SPARSE REPRESENTATION ON GRAPHS BY TIGHT WAVELET FRAMES AND APPLICATIONS

BIN DONG

ABSTRACT. In this paper, we introduce a unified theory of tight wavelet frames on non-flat domains in both continuum setting, i.e. on manifolds, and discrete setting, i.e. on graphs; discuss how fast tight wavelet frame transforms can be computed and how they can be effectively used to process graph data. We start from defining multiresolution analysis (MRA) generated by a single generator on manifolds, and discuss the conditions needed for a generator to form an MRA. With a given MRA, we show how MRA-based tight frames can be constructed and the conditions needed for them to generate a tight wavelet frame on manifolds. In particular, we show that under suitable conditions, framelets on \( \mathbb{R} \) constructed from the unitary extension principle [1] can also generate tight frame systems on manifolds. We also discuss how the transition from the continuum to the discrete setting can be naturally defined, which leads to multi-level discrete tight wavelet frame transforms (decomposition and reconstruction) on graphs. In order for the proposed discrete tight wavelet frame transforms to be useful in applications, we show how the transforms can be computed efficiently and accurately. More importantly, numerical simulations show that the proposed discrete tight wavelet frame transform maps piecewise smooth data to a set of sparse coefficients. This indicates that the proposed tight wavelet frames indeed provide sparse representation on graphs. Finally, we consider two specific applications: graph data denoising and semi-supervised clustering. Utilizing the proposed sparse representation, we introduce \( \ell_1 \)-norm based optimization models for denoising and semi-supervised clustering, which are inspired by the models used in image restoration and image segmentation.

1. INTRODUCTION

In recent years, we are experiencing rapid advances in information and computer technology, which contribute greatly to the exponential growth of data. To properly handle, process and analyze such huge and often unstructured data sets, sophisticated mathematical tools and efficient computing methods need to be developed. Such huge data sets, commonly referred to as “big data”, are generally modelled as huge graphs living in very high dimensional spaces. Graphs are commonly understood as a certain discretization or a random sample from some smooth Riemannian manifold [2–6]. To understand and analyze graphs and data on graphs (shall be called graph data), the graph Laplacian is widely used to reveal the geometric properties of the graph and plays an important role in many applications such as graph clustering.

In signal and image processing, many methods are transform based. Sparsity of the signal/image to be recovered under a certain transform is the key to the success of many existing algorithms. One of the successful examples is the wavelet frame transform, especially the tight wavelet frame transform [7–19]. The power of tight wavelet frames lies in their ability to sparsely approximate piecewise smooth functions and the existence of fast decomposition and reconstruction algorithms. Recently, geometric properties of tight wavelet frames were discovered by connecting them to differential operators under variational and PDE frameworks [17–19].

Key words and phrases. Tight wavelet frames, sparse approximation on graphs, spectral graph theory, big data, graph clustering.

Bin Dong is with the Department of Mathematics, University of Arizona, 617 North Santa Rita Avenue, Tucson, AZ, USA, 85721-0089.

Email: dongbin@math.arizona.edu.

Phone: 1-520-621-2559.

This work was supported in part by NSF DMS-1418772.
The effectiveness of wavelet frames for data defined on flat domains motivates much research on generalizing wavelets and wavelet frames to curved, irregular and unstructured domains. In this paper, we introduce a unified theory of tight wavelet frames on non-flat domains in both continuum (on manifolds) and discrete (on graphs) setting, discuss how fast tight wavelet frame transforms can be computed and how they can be effectively used to process graph data. The basic idea is to understand eigenfunctions of Laplace-Beltrami operator (graph Laplacian in discrete setting) as Fourier basis on manifolds (graphs in discrete setting) and the associated eigenvalues as frequency components. This idea was used earlier by [20] in the discrete setting. In this paper, we further observe that, in fact, many classical concepts in wavelets, such as multiresolution analysis (MRA), refinability and Quasi-affine system generated by translations and dilations of wavelet functions, can be similarly defined on manifolds. Indeed, all of these concepts have an equivalent characterization in Fourier domain. Moreover, with an MRA and the concept of refinability on manifolds, the transition from continuum to discrete setting can be done very naturally, and is analogous to such transition in classical wavelet theory. To be a bit more precise, inner product with wavelet frame functions on manifolds can be approximated in the discrete setting by filtering with the associated masks on graphs. In addition, since most masks we use are trigonometric polynomials, they can be accurately approximated by low-degree polynomials, which enables us to compute the decomposition and reconstruction transforms very efficiently and accurately.

Given a compact Riemannian manifold $(\mathcal{M}, g)$, denote $L^2(\mathcal{M})$ the space of square integrable functions on $\mathcal{M}$. We start from defining multiresolution analysis (MRA) for $L^2(\mathcal{M})$ that is generated by a single generator. We discuss the conditions needed for a generator to form an MRA (Theorem 2.1). With a given MRA, we show how MRA-based tight frames can be constructed and the conditions needed for them to generate a tight wavelet frame for $L^2(\mathcal{M})$ (Theorem 2.2). In particular, we show that under suitable conditions, framelets on $\mathbb{R}$ constructed from the unitary extension principle [1] also generate tight frame systems for $L^2(\mathcal{M})$ (Corollary 2.1).

Thanks to the MRA structure, we discuss how the transition from the continuum to the discrete setting can be naturally defined. This leads to multi-level discrete tight wavelet frame transforms (decomposition and reconstruction) on graphs, which can be interpreted as an iterative “convolution” of graph data with properly dilated wavelet frame masks (or filters). In order for the proposed discrete tight wavelet frame transforms to be useful in applications, we show how the transforms can be computed efficiently and accurately. The proposed fast tight wavelet frame transforms (FTWFT) for graph data have a similar computational complexity as for images, and the reconstruction error is negligible. More importantly, numerical simulations show that the FTWFT maps piecewise smooth data to a set of sparse coefficients, which indicates that the proposed tight wavelet frames indeed provide sparse representation on graphs.

Finally, we consider two specific applications of the proposed FTWFT: graph data denoising and semi-supervised clustering. Utilizing the sparse representation provided by FTWFT, we introduce $\ell_1$-norm based optimization models for denoising and semi-supervised clustering. These models are motivated from models used in image restoration and image segmentation. In fact, now that we have FTWFT as a sparsifying transform for piecewise smooth graph data, modelling on graphs can be easily generalized from modelling in image processing and analysis, whenever the same application is still relevant on graphs.

The rest of the paper is organized as follows. In the rest of this section, we review some existing work on defining wavelets or wavelet-like representations on 2-dimensional surfaces and graphs. In Section 2, we first introduce MRA for $L^2(\mathcal{M})$. Based on an MRA, we discuss how wavelet frames can be constructed and the conditions needed for them to form a tight frame for $L^2(\mathcal{M})$. Section 3 starts with a transition from the continuum to the discrete setting, which leads to the discrete tight wavelet frame transforms. FTWFT is later introduced and numerical simulations showing the efficiency, accuracy and sparsity of the transforms are conducted. Based on the proposed sparse
representation, we design models and efficient numerical algorithms in Section 4 for two specific applications: graph data denoising and semi-supervised clustering.

1.1. Related Work. Wavelets and their generalizations are well studied in the past thirty years [21–24]. Their success in various applications are mostly due to the sparse representation they provide for piecewise smooth functions. In the past few decades, there has been much endeavour in the community to generalize (bio)orthogonal wavelets and wavelet frames to functions or data defined on non-flat domains, such as spheres, surfaces, graphs, etc.

There has been a relatively rich literature on wavelets and wavelet frames on 2-dimensional surfaces. Wavelets on sphere were first introduced in [25] using the lifting scheme [26], and later in [27, 28] via a group-theoretical approach. Biorthogonal wavelets with high symmetry for surface multiresolution processing have been constructed in [29–34]. Loop’s scheme-based biorthogonal wavelets have been considered in [35] with the biorthogonal dual wavelets constructed in [36]. Redundant representations on surfaces were introduced by [37], where 6-fold symmetric bi-frames with 4 framelets (frame generators) for triangulated surfaces were introduced. More recently in [38], tight wavelet frames on triangulated and quad surfaces were constructed and applications in surface denoising were considered.

In recent years, there has been much interest in constructing wavelet-like representation of graph data. Successful examples include the wavelets on unweighted graphs by [39], the multiscale scheme on graphs based on lifting by [40], the Haar wavelet transform for rooted binary trees [41] and its generalization treelets [42], the diffusion wavelets [43] and diffusion polynomial frames [44], the wavelets on compact differentiable manifolds [45], the spectral graph wavelet transform by [20, 46, 47], Haar transform for coherent matrices [48], and orthogonal polynomial systems for weighted trees [49].

2. Tight Wavelet Frames on Manifold \( \{M, g\} \)

We start this section with a brief review of some results in eigenvalue problems of Laplace-Beltrami operator on Riemannian manifolds \( \{M, g\} \). For a comprehensive review of the subject, one can refer to the book [50]. We will define multiresolution analysis of \( L_2(M) \) generated by a single function, followed by a characterization of tight wavelet frames for \( L_2(M) \). We end this section by showing some examples.

2.1. Spectrum and Eigenfunctions of Laplace-Beltrami Operator. Let \( \{M, g\} \) be a compact, connected Riemannian manifold with smooth boundary \( S \). Let \( m \geq 2 \) be the dimension of \( M \). Let \( \Delta \) be the Laplace-Beltrami operator on \( M \) with respect to the metric \( g \). Let \( \{\lambda_p : p = 0, 1, \ldots\} \) and \( \{u_p : p = 0, 1, \ldots\} \) be the eigenvalues and eigenfunctions of the following eigenvalue problem

\[
\Delta u + \lambda u = 0,
\]

with Dirichlet boundary condition \( u = 0 \). As convention, \( 0 < \lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \cdots \). The set of eigenfunctions form an orthonormal basis of \( L_2(M) \), i.e.

\[
\langle u_p, u_{p'} \rangle = \int_M u_p(x)u_{p'}^*(x)dx = \delta_{p,p'},
\]

and \( \{u_p\} \) is complete on \( L_2(M) \). Here, \( f^* \) denotes the complex conjugate of \( f \). Given \( f \in L_2(M) \), we define

\[
\hat{f}[p] = \langle f, u_p \rangle,
\]

with \( \hat{f} \in \ell_2(\mathbb{Z}^+) \) and \( \mathbb{Z}^+ = \{0, 1, 2, \ldots\} \). We have the following Parseval’s identity

\[
\|f\|_{L_2(M)}^2 = \|\hat{f}\|_{\ell_2(\mathbb{Z}^+)}^2.
\]
Finally, we recall the following two results from eigenvalue problems of Riemannian geometry. First, Weyl’s asymptotic formula [50, 51] gives us the growth rate of \( \{\lambda_p\} \):

\[
\lambda_p = O(p^{\frac{s}{m}}),
\]

with \( m \) the dimension of the manifold. The second result is the uniform bound of the eigenfunctions [52]:

\[
\|u_p\|_{L_\infty(M)} \leq C\lambda_p^{\frac{m-1}{4}}.
\]

2.2. Tight Wavelet Frames for \( L_2(M) \). We start with defining a system in \( L_2(M) \) that mimics the quasi-affine system [1, 24] for \( L_2(\mathbb{R}) \) generated by finitely many elements. Define the system \( X(\Psi) \subset L_2(M) \) as

\[
X(\Psi) = \{\psi_{j,n,y}^M \in L_2(M) : 1 \leq j \leq r, n \in \mathbb{Z}, y \in M\},
\]

where \( \psi_{j,n,y}^M \in L_2(M) \) is generated by \( \Psi = \{\psi_j : 1 \leq j \leq r\} \subset L_2(\mathbb{R}) \) as

\[
\psi_{j,n,y}^M(x) = \sum_{p=0}^{\infty} \hat{\psi}_j(2^{-n}\lambda_p)u_p^*(y)u_p(x), \quad \text{with } n \in \mathbb{Z}, \ x \in M, \ y \in M,
\]

where \( \hat{\psi}_j \) denotes that Fourier transform of \( \psi_j \in L_2(\mathbb{R}) \). To ensure \( \psi_{j,n,y}^M \in L_2(M) \), we need to impose a regularity condition on \( \psi_j \in L_2(\mathbb{R}) \) or, in other words, a decay condition on \( \hat{\psi}_j \) as follows:

\[
|\hat{\psi}_j(\xi)| \leq C(1 + |\xi|)^{-s} \quad \text{with } s > \frac{2m - 1}{4},
\]

for all \( \xi \in \mathbb{R} \). Then, the two results in geometry (2.1) and (2.2) guarantee that \( \psi_{j,n,y}^M \) belongs \( L_2(M) \). The index \( n \) of \( \hat{\psi}_{j,n,y}^M \) denotes dilation and \( y \) translation. As one can see that \( \psi_{j,n,y}^M \) is defined in the spectral domain mimicking the dilation and translation of functions on Euclidean domains via Fourier transform, which was first used by [20] in discrete setting, i.e. on graphs. We also note that the dyadic dilation \( 2^{-n} \) used here is nonessential and one may change it to any \( s^{-n} \) with \( s > 1 \).

The main objective of this section is to discuss conditions on \( \psi_j \in L_2(\mathbb{R}) \), \( 1 \leq j \leq r \), such that \( X(\Psi) \) defined by (2.3) is a tight frame of \( L_2(M) \). In case \( X(\Psi) \) is a tight frame of \( L_2(M) \), we shall call \( X(\Psi) \) a tight wavelet frame of \( L_2(M) \) and the elements in \( \Psi \) framelets.

2.2.1. MRA of \( L_2(M) \). The concept of multiresolution analysis (MRA) was first introduced in [53, 54] and was later generalized by [55, 56]. It is a general framework that makes construction of orthonormal wavelet bases for \( L_2(\mathbb{R}) \) painless. The MRA-based compactly supported orthonormal wavelet systems were constructed by [57] and MRA-based biorthogonal wavelet bases were constructed later in [57, 58]. Tight wavelet frames derived from over sampled orthonormal wavelet basis were used in noise removal by [59, 60]. The publication of the unitary extension principle (UEP) of [1] initiated a new wave of theoretical development, as well as exploration of new applications of MRA-based tight wavelet frames. Further theoretical developments on MRA-based wavelet frames can be found in e.g. [61–63] and the reference therein. A short survey on the theory and applications of MRA-based tight wavelet frames is given by [64], and a more detailed survey was given by [24]. Other than supporting easy constructions of bases and redundant systems, the MRA structure also grants fast implementation of wavelet decomposition and reconstruction which makes wavelets a very practical tool for image/signal processing and analysis.

An MRA is a sequence of nested subspaces such that their intersection is trivial and union is dense in the entire space [55, 56]. We start with defining the subspace \( V_0 \subset L_2(M) \) generated by a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) with finitely supported refinement mask \( a \in \ell_0(\mathbb{Z}) \) satisfying the refinement equation

\[
\hat{\phi}(2\xi) = \hat{a}(\xi)\hat{\phi}(\xi).
\]
Here, \( \hat{\phi} \) is the Fourier transform of \( \phi \in L_2(\mathbb{R}) \) and \( \hat{a} \) is the Fourier series of \( a \), which is a trigonometric polynomial since \( a \) is finitely supported. Same as the wavelet functions \( \psi_j \), we assume the following decay property on \( \hat{\phi} \):

\[
|\hat{\phi}(\xi)| \leq C (1 + |\xi|)^{-s} \quad \text{with } s > \frac{2m - 1}{4},
\]

for all \( \xi \in \mathbb{R} \). Similar to the definition of \( \psi_{n,y}^M \in L_2(\mathcal{M}) \), we can define \( \phi_{n,y}^M \in L_2(\mathcal{M}) \) as

\[
\phi_{n,y}^M(x) = \sum_{p=0}^{\infty} \hat{\phi}(2^{-n}\lambda_p)u^*_p(y)u_p(x), \quad \text{with } n \in \mathbb{Z}, \ x \in \mathcal{M}, \ y \in \mathcal{M}.
\]

Then, the refinement equation of \( \hat{\phi} \in L_2(\mathbb{R}) \) is generalized to

\[
\hat{\phi}_{n,y}^M[p] = \hat{a}(2^{-n-1}\lambda_p)\hat{\phi}_{n+1,y}^M[p], \quad p \in \mathbb{Z}^+.
\]

Based on the above generalization, how should we define the subspaces \( V_n \subset L_2(\mathcal{M}) \) analogous to the classical shift-invariant subspaces of \( L_2(\mathbb{R}) \) generated by \( \phi_n = \phi(2^n \cdot) \)? Recall the characterization of \( V_n(\phi_n) \subset L_2(\mathbb{R}) \) from [65],

\[
\mathcal{V}_n(\phi_n) = \{ \tau \hat{\phi}_n \in L_2(\mathbb{R}) : \tau \text{ is } 2^{n+1}\pi\text{-periodic measurable function} \}.
\]

Then naturally, the space \( V_n \subset L_2(\mathcal{M}) \) can be defined as

\[
V_n = \left\{ \sum_{p=0}^{\infty} c[p]\hat{\phi}(2^{-n}\lambda_p)u_p(x) : c \in \ell(\mathbb{Z}^+) \right\},
\]

where the space \( \ell(\mathbb{Z}^+) \) is the space containing sequences with a growth rate not exceeding \( p^{1 - \frac{1}{2m}} \):

\[
\ell(\mathbb{Z}^+) = \left\{ c[p] : p \in \mathbb{Z}^+, \ |c[p]| \leq C(1 + p)^{\frac{1}{2} - \frac{1}{2m}} \right\}.
\]

By the decay assumption (2.5), it is easy to see that \( V_n \) is indeed a subspace of \( L_2(\mathcal{M}) \) for all \( n \in \mathbb{Z} \).

For \( \{V_n : n \in \mathbb{Z}\} \) to form an MRA of \( L_2(\mathcal{M}) \), the following conditions are to be satisfied [55,56].

1. \( V_n \subset V_{n+1} \), for all \( n \in \mathbb{Z} \);
2. \( \bigcup_{n \in \mathbb{Z}} V_n = L_2(\mathcal{M}) \);
3. \( \cap_{n \in \mathbb{Z}} V_n = \{0\} \).

We first show that \( \{V_n : n \in \mathbb{Z}\} \) is nested, i.e. \( V_n \subset V_{n+1} \).

**Proposition 2.1.** Let \( V_n \) be defined by (2.6) associated with a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) satisfying (2.5), with mask \( a \in \ell_0(\mathbb{Z}) \). The sequence of subspaces \( \{V_n : n \in \mathbb{Z}\} \) is nested, i.e. \( V_n \subset V_{n+1} \) for all \( n \in \mathbb{Z} \).

**Proof.** Given \( g \in V_n \), there exists \( c_g \in \ell(\mathbb{Z}^+) \) such that

\[
g = \sum_{p=0}^{\infty} c_g[p]\hat{\phi}(2^{-n}\lambda_p)u_p(x).
\]

Using the refinement equation, we have

\[
\hat{\phi}(2^{-n}\xi) = \hat{a}(2^{-n-1}\xi)\hat{\phi}(2^{-n-1}\xi).
\]

Therefore,

\[
g = \sum_{p=0}^{\infty} c_g[p]\hat{a}(2^{-n-1}\lambda_p)\hat{\phi}(2^{-n-1}\lambda_p)u_p(x).
\]

Let \( \tilde{c}_g[p] = c_g[p]\hat{a}(2^{-n-1}\lambda_p) \). Since \( \hat{a} \) is bounded, we have \( \tilde{c}_g \in \ell(\mathbb{Z}^+) \), which implies \( g \in V_{n+1} \). This shows that \( V_n \subset V_{n+1} \). \( \square \)

Next, we show that the union of \( V_n \) is dense in \( L_2(\mathcal{M}) \).
Proposition 2.2. Let \( V_n \) be defined by (2.6) associated with a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) satisfying (2.5), with mask \( a \in \ell_0(\mathbb{Z}) \). Assume that \(|\hat{a}(\xi) - 1| \leq C|\xi|\) for \( \xi \) near the origin. Then,

\[
\bigcup_{n \in \mathbb{Z}} V_n = L_2(\mathcal{M}).
\]

Proof. By the assumption on \( \hat{a} \) near the origin, we have \( \hat{\phi}(\xi) \to 1 \) as \( \xi \to 0 \) (see e.g. [66, 67]). For any \( f \in L_2(\mathcal{M}) \), let \( f_n \) be defined as

\[
f_n(x) = \sum_{p=0}^{\infty} \hat{f}[p]\hat{\phi}(2^{-n}\lambda_p)u_p(x).
\]

Since \( \hat{f} \in \ell(\mathbb{Z}^+) \), we have \( f_n \in V_n \). Then,

\[
\|f_n - f\|_{L_2(\mathcal{M})}^2 = \sum_{p=0}^{\infty} |\hat{f}[p]\hat{\phi}(2^{-n}\lambda_p) - \hat{f}[p]|^2 = \sum_{p=0}^{\infty} |\hat{f}[p]|^2|\hat{\phi}(2^{-n}\lambda_p) - 1|^2.
\]

Since \( \hat{\phi}(2^{-n}\lambda_p) \to 1 \) for each \( p \geq 0 \) as \( n \to \infty \), and \( \hat{\phi} \) is bounded, we have \( \|f_n - f\|_{L_2(\mathcal{M})}^2 \to 0 \) as \( n \to \infty \). This shows that \( \cup_{n \in \mathbb{Z}} V_n \) is dense in \( L_2(\mathcal{M}) \).

Proposition 2.3. Let \( V_n \) be defined by (2.6) associated with a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) satisfying (2.5), with mask \( a \in \ell_0(\mathbb{Z}) \). Then,

\[
\bigcap_{n \in \mathbb{Z}} V_n = \{0\}.
\]

Proof. Suppose there exists \( f \in L_2(\mathcal{M}) \) such that \( f \neq 0 \) and \( f \in \cap_{n \in \mathbb{Z}} V_n \). Then for each \( n \in \mathbb{Z} \), there exists \( c_n \in \ell(\mathbb{Z}^+) \) such that

\[
\hat{f}[p] = c_n[p]\hat{\phi}(2^{-n}\lambda_p).
\]

By the definition of \( \ell(\mathbb{Z}^+) \) and (2.5), we have

\[
|\hat{f}[p]| \leq C(1 + |p|)^{\frac{1}{2} - \frac{d}{mp}}(1 + |2^{-n}\lambda_p|)^{-s}.
\]

Since \( s > \frac{2m-1}{2} > 0 \) for \( m \geq 2 \), and \( \lambda_p > 0 \) for \( p \geq 0 \), we can let \( n \to -\infty \) in the above inequality and obtain \( \hat{f}[p] = 0 \) for \( p \geq 0 \). Therefore, \( f = 0 \), which contradicts with our hypothesis.

Theorem 2.1. Let \( V_n \subset L_2(\mathcal{M}) \) be defined by (2.6) associated with a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) satisfying (2.5), with mask \( a \in \ell_0(\mathbb{Z}) \). Assume that \(|\hat{a}(\xi) - 1| \leq C|\xi|\) for \( \xi \) near the origin. Then \( \{V_n : n \in \mathbb{Z}\} \) forms an MRA of \( L_2(\mathcal{M}) \).

2.2. MRA-Based Tight wavelet frames for \( L_2(\mathcal{M}) \). Given an MRA \( \{V_n : n \in \mathbb{Z}\} \) of \( L_2(\mathcal{M}) \) generated by a compactly supported refinable function \( \phi \in L_2(\mathbb{R}) \) with mask \( a \in \ell_0(\mathbb{Z}) \), we define a set of compactly supported functions \( \Psi = \{\psi_1, \ldots, \psi_r\} \subset L_2(\mathbb{R}) \) by their associated masks \( a_j \in \ell_0(\mathbb{Z}) \):

\[
\hat{\psi}_j(2\xi) = \hat{a}_j(\xi)\hat{\phi}(\xi), \quad 1 \leq j \leq r.
\]

Letting \( \psi_0 = \phi \) and \( a_0 = a \), we can include the refinement equation for \( \phi \) in the above equations, i.e. \( \hat{\psi}_j(2\xi) = \hat{a}_j(\xi)\hat{\phi}(\xi) \) for \( 0 \leq j \leq r \). Recall the definition of the dilations and translations of \( \psi_j \):

\[
\psi_{j,n,y}^M(x) = \sum_{p=0}^{\infty} \hat{\psi}_j(2^{-n}\lambda_p)u_p^*(y)u_p(x), \quad 0 \leq j \leq r,
\]

where \( \psi_{0,n,y}^M = \phi_{n,y}^M \). Now, let us see that \( \psi_{j,n,y}^M \) defined in (2.7) is indeed associated with the MRA \( \{V_n : n \in \mathbb{Z}\} \) generated by \( \phi \). By (2.7), we have

\[
\hat{\psi}_{j,n,y}^M[p] = \hat{\psi}_j(2^{-n}\lambda_p)u_p^*(y) = \hat{a}_j(2^{-n-1}\lambda_p)\hat{\phi}(2^{-n-1}\lambda_p)u_p^*(y).
\]
By the upper bound of \(\|u_p\|_{L_\infty(M)}\) in (2.2) and boundedness of \(\hat{a}_j(\xi)\), we can easily see that 
\(\hat{a}_j(2^{-n-1}\lambda_p)u_p^*(y) \in \ell(\mathbb{Z}^+)\) for each \(n \in \mathbb{Z}\) and \(y \in M\), and hence, \(\psi_{j,n,y}^M \in V_{n+1}\). This is consistent with how the classical MRA-based wavelet frames for \(L_2(\mathbb{R})\) are constructed.

The inner product between \(f \in L_2(M)\) and \(\psi_{j,n,y}^M\) is given by
\[
\langle f, \psi_{j,n,y}^M \rangle = \langle \hat{f}, \hat{\psi}_{j,n,y}^M \rangle = \sum_{p=1}^{\infty} \hat{f}[p] \hat{\psi}_j^*(2^{-n} \lambda_p)u_p(y).
\]

Now, we can define the following operator \(P_{n,j} : L_2(M) \mapsto L_2(M)\) as
\[
P_{n,j}f = \int_M \langle f, \psi_{j,n,y}^M \rangle \psi_{j,n,y}^M dy, \quad 0 \leq j \leq r,
\]
where
\[
P_n f = P_{n,0} = \int_M \langle f, \phi_{n,y}^M \rangle \phi_{n,y}^M dy,
\]
is a quasi-interpolatory operator with \(P_n : L_2(M) \mapsto V_n\).

By definition, \(\{\psi_j : 0 \leq j \leq r\}\) are uniquely determined by their masks \(\{a_j : 0 \leq j \leq r\}\). Thus, the system \(X(\Psi)\) is uniquely determined by masks \(\{a_j : 0 \leq j \leq r\}\). The following theorem tells us for what conditions on \(\{a_j : 0 \leq j \leq r\}\), the corresponding system \(X(\Psi)\) defined by (2.3) is a tight frame of \(L_2(M)\). In fact, the decay conditions (2.4) and (2.5) can be characterized by the associated masks as well [21]. However, for simplicity and clarity, we will not convert (2.4) and (2.5) to conditions on the masks.

**Theorem 2.2.** Given the set of compactly supported functions \(\{\psi_j : 0 \leq j \leq r\} \subset L_2(\mathbb{R})\) and the associated trigonometric polynomials (or masks) \(\{\hat{a}_j(\xi) : 0 \leq j \leq r\}\) satisfying \(\hat{\psi}_j(2\xi) = \hat{\psi}_j(\xi)\hat{\phi}(\xi)\), assume that the decay conditions (2.4) and (2.5) are satisfied, \(|\hat{a}_0(\xi) - 1| \leq C|\xi|\) for \(\xi\) near the origin, and

\[
(2.8) \sum_{j=0}^{r} |\hat{a}_j(\xi)|^2 = 1.
\]

Then, the system \(X(\Psi)\) is a tight frame of \(L_2(M)\), i.e.
\[
f = \sum_{j=1}^{r} \sum_{n \in \mathbb{Z}} \int_M \langle f, \psi_{j,n,y}^M \rangle \psi_{j,n,y}^M dy \quad \text{for every } f \in L_2(M).
\]
Proof. We first show that $\overline{P_{n,j}f[p]} = \widehat{f[p]} \left| \hat{\psi}_j(2^{-n} \lambda_p) \right|^2$, and in particular, $\overline{P_n f[p]} = \widehat{f[p]} \left| \hat{\phi}(2^{-n} \lambda_p) \right|^2$. Indeed,

$$\overline{P_{n,j}f[p]} = \langle P_{n,j} f, u_p \rangle = \int \langle f, \psi_{j,n,y}^M \rangle \psi_{j,n,y}^M dy, u_p \rangle$$

(by $\langle u_p, u_{p'} \rangle = \delta_{p,p'}$)

$$= \int_\mathcal{M} \langle f, \psi_{j,n,y}^M \rangle \psi_{j,n,y}^M u_p(y) dy$$

$$= \int_\mathcal{M} \left( \sum_{p' = 1}^\infty \widehat{f[p']} \psi_{j,p'}^*(2^{-n} \lambda_{p'}) u_p(y) \right) \psi_j(2^{-n} \lambda_p) u_p^*(y) dy$$

$$= \sum_{p' = 1}^\infty \widehat{f[p']} \psi_{j,p'}^*(2^{-n} \lambda_{p'}) \psi_j(2^{-n} \lambda_p) \left( \int_\mathcal{M} u_{p'}(y) u_p^*(y) dy \right)$$

$$= \widehat{f[p]} \left| \psi_j(2^{-n} \lambda_p) \right|^2.$$ 

Then, we have

$$\overline{P_n f[p]} = \overline{P_{n,0}f[p]} = \widehat{f[p]} \left| \hat{\phi}(2^{-n} \lambda_p) \right|^2$$

(by (2.8))

$$= \sum_{j = 0}^r \widehat{f[p]} \left| \hat{\psi}_j(2^{-n} \lambda_p) \hat{\phi}(2^{-n} \lambda_p) \right|^2$$

$$= \sum_{j = 0}^r \widehat{f[p]} \left| \hat{\psi}_j(2^{-n+1} \lambda_p) \right|^2 = \sum_{j = 0}^r \overline{P_{n-1,j}f[p]}.$$ 

Thus,

$$P_n f = P_{n-1} f + \sum_{j = 1}^r P_{n-1,j} f$$

and hence

$$P_{n_1} f = P_{n_2} f + \sum_{j = 1}^{n_1} \sum_{n = n_2}^r P_{n,j} f.$$ 

Since $\hat{\phi}(\xi) \to 1$ as $\xi \to 0$, and $\hat{\phi}$ is bounded on $\mathbb{R}$, similar arguments as in the proof of Proposition 2.2 lead to

$$P_{n_1} f \to f \text{ in } L_2(\mathcal{M}) \text{ as } n_1 \to \infty.$$ 

Finally, we show that

$$P_{n_2} f \to 0 \text{ in } L_2(\mathcal{M}) \text{ as } n_2 \to -\infty.$$ 

Indeed, since $\hat{\phi}(\xi) \to 0$ as $\xi \to \infty$, we have $\overline{P_{n_2}f[p]} = \widehat{f[p]} \left| \hat{\phi}(2^{-n_2} \lambda_p) \right|^2 \to 0$ as $n_2 \to -\infty$, for $p \geq 0$. Thus, we have $P_{n_2} f \to 0$ in $L_2(\mathcal{M})$ as $n_2 \to -\infty$. This concludes the proof of the theorem.}

By Theorem 2.2, it is immediate that all the compactly supported tight wavelet frames on $L_2(\mathbb{R})$ constructed from the unitary extension principle (UEP) [1] can generate a tight wavelet frame for $L_2(\mathcal{M})$ provided that the decay conditions (2.4) and (2.5) are satisfied. Recall that a set of masks $\{a_j \in \ell_0(\mathbb{Z}) : j = 0, 1, \ldots, r\}$ is said to satisfy the UEP [1] if

$$\sum_{j = 0}^r |\hat{a}_j(\xi)|^2 = 1 \text{ and } \sum_{j = 0}^r \hat{a}_j(\xi)\hat{a}_j^*(\xi + \pi) = 0.$$
 Interested reader should consult [1, 62] for details.

**Corollary 2.1.** Let \( \{ \psi_j : 0 \leq j \leq r \} \) and the associated trigonometric polynomials (or masks) \( \{ \hat{a}_j : 0 \leq j \leq r \} \) be constructed from the UEP. Then \( X(\Psi) \) defined by (2.3) is a tight frame of \( L_2(\mathcal{M}) \) provided that the decay conditions (2.4) and (2.5) are satisfied and \( |\hat{a}_0(\xi) - 1| \leq C|\xi| \) for \( \xi \) near the origin.

**Remark 2.1.** By Theorem 2.2, the second condition of the UEP is not needed for \( X(\Psi) \) defined by (2.3) to form a tight frame for \( L_2(\mathcal{M}) \). This is because the system \( X(\Psi) \) is entirely translation invariant. The second condition of the UEP is to make sure aliasing can be canceled, which is not needed for a translation invariant wavelet system.

### 2.3. Examples of Tight Wavelet Frames on \( \mathcal{M} \).

Corollary 2.1 tells us that, under suitable conditions, tight wavelet frames constructed from the UEP can generate tight wavelet frames for \( L_2(\mathcal{M}) \). This leads us to a huge collection of tight wavelet frames, because in the literature of wavelet frames, numerous sets of masks with various different properties are constructed based on the UEP [1, 62, 63, 68–74]. Here, we shall focus on the sets of masks constructed from B-splines in [1].

Consider B-splines of order \( r \) with \( r \geq 1 \). The corresponding refinement mask \( \hat{a}_0(\xi) = e^{-ir\xi^2} \cos^r(\xi/2) \) with \( r = 0 \) when \( r \) is even and \( r = 1 \) when \( r \) is odd. We define \( r \) wavelet masks as

\[
\hat{a}_j(\xi) = -i^j e^{-ir\xi^2} \sqrt{\frac{r}{j}} \sin^j(\xi/2) \cos^r-j(\xi/2), \quad 1 \leq j \leq r.
\]

It is easy to check that (2.8) is satisfied:

\[
\sum_{j=0}^{r} |\hat{a}_j(\xi)|^2 = (\cos^2(\xi/2) + \sin^2(\xi/2))^r = 1.
\]

Therefore, by Theorem 2.2, the system \( X(\Psi) \) generated by the \( r \) framelets defined by

\[
\hat{\psi}_j = -i^j e^{-ir\xi^2} \sqrt{\frac{r}{j}} \frac{\cos^r-j(\xi/4)\sin^{r+j}(\xi/4)}{(\xi/4)^r}, \quad 1 \leq j \leq r,
\]

forms a tight frame for \( L_2(\mathcal{M}) \) for all \( r \geq \frac{2m-1}{2} \) with \( m \) the dimension of the manifold \( \mathcal{M} \).

**Remark 2.2.** When the dimension of \( \mathcal{M} \) is 2, i.e. \( m = 2 \), the framelets \( \Psi \) given by (2.11) for all \( r \geq 1 \) can generate a tight wavelet frames for \( L_2(\mathcal{M}) \). However, when \( m \geq 3 \), according to the assumption (2.4), only the wavelet frame system generated by higher order B-splines can form a tight frame. Of course, this assumption may be weakened so that the systems generated by low order B-splines also form a tight frame for \( L_2(\mathcal{M}) \). In the next section, we shall show that in the discrete setting, the set of masks (2.10) for any \( r \geq 1 \) always generates a discrete tight frame for all functions on graphs. More generally, any set of masks \( \{ a_j : 0 \leq j \leq r \} \) satisfying (2.8) can generate a discrete tight wavelet frame on graphs (see Theorem 3.1).

### 3. Discrete Tight Wavelet Frame Transforms

#### 3.1. Motivations.

Let \( h_{j,n}(y) = (f, \psi_{j,n,y}^M) \). Observe that

\[
\hat{h}_{j,n-1}[p] = \hat{f}[p] \hat{\psi}_{j}^{*}(2^{-n+1}\lambda_p)
\]

\[
= \hat{f}[p] \hat{a}_j^{*}(2^{-n}\lambda_p) \hat{\phi}^*(2^{-n}\lambda_p)
\]

\[
= \hat{a}_j(2^{-n}\lambda_p) \hat{h}_{0,n}[p].
\]

This shows that \( h_{j,n-1} \), i.e. the continuous wavelet frame coefficients at level \( n-1 \) and band \( j \), can be obtained from \( h_{0,n} \), i.e. the low frequency coefficients at level \( n \), by “convolving” \( h_{0,n} \) with
the masks \( a_j \). Therefore, we can understand discrete function \( f_d \) on a graph (which is a certain discretization of \( \mathcal{M} \)) as a sampling of the underlying function \( f \) by sampling \( h_{0,n}(y) \) at certain scale \( n > 0 \). Then, the discrete wavelet frame transforms can be simply defined as “convolutions” of \( f_d \) with filters \( \{a_j\} \).

3.2. Discrete Tight Wavelet Frame Transforms on Graph \( G = \{E, V, w\} \). Let \( V = \{v_k \in \mathcal{M} : k = 1, \ldots, K\} \) be a discretization of the manifold \( \mathcal{M} \) and \( w : E \mapsto \mathbb{R}^+ \) the weight function. In this paper, we choose the following commonly used weight function

\[
w(v_m, v_n) = e^{-\|v_m - v_n\|^2/\sigma}, \quad \sigma > 0.
\]

Let \( A = (a_{m,n}) \) be the adjacency matrix

\[
a_{m,n} = \begin{cases} w(v_m, v_n) & \text{if } v_m \text{ and } v_n \text{ are connected by an edge in } E \\ 0 & \text{otherwise}, \end{cases}
\]

and \( D = \text{diag}\{d[1], d[2], \ldots, d[K]\} \) where \( d[m] \) is the degree of node \( v_m \) defined by \( d[m] = \sum_n a_{n,m} \).

Let \( \mathcal{L}_{nm} \) be the (unnormalized) graph Laplacian, which takes the following form

\[
\mathcal{L}_{nm} = D - A.
\]

In the literature, normalized graph Laplacians were also considered

\[
\mathcal{L}_{nm} = I - D^{-1}A \quad \text{or} \quad \mathcal{L}_{nm} = I - D^{-1/2}AD^{1/2}.
\]

Here, we shall use \( \mathcal{L} \) to denote one of the above three graph Laplacians. In our numerical computations, we will use the unnormalized graph Laplacian. The consistency of the graph Laplacian to the Laplace-Beltrami operator was studied in [3–5].

Denote \( \{(\lambda_p, u_p)\}_{p=0}^{K-1} \) the set of pairs of eigenvalues and eigenfunctions of \( \mathcal{L} \). Assuming the graph is connected, then we have \( 0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{K-1} \). The eigenfunctions for an orthonormal basis for all functions on the graph:

\[
\langle u_p, u_{p'} \rangle = \sum_{n=1}^{K} u_p[n]u_{p'}[n] = \delta_{p,p'}.
\]

Let \( f_d : V \mapsto \mathbb{R} \) be a function on the graph \( G \). Then its Fourier transform is given by

\[
\hat{f}_d[p] = \sum_{n=1}^{K} f[n]u_p[n].
\]

**Transition from continuum to discrete.** Suppose \( G \) is a certain discretization of \( \mathcal{M} \) with \( V \subset \mathcal{M} \). Given a function \( f_d \) on graph \( G \), we assume that \( f_d \) is sampled from the underlying function \( f : \mathcal{M} \mapsto \mathbb{R} \) by

\[
f_d[k] = \langle f, \phi_{J,v_k} \rangle, \quad v_k \in V,
\]

with some dilation scale \( J \) such that \( \lambda_{\text{max}} = \lambda_{K-1} = 2^J \pi \). Note that the scale \( J \) is selected such that \( 2^{-J}\lambda_p \in [0, \pi] \) for \( 0 \leq p \leq K - 1 \). After this point, since we will entirely focus on the discrete graph data \( f_d \), we shall drop the subscript “\( d \)” and simply denote it as \( f \).

Based on our earlier observation (3.1), we define the **discrete tight wavelet frame decomposition** as

\[
Wf = \{W_{j,l}f : (j, l) \in \mathbb{B} \}
\]

with

\[
(3.2) \quad \mathbb{B} = \{(1, 1), (2, 1), \ldots, (r, 1), (1, 2, \ldots, (r, L) \} \cup \{(0, L)\}
\]

and

\[
(3.3) \quad \widehat{W_{j,l}f}[p] = \begin{cases} \hat{a}_j^s(2^{-J}\lambda_p)\hat{f}[p] & l = 1, \\ \hat{a}_j^l(2^{-J-l+1}\lambda_p)\hat{a}_0^s(2^{-J-l+2}\lambda_p) \cdots \hat{a}_0^s(2^{-J}\lambda_p)\hat{f}[p] & 2 \leq l \leq L. \end{cases}
\]
The index $j$ denotes the band of the transform with $j = 0$ the low frequency component and $1 \leq j \leq r$ the high frequency components. The index $1 \leq l \leq L$ denotes levels of the transform.

Given a graph function $f$, let $\alpha = Wf = \{\alpha_{j,l} : (j,l) \in B\}$ be its tight wavelet frame coefficients. We denote the discrete tight wavelet frame transforms $W^\top$ as $W^\top \alpha$, which is defined by

\[(3.4) \quad W^\top \alpha[p] = \sum_{l=L}^{1} \sum_{j=0}^{r} \hat{a}_j(2^{-J-J-l+1} \lambda_p) \alpha_{j,l}[p].\]

Note from the definition of $B$ in (3.2), the only low frequency component we store for an $L$-level decomposition is $\alpha_{0,L}$. Therefore, the values $\alpha_{0,l}$ used in (3.4) for $1 \leq l < L$ are computed iterative from

\[\alpha_{0,l} = \sum_{j=0}^{r} \hat{a}_j(2^{-J-J-l+1} \lambda_p) \alpha_{j,l+1}, \quad l = L - 1, L - 2, \ldots 1.\]

Perfect reconstruction from $\alpha$ to $f$ through $W^\top$ can be easily verified as long as the masks $\{a_j : 0 \leq j \leq r\}$ satisfy the condition (2.8) in Theorem 2.2. Then, we have the following theorem indicating that $\{a_j : 0 \leq j \leq r\}$ generates a discrete tight wavelet frame on graphs.

**Theorem 3.1.** Given a set of masks $\{a_j : 0 \leq j \leq r\} \subset \ell_2(\mathbb{Z})$, suppose condition (2.8) is satisfied. Then, the discrete tight wavelet frame transforms $W$ and $W^\top$ defined on $G = \{E,V,w\}$ by (3.3) and (3.4) satisfy

\[W^\top W f = f, \quad \text{for all } f : V \rightarrow \mathbb{R}.\]

We present some sets of masks satisfying (2.8), and hence generate discrete tight wavelet frames on graphs according to Theorem 3.1. Note that these masks are modified from the those given by (2.10) by removing the complex factors.

**Example 3.1.** We present three sets of masks that satisfy (2.8) and hence generate discrete tight wavelet frames on graphs. We shall refer to them as the “Haar”, “linear” and “cubic” tight wavelet frame systems, and the masks as the “Haar”, “linear” and “cubic” masks. These masks slice the spectrum in a different but similar way as shown in Figure 1.

(1) **Haar.**

\[\hat{a}_0(\xi) = \cos(\xi/2) \quad \text{and} \quad \hat{a}_1(\xi) = \sin(\xi/2).\]

(2) **Linear.**

\[\hat{a}_0(\xi) = \cos^2(\xi/2), \quad \hat{a}_1(\xi) = \frac{1}{\sqrt{2}} \sin(\xi) \quad \text{and} \quad \hat{a}_2(\xi) = \sin^2(\xi/2).\]

(3) **Cubic.**

\[\hat{a}_0(\xi) = \cos^3(\xi/2), \quad \hat{a}_1(\xi) = \sqrt{3} \sin(\xi/2) \cos^2(\xi/2) \quad \hat{a}_2(\xi) = \sqrt{3} \sin^2(\xi/2) \cos(\xi/2) \quad \text{and} \quad \hat{a}_3(\xi) = \sin^3(\xi/2).\]

### 3.3. Fast Tight Wavelet Frame Transform (FTWFT) on Graphs.

The discrete tight wavelet frame transforms given by (3.3) and (3.4) require the full set of eigenvectors and eigenvalues of the graph Laplacian, which is computationally expensive to find for large graphs. A solution to such computation challenge is to use polynomial approximation of the masks, such as the Chebyshev polynomials used by [20], so that eigenvalue decomposition of the graph Laplacian is not needed. Note that the masks $\hat{a}_j(\xi)$ that we use, as well as those constructed in the literature from the UEP, are trigonometric polynomials (such as (2.10)). Therefore, $\hat{a}_j(\xi)$ can be accurately approximated by low-degree Chebyshev polynomials (see e.g. [75]) which significantly reduces the computation cost of the decomposition and reconstruction algorithms. In this section, we describe the details of the fast tight wavelet frame transform (FTWFT) based on polynomial approximation.
Figure 1. Plots of $\hat{a}_j(\xi)$ for $\xi \in [0, \pi]$. Plot (a) presents the “Haar” masks with $\hat{a}_0$ in blue and $\hat{a}_1$ in green. Plot (b) presents the “linear” masks with $\hat{a}_0$ in blue, $\hat{a}_1$ in green and $\hat{a}_2$ in red. Plot (c) presents the “cubic” masks with $\hat{a}_0$ in blue, $\hat{a}_1$ in green, $\hat{a}_2$ in red and $\hat{a}_3$ in light blue.

We start with approximating $g(\xi)$ with $\xi \in [0, \pi]$ using Chebyshev polynomials $\{T_k(\xi) : k = 0, 1, \ldots\}$. Recall that the Chebyshev polynomials on $[0, \pi]$ are defined iteratively by

$$T_0 = 1, \quad T_1(\xi) = \frac{\xi - \pi/2}{\pi/2}, \quad T_k(\xi) = \frac{4}{\pi} (\xi - \pi/2) T_{k-1}(\xi) - T_{k-2}(\xi), \text{ for } k = 2, 3, \ldots.$$  

Then, given $g(\xi)$, we have the following expansion

$$g(\xi) = \frac{1}{2} c_0 + \sum_{k=1}^{\infty} c_k T_k,$$

where

$$c_k = \frac{2}{\pi} \int_0^\pi \cos(k\theta) g \left( \frac{\pi}{2} (\cos(\theta) + 1) \right) d\theta.$$  

When $g(\xi)$ is smooth, we can use partial sums to accurately approximate $g(\xi)$:

$$g(\xi) \approx T^n(\xi) = \frac{1}{2} c_0 + \sum_{k=1}^{n-1} c_k T_k.$$  

We denote the Chebyshev approximation of $\hat{a}_j(\xi)$ as

$$\tilde{a}_j(\xi) \approx T^n_j(\xi) = \frac{1}{2} c_{j,0} + \sum_{k=1}^{n-1} c_{j,k} T_k,$$

where

$$c_{j,k} = \frac{2}{\pi} \int_0^\pi \cos(k\theta) \tilde{a}_j \left( \frac{\pi}{2} (\cos(\theta) + 1) \right) d\theta.$$  

We denote the Chebyshev approximation of $\hat{a}^*_j$ as $T^{n*}_j$. Since the masks $\{\hat{a}_j\}$ we will be using are real-valued (those in Example 3.1), we have $T^n_j = T^{n*}_j$.

Let us see how to speed up the transformation (3.3) using the above approximation. The Laplacian $L$ admits the eigenvalue decomposition $L = U \Lambda U^\top$ where $\Lambda = \text{diag}\{\lambda_0, \lambda_1, \ldots, \lambda_{K-1}\}$ and columns of $U$ are the eigenvectors. Then we can rewrite (3.3) in the following matrix form in physical domain:

$$W_{j,l} f = \begin{cases} U \tilde{a}^*_j (2^{-J}\Lambda)^{l} U^\top f & l = 1, \\ U \tilde{a}^*_j (2^{-J-l+1}\Lambda) \hat{a}_0^*(2^{-J-l+2}\Lambda) \cdots \hat{a}_0^*(2^{-J}\Lambda) U^\top f & l \geq 2. \end{cases}$$  

If we substitute the approximation $\tilde{a}_j(\xi) \approx T^n_j(\xi)$ in (3.6) and use the fact that $T^n_j$ are polynomials, we obtain the FTWFT:
Fast Tight Wavelet Frame Transform (FTWFT). $Wf = \{W_{jl}f : (j,l) \in B\}$ where

$$W_{jl}f = \begin{cases} T_j^n(2^{-j}\mathcal{L})f & l = 1, \\ T_j^n(2^{-j-1}\mathcal{L})T_0^n(2^{-j-2}\mathcal{L})\cdots T_0^n(2^{-j}\mathcal{L})f & l \geq 2. \end{cases}$$

Reconstruction transform $W^T$ can be defined similarly and we have $W^TW \approx I$. Note that, for computational efficiency, the operations $T_j^n(s\mathcal{L})f$ for decomposition and $T_j^n(s\mathcal{L})f$ for reconstruction are computed via the iterative definition of the Chebyshev polynomial (3.5). Therefore, in FTWFT, only (sparse) matrix-vector multiplications are involved.

**Remark 3.1.** As we mentioned earlier, since the masks $\hat{\alpha}_j(\xi)$ are trigonometric polynomials, they can be accurately approximated by low-degree Chebyshev polynomials (see Table 1 for example). In our simulations, we choose $n = 8$ which is sufficient for the applications we considered. Note that, if a higher order B-spline tight wavelet frame system is used, Chebyshev polynomials with a higher degree may be needed to achieve a given approximation error. However, just as in image processing, only the lower order systems are mostly used because they provide a better balance between quality and computation efficiency.

### 3.4. Numerical Simulations of FTWFT.

Given a graph $G = \{E,V,w\}$ with $|V| = K$, we shall use the unnormalized graph Laplacian $\mathcal{L} = D - A$ as an example. We compute only the largest eigenvalue $\lambda_{K-1}$, and choose the initial dilation scale as $J = \log_2(\frac{\lambda_{K-1}}{\pi})$.

In our simulations, we choose the “linear” tight wavelet frame system given in Example 3.1, i.e.

$$\hat{\alpha}_0(\xi) = \cos^2(\xi/2), \quad \hat{\alpha}_1(\xi) = \frac{1}{\sqrt{2}}\sin(\xi) \quad \text{and} \quad \hat{\alpha}_2(\xi) = \sin^2(\xi/2).$$

We present approximation errors of $T_j^n$ to $\hat{\alpha}_j$ for several values of $n$ in the following Table 1, where one can easily see that the $\hat{\alpha}_j$ can be accurately approximated by low-degree Chebyshev polynomials. In our simulations, we fix $n = 8$, i.e. we use Chebyshev polynomials of degree 7 to approximate the masks given in Example 3.1.

**Table 1.** Approximation error $\|T_j^n - \hat{\alpha}_j\|_{\infty}$ with $\hat{\alpha}_j$ the ‘linear” masks given in Example 3.1.

<table>
<thead>
<tr>
<th>Errors</th>
<th>$\hat{\alpha}_0$</th>
<th>$\hat{\alpha}_1$</th>
<th>$\hat{\alpha}_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 4$</td>
<td>2.733x10^{-3}</td>
<td>2.022x10^{-2}</td>
<td>2.273x10^{-3}</td>
</tr>
<tr>
<td>$n = 5$</td>
<td>2.273x10^{-3}</td>
<td>4.267x10^{-4}</td>
<td>2.273x10^{-3}</td>
</tr>
<tr>
<td>$n = 6$</td>
<td>3.417x10^{-5}</td>
<td>4.267x10^{-4}</td>
<td>3.417x10^{-5}</td>
</tr>
<tr>
<td>$n = 7$</td>
<td>3.417x10^{-5}</td>
<td>4.775x10^{-6}</td>
<td>3.417x10^{-5}</td>
</tr>
<tr>
<td>$n = 8$</td>
<td>3.762x10^{-7}</td>
<td>4.775x10^{-6}</td>
<td>3.762x10^{-7}</td>
</tr>
</tbody>
</table>

To illustrate the effects of the FTWFT using the “linear” masks in Example 3.1, we consider a graph $G = \{E,V,w\}$, with $V$ being sampled from a unit sphere. The number of vertices in $V$ is 16,728. The adjacency matrix $A$ is generated using the weight function $w(v_i,v_j) = e^{-\|v_i-v_j\|^2/\sigma}$ for $v_i,v_j \in V$. In our experiments, we fix $\sigma = 10$. Then we threshold $A$ to limit the number of nearest neighbors of each vertex to 10. The functions $f : V \mapsto \mathbb{R}$ are generated by mapping two images, “Slope” and “Eric”, onto the graph $G$ (see Figure 2). We perform 4 levels of FTWFT and use $n = 8$ for the Chebyshev polynomial approximation of the masks. The tight wavelet frame coefficients $W_{jl}f$ for $0 \leq j \leq 2$ and $1 \leq l \leq 4$ are shown in Figure 3 and 4. It is worth noticing that the high frequency coefficients $W_{jl}f$ for $0 \leq j \leq 2$ with $j = 1,2$ are very sparse, which is consistent with what we normally observe from wavelet frame coefficients of images. In other words, the proposed tight wavelet frames provide a *sparse* representation for graph functions. Sparsity is crucial for many applications. In the next section, we shall consider the applications to denoising and semi-supervised clustering utilizing the sparse representation provided by our FTWFT.
We finally note that the reconstruction errors from the 4-level decomposition shown in Figure 3 and 4 are $4.432 \times 10^{-6}$ and $4.365 \times 10^{-6}$ respectively in $\ell_\infty$-norm. The total computation time is 0.417 seconds on a laptop with Intel(R) Core(TM) i7-4600U CPU (2.1GHz & 2.7GHz) and 8GB RAM running Windows 7 (64-bit).

**Figure 2.** This figure shows two images (first row), “Slope” and “Eric”, that are mapped to the graph of the unit sphere to for the graph data $f$ (second row).

**Figure 3.** This figure shows the images of the tight wavelet frame coefficients $W_{j,l}f$ for $0 \leq j \leq 2$ (row 1-3) and $1 \leq l \leq 4$ (column 1-4).
Figure 4. This figure shows the images of the tight wavelet frame coefficients $W_{j,l}f$ for $0 \leq j \leq 2$ (row 1-3) and $1 \leq l \leq 4$ (column 1-4).

4. Applications to Denoising and Semi-Supervised Classification

In the previous section, we discussed how we can define sparse representations on graphs by tight wavelet frames, and how to efficiently compute the transforms using polynomial approximation of the masks. There are many potential applications of the proposed representation. For example, one may consider a general linear inverse problem on graphs whenever the solution we seek is sparse under the proposed tight wavelet frame transform. In this section, we shall focus on the applications of the tight wavelet frame transforms to graph data denoising and semi-supervised clustering. Note that the focus of this paper is on some theoretical aspects of the proposed tight wavelet frames on graphs and the design of fast transforms. This section is to show the great potential of the proposed transform in some applications. A more comprehensive study on applications such as clustering is obviously needed. For example, the proposed clustering model in this section needs to be tested on some benchmark data sets and compared with existing clustering algorithms. We postpone such study to a separate paper.

4.1. Denoising. Denoising on graphs may not be a problem as important as image denoising. However, a good representation together with the modelling based on such representation should be robust to noise in order for them to be useful in practice. Therefore, denoising is a necessary test problem. On the other hand, if the graph data collected in practice is corrupted by a certain type of noise that needs to be removed, denoising is a helpful step before any further data analysis. Given a graph $G = \{E, V, w\}$ with $|V| = K$, and a graph function $\bar{u} : V \rightarrow \mathbb{R}$, the observed data or the data we collect is $f = \bar{u} + \eta$, where $\eta$ is some noise, which shall be assumed to be Gaussian white noise.
Given a graph data $f$, its Euclidean $\ell_p$-norm ($p = 1$ or $2$) is defined by
\[
\|f\|_p = \left( \sum_{k=1}^{K} |f[k]|^p \right)^{\frac{1}{p}},
\]
and its graph-$\ell_p$-norm $\ell_{p,G}$ is defined by
\[
\|f\|_{p,G} = \left( \sum_{k=1}^{K} |f[k]|^p d[k] \right)^{\frac{1}{p}},
\]
de\textbf{were} $d[k]$ is the degree of the node $v_k \in V$. Let $D = \text{diag}\{d[1], d[2], \ldots, d[K]\}$. Obviously, we have $\|D^{1/p} f\|_p = \|f\|_{p,G}$ with $D^s = \text{diag}\{d^s[1], d^s[2], \ldots, d^s[K]\}$ for $s \in \mathbb{R}$.

Now, we propose to use the following analysis based model for denoising, which was originally proposed in image processing [16, 76, 77]:
\[
(4.1) \quad \min_u \|\nu \cdot W u\|_{1,G} + \frac{1}{2} \|u - f\|_{2,G}^2,
\]
where $W$ is the tight wavelet frame transform (3.3) and
\[
\|\nu \cdot W u\|_{1,G} = \sum_{(j,l) \in \mathbb{B}} \nu_{j,l} \|W_{j,l} u\|_{1,G}
\]
with $\nu_{j,l} \geq 0$ being the tuning parameters and $\mathbb{B}$ defined in (3.2). In our simulations for denoising, we choose
\[
(4.2) \quad \nu_{j,l} = \begin{cases} 
4^{-l+1}\nu & \text{for } j \neq 0, 1 \leq l \leq L, \\
0 & \text{for } j = 0, l = L,
\end{cases}
\]
where $\lambda > 0$ is a scalar tuning parameter. Note that model (4.1) can be solved efficiently using the split Bregman algorithm [16, 78], which is also equivalent to the alternating direction method of multipliers (ADMM) [79–81]. Applying the derivation of split Bregman algorithm to model (4.1), we have
\[
(4.3) \quad \begin{cases} 
\alpha^k = \arg \min_u \frac{1}{2} \|u - f\|_{2,G}^2 + \frac{\mu}{2} \|W u - d_k + b^k\|_{2}^2, \\
\beta^k = \arg \min_d \|\nu \cdot d\|_{1,G} + \frac{\mu}{2} \|d - (W u^k + b^k)\|_{2}^2, \\
b^{k+1} = b^k + W u^{k+1} + d^{k+1}.
\end{cases}
\]
Both of the sub-optimization problems in (4.3) have closed-form solutions. Replacing them by their closed-form solutions, we obtain the following denoising algorithm solving (4.1):
\[
(4.4) \quad \begin{cases} 
\alpha^k = (D + \mu I)^{-1} (D f + \mu W^\top (d^k - b^k)), \\
\beta^k = S_{\nu \cdot D} (W u^k + b^k), \\
b^{k+1} = b^k + W u^{k+1} + d^{k+1},
\end{cases}
\]
where $(D + \mu I)^{-1} = \text{diag}\{\frac{1}{d[1]+\mu}, \ldots, \frac{1}{d[K]+\mu}\}$, $\nu_{j,l} D = \{\nu_{j,l} D : (j,l) \in \mathbb{B}\}$ and the soft-thresholding operator $S$ is defined pointwise as $S_a(y) = \frac{a}{|y|} \max\{|y| - a, 0\}$. In our numerical implementation, we choose zero initialization of the algorithm, i.e. $u^0 = 0$ and $b^0 = d^0 = 0$.

**Remark 4.1.** We specifically choose the Euclidean $\ell_2$-norm in the second term of the first two equations of (4.3), which corresponds to the choice of the Euclidean linear and augmented terms in the derivation of the split Bregman algorithm using the augmented Lagrangian method (see [79–81] or [24, Chapter 4] for details). Another choice, perhaps a more straightforward choice, is to use $\|\cdot\|_{2,G}^2$. However, this will lead to an inversion of the non-diagonal matrix $D + W^\top D W$ in solving for $u^{k+1}$, where $D$ is a $(rL+1)K \times (rL+1)K$ diagonal matrix given by $D = \text{diag}\{D, D, \ldots, D\}$. Therefore, for computation efficiency, we shall use the algorithm (4.4) to solve (4.1).
The graph we consider is the one formed by sampling a unit sphere as described in Section 3.4. We select four examples of graph data as the noise-free data $\bar{u}$, where “Slope” and “Eric” were given in Figure 2, and the other two graph data, “Barbara” and “Lena”, are generated by mapping these two widely used images on the discrete sphere. Gaussian white noise is added to $\bar{u}$ with a standard deviation 0.05. The noise-free and noisy data are visualized in Figure 5, first and second row respectively.

Denoising results by algorithm (4.4) using the “linear” tight wavelet frame system in Example 3.1 are shown in the third row of Figure 5. The level of decomposition is chosen to be 1. We fixed the number of iteration to be 100, and the computation time for all of the examples is 13.7 seconds. The parameters are manually adjusted for optimal denoising quality. To see the denoising effect using different tight wavelet frame systems, we list the denoising errors in Table 2 using the “Haar”, “linear” and “cubic” tight wavelet frame systems. The denoising error is defined by

$$\text{Denoising Error} = \frac{\|u - \bar{u}\|_2}{\|\bar{u}\|_2},$$

where $u$ is the denoised data and $\bar{u}$ is the original data. From Table 2, we can see that “linear” produces better results than “Haar”, while “cubic” is comparable with “linear”. However, FWTFT using “Cubic” masks is slower than “Haar” and “Linear”. Therefore, the “linear” tight wavelet frame system on graphs seems to be a good balance between computational cost and reconstruction quality for denoising problems. However, when the graph data can be well approximated by a piecewise constant function, such as “Slope”, the “Haar” system seems to be the best choice.

Finally, we note that if one has a linear inverse problem to solve on graph, i.e. $f = A\bar{u} + \eta$, then we can change the second term of (4.1) to $\frac{1}{2}\|Au - f\|_2^2_G$ (see e.g. [16, 17]). Also, if the additive noise $\eta$ is non-Gaussian, such as Poisson, one may modify the norm of the second term of (4.1) to properly incorporate the correct noise statistics (see e.g. [82–84]).

![Figure 5](image-url)
Table 2. Errors of denoising. Computation time for 100 iterations is 13.7 seconds.

<table>
<thead>
<tr>
<th>Errors</th>
<th>“Haar”</th>
<th>“Linear”</th>
<th>“Cubic”</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Slope”</td>
<td>0.04279</td>
<td>0.04224</td>
<td>0.04307</td>
</tr>
<tr>
<td>“Eric”</td>
<td>0.02995</td>
<td>0.02872</td>
<td>0.02886</td>
</tr>
<tr>
<td>“Barbara”</td>
<td>0.07587</td>
<td>0.07419</td>
<td>0.07376</td>
</tr>
<tr>
<td>“Lena”</td>
<td>0.07972</td>
<td>0.07864</td>
<td>0.07829</td>
</tr>
</tbody>
</table>

4.2. Semi-Supervised Clustering. Here, we consider semi-supervised clustering, where the labeling of a small set of the data is provided in advance. We introduce an optimization model based on the proposed tight wavelet frame transform on graphs along with a fast numerical algorithm. The proposed model and algorithm are motivated by the earlier work on wavelet frame based image segmentation [85] and surface reconstruction [86], as well as the variational frameworks for image segmentation [87, 88] and graph clustering [89, 90]. We shall focus on the 2-classes clustering problem. Generalization to multi-class clustering can be done following a similar idea as in multiphase image segmentation [91–93].

Suppose the given graph $G = \{E, V, w\}$ contains two clusters. Let $|V| = K$ and $\Gamma \subset \{1, 2, \ldots, K\}$ be the set of labels where, for $i \in \Gamma$, we know which cluster $v_i$ belongs to. Let $\Gamma = \Gamma_0 \cup \Gamma_1$ where $\Gamma_0$ is the index set for cluster 1 and $\Gamma_1$ for cluster 2. Define $f : \Gamma \mapsto \mathbb{R}$ as

$$f[i] = \begin{cases} 0 & \text{for } i \in \Gamma_0, \\ 1 & \text{for } i \in \Gamma_1. \end{cases}$$

Our objective is to recover a function $u : V \mapsto [0, 1]$ with $u|_\Gamma \approx f$, such that the two sets $\{i \mid u[i] \geq \alpha\}$ and $\{i \mid u[i] < \alpha\}$ provide a good clustering of the given graph for some $\alpha \in (0, 1)$. To find such $u$, we propose the following model

$$\min_{u \in [0,1]} \| \nu \cdot Wu \|_{1,G} + \frac{1}{2} \| u|_\Gamma - f \|_{2,G},$$

where the first term impose a regularization of the level sets of $u$, while the second is the fidelity term making sure that $u|_\Gamma \approx f$. The parameter $\nu$ is chosen as in (4.2). It has been shown in [17–19] that $Wu$ is a discrete approximation to $D\nu$, in particular $\nabla u$, in the context of variational and nonlinear PDE modeling in image restoration. Furthermore, the discretization provided by $W$ is better than some standard finite difference approximations for image restoration [17, 18], image segmentation [85] and surface reconstruction [86]. Therefore, we may think $Wu$ in (4.5) as a certain approximation of $\nabla$, or a generic differential operator, on the underlying manifold from which the graph is sampled. Of course, this needs to be rigorously justified. However, based on numerical experiments such as in Figure 3 and 4, one can see that $Wu$ does perform similarly as differential operators.

To solve (4.5), we apply the derivation of the split Bregman algorithm again, and obtain the following algorithm

$$\begin{align*}
u^{k+1} &= \arg \min_{\nu \in [0,1]} \frac{1}{2}\| u|_\Gamma - f \|_2^2 + \frac{\mu}{2} \| Wu - d^k + b^k \|_2^2, \\
d^{k+1} &= S_{\nu \Gamma,\mu}(Wu^{k+1} + b^k), \\
b^{k+1} &= b^k + Wu^{k+1} + d^{k+1},
\end{align*}$$

where the subproblem for $u^{k+1}$ is approximated by the following steps

$$\begin{align*}
\left( u^{k+\frac{1}{2}} \right)|_{\Gamma^c} &= (W^T(d^k - b^k))|_{\Gamma^c}, \\
\left( u^{k+\frac{1}{2}} \right)|_{\Gamma} &= (D\Gamma + \mu I)^{-1} \left( f + \mu (W^T(d^k - b^k))|_{\Gamma} \right), \\
u^{k+1} &= \max \left\{ \min \left\{ u^{k+\frac{1}{2}}, 1 \right\}, 0 \right\},
\end{align*}$$
where $D_{\Gamma} = \text{diag}\{d[k] : k \in \Gamma\}$.

Now, we test algorithm (4.6) to a synthetic graph known as the “two moons” first used in [94]. This data set is constructed from two half unit circles in $\mathbb{R}^2$ with centers at (0, 0) and (1, 0.5). For each circle, 1000 points are uniformly chosen and lifted to $\mathbb{R}^{100}$ by adding i.i.d. Gaussian white noise with variance $= 0.02$ to each coordinate. The graph is formed from the point set using the exact same way as we discussed earlier in Section 3.4.

Since the solution $u$ of (4.5) is ideally binary, it is more suitable to use the “Haar” tight wavelet frame system given in Example 3.1. The level of decomposition is chosen to be 1. The number of iteration is fixed to be 500, and the computation time for all of the cases is 14.7 seconds. Parameters are properly tuned for optimal clustering results. Algorithm 4.6 is initialized by

$$u^0[k] = \begin{cases} 0, & u_1[k] > 0, \\ 1, & u_1[k] \leq 0, \end{cases}$$

where $u_1$ is the second eigenvector of $\mathcal{L}$, and $d^0 = b^0 = W u^0$.

Figure 6 shows the ground truth of the two clusters and the clustering result by our algorithm assuming 10% of the labels of the vertices (chosen randomly) are known. More clustering results assuming different percentages of known labels are summarized in Table 3. Note that these results are only slightly worse than the very latest results obtained by [90]. This shows great potential of the proposed sparse representation and the clustering model (4.5). More serious numerical studies are necessary and will be postponed to a separate paper.

**Ground Truth**

![Ground Truth](image1)

**Estimated with error = 4.5556%**

![Estimated](image2)

**Figure 6.** This figure shows the ground truth of the two clusters (left) and the estimated clusters (right) with error rate 4.5556%.

**Table 3.** Clustering errors of the dataset “two moons”. Computation time for 500 iterations is 14.7 seconds.

<table>
<thead>
<tr>
<th>Known Labels (%)</th>
<th>15</th>
<th>10</th>
<th>5</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error (%)</td>
<td>4.1765</td>
<td>4.5556</td>
<td>5.9474</td>
<td>6.3402</td>
</tr>
</tbody>
</table>

**References**


