convolutional matrix completion. In *KDD*, 2018.

Andrea L Bertozzi and Arjuna Flenner. Diffuse interface models on graphs for classification of high dimensional data. *Multiscale Modeling & Simulation*, 10(3):1090–1118, 2012.

Cristian Bodnar, Fabrizio Frasca, Nina Otter, Yu Guang Wang, Pietro Liò, Guido Montufar, and Michael M. Bronstein. Weisfeiler and Lehman go cellular: CW networks. In *NeurIPS*, 2021a. URL `https://openreview.net/forum?id=uVPZCMVtsSG`.

Cristian Bodnar, Fabrizio Frasca, Yu Guang Wang, Nina Otter, Guido Montúfar, Pietro Lio, and Michael Bronstein. Weisfeiler and Lehman go topological: Message passing simplicial networks. 2021b.

Karsten M Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In *ICDM*, pages 74–81, 2005.

Karsten M Borgwardt, Cheng Soon Ong, Stefan Schönauer, SVN Vishwanathan, Alex J Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl_1):i47–i56, 2005.

Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11): 1222–1239, 2001.

J. S. Brauchart, J. Dick, E. B. Saff, I. H. Sloan, Y. G. Wang, and R. S. Womersley. Covering of spheres by spherical caps and worst-case error for equal weight cubature in Sobolev spaces. *Journal of Mathematical Analysis and Applications*, 431(2):782–811, 2015.

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *ICLR (CBLS)*, 2014.

Emmanuel Candès, Laurent Demanet, David Donoho, and Lexing Ying. Fast discrete curvelet transforms. *Multiscale Modeling & Simulation*, 5(3):861–899, 2006.

Emmanuel J Candès and Carlos Fernandez-Granda. Towards a mathematical theory of super-resolution. *Communications on Pure and Applied Mathematics*, 67(6):906–956, 2014.

Kamalika Chaudhuri and Sanjoy Dasgupta. Rates of convergence for the cluster tree. In *NIPS*, pages 343–351, 2010.

Z. Che and X. Zhuang. Digital affine shear filter banks with 2-layer structure and their applications in image processing. *IEEE Transactions on Image Processing*, 27(8):3931–3941, Aug 2018.

Xu Chen, Xiuyuan Cheng, and Stéphane Mallat. Unsupervised deep Haar scattering on graphs. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *NIPS*, volume 27, 2014.

Gene Cheung, Enrico Magli, Yuichi Tanaka, and Michael K Ng. Graph spectral image processing. *Proceedings of the IEEE*, 106(5):907–930, 2018.

Mark Cheung, John Shi, Oren Wright, Lavendar Y Jiang, Xujin Liu, and José MF Moura. Graph signal processing and deep learning: Convolution, pooling, and topology. *IEEE Signal Processing Magazine*, 37(6):139–149, 2020.

Ole Christensen and Say Song Goh. The unitary extension principle on locally compact abelian groups. *Applied and Computational Harmonic Analysis*, 47(1):1 – 29, 2019. URL http://www.sciencedirect.com/science/article/pii/S1063520317300751.

C. K. Chui, F. Filbir, and H. N. Mhaskar. Representation of functions on big data: graphs and trees. *Applied and Computational Harmonic Analysis*, 38(3):489–509, 2015.

Charles K. Chui. *An introduction to wavelets*, volume 1 of *Wavelet Analysis and its Applications*. Academic Press, Inc., Boston, MA, 1992.

Charles K Chui, Xianliang Shi, and Joachim Stöckler. Affine frames, quasi-affine frames, and their duals. *Advances in Computational Mathematics*, 8(1-2):1–17, 1998.

Charles K. Chui, Wenjie He, and Joachim Stöckler. Compactly supported tight and sibling frames with maximum vanishing moments. *Applied and Computational Harmonic Analysis*, 13(3):224–262, 2002. URL `http://dx.doi.org.ezproxy.cityu.edu.hk/10.1016/S1063-5203(02)00510-9`.

Charles K Chui, HN Mhaskar, and Xiaosheng Zhuang. Representation of functions on big data associated with directed graphs. *Applied and Computational Harmonic Analysis*, 44 (1):165–188, 2018.

Fan R. K. Chung. *Spectral graph theory*, volume 92 of *CBMS Regional Conference Series in Mathematics*. American Mathematical Society, Providence, RI, 1997.

Ronald R. Coifman and Mauro Maggioni. Diffusion wavelets. *Applied and Computational Harmonic Analysis*, 21(1):53–94, 2006. URL `http://dx.doi.org.ezproxy.cityu.edu.hk/10.1016/j.acha.2006.04.004`.

Ronald R Coifman, Stephane Lafon, Ann B Lee, Mauro Maggioni, Boaz Nadler, Frederick Warner, and Steven W Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences*, 102(21):7426–7431, 2005.

Mark Crovella and Eric Kolaczyk. Graph wavelets for spatial traffic analysis. In *Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1848–1857, 2003.

Ingrid Daubechies. *Ten lectures on wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.

Ingrid Daubechies, Bin Han, Amos Ron, and Zuowei Shen. Framelets: MRA-based constructions of wavelet frames. *Applied and Computational Harmonic Analysis*, 14(1):1 – 46, 2003.

Asim Kumar Debnath, Rosa L Lopez de Compadre, Gargi Debnath, Alan J Shusterman, and Corwin Hansch. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of Medicinal Chemistry*, 34(2):786–797, 1991.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *NIPS*, pages 3844–3852, 2016.

Xin Deng, Ren Yang, Mai Xu, and Pier Luigi Dragotti. Wavelet domain style transfer for an effective perception-distortion tradeoff in single image super-resolution. In *ICCV*, October 2019.

Paul D Dobson and Andrew J Doig. Distinguishing enzyme structures from non-enzymes without alignments. *Journal of Molecular Biology*, 330(4):771–783, 2003.

Bin Dong. Sparse representation on graphs by tight wavelet frames and applications. *Applied and Computational Harmonic Analysis*, 42(3):452–479, 2017.

Stijn Dongen. A cluster algorithm for graphs. Technical report, Amsterdam, 2000.

David B Dunson, Hau-Tieng Wu, and Nan Wu. Graph based Gaussian processes on restricted domains. *arXiv preprint arXiv:2010.07242*, 2020.

David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *NIPS*, pages 2224–2232, 2015.

Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard, and Stefano Soatto. Empirical study of the topology and geometry of deep networks. In *CVPR*, pages 3762–3770, 2018.

Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. A survey of kernel and spectral methods for clustering. *Pattern Recognition*, 41(1):176–190, 2008.

Joseph Fourier. *Theorie Analytique de la Chaleur*. Firmin Didot, 1822.

Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Stability of graph scattering transforms. In *NeurIPS*, pages 8038–8048, 2019.

Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *ICML*, pages 2122–2131. PMLR, 2019.

Hongyang Gao and Shuiwang Ji. Graph U-Nets. In *ICML*, 2019.

Cristina Garcia-Cardona, Ekaterina Merkurjev, Andrea L Bertozzi, Arjuna Flenner, and Allon Percus. Multiclass data segmentation using diffuse interface methods on graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1600–1613, 2014.

Thomas Gärtner, Peter Flach, and Stefan Wrobel. On graph kernels: Hardness results and efficient alternatives. In *Learning Theory and Kernel Machines*, pages 129–143. Springer, 2003.

Matan Gavish and Ronald R. Coifman. Sampling, denoising and compression of matrices by coherent matrix organization. *Applied and Computational Harmonic Analysis*, 33(3): 354 – 369, 2012.

Matan Gavish, Boaz Nadler, and Ronald R Coifman. Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning. In *ICML*, pages 367–374, 2010.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *NeurIPS*, volume 70, pages 1263–1272, 2017.

Kanghui Guo, Demetrio Labate, Wang-Q Lim, Guido Weiss, and Edward Wilson. Wavelets with composite dilations. *Electronic Research Announcements of the American Mathematical Society*, 10(9):78–87, 2004.

Alfred Haar. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69 (3):331–371, 1910.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NIPS*, pages 1024–1034, 2017.

David K. Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, 2011.

Bin Han. On dual wavelet tight frames. *Applied and Computational Harmonic Analysis*, 4 (4):380–413, 1997.

Bin Han. Pairs of frequency-based nonhomogeneous dual wavelet frames in the distribution space. *Applied and Computational Harmonic Analysis*, 29(3):330 – 353, 2010. URL http://www.sciencedirect.com/science/article/pii/S1063520310000060.

Bin Han. Nonhomogeneous wavelet systems in high dimensions. *Applied and Computational Harmonic Analysis*, 32(2):169–196, 2012.

Bin Han. Matrix splitting with symmetry and symmetric tight framelet filter banks with two high-pass filters. *Applied and Computational Harmonic Analysis*, 35(2):200–227, 2013a.

Bin Han. Properties of discrete framelet transforms. *Mathematical Modelling of Natural Phenomena*, 8(1):18–47, 2013b.

Bin Han. *Framelets and Wavelets: Algorithms, Analysis, and Applications*. Birkhäuser, 2017. ISBN 978-3-319-68529-8.

Bin Han and Xiaosheng Zhuang. Matrix extension with symmetry and its application to symmetric orthonormal multiwavelets. *SIAM Journal on Mathematical Analysis*, 42(5): 2297–2317, 2010.

Bin Han and Xiaosheng Zhuang. Smooth affine shear tight frames with MRA structure. *Applied and Computational Harmonic Analysis*, 39(2):300 – 338, 2015. URL http://www.sciencedirect.com/science/article/pii/S1063520314001213.

Bin Han, Zhenpeng Zhao, and Xiaosheng Zhuang. Directional tensor product complex tight framelets with low redundancy. *Applied and Computational Harmonic Analysis*, 41(2): 603–637, 2016a.

Bin Han, Zhenpeng Zhao, and Xiaosheng Zhuang. Directional tensor product complex tight framelets with low redundancy. *Applied and Computational Harmonic Analysis*, 41(2): 603–637, 2016b.

Bin Han, Tao Li, and Xiaosheng Zhuang. Directional compactly supported box spline tight framelets with simple geometric structure. *Applied Mathematics Letters*, 91:213 – 219, 2019. URL `http://www.sciencedirect.com/science/article/pii/S0893965918304221`.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

T. Jiang, M. K. Ng, X. Zhao, and T. Huang. Framelet representation of tensor nuclear norm for third-order tensor completion. *IEEE Transactions on Image Processing*, 29: 7233–7244, 2020.

Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1999.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

Nils Kriege and Petra Mutzel. Subgraph matching kernels for attributed graphs. In *ICML*, 2012.

Stephane Lafon and Ann B Lee. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1393–1403, 2006.

Quoc T Le Gia, Ian H Sloan, Yu Guang Wang, and Robert S Womersley. Needlet approximation for isotropic random fields on the sphere. *Journal of Approximation Theory*, 216: 86–116, 2017.

Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *ICML*, 2019.

Ming Li, Philip Broadbridge, Andriy Olenko, and Yu Guang Wang. Fast tensor needlet transforms for tangent vector fields on the sphere. *arXiv preprint arXiv:1907.13339*, 2019.

Ming Li, Zheng Ma, Yu Guang Wang, and Xiaosheng Zhuang. Fast Haar transforms for graph neural networks. *Neural Networks*, 128:188–198, 2020. URL `http://www.sciencedirect.com/science/article/pii/S0893608020301568`.

Yan-Ran Li and Xiaosheng Zhuang. Parallel magnetic resonance imaging reconstruction algorithm by three-dimension directional Haar tight framelet regularization. In Dimitri Van De Ville, Manos Papadakis, and Yue M. Lu, editors, *Wavelets and Sparsity XVIII*, volume 11138, pages 400 – 407. International Society for Optics and Photonics, SPIE, 2019. URL `https://doi.org/10.1117/12.2528788`.

Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

Yao Ma, Suhang Wang, Charu C Aggarwal, and Jiliang Tang. Graph convolutional networks with eigenpooling. In *KDD*, pages 723–731, 2019.

Yao Ma, Xiaorui Liu, Tong Zhao, Yozen Liu, Jiliang Tang, and Neil Shah. A unified view on graph neural networks as graph signal denoising. In *CIKM'21: Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021.

Zheng Ma, Junyu Xuan, Yu Guang Wang, Ming Li, and Pietro Liò. Path integral based convolution and pooling for graph neural networks. *NeurIPS*, 33, 2020.

James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

Mauro Maggioni and Hrushikesh H. Mhaskar. Diffusion polynomial frames on metric measure spaces. *Applied and Computational Harmonic Analysis*, 24(3):329 – 353, 2008.

Stéphane Mallat. *A wavelet tour of signal processing. The sparse way, With contributions from Gabriel Peyré*. Elsevier/Academic Press, Amsterdam, Third edition, 2009.

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

Stéphane Mallat. Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065): 20150203, 2016.

Stéphane G Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7): 674–693, 1989.

Barry Merriman, James Kenyard Bence, and Stanley Osher. Diffusion generated motion by mean curvature. In J.E. Taylor, editor, *Computational Crystal Growers Workshop*, pages 73–83. American Mathematical Society, Providence, Rhode Island, 1992.

Yves Meyer. *Ondelettes et opérateurs. I*. Actualités Mathématiques. [Current Mathematical Topics]. Hermann, Paris, 1990.

H.N. Mhaskar. Eignets for function approximation on manifolds. *Applied and Computational Harmonic Analysis*, 29(1):63 – 87, 2010. URL http://www.sciencedirect.com/science/article/pii/S106352030900089X.

HN Mhaskar and J Prestin. Polynomial frames: a fast tour. In *Approximation Theory XI: Gatlinburg*, pages 101–132, 2004.

Federico Monti, Michael Bronstein, and Xavier Bresson. Geometric matrix completion with recurrent multi-graph neural networks. In *NIPS*, pages 3697–3707, 2017.

Maria CV Nascimento and Andre CPLF De Carvalho. Spectral methods for graph clustering–a survey. *European Journal of Operational Research*, 211(2):221–231, 2011.

Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *ICML*, pages 2014–2023, 2016.

Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. Deepsphere: Efficient spherical convolutional neural network with healpix sampling for cosmological applications. *Astronomy and Computing*, 27:130–146, 2019.

Amos Ron and Zuowei Shen. Affine systems in $L_2(\mathbb{R}^d)$: The analysis of the analysis operator. *Journal of Functional Analysis*, 148(2):408 – 447, 1997a.

Amos Ron and Zuowei Shen. Affine systems in $L_2(\mathbb{R}^d)$ II: Dual systems. *Journal of Fourier Analysis and applications*, 3(5):617–637, 1997b.

Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL http://networkrepository.com.

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

Rui Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678, May 2005.

Aliaksei Sandryhaila and José MF Moura. Discrete signal processing on graphs. *IEEE transactions on signal processing*, 61(7):1644–1656, 2013.

Aliaksei Sandryhaila and Jose MF Moura. Discrete signal processing on graphs: Frequency analysis. *IEEE Transactions on Signal Processing*, 62(12):3042–3054, 2014.

Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

Geoffrey Schiebinger, Elina Robeva, and Benjamin Recht. Superresolution without separation. *Information and Inference: A Journal of the IMA*, 7(1):1–30, 05 2017. URL https://doi.org/10.1093/imaiai/iax006.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *NeurIPS Workshop on Relational Representation Learning*, 2018.

Nino Shervashidze, SVN Vishwanathan, Tobias Petri, Kurt Mehlhorn, and Karsten Borgwardt. Efficient graphlet kernels for large graph comparison. In *AISTATS*, pages 488–495, 2009.

Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-Lehman graph kernels. *Journal of Machine Learning Research*, 12(77):2539–2561, 2011.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Jonathan Shlomi, Peter Battaglia, et al. Graph neural networks in particle physics. *Machine Learning: Science and Technology*, 2020.

David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.

David I Shuman, Christoph Wiesmeyr, Nicki Holighaus, and Pierre Vandergheynst. Spectrum-adapted tight graph wavelet and vertex-frequency frames. *IEEE Transactions on Signal Processing*, 63(16):4223–4235, 2015.

Amit Singer. From graph to manifold laplacian: The convergence rate. *Applied and Computational Harmonic Analysis*, 21(1):128 – 134, 2006. Special Issue: Diffusion Maps and Wavelets.

Elias M. Stein. *Singular Integrals and Differentiability Properties of Functions*. Princeton University Press, 1970.

Elias M. Stein. *Harmonic Analysis: Real-Variable Methods, Orthogonality, and Oscillatory Integrals*. Princeton University Press, 1993.

Elias M Stein and Rami Shakarchi. *Fourier analysis: an introduction*, volume 1. Princeton University Press, 2011.

Elias M. Stein and Guido Weiss. *Introduction to Fourier analysis on Euclidean spaces*. Princeton University Press, 1971.

X Yu Stella and Jianbo Shi. Multiclass spectral clustering. In *ICCV*, pages 313–319 vol.1, 2003.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Yu Guang Wang and Houying Zhu. Analysis of framelet transforms on a simplex. In *Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan*, pages 1175–1189. Springer, 2018.

Yu Guang Wang and Xiaosheng Zhuang. Tight framelets and fast framelet filter bank transforms on manifolds. *Applied and Computational Harmonic Analysis*, 2018.

Yu Guang Wang and Xiaosheng Zhuang. Tight framelets on graphs for multiscale data analysis. In *Wavelets and Sparsity XVIII*, volume 11138, page 111380B. International Society for Optics and Photonics, 2019.

Yu Guang Wang, Quoc T Le Gia, Ian H Sloan, and Robert S Womersley. Fully discrete needlet approximation on the sphere. *Applied and Computational Harmonic Analysis*, 43 (2):292–316, 2017.

Yu Guang Wang, Ming Li, Zheng Ma, Guido Montufar, Xiaosheng Zhuang, and Yanan Fan. Haar graph pooling. In *ICML*, 2020.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

Bingbing Xu, Huawei Shen, Qi Cao, Yunqi Qiu, and Xueqi Cheng. Graph wavelet neural network. In *ICLR*, 2019a.

Dongkuan Xu and Yingjie Tian. A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2):165–193, 2015.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *ICLR*, 2019b. URL `https://openreview.net/forum?id=ryGs6iA5Km`.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *NeurIPS*, pages 4800–4810, 2018.

Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. In *AAAI*, 2018.

Zhenyue Zhang and Hongyuan Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal on Scientific Computing*, 26(1): 313–338, 2004.

Xuebin Zheng, Bingxin Zhou, Ming Li, Yu Guang Wang, and Junbin Gao. MathNet: Haar-like wavelet multiresolution-analysis for graph representation and learning. *arXiv preprint arXiv:2007.11202*, 2020.

Xuebin Zheng, Bingxin Zhou, Junbin Gao, Yu Guang Wang, Pietro Liò, Ming Li, and Guido Montúfar. How framelets enhance graph neural networks. *ICML*, 2021.

Bingxin Zhou, Ruikun Li, Xuebin Zheng, Yu Guang Wang, and Junbin Gao. Graph denoising with framelet regularizer. *arXiv preprint arXiv:2111.03264*, 2021a.

Bingxin Zhou, Xinliang Liu, Yuehua Liu, Yunying Huang, Pietro Liò, and YuGuang Wang. Spectral transform forms scalable transformer. *arXiv preprint arXiv:2111.07602*, 2021b.

Xiaosheng Zhuang. Digital affine shear transforms: fast realization and applications in image/video processing. *SIAM Journal on Imaging Sciences*, 9(3):1437–1466, 2016.

Dongmian Zou and Gilad Lerman. Graph convolutional neural networks via scattering. *Applied and Computational Harmonic Analysis*, 49(3):1046–1074, 2020.