Title

Efficient Approximation and Denoising of Graph Signals Using the Multiscale Basis Dictionaries

Journal

IEEE Transactions on Signal and Information Processing over Networks, 3(3)

ISSN

2373-7778

Authors

Irion, Jeff
Saito, Naoki

Publication Date

2017

DOI

10.1109/tsipn.2016.2632039

Peer reviewed

Efficient Approximation and Denoising of Graph Signals Using the Multiscale Basis Dictionaries

Jeff Irion and Naoki Saito, *Senior Member, IEEE*

***Abstract*—We propose methods to efficiently approximate and denoise signals sampled on the nodes of graphs using our overcomplete multiscale transforms/basis dictionaries for such graph signals: the hierarchical graph Laplacian eigen transform (HGLET) and the generalized Haar–Walsh transform (GHWT). These can be viewed as generalizations of the hierarchical discrete cosine transform and the Haar-Walsh wavelet packet transform, respectively, from regularly sampled signals to graph signals. Both of these transforms generate dictionaries containing an immense number of choosable bases, and in order to select a particular basis most suitable for a given task, we have generalized the best basis algorithm from classical signal processing. After briefly reviewing these transforms and the best basis algorithm, we precisely prove their efficiency in approximating graph signals belonging to discrete analogs of the space of Hölder continuous functions and the Besov spaces. Then, we validate their effectiveness with numerical experiments on real datasets, in which we compare them against other graph-based transforms. Building upon this approximation efficiency of our transforms, we devise a signal denoising method using the HGLET and GHWT and again compare against other transforms. Our results clearly demonstrate the superiority of our transforms over those other transforms in terms of both approximation and denoising.**

***Index Terms*—Best basis selection, graph wavelets and wavelet packets, graph signal approximation and denoising, multiscale basis dictionaries on graphs.**

I. Introduction

In CLASSICAL signal processing, the signals considered possess simple, regular structures that are known a priori. Examples of such signals include audio, images, time series data, matrices, etc. All of these signals lie on regular grids, which makes it easy to exploit their underlying structure in order to analyze them. To this end, a number of highly successful tools have been developed, with wavelets being one of the crowning achievements.

Of course, as advancements in signal processing were being made, so too were advancements made in computing power. This made possible both the collection and processing of signals on a new domain: graphs. Here, a signal's structure is no longer confined to the equispaced, regularly connected domains of classical signal processing. Such freedom allows for much richer classes of signals to be considered and analyzed.

But this increased versatility does not come without challenges. Nearly all of the theory and tools developed for classical signals cannot be generalized easily, if at all, to signals on graphs.[1] Current methods must change and evolve, and new methods must be developed. However, many of the questions remain the same. How can we efficiently approximate a signal on a graph? How can we quantitatively describe a signal? How can we identify and remove noise from a signal on a graph? In this work, we present strategies for tackling these questions and more. Drawing motivation from concepts and techniques used in classical signal processing, we develop new tools and methods for analyzing signals on graphs which can rightly be viewed as generalizations of classical techniques.

The organization of this article is as follows. In Section II, we cover some basics of graph theory and recursive graph-partitioning. We briefly review some transforms for signals on graphs developed by other researchers, and then we provide an overview of our own HGLET and GHWT transforms. In Section III, we present theoretical and experimental results concerning the use of the HGLET and GHWT for approximation of signals on graphs. Then in Section IV, we demonstrate the effectiveness of our transforms for denoising. The methods and tools discussed herein are freely available in the MTSG Toolbox,[2] which includes scripts for recreating Fig. 2–8 and Table I. The experiments in this paper were performed on a personal laptop with a 2.20 GHz Intel Core i5-5200U CPU with 12.0 GB RAM.

II. Background

A. Graph Theory

Let $G = (V, E)$ be an undirected connected graph. Let $V = V(G) = \{v_1, v_2, \ldots, v_N\}$ denote the set of vertices (or

Manuscript received February 1, 2016; revised July 19, 2016 and October 5, 2016; accepted November 3, 2016. Date of publication; date of current version. This work was supported in part by Naoki Saito's ONR Grants N00014-12-1-0177 and N00014-16-1-2255, and NSF Grant DMS-1418779, and was conducted with Government support under Contract FA9550-11-C-0028 and awarded by the Department of Defense, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate Fellowship, 32 CFR 168a. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Venkatesh Saligrama. (*Corresponding author: Jeff Irion.*)

J. Irion was with the Department of Mathematics, University of California, Davis, CA 95616 USA. He is now with Bosch Research and Technology Center, Palo Alto, CA 94304 USA (e-mail: jlirion@ucdavis.edu).

N. Saito is with the Department of Mathematics, University of California, Davis, CA 95616 USA (e-mail: saito@math.ucdavis.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSIPN.2016.2632039

[1] It has been proposed in [1] that one can generalize the Fourier transform to the graph setting by using the Laplacian eigenvectors as a generalization of the Fourier basis. However, as explained in [2], it is a mistake to interpret the graph Laplacian eigenvalues and eigenvectors as the (squared) frequencies and the Fourier basis functions, respectively.

[2] https://github.com/JeffLIrion/MTSG_Toolbox

nodes) of the graph, where $N := |V(G)|$. For simplicity, we typically associate each vertex with its index and write i in place of v_i. $E = E(G) = \{e_1, e_2, \ldots, e_M\}$ is the set of edges, where each e_k connects two vertices i and j, and $M := |E(G)|$. In this article we consider only finite graphs (i.e., $M, N < \infty$). Moreover, we restrict to the case of simple graphs; that is, graphs without loops (an edge connecting a vertex to itself) and multiple edges (more than one edge connecting a pair of vertices i and j). We use $\boldsymbol{f} \in \mathbb{R}^N$ to denote a signal on G, and we define $\mathbf{1} := (1, \ldots, 1)^\mathsf{T} \in \mathbb{R}^N$.

We now discuss several matrices associated with graphs. The information in both V and E is captured by the *edge weight matrix* $W(G) \in \mathbb{R}^{N \times N}$, where $W_{ij} \geq 0$ is the edge weight between nodes i and j. In an unweighted graph, this is restricted to be either 0 or 1, depending on whether nodes i and j are connected, and we may refer to $W(G)$ as an *adjacency matrix*. In a weighted graph, W_{ij} indicates the affinity between i and j. In either case, since G is undirected, $W(G)$ is a symmetric matrix. We then define the *degree matrix* $D(G)$ as the diagonal matrix with entries $d_i = \sum_j W_{ij}$. With this in place, we are now able to define the *(unnormalized) Laplacian* matrix, *random-walk normalized Laplacian* matrix, and *symmetric normalized Laplacian* matrix, respectively, as

$$\begin{aligned} L(G) &:= D(G) - W(G) \\ L_{\mathrm{rw}}(G) &:= D(G)^{-1} L(G) \\ L_{\mathrm{sym}}(G) &:= D(G)^{-1/2} L(G) D(G)^{-1/2}. \end{aligned}$$

We use $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{N-1}$ to denote the sorted Laplacian eigenvalues and $\boldsymbol{\phi}_0, \boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_{N-1}$ to denote their corresponding eigenvectors, where the specific Laplacian matrix to which they refer will be clear from either context or superscripts.

These matrices have been studied extensively, and we now highlight three key properties (further information can be found in [3], [4]). First, both L and L_{sym} are symmetric matrices and therefore their eigenvectors form orthonormal bases for $\mathbb{R}^N$. Second, L_{rw} and L_{sym} have the same eigenvalues, and their eigenvectors are related in the following way:

$$\boldsymbol{\phi}_l^{\mathrm{rw}} = D(G)^{-1/2} \boldsymbol{\phi}_l^{\mathrm{sym}} \qquad l = 0, 1, \ldots, N-1. \tag{1}$$

From this, it is easily seen that the eigenvectors of L_{rw} are orthonormal with respect to the weighted inner product $\langle , \rangle_{D(G)}$; that is, $(\boldsymbol{\phi}_{l_1}^{\mathrm{rw}})^* D(G) \boldsymbol{\phi}_{l_2}^{\mathrm{rw}} = \delta_{l_1, l_2}$. Third, for all three matrices the smallest eigenvalue is zero, and for a connected graph all the other eigenvalues are strictly positive. Furthermore, for both L and L_{rw} the eigenvector associated to eigenvalue zero is the normalized constant vector: $\boldsymbol{\phi}_0 = \mathbf{1}/\sqrt{N}$ and $\boldsymbol{\phi}_0^{\mathrm{rw}} = \mathbf{1}/\sqrt{\sum_{i=1}^N d_i}$.

B. Recursive Graph Partitioning

In addition to serving as bases for signals on a graph, Laplacian eigenvectors can also be used for graph partitioning. For a connected graph G, Fiedler showed in [5] that an eigenvector corresponding to the first nonzero eigenvalue of the unnormalized Laplacian (i.e., $\boldsymbol{\phi}_1$) partitions the vertices into two sets,

$$\begin{aligned} V_1 &= \left\{ i \;\middle|\; \boldsymbol{\phi}_1(i) \geq 0 \right\} \\ V_2 &= V \setminus V_1, \end{aligned}$$

such that the subgraphs induced on V_1 and V_2 by G are both connected graphs. Thus, the *Fiedler vector*, as it has come to be known, provides a simple means of bipartitioning. This result also holds when using $\boldsymbol{\phi}_1^{\mathrm{rw}}$ (which is equivalent to using $\boldsymbol{\phi}_1^{\mathrm{sym}}$, since (1) reveals that the eigenvector entries will have the same signs). Justification of this approach comes from the fact that it yields an approximate minimizer of the bipartitioning criterion called the *RatioCut* (or the *Normalized Cut*) when L (or L_{rw}, respectively) is used [4], [6]. This result can be seen as a corollary of the Discrete Nodal Domain Theorem [7], [8], and by utilizing more of the Laplacian eigenvectors we can partition the graph into more subgraphs.

A common strategy used to develop transforms for signals on graphs, and one that we employ, is to utilize a hierarchical tree. Unless the hierarchical tree is provided along with the graph, it must be generated in one of two ways. The first is to utilize a bottom-up clustering approach in which we start with the individual vertices of the graph and recursively group them together according to their similarity, as indicated by the weight matrix W. The second method is to use a top-down partitioning approach in which we start with the entire graph and repeatedly partition it into subgraphs, typically in a manner that strives to generate subgraphs that are roughly equal in size while keeping similar vertices grouped together. We now set forth a set of conditions for hierarchical trees. For some transforms these requirements are stricter than necessary, but we maintain them because the resulting trees are compatible with all of the hierarchical tree-based transforms that we mention in Section II-C ([9]–[16]), as well as our own HGLET [17] and GHWT [18].

Starting with notation, we use j to denote the levels of the hierarchical tree, with $j = 0$ denoting the coarsest level and $j = j_{\max}$ denoting the finest level. We use K^j to denote the number of sets of vertices on level j of the tree, and we use $k \in [0, K^j)$ to index these sets. We use V_k^j to denote the kth set of vertices on level j, and we set $N_k^j := |V_k^j|$. We define G_k^j to be the subgraph formed by restricting to the vertices in V_k^j and the edges between them. We often use the term "region" to refer to a subgraph G_k^j, especially when the nodes of the graph lie in $\mathbb{R}, \mathbb{R}^2$, or $\mathbb{R}^3$ because this emphasizes the spatial organization of the subgraphs. In addition, we use the term "subregion" to refer to a child subgraph. This notation is illustrated in the hierarchical tree for a graph with $N = 6$ vertices in Fig. 1.

We impose the following four requirements for a hierarchical tree:

i) The coarsest level is the entire graph; that is, $G_0^0 = G$.
ii) At the finest level, each region is a single node; that is, $N_k^{j_{\max}} = 1$ for $0 \leq k < K^{j_{\max}} = N$.
iii) All regions on a given level are disjoint; that is, $V_k^j \cap V_{\tilde{k}}^j = \emptyset$ if $k \neq \tilde{k}$.
iv) Each region on level $j < j_{\max}$ containing two or more nodes is partitioned into exactly two regions on level $j+1$.

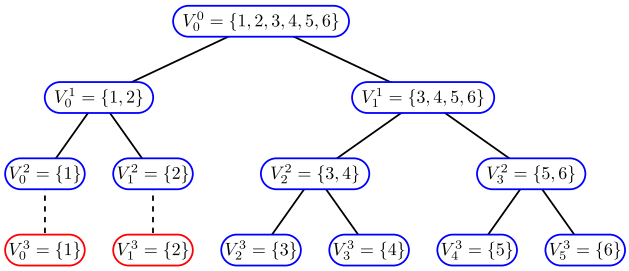

Fig. 1. An example of a hierarchical tree for a graph with $N = 6$ nodes that conforms to our notation and requirements. The nodes encircled in red and connected by dashed lines are "copies" of singleton nodes, which we include because our HGLET and GHWT require that all N nodes of the graph are present at each level j of the hierarchical tree.

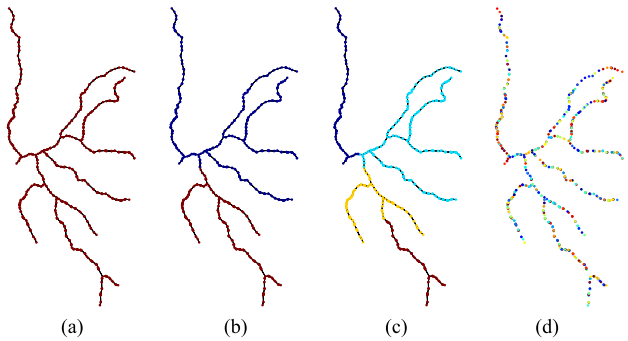

Fig. 2. A demonstration of recursive partitioning on a subset of a dendritic tree (the full tree is shown in Fig. 8. In (a)-(c), colors correspond to different regions. In (d), each region is a single node, and as such all nodes are disconnected. The level index: (a) $j = 0$; (b) $j = 1$; (c) $j = 2$; (d) $j_{\max} = 12$.

One method for generating a suitable recursive partitioning of a graph is to repeatedly partition the graph and subgraphs according to the signs of their respective Fiedler vectors (see [2], [17]–[19] for details); this is illustrated in Fig. 2.

Generating a recursive bipartitioning of a graph using Fiedler vectors is obviously not a novel idea – Simon discussed such a method already in [20]. What is novel is our use of such a recursive bipartitioning to generate *overcomplete* dictionaries of orthonormal bases for analyzing signals on the graph. That is, while [9]–[16] each generate a single wavelet-like basis for signals on the graph, we generate an entire dictionary of bases from which one can choose the particular basis that is best suited for the task at hand (e.g., via our generalization of the best basis selection algorithm). Moreover, our transforms are compatible with hierarchical trees generated using different approaches, such as the diffuse interface model of Bertozzi and Flenner [21] or the local spectral method of Mahoney *et al.* [22]. This flexibility is certainly advantageous, since graph clustering and partitioning are quite active areas of research and new algorithms continue to be developed.

C. Previous Work

A comprehensive review of transforms for signals on graphs can be found in [23]. In their review, Shuman *et al.* divide transforms into two general categories. The first category consists of those transforms that are based on the graph Fourier transform [1], which essentially uses Laplacian eigenvectors as the analogs of the complex exponentials in the classical Discrete Fourier Transform (DFT). Thus, these transforms rely upon a notion of frequency. In contrast, the second category are those methods which operate according to the connectivity of the vertices. Our transforms fall into this latter group, so that is where we shall focus our discussion (see [19, Sec. 2.3] for a more in-depth review).

Using a hierarchical tree, several groups of researchers have generalized the Haar wavelet transform to the graph setting [9]–[12]. Recall that classical Haar scaling coefficients are scaled averages of a function on an interval and that the wavelet coefficients are the differences of these averages on the two subintervals. Accordingly, each of these generalized Haar transforms proceeds by assigning one "wavelet" coefficient to each of the $N - 1$ parent (i.e., non-leaf) nodes in the hierarchical tree, which is computed by taking the difference of the scaled averages on its two children nodes. The remaining expansion coefficient is the scaling coefficient on the root node of the tree, which is equal to $\sqrt{N}$ times the average of the signal over the entire graph. The generalized Haar basis is orthonormal, and its coefficients range in scale from local to global.

Along with these generalizations of the Haar basis, a number of other transforms also utilize a recursive partitioning of a graph. Szlam *et al.* generate an orthonormal basis for signals on graphs in two different ways [13]. Their first method entails constructing the generalized Haar basis, smoothing the basis functions using diffusion operators, and then performing an orthogonalization procedure. Their second approach is to generalize the local cosine dictionary on each subgraph using the graph/manifold version of the folding and unfolding operators initially proposed by Coifman and Meyer for functions on the real line (or on the regular 1-D lattice) [24]. Sharon and Shkolnisky use a subset of the Laplacian eigenvectors and a recursive partitioning tree to construct a multiresolution analysis and consequently multiwavelet bases [14]. For a user-specified constant $m \in [1, N]$, their orthonormal basis is such that (i) all but m basis vectors are orthogonal to the first m Laplacian eigenvectors of $L_{\mathrm{rw}}(G)$, and (ii) all but $O(m)$ basis vectors have small support. Another transform that utilizes a hierarchical tree is that of Rustamov [15], which is a generalization of the average-interpolating transform of Donoho *et al.* for manifold-valued data [25] to the setting of graphs. Rustamov and Guibas developed a second transform [16], which is based on the lifting scheme for classical wavelets (see, e.g., [26], [27]).

Of course, not all methods are based on a recursive partitioning of the graph. Jansen *et al.* have designed a wavelet transform for signals on graphs that is based on the lifting scheme, with the distinction that they are "lifting one coefficient at a time" [28]. Coifman and Maggioni take a unique approach, using the diffusion/random walk on a graph to build diffusion wavelets [29] and diffusion wavelet packets [30]. The underlying idea is that by taking dyadic powers of a diffusion operator U for which high powers have low numerical rank, they are able to coarsen the graph and construct a multiresolution approximation.

D. HGLET, GHWT, and the Best Basis Algorithm

Having reviewed existing transforms and techniques for signals on graphs, we will now briefly review our own contributions: the Hierarchical Graph Laplacian Eigen Transform

(HGLET) and Generalized Haar-Walsh Transform (GHWT), along with the best basis algorithm. Like many of the transforms covered in this subsection, the HGLET and GHWT utilize a recursive partitioning of the graph. (For a more thorough discussion of these three techniques, see [2], [17]–[19].)

Using a recursive partitioning of the graph, the HGLET generates an overcomplete dictionary whose basis vectors' supports vary in size from a single node to the entire graph. We use $\boldsymbol{\phi}_{k,l}^j$ to denote the HGLET basis vectors, and we use $c_{k,l}^j$ to denote the corresponding expansion coefficients. As with the recursive partitioning, $j \in [0, j_{\max}]$ and $k \in [0, K^j)$ denote the level and region, respectively, to which a basis vector/coefficient corresponds. $l \in [0, N_k^j)$ indexes the vectors/coefficients from G_k^j. The basis vectors are formed by computing Laplacian eigenvectors on subgraphs G_k^j and extending them by zeros to the entire graph; these may be the extended eigenvectors of L, L_{rw}, or L_{sym}. A benefit of considering all three dictionaries is that we are able to construct a hybrid basis, as described in Remark 2.1. In [2], we demonstrated the use of hybrid bases for simultaneous segmentation, denoising, and compression of classical 1D signals. The computational complexity of the HGLET is $O(N^3)$, which is due to computing the full set of eigenvectors of the $N \times N$ Laplacian matrix on level $j = 0$. Of course, when such a cost is prohibitively expensive, one could perform the HGLET only on subgraphs G_k^j with $N_k^j \leq N_{\max} < N$ nodes where $N_{\max}$ is a user-specified number depending on the computational budget, in which case the cost is reduced to $O(N_{\max}^2 N)$.

Like the HGLET, the GHWT uses a recursive partitioning of the graph to generate an overcomplete dictionary, but in this case the basis vectors are piecewise constant on their support. We use $\boldsymbol{\psi}_{k,l}^j$ and $d_{k,l}^j$ to denote the GHWT basis vectors and expansion coefficients, respectively. As with the HGLET, $j \in [0, j_{\max}]$ and $k \in [0, K^j)$ denote level and region, respectively. In the case of the GHWT, we refer to l as the basis vector's/coefficient's *tag*, and it assumes N_k^j distinct values within the range $[0, 2^{j_{\max}-j})$. We refer to coefficients with tag $l = 0$ as *scaling coefficients*, those with tag $l = 1$ as *Haar coefficients*, and those with tag $l \geq 2$ as *Walsh coefficients*. Given a hierarchical tree with $O(\log N)$ levels, the computational cost of the GHWT is $O(N \log N)$.

A key feature of the GHWT is that we can arrange the coefficients in two ways. On each level j, we can group them by their k index, yielding the *coarse-to-fine dictionary*; this dictionary has the same structure as the HGLET dictionary. Alternatively, we can group them by their tag l to obtain the *fine-to-coarse dictionary*, the significance of which is that it affords us more bases from which to choose. Generally speaking, for a graph with N nodes, the HGLET, GHWT coarse-to-fine, and GHWT fine-to-coarse dictionaries each contain $> 2^{\lfloor N/2 \rfloor}$ choosable bases. (See [19, Table 6.1]; exceptions can occur when the recursive partitioning is highly imbalanced.)

For the task of selecting one basis from the immense number of choosable bases, we have generalized the best basis algorithm of Coifman and Wickerhauser [31] for our transforms. The algorithm requires a user-specified cost functional, and the search starts at the bottom level of the dictionary and proceeds upwards, comparing the cost of the children coefficients to the cost of the parent coefficients. As justification of the term "best basis," we have also generalized the corresponding proposition of Coifman and Wickerhauser:

Proposition 2.1. [19, Ch. 6] Suppose that $\mathcal{J}$ is a cost functional such that for all sequences $\{x_i\}$ and $\{y_i\}$ and integers $\alpha < \beta < \gamma$,

$$\begin{aligned} \text{if} \quad & \mathcal{J}\left(\{x_i\}_{i\in[\alpha,\beta)}\right) \leq \mathcal{J}\left(\{y_i\}_{i\in[\alpha,\beta)}\right) \\ \text{and} \quad & \mathcal{J}\left(\{x_i\}_{i\in[\beta,\gamma)}\right) \leq \mathcal{J}\left(\{y_i\}_{i\in[\beta,\gamma)}\right), \\ \text{then} \quad & \mathcal{J}\left(\{x_i\}_{i\in[\alpha,\gamma)}\right) \leq \mathcal{J}\left(\{y_i\}_{i\in[\alpha,\gamma)}\right). \end{aligned} \tag{2}$$

Given a signal $\boldsymbol{f}$ on a graph G and a hierarchical tree for the graph, the set $\{b_i\}_{i\in[0,N)}$ of expansion coefficients returned by the best basis algorithm is the set that minimizes $\mathcal{J}$ over all choosable sets of coefficients in the dictionary (or dictionaries) considered. (We refer the reader to [19] for the proof.)

Remark 2.1. The three HGLET dictionaries (using L, L_{rw}, and L_{sym}) and the GHWT coarse-to-fine dictionary all conform to the same hierarchical structure. We can take advantage of this by using a *"hybrid" best basis algorithm* in which we choose different transforms to capture the various regions of the signal. While the structure of the GHWT fine-to-coarse dictionary is incompatible with the structure of the other four dictionaries, we can select a best basis from the fine-to-coarse dictionary and compare its cost to that of the GHWT coarse-to-fine best basis or the hybrid best basis.

III. Approximation of Signals on Graphs

A. Theoretical Results

Classical wavelets have been highly successful for approximation and compression. Examples of their use include the JPEG 2000 image compression standard [32] and wavelet orthogonal frequency-division multiplexing (OFDM), which is a means of data encoding commonly used in digital communication [33]. As theoretical justification for their use, results on approximation error bounds and wavelet coefficient decay rates have been proven for signals of various types (e.g., Lipschitz, Hölder, Sobolev, Besov, and bounded variation; see [34], [35] and [36, Ch. 9]).

Proving similar results for signals on graphs is challenging because we lack the concepts and tools used for classical signals, but there have been some developments. For a graph equipped with a hierarchical tree, Coifman *et al.* [11], [12], [37] define a Hölder seminorm and use it to prove various results for the graph Haar basis (which is a choosable basis from the GHWT fine-to-coarse dictionary). They begin by using the hierarchical tree to define a distance function between nodes of a graph:

$$d(m,n) := \min\{N_k^j \;\big|\; m,n \in V(G_k^j)\}.$$

Thus, the distance between two nodes is the size of the smallest subgraph to which both nodes belong. For a constant $0 < \alpha \leq 1$, they define the Hölder seminorm of a function $\boldsymbol{f}$ on the graph as

$$C_H(\boldsymbol{f}) := \sup_{m\neq n} \frac{|\boldsymbol{f}(n) - \boldsymbol{f}(m)|}{d(m,n)^\alpha}.$$

With these definitions in place, we now extend their result for the generalized Haar transform to our own transforms.

Theorem 3.1. For a graph G equipped with a hierarchical tree, suppose that a signal $\boldsymbol{f}$ is Hölder continuous with exponent α and constant $C_H(\boldsymbol{f})$. Then the coefficients with $l \geq 1$ for the HGLET (with unnormalized Laplacian L) and the GHWT satisfy

$$|c_{k,l}^j| \leq C_H(\boldsymbol{f})(N_k^j)^{\alpha+1/2}$$
$$|d_{k,l}^j| \leq C_H(\boldsymbol{f})(N_k^j)^{\alpha+1/2}.$$

The coefficients with $l \geq 1$ for the HGLET with L_{rw} and L_{sym} satisfy

$$|c_{k,l}^{j,\mathrm{rw}}| \leq \frac{C_k^j}{\sqrt{2}} \cdot C_H(\boldsymbol{f})(N_k^j)^{\alpha+1/2} + \tilde{C}_k^j \left\| \boldsymbol{f}|_{V_k^j} \right\|_{D(G_k^j)}$$
$$|c_{k,l}^{j,\mathrm{sym}}| \leq \sqrt{C_k^j} \cdot C_H(\boldsymbol{f})(N_k^j)^{\alpha+1/2} + \sqrt{C_k^j} \left\| \boldsymbol{f}|_{V_k^j} \right\|_2,$$

where $\boldsymbol{f}|_{V_k^j} \in \mathbb{R}^{N_k^j}$ denotes the restriction of $\boldsymbol{f}$ to the vertices in V_k^j, and C_k^j and $\tilde{C}_k^j$ are constants that are independent from α.

Proof. Below, we present the proof for the HGLET with L; the proof for the GHWT bound is identical, with $c_{k,l}^j$ and $\boldsymbol{\phi}_{k,l}^j$ replaced by $d_{k,l}^j$ and $\boldsymbol{\psi}_{k,l}^j$, respectively. Our proof follows that of [37], although here we use vectors and summations rather than functions and integrals. For the proofs for the HGLET with L_{rw} and that with L_{sym}, due to the page limitation, we refer the interested readers to our online supplementary note [38].

For the coefficients from the HGLET with unnormalized Laplacian L and with tag $l \geq 1$, we have

$$\begin{aligned}
|c_{k,l}^j| &= \left| \left\langle \boldsymbol{f}, \boldsymbol{\phi}_{k,l}^j \right\rangle \right| \\
&= \left| \left\langle \boldsymbol{f} - \left\langle \boldsymbol{f}, \boldsymbol{\phi}_{k,0}^j \right\rangle \boldsymbol{\phi}_{k,0}^j, \boldsymbol{\phi}_{k,l}^j \right\rangle \right| \\
&\leq \left\| \boldsymbol{f} - \left\langle \boldsymbol{f}, \boldsymbol{\phi}_{k,0}^j \right\rangle \boldsymbol{\phi}_{k,0}^j \right\|_2 \|\boldsymbol{\phi}_{k,l}^j\|_2 \\
&= \left(\sum_{n \in V_k^j} \left| \boldsymbol{f}(n) - \sum_{m \in V_k^j} \frac{\boldsymbol{f}(m)}{N_k^j} \right|^2 \right)^{1/2} \\
&= \left(\sum_{n \in V_k^j} \left| \sum_{m \in V_k^j} \frac{\boldsymbol{f}(n) - \boldsymbol{f}(m)}{N_k^j} \right|^2 \right)^{1/2} \\
&\leq \left(\sum_{n \in V_k^j} \left| \sum_{m \in V_k^j} \frac{C_H(\boldsymbol{f}) d(m,n)^\alpha}{N_k^j} \right|^2 \right)^{1/2} \\
&\leq \left(\sum_{n \in V_k^j} \left| \sum_{m \in V_k^j} \frac{C_H(\boldsymbol{f})(N_k^j)^\alpha}{N_k^j} \right|^2 \right)^{1/2} \\
&= \left(\sum_{n \in V_k^j} \left(C_H(\boldsymbol{f})(N_k^j)^\alpha \right)^2 \right)^{1/2} \\
&= C_H(\boldsymbol{f})(N_k^j)^{\alpha+1/2}.
\end{aligned}$$

■

Sharon and Shkolnisky derive an n-term nonlinear approximation bound by defining a generalization of Besov spaces in the graph setting [14]. For a fixed orthonormal basis $\{\varphi_l\}_{l=0}^{N-1}$ and a parameter $\tau \in (0,2)$, they define the τ-measure of a function $\boldsymbol{f}$ as

$$|\boldsymbol{f}|_\tau := \left(\sum_{l=0}^{N-1} |\langle \boldsymbol{f}, \varphi_l \rangle|^\tau \right)^{1/\tau}. \tag{3}$$

They note that for all signals, the τ-measure satisfies

$$\|\boldsymbol{f}\|_2 \leq |\boldsymbol{f}|_\tau \leq N^{\frac{1}{\tau}-\frac{1}{2}} \|\boldsymbol{f}\|_2.$$

They define discrete analogs of the Besov spaces as

$$B_{\tau,M} := \{\boldsymbol{f} \mid |\boldsymbol{f}|_\tau < M \text{ and } \|\boldsymbol{f}\| = 1\},$$

where $0 < \tau < 2$ and $1 \leq M \leq N^{\frac{1}{\tau}-\frac{1}{2}}$. Following the notation of [34], let $P_n \boldsymbol{f}$ denote the best nonlinear n-term approximation of $\boldsymbol{f}$ in the basis. Sharon and Shkolnisky prove the following bound on the approximation error.

Theorem 3.2. [14] For a fixed orthonormal basis $\{\varphi_l\}_{l=0}^{N-1}$ and a parameter $0 < \tau < 2$,

$$\|\boldsymbol{f} - P_n \boldsymbol{f}\|_2 \leq \frac{|\boldsymbol{f}|_\tau}{n^\beta}, \tag{4}$$

where $|\boldsymbol{f}|_\tau$ corresponds to $\{\varphi_l\}_{l=0}^{N-1}$ and $\beta = \frac{1}{\tau} - \frac{1}{2}$.

As the HGLET (with L and L_{sym} but not with L_{rw}) and GHWT yield overcomplete dictionaries of orthonormal bases, this theorem applies directly to any basis we select from those dictionaries; for the GHWT, this includes both the coarse-to-fine and fine-to-coarse dictionaries. Furthermore, note that the τ-measure satisfies the requirements (2) from Proposition 2.1 for our best basis algorithms. Therefore, we have the following corollary.

Corollary 3.1. For a signal $\boldsymbol{f}$, consider one or more dictionaries of orthonormal expansion coefficients (i.e., those corresponding to the HGLET with L, the HGLET with L_{sym}, GHWT coarse-to-fine, or GHWT fine-to-coarse). For $\tau \in (0,2)$, using the τ-measure as the cost functional for the ("hybrid") best basis algorithm yields the choosable orthonormal basis that minimizes $|\boldsymbol{f}|_\tau$ and therefore has the best bound for nonlinear approximation error in (4).

Of course, this corollary does not tell us which τ-measure should be used as the best basis cost functional in order to achieve the best approximation bound in (4). Fortunately, the best basis search is quick and inexpensive, and thus we can perform the search over a range of τ values (e.g., $\tau = 0.1, 0.2, \ldots, 1.9$), obtaining a set of best basis coefficients for each one. We can then specify a constant n (e.g., $n = [0.1N]$) and choose the particular τ and corresponding basis which minimizes the upper bound $|\boldsymbol{f}|_\tau / n^\beta$. However, in practice this does not always lead to the best choice of τ because the bound (4) is not tight enough.

What we can do instead is to search over a range of τ values and choose the particular best basis that yields the smallest cumulative relative error. To do this, we find the N best basis expansion coefficients for each τ and then compute a vector of length N containing the relative approximation errors when $1, 2, \ldots, N$ coefficients are retained. This is easily done for

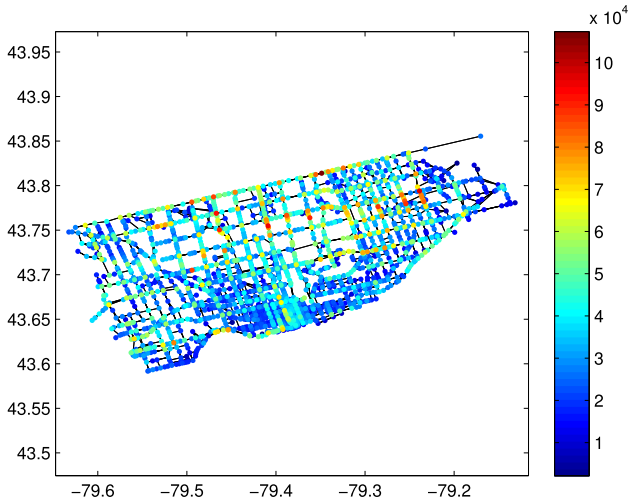

Fig. 3. Traffic volume data over a 24 hour period at intersections in the road network of Toronto ($N = 2202$ nodes and $M = 4877$ edges).

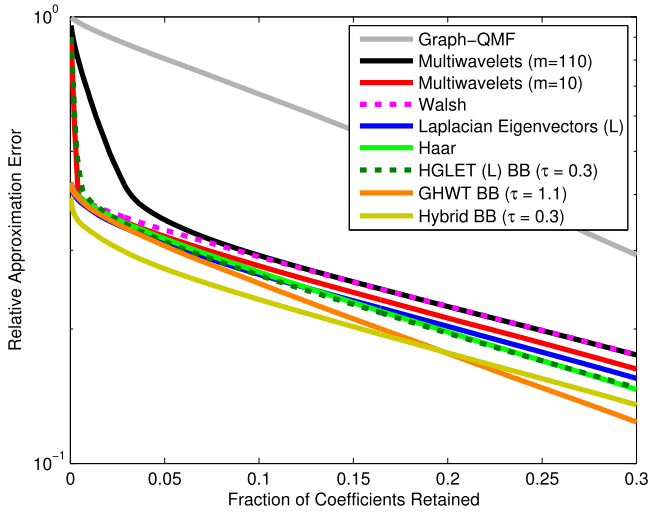

Fig. 4. Relative approximation error as a function of coefficients kept for the Toronto traffic volume data set.

orthonormal bases; for bases that are not orthonormal, this can still be accomplished in a simple manner by forming the $N \times N$ matrix of best basis vectors. We then take the sum of this vector of relative errors; in other words, letting $P_n \boldsymbol{f}$ denote the best n-term nonlinear approximation of $\boldsymbol{f}$ with respect to the basis, we compute

$$\text{cumulative relative error } = \sum_{n=1}^{N} \|\boldsymbol{f} - P_n \boldsymbol{f}\|_2 / \|\boldsymbol{f}\|_2. \qquad (5)$$

We search over the range of τ values and select the basis which minimizes this sum. In terms of Fig. 4, we are selecting the τ that yields the smallest area under the relative error curve. As we will use this strategy often, we refer to it as the *minimum relative error best basis algorithm*. Note that we can use this method for the HGLET with L_{rw} even though the basis is not orthonormal with respect to the standard inner product. However, Theorem 3.2 and Corollary 3.1 will not apply to the resulting basis.

B. Experimental Results

Having proven some theoretical approximation results for our transforms, we now present an experiment comparing our methods to other transforms. For our signal, we use vehicular traffic volume data on the Toronto road network,[3] as seen in Fig. 3. The data was collected over 24 hour windows (i.e., it is not the case that all intersections were monitored over the same 24 hour time span). Using the street names and intersection coordinates included in the data set, we generated the road network of Toronto. This graph and its corresponding signal are freely distributed as part of the MTSG Toolbox. We emphasize that this is a real data set, thereby avoiding the concern of designing a synthetic signal that is either unrealistic or biased towards certain transforms.

In addition to the graph Haar basis, the graph Walsh basis (i.e., level $j = 0$ of the GHWT coarse-to-fine dictionary), and the eigenvectors of the unnormalized Laplacian $L(G)$ of the entire graph, we compare our methods to two other graph transforms. Granted, the transforms considered use a fixed basis while our methods involve adaptively choosing a basis from an overcomplete dictionary, but this is the fairest comparison we can make. The two transforms that we selected were the graph-QMF [39] (which is based on the graph Fourier transform; see [1]) and Laplacian multiwavelets [14]. As we mentioned in Section II-C, a parameter m needs to be specified for these multiwavelets. We used two values, both of which are used in example code that the authors provide: $m = 10$ and $m = \lfloor N/20 \rfloor$. The cost of generating the multiwavelet basis is $O(m^2 N \log N + T(N, m) \log N)$, where $T(N, m)$ is the cost of computing the first m global Laplacian eigenvectors [14]; a computational cost for the graph-QMF is not mentioned in [39].

As for our own transforms, we use the HGLET best basis (with unnormalized Laplacian L), the GHWT best basis, and the hybrid best basis. For the hybrid best basis algorithm, we consider all four dictionaries: HGLET with L, HGLET with L_{rw}, HGLET with L_{sym}, and GHWT coarse-to-fine. In order to avoid the need to specify a cost functional, we utilize the minimum relative error best basis algorithm, which determines the best τ-measure to be used as the cost functional. To generate the partitioning tree for our transforms, we perform recursive bipartitioning using the Fiedler vector of L_{rw}, as described in Section II-B; we use this same method to generate the partitioning tree required by Laplacian multiwavelets.

Fig. 4 shows the relative approximation errors for the Toronto data set as a function of the fraction of coefficients retained. The best performances are achieved by the hybrid best basis[4] and the GHWT best basis (which originates from the fine-to-coarse dictionary), with the hybrid basis performing better when fewer than 19.7% of the coefficients are kept and the fine-to-coarse GHWT best basis performing better thereafter. To explain why this crossover occurs, we need to examine the structure of these

[3]This information is made publicly available by the city of Toronto at http://www1.toronto.ca/wps/portal/contentonly?vgnextoid=417aed3c99cc7310VgnVCM1000003dd60f89RCRD.

[4]In the hybrid best basis algorithm, we did not include the GHWT fine-to-coarse dictionary as a possible choice, as mentioned in Remark 2.1. Although the GHWT best basis has a lower cumulative relative error, we display the results for the hybrid best basis so that the two can be compared.

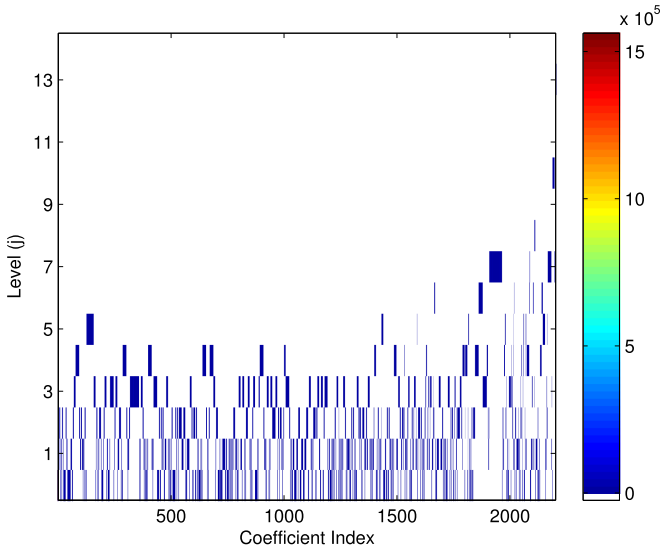

Fig. 5. The locations of the GHWT best basis coefficients within the fine-to-coarse dictionary for the Toronto traffic data. (See [19, Sec. 5.2] for a detailed explanation of this visualization.)

bases. Fig. 5 illustrates the levels of the selected GHWT coefficients from within the fine-to-coarse dictionary. By contrast, the hybrid best basis is actually the set of global eigenvectors of $L_{\mathrm{sym}}(G)$. Intuitively, this makes sense because we expect that intersections involving more streets will have more traffic volume, and the degree normalization of L_{sym} should help its eigenvectors to capture this. Since the vectors in this hybrid best basis are global in scale, this basis is well-suited for very sparse, coarse approximation of the signal, which is why it outperforms the GHWT best basis when fewer than 19.7% of the coefficients are retained. However, the more localized basis vectors in the GHWT best basis enable it to better capture details on finer scales, and thus it surpasses the hybrid best basis after the 19.7% mark.

It is also important to note from Fig. 5 that the structure of the GHWT best basis differs radically from that of the Haar basis, which has one block of $\approx 2^j$ coefficients on levels $j = 0, 1, ..., j_{\max} - 1$. Recalling that the basis vectors are global on level $j = 0$ and become more localized as j increases, we see that the GHWT best basis has far more basis vectors with large supports. Furthermore, given that the number of oscillations in the basis vectors on a particular level j generally increases from left to right in this table, i.e., as l increases (see [18] and [19, Ch. 5]), we note that the GHWT best basis contains basis vectors with much more oscillation than those in the Haar basis, which assume only two distinct nonzero values. Thus, the best basis algorithm validates what we would expect: more oscillatory basis vectors are advantageous for representing this signal. However, it is also necessary to have some basis vectors which are more localized, as evidenced by the fact that the Walsh basis is outperformed by the GHWT best basis and the Haar basis.

This experiment demonstrates the effectiveness of adaptively selecting a basis for a signal on a graph, as opposed to using a fixed basis. It also illustrates some of the insights afforded by selected bases, such as whether the nature of the signal is smooth or oscillatory, or whether its features are local or global in scale.

TABLE I
DENOISING RESULTS FOR THE NOISY VERSIONS OF THE TRAFFIC VOLUME DATA FOR TORONTO (FIG. 6) AND THE DENDRITIC TREE THICKNESS DATA (FIG. 8)

	Dendritic Tree (8.00 dB)	Toronto (7.00 dB)
HGLET (L) Best Basis	20.85 dB ($\tau = 0.1$)	**8.96 dB** ($\tau = 0.3$)
Laplacian Eigenvectors (L)	22.56 dB	8.26 dB
GHWT Best Basis	**23.03 dB** ($\tau = 0.9$)	8.27 dB ($\tau = 1.0$)
Haar Basis	22.68 dB	8.29 dB
Hybrid Best Basis	22.29 dB ($\tau = 0.3$)	8.82 dB ($\tau = 0.3$)
Walsh Basis	21.57 dB	8.14 dB
Graph-QMF	2.85 dB	8.09 dB
Multiwavelets ($m = 10$)	21.76 dB	8.61 dB
Multiwavelets ($m = \lfloor N/20 \rfloor$)	15.37 dB	7.47 dB

For Laplacian multiwavelets [14], we used the two values of m that were used in the example code provided by the authors: $m = 10$ and $m = \lfloor N/20 \rfloor$; it was not necessary to specify parameters for the Graph-QMF [39].

IV. DENOISING OF SIGNALS ON GRAPHS

Building upon their effectiveness for approximation, classical wavelets have also been applied to the task of denoising with much success. The reasons why this works are because (i) a basis that is efficient for approximation concentrates the majority of a signal's energy into a small number of large coefficients; and (ii) "Gaussian white noise in any one orthogonal basis is again a white noise in any other (and with the same amplitude)" [40]. Based on these insights, Donoho *et al.* devised wavelet shrinkage [41], which yields nearly optimal nonlinear estimators. Their method is simple and straightforward: apply the wavelet transform to the signal, soft-threshold the coefficients (excluding the scaling coefficients), and then reconstruct.

We employ this same strategy in order to denoise signals on graphs using our transforms. Of course, a precursor step when denoising with the HGLET and GHWT is to first select a best basis. As with our approximation experiment, we do so by using the minimal relative error best basis algorithm.

Consider a noisy signal $\boldsymbol{g} = \boldsymbol{f} + \boldsymbol{\epsilon}$, where $\boldsymbol{f}$ is the noise-free signal and $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 I)$ is Gaussian noise. For the sake of transparency, the formula that we use to compute the signal-to-noise ratio is

$$\mathrm{SNR} = 20 \log_{10} \frac{\|\boldsymbol{f}\|_2}{\|\boldsymbol{g} - \boldsymbol{f}\|_2}.$$

We analyze the signal with the transform(s) of our choice and select a basis using the minimal relative error best basis algorithm. Having selected a basis, the next step is to threshold the coefficients. For a threshold $T > 0$, we soft-threshold HGLET expansion coefficients $c^j_{k,l}$ (and likewise for GHWT coefficients $d^j_{k,l}$) as

$$\tilde{c}^j_{k,l} = \begin{cases} c^j_{k,l} & \text{if } l = 0 \\ \mathrm{sign}(c^j_{k,l})(|c^j_{k,l}| - T)_+ & \text{otherwise.} \end{cases}$$

A key aspect of this denoising procedure is to determine the appropriate threshold T. To do this, we generate a curve of the relative reconstruction errors (i.e., $\|\boldsymbol{g} - \widehat{\boldsymbol{g}}\|_2 / \|\boldsymbol{g}\|_2$, where $\widehat{\boldsymbol{g}}$ is a reconstruction of $\boldsymbol{g}$) in which we use the magnitudes of the coefficients as thresholds; specifically, the smallest threshold is

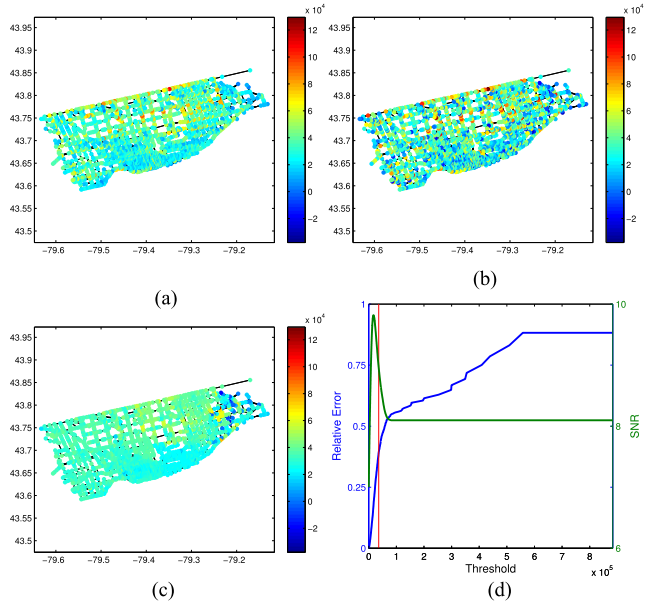

Fig. 6. The (a) original, (b) noisy, and (c) denoised versions of the traffic volume data on the Toronto road network. The HGLET (L) best basis ($\tau = 0.3$) was used here. (d) Relative error and SNR curves, with the red line indicating the selected threshold. The SNR values of (b) and (c) are 7.00 dB and 8.96 dB, respectively.

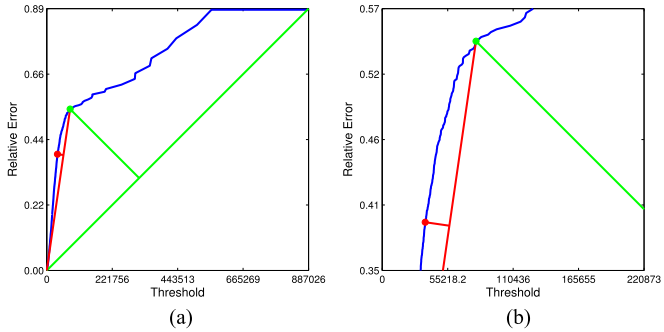

Fig. 7. (a) An illustration of the method that we use to determine a threshold from the relative error curve. The curve seen here is a rescaled version of the relative error curve for the Toronto traffic data (Fig. 6(d)). (b) A zoomed-in version of (a).

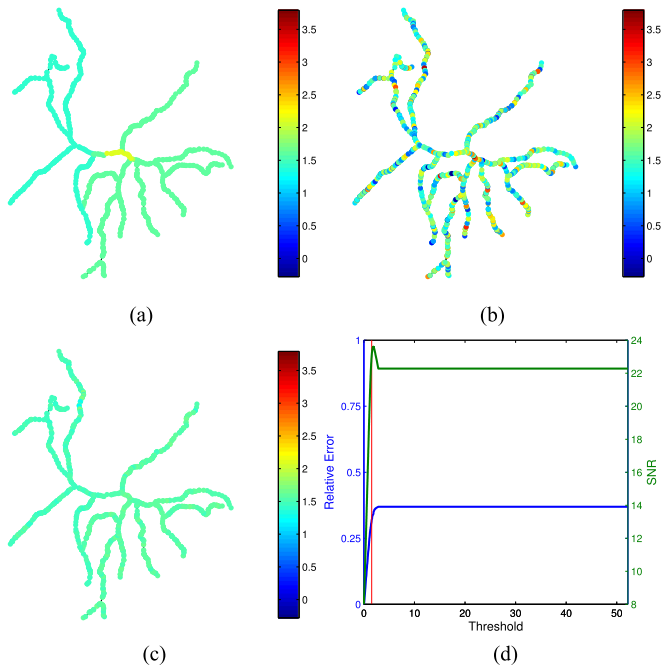

Fig. 8. The (a) original, (b) noisy, and (c) denoised versions of the thickness data on the dendritic tree. This denoising was done using the GHWT best basis ($\tau = 0.9$). (d) Relative error and SNR curves, with the red line indicating the selected threshold.

zero and the biggest is the magnitude of the second largest coefficient. For this task we use hard-thresholding, and thus the best n term nonlinear approximation of the signal corresponds to hard-thresholding with the $(n+1)$st largest coefficient magnitude. An example of such a curve can be seen in Fig. 6(d), where the signal is a noisy version (7.00 dB) of the Toronto traffic data and the basis being considered is the HGLET best basis. Although we do not use it to denoise the signal, we also display a curve of the signal-to-noise ratios obtained by using soft-thresholding with each of the N coefficient magnitudes as the thresholds.

Note the behavior of the two curves in Fig. 6(d): the SNR curve rises quickly as the threshold increases from zero, while the relative error curve starts dropping rapidly when the threshold decreases towards zero. After attaining its maximum, the SNR curve falls quickly to the SNR of the noisy signal (7.00 dB). In Fig. 8, we observe similar behavior for a noisy version of thickness data on a dendritic tree. The value of the signal at each node is the thickness of the dendrite at that point, as measured by Coombs *et al.* [42]. As we lower the threshold (i.e., proceed from right to left in the plots), the reconstruction error steadily declines while the threshold is relatively large. This is because, as mentioned at the start of this section, a basis that is efficient for approximation concentrates the majority of the signal's energy into a small number of large coefficients. When the threshold is high, only a few coefficients are retained, which explains why the relative error curve is constant on the right side of the plot and fairly flat in the middle of the plot. On the other hand, there are a large number of small coefficients which capture the detail and noise in the signal. As the threshold decreases, more and more of these are retained, which explains the rapid decrease in the relative error of the reconstructions of the noisy signal.

As we see from Figs. 6(d) and 8(d), the peak SNR occurs soon after the relative error starts to drop quickly as the threshold decreases toward zero. The intuition here is simple: we want to retain the coefficients that capture detail in the signal while thresholding those which capture the noise; without thresholding, these coefficients ultimately lead to a relative reconstruction error of zero and the original SNR value of the noisy signal. Empirically, we have found the following elbow detection scheme to work well for determining a threshold, which we illustrate in Fig. 7 using the case of the HGLET best basis relative error curve for the noisy Toronto traffic data (Fig. 6). First, we draw a line (shown in green) from the first point on the relative error curve to the last. We then find the point on the curve with the largest orthogonal distance from this line. We repeat the process a second time, drawing a line from this point to the first point (shown in red) and finding the point on the relative error curve

with the greatest orthogonal distance from that line. This point on the relative error curve (again shown in red) is the threshold that we use for denoising. The reason why we iterate this elbow detection scheme twice is because we seek a threshold that is lower than that at which the relative error curve starts to drop rapidly towards zero. We do not iterate a third time because doing so would drive the threshold too low, causing too much of the noise to be retained.

At this point we now formally describe our denoising experiments. We consider two signals: the traffic volume data for Toronto (Fig. 6(a)) and thickness data on the dendritic tree (Fig. 8(a)). We add Gaussian noise to these signals such that the signal-to-noise ratios are 7.00 dB for the Toronto traffic data and 8.00 dB for the dendritic tree; the resulting signals are displayed in Figs. 6(b) and 8(b), respectively. (Lower SNR values for both signals were investigated, but in such cases it was found that the noise obscured the signal and denoising was infeasible.) We recursively partition the graphs using Fiedler vectors of L_{rw}, as described in Section II-B, and we analyze the noisy signals using each of the three HGLET variations (L, L_{rw}, and L_{sym}) and the GHWT. Using the minimal relative error best basis algorithm, we compute the HGLET (L) best basis, the GHWT best basis, and the hybrid best basis selected from the three HGLET dictionaries and the GHWT coarse-to-fine dictionary. For comparison, we also consider the Haar basis, the Walsh basis (i.e., level $j = 0$ of the GHWT coarse-to-fine dictionary), the eigenvectors of the unnormalized Laplacian $L(G)$ of the entire graph, the graph-QMF transform, and Laplacian multiwavelets. For each of these bases we generate a relative error curve, and from this curve we determine the threshold using the aforementioned elbow detection scheme. We soft-threshold the coefficients (leaving coefficients with $l = 0$ unchanged), reconstruct the signal, and compute the SNR.

Figs. 6(d) and 8(d) show the results of our threshold selection method for the relative error and SNR curves of the noisy Toronto and dendritic tree data sets. These curves correspond to use of the HGLET (L) best basis for the Toronto traffic data and the GHWT best basis for the dendritic tree data. The denoised signals are displayed in Figs. 6(b) and 8(b). In addition to these results, a summary of the full results from this experiment can be found in Table I.

These experimental results demonstrate the effectiveness of the HGLET and GHWT, along with the best basis algorithms, for denoising signals on graphs. It is worth noting that for both of these signals, the GHWT best basis originated from the fine-to-coarse dictionary. An advantage of this dictionary is that, unlike the coarse-to-fine and HGLET dictionaries, it contains choosable bases for which basis vectors from different levels have overlapping supports. Thus, global basis vectors can capture the general characteristics of the signal while localized basis vectors contribute the finer scale details. We also note that for the Toronto traffic data the HGLET (L) best basis performed better than the hybrid best basis selected from the dictionaries that include the HGLET (L) dictionary. Why did this happen in this case? While this could be because the chosen threshold for the hybrid best basis was not optimal, the real answer is that the best basis algorithm merely finds the basis that minimizes its cost functional, which in this case is based on the τ-norm of the expansion coefficients, where $0 < \tau < 2$, and the relative ℓ^2 errors. When we compute the relative ℓ^2 errors, the noise-free signal $\boldsymbol{f}$ is not available. Hence, for the best basis selection we must use the relative ℓ^2 errors between the noisy observed signal and the denoised signal that is constructed using the bases in our dictionaries. In contrast, the SNR values in Table I were computed using the noise-free signals and the denoised signals. As the best basis algorithm is not privy to the noise-free signal, there is no guarantee that it will select the optimal basis for maximizing SNR, which explains this seemingly impossible result.

Remark 4.1. Our denoising strategy using the HGLET and GHWT dictionaries can be generalized to cope with non-Gaussian noise. To properly handle such noise models, however, it is necessary to consider precise statistical models of the coefficients and adopt a level-dependent thresholding scheme as suggested, e.g., in [43], [44].

V. Conclusion

In this article, we precisely proved the efficiency of our HGLET and GHWT transforms, in conjunction with the best basis selection algorithm, for approximating signals on graphs belonging to discrete analogs of the space of Hölder continuous functions and the Besov spaces. We then proposed quite natural methods to approximate and denoise a given graph signal and performed numerical experiments. Our transforms performed favorably when pitted against various other transforms for the real signals on graphs we used. Indeed, such direct comparisons between methods are especially important as the field of signal processing on graphs continues to advance and mature. In future work we plan to showcase the versatility and advantages of our graph-based transforms on certain classical problems such as signal segmentation and matrix data analysis where the conventional non-graph-based methods encounter difficulty.

Acknowledgment

The comments and criticism of anonymous reviewers greatly helped the authors improve this article.

References

[1] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.

[2] J. Irion and N. Saito, "Applied and computational harmonic analysis on graphs and networks," *Proc. SPIE*, vol. 9597, 2015, Art. no 95971F.

[3] F. R. K. Chung, *Spectral Graph Theory* (ser. CBMS Regional Conference Series in Mathematics), vol. 92. Providence, RI, USA: Amer. Math. Soc., 1997.

[4] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[5] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory," *Czechoslovak Math. J.*, vol. 25, no. 4, pp. 619–633, 1975.

[6] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[7] E. B. Davies, G. M. L. Gladwell, J. Leydold, and P. F. Stadler, "Discrete nodal domain theorems," *Linear Algebra Appl.*, vol. 336, pp. 51–60, 2001.

[8] T. Bıyıkoğlu, W. Hordijk, J. Leydold, T. Pisanski, and P. F. Stadler, "Graph Laplacians, nodal domains, and hyperplane arrangements," *Linear Algebra Appl.*, vol. 390, pp. 155–174, 2004.

[9] F. Murtagh, "The Haar wavelet transform of a dendrogram," *J. Classification*, vol. 24, no. 1, pp. 3–32, 2007.

[10] A. B. Lee, B. Nadler, and L. Wasserman, "Treelets—an adaptive multi-scale basis for sparse unordered data," *Ann. Appl. Statist.*, vol. 2, no. 2, pp. 435–471, 2008.
[11] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning," in *Proc. 27th Int. Conf. Mach. Learn.*, Jun. 2010, pp. 367–374.
[12] R. R. Coifman and M. Gavish, "Harmonic analysis of digital data bases," in *Wavelets and Multiscale Analysis*, J. Cohen and A. I. Zayed, Eds. New York, NY, USA: Birkhäuser/Springer, 2011, pp. 161–197.
[13] A. D. Szlam, M. Maggioni, R. R. Coifman, and J. C. Bremer, "Diffusion-driven multiscale analysis on manifolds and graphs: Top-down and bottom-up constructions," *Proc. SPIE*, vol. 5914, 2005, Art. no. 59141D. [Online]. Available: http://dx.doi.org/10.1117/12.616931
[14] N. Sharon and Y. Shkolnisky, "A class of laplacian multiwavelets bases for high-dimensional data," *Appl. Comput. Harmon. Anal.*, vol. 38, no. 3, pp. 420–451, 2015.
[15] R. M. Rustamov, "Average interpolating wavelets on point clouds and graphs," *arXiv preprint arXiv:1110.2227*, 2011.
[16] R. Rustamov and L. Guibas, "Wavelets on graphs via deep learning," in *Proc. Adv. Neural Inf. Process. Syst. Conf.*, 2013, pp. 998–1006.
[17] J. Irion and N. Saito, "Hierarchical graph Laplacian eigen transforms," *JSIAM Lett.*, vol. 6, pp. 21–24, 2014.
[18] J. Irion and N. Saito, "The generalized Haar-Walsh transform," *Proc. IEEE Stat. Signal Process. Workshop*, 2014, pp. 472–475.
[19] J. Irion, "Multiscale transforms for signals on graphs: Methods and applications," Ph.D. dissertation, Univ. California, Davis, CA, USA, 2015. [Online]. Available: https://github.com/JeffLIrion/MTSG_Toolbox/tree/master/Publications
[20] H. D. Simon, "Partitioning of unstructured problems for parallel processing," *Comput. Syst. Eng.*, vol. 2, no. 2, pp. 135–148, 1991.
[21] A. L. Bertozzi and A. Flenner, "Diffuse interface models on graphs for classification of high dimensional data," *Multiscale Model. Simul.*, vol. 10, no. 3, pp. 1090–1118, 2012.
[22] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi, "A local spectral method for graphs: With applications to improving graph partitions and exploring data graphs locally," *J. Mach. Learn. Res.*, vol. 13, pp. 2339–2365, 2012.
[23] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.
[24] R. R. Coifman and Y. Meyer, "Remarques sur l'analyse de Fourier à fenêtre," *C. R. Acad. Sci. Paris Sér. I Math.*, vol. 312, no. 3, pp. 259–261, 1991.
[25] I. U. Rahman, I. Drori, V. C. Stodden, D. L. Donoho, and P. Schröder, "Multiscale representations for manifold-valued data," *Multiscale Model. Simul.*, vol. 4, no. 4, pp. 1201–1232, 2005.
[26] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, 1996.
[27] W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *SIAM J. Math. Anal.*, vol. 29, no. 2, pp. 511–546, 1998. [Online]. Available: http://dx.doi.org/10.1137/S0036141095289051
[28] M. Jansen, G. P. Nason, and B. W. Silverman, "Multiscale methods for data on graphs and irregular multidimensional situations," *J. Roy. Statist. Soc. Ser. B Statist. Methodol.*, vol. 71, no. 1, pp. 97–125, 2009.
[29] R. R. Coifman and M. Maggioni, "Diffusion wavelets," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 53–94, 2006.
[30] J. C. Bremer, R. R. Coifman, M. Maggioni, and A. D. Szlam, "Diffusion wavelet packets," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 95–112, 2006.
[31] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 713–718, Mar. 1992.
[32] M. Marcellin, M. Gormish, A. Bilgin, and M. Boliek, "An overview of JPEG-2000," in *Proc. IEEE Data Compression Conf.*, 2000, pp. 523–541.
[33] B. Negash and H. Nikookar, "Wavelet based OFDM for wireless channels," in *Proc. IEEE VTS 53rd Veh. Technol. Conf.*, Rhodes, Greece, May 6–9, 2001, vol. 1, pp. 688–691.
[34] R. A. DeVore, "Nonlinear approximation," in *Acta Numerica*. Cambridge, U.K.: Cambridge Univ. Press, vol. 7, 1998, pp. 51–150.
[35] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 719–746, Mar. 1992.
[36] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed. Amsterdam, The Netherlands: Elsevier/Academic, 2009.
[37] R. Coifman and W. Leeb, "Earth mover's distance and equivalent metrics for spaces with hierarchical partition trees," Yale Univ., New Haven, CT, USA, Tech. Rep. YALEU/DCS/TR-1482, 2013.
[38] J. Irion and N. Saito, "The proof of Theorem 3.1," Univ. California, Davis, CA, USA, Tech. Rep., Jul. 2016. [Online]. Available: https://www.math.ucdavis.edu/∼saito/publications/Proof31.pdf
[39] S. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2786–2799, Jun. 2012.
[40] D. L. Donoho, "Wavelet shrinkage and W.V.D.: A 10-minute tour," in *Progress in Wavelet Analysis and Applications*, Y. Meyer and S. Roques, Eds. Gif sur Yvette, France: Editions Frontieres, 1993, pp. 109–128.
[41] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, "Wavelet shrinkage: Asymptopia?" *J. Roy. Statist. Soc. Ser. B*, vol. 57, no. 2, pp. 301–369, 1995.
[42] J. Coombs, D. van der List, G.-Y. Wang, and L. Chalupa, "Morphological properties of mouse retinal ganglion cells," *Neuroscience*, vol. 140, no. 1, pp. 123–136, 2006.
[43] A. Antoniadis, D. Leporini, and J.-C. Pesquet, "Wavelet thresholding for some classes of non-Gaussian noise," *Statist. Neerlandica*, vol. 56, no. 4, pp. 434–453, 2002.
[44] I. M. Johnstone and B. W. Silverman, "Empirical Bayes selection of wavelet thresholds," *Ann. Statist.*, vol. 33, no. 4, pp. 1700–1752, 2005.

Jeff Irion received the B.S. degree in chemical engineering from the University of California, San Diego, CA, USA, in 2009, and the Ph.D. degree in applied mathematics from the University of California, Davis, CA, USA, in 2015. He was a Regents Scholar and a National Merit Scholar at University of California, San Diego. He is currently a Research Scientist at Bosch Research and Technology Center, Palo Alto, CA. His research interests include graph optimization and robotics. He received the National Defense Science and Engineering Graduate Fellowship at University of California, Davis, in 2011 and the Japan Society for Industrial and Applied Mathematics (JSIAM) Letters Best Paper Award jointly with Dr. N. Saito from the JSIAM in 2016. He is a member of the Tau Beta Pi Engineering Honors Society and the Association of Computing Machinery.

Naoki Saito (M'87–SM'99) received the B.Eng. and M.Eng. degrees in mathematical engineering from the University of Tokyo, Tokyo, Japan, in 1982 and 1984, respectively, and the Ph.D. degree in applied mathematics from Yale University, New Haven, CT, USA, in 1994. In 1984, he joined Nippon Schlumberger K.K., Fuchinobe, Japan, and in 1986, he joined Schlumberger-Doll Research (SDR), Ridgefield, CT, USA, where he worked as a Research Scientist until 1997. In 1997, he joined the Department of Mathematics, University of California, Davis, CA, USA, where he is currently a Professor. He also served as the Chair of the Graduate Group in Applied Mathematics at the University of California, Davis, from 2007 to 2012. His research interests include applied and computational harmonic analysis, feature extraction, pattern recognition, data analysis, Laplacian eigenvalue problems, elliptic boundary value problems, data compression, statistical signal processing and analysis, human and machine perception, and geophysical inverse problems.

Dr. Saito received the Best Paper Award from SPIE for the wavelet applications in signal and image processing conference in 1994 and the Henri Doll Award (the highest honor for technical papers presented at the annual symposium within the Schlumberger organization) in 1997. He also received the Young Investigator Award from the Office of Naval Research, and the Presidential Early Career Award for Scientists and Engineers from the White House, both in 2000. In addition, he was awarded the Best Author Award from the Japan Society for Industrial and Applied Mathematics (JSIAM) as well as the JSIAM Letters Best Paper Award jointly with Dr. J. Irion both in 2016. He is a member of the Institute of Mathematical Statistics, Society for Industrial and Applied Mathematics, and JSIAM. He also served as Chair of the SIAM Activity Group on Imaging Science from 2013 to 2015, and is as a member of the Editorial Board of the three international journals: *Applied and Computational Harmonic Analysis*, *Inverse Problems and Imaging*, and *Journal of Mathematical Imaging and Vision*.