SAMPLING THEORY FOR GRAPH SIGNALS

WITH APPLICATIONS TO SEMI-SUPERVISED LEARNING

by

AAMIR ANIS

DISSERTATION

presented to the

FACULTY OF THE USC GRADUATE SCHOOL

UNIVERSITY OF SOUTHERN CALIFORNIA

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

December 2017

To my family.

Abstract

The representation, processing and analysis of large-scale data as signals defined over graphs has drawn much interest recently. Graphs allow us to embed natural inter-connectivities between data points and exploit them during processing. As a result, graph signal processing has laid a strong foothold in various modern application domains such as machine learning, analysis of social, transportation, web and sensor networks, and even traditional areas such as image processing and video compression. Although powerful, this research area is still in its infancy. Recent efforts have therefore focused on translating well-developed tools of traditional signal processing for handling graph signals.

An important aspect of graph signal processing is defining a notion of frequency for graph signals. A frequency domain representation for graph signals can be defined using the eigenvectors and eigenvalues of variation operators (e.g., graph Laplacian) that take into account the underlying graph connectivity. These operators can also be used to design graph spectral filters. The primary focus of our work is to develop a theory of sampling for graph signals that answers the following questions: 1. When can one recover a graph signal from its samples on a given subset of nodes of the graph? 2. What is the best choice of nodes to sample a given graph signal? Our formulation primarily works under the assumption of bandlimitedness in the graph Fourier domain, which amounts to smoothness of the signal over the graph. The techniques we employ to answer these questions are based on the introduction of special quantities called graph spectral proxies that allow our algorithms to operate in the vertex domain, thereby admitting efficient, localized implementations.

We also explore the sampling problem in the context of designing wavelet filterbanks on graphs. This problem is fundamentally different since one needs to choose a sampling scheme jointly over multiple channels of the filterbank. We explore constraints

for designing perfect reconstruction two-channel critically-sampled filterbanks with low-degree polynomial filters, and conclude that such a design is in general not possible for arbitrary graphs. This leads us to propose an efficient technique for designing a critical sampling scheme that, given predesigned filters, aims to minimize the overall reconstruction error of the filterbank. We also explore M-channel filterbanks over M-block cyclic graphs (that are natural extensions of bipartite graphs), and propose a tree-structured design in a simpler setting when M is a power of 2.

As an application, we study the graph-based semi-supervised learning problem from a sampling theory point of view. A crucial assumption here is that class labels form a smooth graph signal over a similarity graph constructed from the feature vectors. Our analysis justifies this premise by showing that in the asymptotic limit, the bandwidth (a measure of smoothness) of any class indicator signal is closely related to the geometry of the dataset. Using the sampling theory perspective, we also quantitatively show that the label complexity (i.e., the amount of labeling required for perfect prediction of unknown labels) matches its theoretical value, thereby adding to the appeal of graph-based techniques for semi-supervised learning.

Acknowledgments

The journey of my doctoral studies would not have been a success without the help and support of a number of people around me. First and foremost, I would like to express my sincerest gratitude towards my advisor Prof. Antonio Ortega, for giving me the opportunity of undertaking this endeavor. His guidance and encouragement have been pivotal for my progress, especially when I have fumbled along the way. I am deeply indebted to him for inculcating in me the discipline necessary for pursuing research, through his wisdom and even through his everlasting red pen.

I am very grateful to Prof. Salman Avestimehr for being a part of my committee and for stimulating my interest in theoretical problems through our collaborations. I would like to thank Prof. Shri Narayanan, Prof. Mahdi Soltanolkotabi, Prof. Yan Liu for serving on my qualifying exam committee and providing me with valuable feedback necessary for improving my thesis. I would also like to thank Prof. Jinchi Lv for being a part of my defense committee. I am indebted to all my teachers at USC for imparting the finest quality of teaching through a wide variety of courses. I am thankful to USC and its staff for providing all the facilities conducive to a good research atmosphere. I must also thank my research collaborators Akshay Gadde, Aly El Gamal, Eduardo Pavez, Nicolo Michelusi, and Prof. Urbashi Mitra. My discussions with them have been immensely enlightening.

It would be remiss of me not to mention the crucial role of my friends Praveen, Akshay, Ruchir, Arvind and Naveen. Their company and moral support has alleviated much of the psychological stress associated with a life in research, for which I am thoroughly grateful. I would also like to thank my labmates Hilmi, Eduardo, Jessie, Joanne and Eric for fulfilling a similar role in the office.

Most importantly, to my Mom, Dad and Little Brother Fahad, for being a never-ending source of love, inspiration, motivation, strength, and encouragement, and for their blind faith in me no matter what – I cannot thank them enough!

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recent improvements in communication and electronic storage capabilities have created an explosion in the amount of data generated from a variety of sources, so much so that we have already stretched the boundaries of current data processing techniques and ventured into unchartered territory. The exponential improvement in raw computing power can only do so much to alleviate this problem – partly because current processing techniques do not scale very well with the size of datasets, but more importantly because "modern" data is quite unstructured and irregular, and hence, there is an alarming shortage of techniques required for understanding and processing such data.

Graph-structured data is one such form of unconventional data that cannot be easily represented by regularly or irregularly spaced samples in a metric space. In many application domains, such as social networks, web information analysis, sensor networks and machine learning, a natural representation of the dataset can be provided by a graph. Each observation or data point is attached to one of the vertices/nodes in the graph, and the edges (directed or undirected, weighted or unweighted) capture important characteristics of the dataset, *e.g.*, similarity or proximity between the vertices, link capacity *etc.*. Specifically, such a mapping from the vertices of a graph to the set of scalars is known as a graph signal. In order to study various properties of these signals (such as smoothness, anomalous behavior *etc.*), one needs to take into account the underlying connectivity of the data points. Common examples of graph signals include user ratings or preferences on social graphs, class labels on similarity graphs in machine learning, signal values on sensor nodes in a sensor network, collaborative filtering, *etc.* (Figure 1.1). Further, even traditional applications such as image and video processing, and compression, can benefit from a graph-based formulation [47, 30]. The set of tools that facilitate understanding and processing graph signals are broadly covered by the emerging research area known as *Graph Signal Processing* [66, 62].

Traditional signals in the Euclidean space can be analyzed and processed using well-established techniques such as sampling, filtering, frequency transforms, *etc.*. Graph Signal Processing aims to extend these tools for signals on graphs by exploiting the underlying connectivity information. These extensions in most cases are non-trivial since
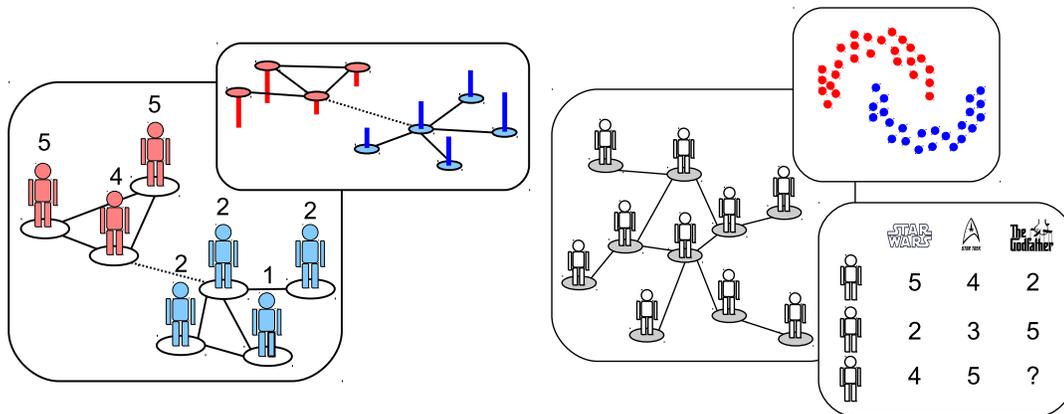
Figure 1.1: Commonly occurring examples of graph-structured data in modern application areas such as ranking, user ratings prediction over social networks, collaborative filtering and semi-supervised learning.

graph signals lie in irregular non-Euclidean spaces. For example, defining analogs of even the simplest signal processing operations such as time shift, scaling, and downsampling is not immediately obvious for graph signals. An additional consideration in graph signal processing is complexity, therefore greater emphasis is laid on the design of localized algorithms, where the output at each vertex depends only on its local neighborhood. This allows algorithms to scale well with large graph sizes that have become increasingly common in modern big data applications.

The primary focus of this dissertation is to formulate a sampling theory for graph signals that is meant to form a Graph Signal Processing analog of the Nyquist-Shannon sampling theorem. Contrary to the more common uniformly-spaced sampling strategy in traditional signals, sampling in graphs can consist of choosing any subset of vertices over the graph. Therefore, in our formulation, we seek an answer for the following questions:

1. When can one recover a graph signal from a given subset of sampled vertices on the graph?

2. How can one choose the best vertices to sample a given graph signal?

Next, we consider the sampling problem in the context of designing wavelet filterbanks on graphs. Finally, as an application, we view the problem of graph-based semi-supervised learning from the perspective of sampling theory. A detailed description of the research problems studied in this dissertation is provided in the following sections.

1.1 Sampling theory for graph signals

The Nyquist-Shannon sampling theorem is a landmark result in signal processing theory that forms a bridge between continuous-time and discrete-time signals. It establishes a sufficient condition on the sampling rate required to capture all the information of continuous-time signals satisfying certain modeling assumptions (most commonly, bandlimitedness in the spectral domain). For example, the simplest form of the sampling theorem states that a signal with bandwidth f in the Fourier domain can be uniquely represented by its samples captured uniformly at a rate of at least $2f$. An analogue of the sampling theorem in the digital signal processing domain specifies conditions on downsampling bandlimited discrete-time signals without losing information. There also exist alternate variants of the sampling theorem under different sampling regimes such as non-uniform sampling [31], or different modeling assumptions (for instance, non-baseband signals, signals with finite rate of innovation [73]), or a combination of these in the more recent compressed sensing literature [18].

Similarly, in the context of Graph Signal Processing, the sampling theorem gives conditions under which a graph signal can be uniquely represented by its samples on a subset of nodes. This problem is fundamentally different from traditional sampling due to a lack of "ordering" of the samples, i.e., one cannot select uniformly spaced samples. Further, signal properties such as smoothness/bandlimitedness are determined by the topology or connectivity of the graph. As we shall see in Chapter 2, similar to the Fourier domain for traditional signals, a graph Fourier domain is obtained if one is able to represent graph signals in a basis that takes into account the underlying connectivity of the graph. Such a basis can be obtained using the eigenvalues and eigenvectors of special matrices associated with the graph that capture variation in a graph signal (example of such matrices include the adjacency [63], the Laplacian [66], and its normalized versions [4, 24]). The representation of a graph signal in this basis is known as the Graph Fourier Transform (GFT) and is used to provide a notion of frequency for graph signals, which is crucial for developing the theory of sampling.

In order to formulate a sampling theorem for graph signals, one needs to consider the following questions –

P1. *When can one recover a graph signal from its samples on a given subset of nodes?*
Not all graph signals can be uniquely represented from their samples on a subset of nodes. Since sampling results in an inherent reduction of dimensionality, we need to impose certain modeling assumptions on signals for recovery. In this work,

we assume that our signals of interest are smooth over the graph. In quantitative terms, this can be imposed by enforcing signals to be bandlimited or *lowpass* in the graph Fourier domain. Therefore, for a given subset of nodes, we would like to find the minimum bandwidth a signal can have in the graph Fourier domain (also known as the cutoff frequency) so that it can be perfectly recovered from its samples on the subset.

P2. *What is the best choice of nodes to sample a given graph signal?*
Under the modeling assumption of bandlimitedness, we would like to find a set of nodes of smallest possible size that uniquely represents all signals with bandwidth lower than a given value. However, not all choices of sampling sets exhibit robustness of reconstruction in the presence of noise and model mismatch, therefore we would also like to focus on choosing sets that promote stable reconstructions of the original signal. Note that the algorithms we consider for choosing the sets should be scale well with graph sizes, since graphs in modern applications are quite large.

P3. *How can we reconstruct a signal from its samples?*
Given samples of a graph signal on a set that satisfies the criteria posed by the two questions above, one would like to design efficient algorithms for recovering the entire signal. This is equivalent to the interpolation problem in traditional signal processing, however, here we must focus on the design of localized algorithms, where the signal value on a particular node can be computed from values on neighboring nodes upto a certain connection depth.

Sampling theory for graph signals has applications in many research areas, most prominent being machine learning and image processing. In the former case, class labels are treated as graph signals over the similarity graph and predicting unknown labels can be posed as an interpolation problem. In image processing, graphs can be used to incorporate edge information in the images and are useful for designing edge-adaptive wavelet transforms over images.

1.1.1   Existing work

Prior to our work, sampling theory for graph signals was considered first in [55], where a sufficient condition was given for unique recovery of signals from a given subset of nodes (called a sampling set). Using this condition, [49] computes a bound on the maximum bandwidth that a signal can have, so that it can be uniquely reconstructed

from its samples on a given subset of nodes. The uniqueness conditions are in fact special cases of results pertaining to sampling signals from arbitrary subspaces presented in [28]. Results along the same lines have been considered recently in [65, 22] after our first work [3]. However, in order to apply results from the aforementioned papers, one must have an explicit representation of the GFT basis corresponding to the bandlimited space of interest thereby limiting their practical utility when the graphs are large. To our knowledge, our approach is the first to circumvent this issue. Sampling theory for graph signals has also been studied in other contexts such as designing filterbanks on graphs [44, 52], or more recently along the lines of alternative sampling strategies such as shift-based sampling [43] and random sampling [57], and in connection with the uncertainty principle for graph signals [71, 72].

1.1.2   Our contributions

The primary contribution of this dissertation is to develop a sampling theorem for bandlimited graph signals by answering the questions posed in Section 1.1. We proceed by first stating a necessary and sufficient condition under which a bandlimited signal can be perfectly recovered from a given subset of nodes on the graph. This condition guarantees uniqueness of sampling and is used to compute, for a given sampling set, the maximum bandwidth (i.e., the cutoff frequency) that a signal can have so that it can be perfectly reconstructed from its samples (thereby answering P1). Further, this formulation also avoids explicit computation of the GFT basis for the graph by defining quantities called *Graph Spectral Proxies* that allow a trade-off between complexity and accuracy while computing the cutoff frequency. These quantities are based on a method for estimating the bandwidth of any signal, upto different orders of accuracy, through simple localized operations in the vertex domain.

In order to obtain the best sampling set (i.e., problem P2), we propose to use the cutoff frequency, estimated via spectral proxies, as the objective function that is maximized subject to a bandwidth constraint (or equivalently, a cardinality constraint that is determined by the dimensionality of the space of signals to be sampled). This optimization problem is however combinatorial in nature, therefore we resort to a greedy algorithm that uses binary relaxation and gradient ascent. Although the spectral proxies-based objective function is designed to obtain sampling sets that guarantee uniqueness, we show that it also considers robustness of reconstruction with respect to sampling noise and model mismatch. This is because maximizing the objective function is shown to

be closely related to minimizing a bound on the worst case reconstruction error associated with a given sampling set. Further, our method is iterative and requires finding a minimum eigenpair as the atomic operation, thereby making it efficient and amenable for localized/distributed implementations. This is demonstrated through various experiments that compare our sampling set selection method against other existing methods in terms of robustness and complexity.

In this dissertation, we exclude the study of techniques for bandlimited reconstruction of signals from their samples (i.e., problem P3) since they have been studied independently elsewhere [50, 74, 76]. An example of a real-world application of our sampling theory is the problem of active semi-supervised learning where the learning algorithm can specify beforehand specific data points to label given a budget. Under the assumption that class labels are smooth over the similarity graph, active learning is essentially equivalent to a sampling set selection problem. It is shown in [29] that our method performs well in comparison to other state-of-the art methods. The details of this work are excluded from this dissertation for brevity.

1.2 Wavelet filterbanks on graphs

Graph wavelet transforms have recently been used for a variety of applications, ranging from multiresolution analysis [47, 60], compression [48, 53, 19], denoising [25], and classification [27]. These transforms allow one to analyze and process signals defined over graphs while taking into consideration the underlying relationships between signal values. The designs of these transforms are generally inspired by traditional wavelet construction schemes and leverage principles from graph signal processing.

One of the recent techniques for constructing wavelet transforms on graphs is based on filterbanks. This approach is quite appealing because it makes use of spectral graph filters [66] that have a low complexity and enable a trade-off between vertex-domain and frequency-domain localization. Besides designing the filters, sampling set selection is an important aspect of graph wavelet filterbank design. The requirements are fundamentally different in this case – instead of working with bandlimited signals for perfect and stable recovery, one looks for a sampling scheme over two or more channels of a filterbank that allows a multiresolution analysis, followed by recovery (or synthesis) of all signals over the graph. An important property taken into consideration while designing these filterbanks is critical sampling, which requires the total number of samples across channels to be equal to the number of nodes in the graph [45, 46]. Other desirable properties

of the sampling scheme involve ensuring near perfect reconstruction, near orthogonality, and most importantly compact support that allows localized implementations of the filterbank. These properties are suitable for compression applications where one would like to obtain compact respresentations of signals of interest efficiently at the encoder, followed by near lossless recovery at the decoder. Unlike traditional filterbanks that boast numerous well-studied designs with wide-ranging properties, graph filterbanks are more difficult to design as the structure of the Fourier basis is influenced by the structure of the graph. Therefore, it is useful to understand and characterize graphs that admit filterbanks with one or more of these properties.

1.2.1 Existing work

In order to ensure perfect reconstruction and compact support, state-of-the-art wavelet filterbanks require imposing certain structural constraints on the underlying graph. For example, the recently proposed two-channel filterbanks in [45, 46] are designed specifically for bipartite graphs. The special structure leads to a natural downsampling-upsampling scheme (on one of the two partitions) in each channel, accompanied by a spectral folding phenomenon that is exploited while designing the filters. The design can be extended to arbitrary graphs through a multidimensional approach that involves hierarchically decomposing the original graph into a sequence of bipartite subgraphs. Techniques for bipartite subgraph decomposition such as those in [52, 77] form an active area of research. There also exist filterbanks that exploit circulant graph architectures, albeit with non-polynomial synthesis filterbanks [26, 27]. Recently, yet another class of graphs suitable for the design of M-channel polynomial filterbanks have been studied [69, 70]. These are called M-block cyclic graphs and are a natural directed extension of bipartite graphs. The topology and the eigenstructure of these graphs also induces a spectral folding upon downsampling-upsampling on one of the blocks, thus providing a means to extend several concepts from classical multirate signal processing to the graph signal processing domain. Perfect reconstruction conditions for M-channel filterbanks on these graphs are stated in [70], albeit with a lack of concrete solutions.

1.2.2 Our contributions

Following the lines of [45], our work begins by spelling out design criteria for two-channel filterbanks that satisfy the critical sampling, compact support, and perfect reconstruction properties. We realize immediately that without the presence of special

structure in the Fourier basis associated with the graph, it is in general difficult to satisfy critical sampling and perfect reconstruction simultaneously with low-degree polynomial filterbanks. Our analysis shows that the structural requirement on the Fourier basis is identical to that of traditional signal processing – downsampling-upsampling should induce a spectral folding in the graph Fourier domain. This makes bipartite graphs particularly amenable to the design of perfect reconstruction two-channel filterbanks.

We shift our focus and seek to design filterbanks on arbitrary graphs (without altering their topology) that are critically-sampled, have compact support, and satisfy the perfect reconstruction and orthogonality conditions as closely as possible. The lack of a special structure makes the problem of jointly designing low-degree polynomial filters and the sampling scheme impossible. Therefore, as a starting step, we decouple the two by focusing only on designing a critical sampling scheme while assuming that the analysis and synthesis filters are predesigned for given frequency localization constraints. Our main contributions are the following:

- A criterion based on the reconstruction error to evaluate any sampling scheme for given predesigned filters.

- A greedy but computationally efficient algorithm to minimize the error criterion that approximates the optimal solution, along with some theoretical guarantees.

Experiments show that our algorithm for choosing the sampling scheme performs better than existing heuristics.

Finally, we turn our attention to the design of M-channel filterbanks on M-block cyclic graphs. The perfect reconstruction conditions stated in [70] are meant for graphs with balanced block sizes, where signals in each channel are sampled on the same block. We first modify the conditions to work for disjoint sampling sets across different channels. Next, we look into the design of possible polynomial solutions satisfying these conditions. This problem is more complex than traditional M-channel filterbanks since the graph Fourier domain (of graphs with a normalized adjacency) spans the complete unit-disc instead of the unit-circle. In our work, we consider a simpler version of the problem where M is a power of 2, and propose a tree-structured design composed of hierarchically arranged two-channel designs. Our design allows disjoint sampling sets for different channels, and also works for unbalanced block sizes. Our main contributions are:

- A scheme to represent a 2^L-channel filterbank on 2^L-block cyclic graphs as a hierarchical tree-structure composed of 2-channel filterbanks on smaller directed bipartite graphs.

8

- A perfect reconstruction 2-channel filterbank solution for a 2-block cyclic graph (i.e., a directed bipartite graph). This is a generalization of the designs in [45, 46, 68], since the spectrum of directed bipartite graphs can be complex.

1.3 Sampling theory perspective of graph-based semi-supervised learning

The abundance of unlabeled data in various machine learning applications, along with the prohibitive cost of labeling, has led to growing interest in semi-supervised learning. This paradigm deals with the task of classifying data points in the presence of very little labeling information by relying on the geometry of the dataset. Assuming that the features are well-chosen, a natural assumption in this setting is to consider the marginal density $p(\mathbf{x})$ of the feature vectors to be informative about the labeling function $f(\mathbf{x})$ defined on the points. This assumption is fundamental to the semi-supervised learning problem both in the classification and the regression settings, and is also known as the *semi-supervised smoothness assumption* [20], which states that the label function is smoother in regions of high data density. There also exist other similar variants of this assumption specialized for the classification setting, namely, the *cluster assumption* [78] (points in a cluster are likely to have the same class label) or the *low density separation assumption* [51] (decision boundaries pass through regions of low data density). Most present day algorithms for semi-supervised learning are based on one or more of these assumptions.

In practice, graph-based methods have been found to be quite suitable for geometry-based learning tasks, primarily because they provide an easy way of exploiting information from the geometry of the dataset. These methods involve constructing a distance-based similarity graph whose vertices represent the data points and edge weights are computed using a decreasing function of Euclidean distances between the points in the feature space. The class labels are treated as a graph signal, and the known labels as its samples. The semi-supervised smoothness assumption for the class labels translates into a notion of "smoothness" of the signal over the graph, in the sense that labels of vertices do not vary much over edges with high weights (i.e., edges that connect close or similar points). This smoothness assumption can be imposed through quantities such as the *graph cut*, or the *graph Laplacian regularizer*, and more recently, through *bandwidth*

constraints in the graph spectral domain. Further, predicting labels of the unknown points can be considered as an intepolation problem over the similarity graph.

Although using the bandwidth of signals to impose smoothness is well-motivated in graph-based learning methods, it is important to understand its connection to the underlying geometry of the dataset in a theoretical sense. One way of justifying this approach is to explore its geometrical interpretation in the limit of infinitely available unlabeled data. This typically involves assuming a probabilistic generative model for the dataset and analyzing the bandwidth of class indicator functions in the asymptotic setting for certain commonly-used graph construction schemes. Specifically, in this setting, we seek answers for the following questions in the asymptotic setting:

1. What is the connection between the bandwidth of class indicator signals over the similarity graph and the underlying geometry of the data set?

2. What is the interpretation of the bandlimited recontruction approach for label prediction?

3. How many labeled examples does one require for perfect prediction?

The answers to these questions would help complete our theoretical understanding of graph-based semi-supervised classification approaches, specifically bandlimited interpolation over the graph, and strengthen their link with the semi-supervised smoothness assumption and its variants.

1.3.1 Existing work

In the graph-based semi-supervised learning paradigm, there have been numerous ways of quantitatively imposing smoothness constraints over label functions defined on vertices of a similarity graph. Most graph-based semi-supervised classification algorithms incorporate one of these criteria as a penalty against the fitting error in a regularization problem, or as a constraint term while minimizing the fitting error in an optimization problem. For example, a commonly used measure of smoothness for a label function $\mathbf{f}$ is the graph Laplacian regularizer $\mathbf{f}^T \mathbf{L} \mathbf{f}$ ($\mathbf{L}$ being the graph Laplacian), and many algorithms involve minimizing this quadratic energy function while ensuring that $\mathbf{f}$ satisfies the known set of labels [83, 78]. There also exist higher-order variants of this measure known as iterated graph Laplacian regularizers $\mathbf{f}^T \mathbf{L}^m \mathbf{f}$, that have been shown to make the problem more well-behaved [81]. On the other hand, a spectral theory based classification algorithm restricts $\mathbf{f}$ to be spanned by the first few eigenvectors of the graph

Laplacian [10, 11], that are known to form a representation basis for smooth functions on the graph. In each of the examples, the criterion enforces smoothness of the labels over the graph – a lower value of the regularizer $\mathbf{f}^T\mathbf{L}\mathbf{f}$, and a smaller number of leading eigenvectors to model $\mathbf{f}$ imply that vertices that are close neighbors on the graph are more likely to have the same label.

Recent works have therefore focused on justifying these approaches by exploring their geometrical interpretation in the limit of infinitely available unlabeled data. This is typically done by assuming a probabilistic generative model for the dataset and analyzing the graph smoothness criteria in the asymptotic setting for certain commonly-used graph construction schemes. For example, it has been shown that for data points drawn from a smooth distribution with an associated smooth label function (i.e., the regression setting), the graph Laplacian regularizer converges in the limit of infinite data points to a density-weighted variational energy functional that penalizes large variations of the labels in high density regions [11, 15, 34, 12, 81, 82]. A similar connection ensues for semi-supervised learning problems in the classification setting (i.e., when labels are discrete in the feature space). If points drawn from a smooth distribution are separated by a smooth boundary into two classes, then the graph cut for the partition converges to a weighted volume of the boundary [51, 42]. This is consistent with the low density separation assumption – a low value of the graph cut implies that the boundary passes through regions of low data density.

1.3.2 Our contributions

The main contribution of our work is a novel theoretical justification of the sampling theoretic approach to semi-supervised learning. This approach involves treating the class label function/indicator signal as a *bandlimited* graph signal, and label prediction as a *bandlimited reconstruction* problem. Our work is the first to consider, using sampling theory, the *label complexity* of semi-supervised classification, that is, the minimum fraction of labeled examples required for perfect prediction of the unknown labels. A key ingredient in this formulation is the *bandwidth* of signals on the graph – signals with lower bandwidth tend to be smoother on the graph, and are useful for modeling label functions over similarity graphs. Label prediction using bandlimited reconstruction then involves estimating a label function/indicator signal that minimizes prediction error on the known set under a bandwidth constraint.

In order to provide a geometrical interpretation of bandwidth of class indicator signals, we leverage our work on sampling theory and make use of spectral proxies. These spectral proxies are computed over similarity graphs constructed from a two-class statistical model for the feature vectors. To make our analysis as general as possible, we consider two data models: *separable* and *nonseparable*. These generative models are quite practical and mimic most datasets in the real world. The separable model assumes that data points are drawn from an underlying probability distribution in the feature space and each class is separated from the others by a smooth boundary. On the other hand, the nonseparable model assumes a mixture distribution for the data where the data points are drawn randomly and independently with certain probability from separate class conditional distributions. We also introduce a notion of "boundaries" for classes in the nonseparable model in the form of *overlap* regions, defined as the set of points where the probability of belonging and not belonging to a class are both non-zero (i.e., the region of ambiguity). This definition is quite practical and useful for characterizing the geometry of such datasets.

Using the data points, we consider a specific graph construction scheme that applies the Gaussian kernel over Euclidean distances between feature vectors for computing their similarities (our analysis can be generalized easily to arbitrary kernels under simple assumptions). In order to compute the bandwidth of the indicator, we use graph spectral proxies from our work on sampling theory. A significant portion of this work focuses on analyzing the stochastic convergence of this bandwidth estimate (using variance-bias decomposition) in the limit of infinite data points for any class indicator signal on the graph. The analysis in our work suggests a novel sampling theoretic interpretation of graph-based semi-supervised learning and the main contributions can be summarized as follows:

1. *Relationship between bandwidth and data geometry.* For the separable model, we show that under certain rate conditions, the bandwidth estimate for any class converges to the supremum of the data density over the class boundary. Similarly, for the nonseparable model, we show that the bandwidth estimate converges to the supremum of the density over the overlap region.

2. *Interpretation of bandlimited reconstruction.* Using the geometrical interpretation of the bandwidth, we conclude that bandlimited reconstruction allows one to choose the complexity of the hypothesis space while predicting unknown labels (i.e., a larger bandwidth allows more complex class boundaries).

3. *Quantification of label complexity.* We also show that the bandwidth of class indicator signals is closely linked theoretically to the fraction of labeled points required for perfect classification which is in turn related to the geometry of the data.

Our analysis has significant implications: Firstly, class indicator signals have a low bandwidth if class boundaries lie in regions of low data densities, that is, the semi-supervised assumption holds for graph-based methods. And secondly, our analysis also helps quantify the impact of bandwidth and data geometry in semi-supervised learning problems, an aspect that was lacking in existing work. Specifically, our results enable us to theoretically assert that for the sampling theoretic approach to graph-based semi-supervised learning, the label complexity (minimum fraction of labeled points required) of learning classifiers matches the theoretical estimate and is indeed lower if the boundary lies in regions of low data density, as demonstrated empirically in earlier works [10, 11].

1.4   Outline

The rest of this dissertation is organized as follows:

- Chapter 2 begins by introducing notations used throughout this dissertation, along with several useful definitions. This is followed by a brief review of graph signal processing concepts and the introduction of notions of frequencies over graphs.

- In Chapter 3, we formulate our theory of sampling for bandlimited graph signals. We begin by stating necessary and sufficient conditions for unique sampling, using which we formulate methods to compute the cutoff frequency for a given sampling set and conversely, the best sampling set of a given size. We also briefly discuss stability considerations in sampling.

- Chapter 4 considers the sampling problem in a different, but related, problem of designing graph filterbanks. We consider a general formulation for designing two-channel filterbanks on arbitrary graphs and study conditions required for satisying desirable properties such as critical sampling, compact support, perfect reconstruction and orthogonality. We characterize graphs that admit solutions satisfying these properties. We then present an efficient algorithm for selecting the best sampling scheme over the channels that aims to minimize reconstruction error,

given predesigned filters. Finally, we also consider the design of M-channel filterbanks on M-block cyclic graphs and propose a simple tree-structured design.

- Finally, in Chapter 5, we view graph-based semi-supervised learning from the perspective of sampling theory. This point of view allows us to explain why graph-based methods are suited for this problem by looking at the geometrical interpretations of bandwidth of class indicator signals in the asymptotic limit. Further, using sampling theory, we also shed light on the label complexity of the semi-supervised learning problem.

Chapter 2

Review of Graph Signal Processing

In this chapter, we review relevant concepts from Graph Signal Processing that will be helpful in formulating our sampling theory. The most important concept is the notion of frequency for graph signals that will help us characterize their level of smoothness. The key idea is to introduce analogs of operators such as variation or shift from traditional signal processing, that allow one to transform a signal or measure its properties while taking into account the underlying connectivity on the graph. Such operators create a notion of smoothness for graph signals through their spectrum. There can be many different choices of variation operators depending the application at hand, we review a few examples after introducing notations used throughout this proposal.

2.1 Basic concepts

A graph $G = (\mathcal{V}, \mathcal{E})$ is a collection of nodes indexed by the set $\mathcal{V} = \{1, \ldots, N\}$ and connected by links $\mathcal{E} = \{(i, j, w_{ij})\}$, where (i, j, w_{ij}) denotes a link of weight $w_{ij} \in \mathbb{R}^+$ pointing from node i to node j. The adjacency matrix $\mathbf{W}$ of the graph is an $N \times N$ matrix with elements w_{ij}. The degree d_i of a node i in an undirected graph is defined as $\sum_j w_{ij}$, and the degree matrix of the graph is defined as $\mathbf{D} = \mathrm{diag}\{d_1, d_2, \ldots, d_N\}$. For directed graphs, one can define in-degrees and out-degrees separately.

A graph signal is a function $f : \mathcal{V} \to \mathbb{R}$ defined on the vertices of the graph, (i.e., a scalar value assigned to each vertex, such that $f(i)$ is the value of the signal on node i). It can be represented as a vector $\mathbf{f} \in \mathbb{R}^N$ where $\mathbf{f}_i$ represents the function value on the ith vertex.

Sampling sets are subset of nodes $\mathcal{S} \subset \mathcal{V}$ over which the values of a signal are measured. For any signal $\mathbf{x} \in \mathbb{R}^N$ and a set $\mathcal{S} \subseteq \{1, \ldots, N\}$, we use $\mathbf{x}_{\mathcal{S}}$ to denote its sampled version, which is an $|\mathcal{S}|$-dimensional sub-vector of $\mathbf{x}$ consisting of components indexed by $\mathcal{S}$. Similarly, for any matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{A}_{\mathcal{S}_1 \mathcal{S}_2}$ is used to denote the sub-matrix of $\mathbf{A}$ with rows indexed by $\mathcal{S}_1$ and columns indexed by $\mathcal{S}_2$. For simplicity, we denote $\mathbf{A}_{\mathcal{S}\mathcal{S}}$ by $\mathbf{A}_{\mathcal{S}}$. The complement of $\mathcal{S}$ in $\mathcal{V}$ is denoted by $\mathcal{S}^c = \mathcal{V} \setminus \mathcal{S}$.
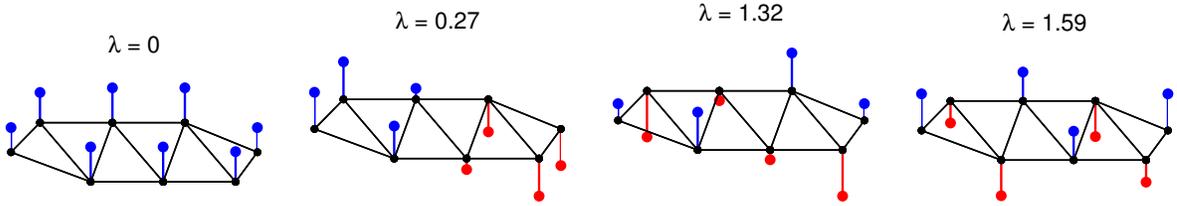
Figure 2.1: Variation in the eigenvectors of the Laplacian of a graph ordered according to the eigenvalues. Note the increasing variation over edges.

2.2 Notion of frequency for graph signals

In order to formulate a sampling theorem for graph signals, we need a notion of frequency that helps us in characterizing the degree of smoothness of various signals with respect to the graph. This can be done by defining shift or variation operators that allow one to transform a signal or measure its properties while taking into account the underlying connectivity over the graph. We denote any operator that measures variation of signals over the graph by $\mathbf{L}$, an $N \times N$ matrix[1]. This operator essentially creates a notion of smoothness for graph signals through its spectrum. Specifically, assume that $\mathbf{L}$ has eigenvalues $|\lambda_1| \leq \ldots \leq |\lambda_N|$ and corresponding eigenvectors $\{\mathbf{u}^1, \ldots, \mathbf{u}^N\}$. Then, these eigenvectors provide a Fourier-like basis for graph signals with the frequencies given by the corresponding eigenvalues. For each $\mathbf{L}$, one can also define a variation functional $\mathrm{Var}(\mathbf{L}, \mathbf{f})$ that measures the variation in any signal $\mathbf{f}$ with respect to $\mathbf{L}$. Such a definition should induce an ordering of the eigenvectors which is consistent with the ordering of eigenvalues (see for example, Figure 2.1). More formally, if $|\lambda_i| \leq |\lambda_j|$, then $\mathrm{Var}(\mathbf{L}, \mathbf{u}^i) \leq \mathrm{Var}(\mathbf{L}, \mathbf{u}^j)$.

The GFT $\tilde{\mathbf{f}}$ of a signal $\mathbf{f}$ is given by its representation in the above basis, $\tilde{\mathbf{f}} = \mathbf{U}^{-1}\mathbf{f}$, where $\mathbf{U} = [\mathbf{u}^1 \ldots \mathbf{u}^N]$. Note that a GFT can be defined using any variation operator. Examples of possible variation operators are reviewed in the next section. If the variation operator $\mathbf{L}$ is symmetric then its eigenvectors are orthogonal, leading to an orthogonal GFT. In some cases, $\mathbf{L}$ may not be diagonalizable. In such cases, one can resort to the Jordan normal form [63] and use generalized eigenvectors.

---

[1]Although $\mathbf{L}$ has been extensively used to denote the combinatorial Laplacian in graph theory, we overload this notation to make the point that any such variation operator can be defined to characterize signals of interest in the application at hand.

Using this notion of frequency, filters can be designed on the graph using any of the variation operators by manipulating its spectral response to satisfy desired properties. One useful way is to consider polynomial graph filters $\mathbf{H} = h(\mathbf{L}) = \sum_{i=0}^{k} h_i \mathbf{L}^i$ of different degrees, whose response in the spectral domain is given by the polynomial $h(\lambda) = \sum_{i=0}^{k} h_i \lambda^i$. These are particularly useful because of their simplicity – a k-degree polynomial filter can be implemented in $O(k|\mathcal{E}|)$ complexity. Further, note that for undirected graphs, $\mathbf{L}$ is symmetric, and hence $\mathbf{H}$ is symmetric.

Bandlimited graph signals

A signal $\mathbf{f}$ is said to be ω-bandlimited if $\tilde{\mathbf{f}}_i = 0$ for all i with $|\lambda_i| > \omega$. In other words, GFT of an ω-bandlimited[2] $\mathbf{f}$ is supported on frequencies in $[0, \omega]$. If $\{\lambda_1, \lambda_2, \ldots, \lambda_r\}$ are the eigenvalues of $\mathbf{L}$ less than or equal to ω in magnitude, then any ω-bandlimited signal can be written as a linear combination of the corresponding eigenvectors:

$$\mathbf{f} = \sum_{i=1}^{r} \tilde{\mathbf{f}}_i \mathbf{u}^i = \mathbf{U}_{\mathcal{VR}} \tilde{\mathbf{f}}_{\mathcal{R}}, \tag{2.1}$$

where $\mathcal{R} = \{1, \ldots, r\}$. The space of ω-bandlimited signals is called Paley-Wiener space and is denoted by $PW_\omega(G)$ [55]. Note that $PW_\omega(G) = \mathrm{range}(\mathbf{U}_{\mathcal{VR}})$ (*i.e.*, the span of columns of $\mathbf{U}_{\mathcal{VR}}$). Bandwidth of a signal $\mathbf{f}$ is defined as the largest among absolute values of eigenvalues corresponding to non-zero GFT coefficients of $\mathbf{f}$, i.e.,

$$\omega(\mathbf{f}) \triangleq \max_i \{|\lambda_i| \mid \tilde{\mathbf{f}}_i \neq 0\}. \tag{2.2}$$

A key ingredient in our theory is an approximation of the bandwidth of a signal using powers of the variation operator $\mathbf{L}$, as explained in Section 3.3. Since this approximation holds for any variation operator, the proposed theory remains valid for all of the choices of GFT in Table 2.1.

---

[2]One can also define highpass and bandpass signals in the GFT domain. Sampling theory can be generalized for such signals by treating them as lowpass in the eigenbasis of a shifted variation operator, *e.g.*, one can use $\mathbf{L}' = |\lambda_N|\mathbf{I} - \mathbf{L}$ for highpass signals.

2.3 Examples of variation operators

2.3.1 Variation on undirected graphs

In undirected graphs, the most commonly used variation operator is the combinatorial Laplacian [24] given by:

$$\mathbf{L} = \mathbf{D} - \mathbf{W}, \tag{2.3}$$

where $\mathbf{D}$ is the diagonal degree matrix $\text{diag}\{d_1, \ldots, d_N\}$ with $d_i = \sum_j w_{ij}$. Since, $w_{ij} = w_{ji}$ for undirected graphs, this matrix is symmetric. As a result, it has real non-negative eigenvalues $\lambda_i \geq 0$ and an orthogonal set of eigenvectors. The variation functional associated with this operator is known as the *graph Laplacian quadratic form* [66] and is given by

$$\text{Var}_{QF}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2. \tag{2.4}$$

One can normalize the combinatorial Laplacian to obtain the symmetric normalized Laplacian and the (asymmetric) random walk Laplacian given as

$$\mathbf{L}_{sym} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2}, \quad \mathbf{L}_{rw} = \mathbf{D}^{-1} \mathbf{L}. \tag{2.5}$$

Both $\mathbf{L}_{sym}$ and $\mathbf{L}_{rw}$ have non-negative eigenvalues. However the eigenvectors of $\mathbf{L}_{rw}$ are not orthogonal as it is asymmetric. The eigenvectors of $\mathbf{L}_{sym}$, on the other hand, are orthogonal. The variation functional associated with $\mathbf{L}_{sym}$ has a nice interpretation as it normalizes the signal values on the nodes by the degree:

$$\text{Var}_{QFsym}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{sym} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2. \tag{2.6}$$

2.3.2 Variation on directed graphs

Note that variation operators defined for directed graphs can also be used for undirected graphs since each undirected edge can be thought of as two oppositely pointing directed edges.

Variation using the adjacency matrix This approach involves posing the adjacency matrix as a *shift operator* over the graph (see [63] for details). For any signal $\mathbf{f} \in \mathbb{R}^n$,

Table 2.1: Different choices of the variation operator $\mathbf{L}$ for defining GFT bases.

Operator	Expression	Graph type	Variation functional	Properties
Combinatorial	$\mathbf{L} = \mathbf{D} - \mathbf{W}$	Undirected	$\mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2}\sum_{i,j} w_{ij}(\mathbf{f}_i - \mathbf{f}_j)^2$	Symmetric, $\lambda_i \geq 0$, $\mathbf{U}$ orthogonal
Symmetric normalized	$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$	Undirected	$\mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2}\sum_{i,j} w_{ij}\left(\frac{\mathbf{f}_i}{\sqrt{d_i}} - \frac{\mathbf{f}_j}{\sqrt{d_j}}\right)^2$	Symm. , $\lambda_i \in [0,2]$, $\mathbf{U}$ orthogonal
Random walk (undirected)	$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$	Undirected	$\|\mathbf{L}\mathbf{f}\|$	Asymmetric, $\lambda_i \geq 0$, $\mathbf{U}$ non-orthogonal
Adjacency-based	$\mathbf{L} = \mathbf{I} - \frac{1}{|\mu_{\max}|}\mathbf{W}$, $\mu_{\max}$: maximum eigenvalue of $\mathbf{W}$	Undirected/ Directed	$\|\mathbf{L}\mathbf{f}\|_p, p = 1, 2$	Asymm., non-orthog. $\mathbf{U}$ for directed graphs, $\mathrm{Re}\{\lambda_i\} \geq 0$
Hub-authority	$\mathbf{L} = \gamma(\mathbf{I} - \mathbf{T}^\top\mathbf{T}) + (1-\gamma)(\mathbf{I} - \mathbf{T}\mathbf{T}^\top)$, $\mathbf{T} = \mathbf{D}_p^{-1/2}\mathbf{W}\mathbf{D}_q^{-1/2}$, $\mathbf{D}_p = \mathrm{diag}\{p_i\}, p_i = \sum_j w_{ji}, \mathbf{D}_q = \mathrm{diag}\{q_i\}, q_i = \sum_j w_{ij}$	Directed	$\mathbf{f}^\top\mathbf{L}\mathbf{f}$, see text.	Symmetric, $\lambda_i \geq 0$, $\mathbf{U}$ orthogonal
Random walk (directed)	$\mathbf{L} = \mathbf{I} - \frac{1}{2}\left(\mathbf{\Pi}^{1/2}\mathbf{P}\mathbf{\Pi}^{-1/2} + \mathbf{\Pi}^{-1/2}\mathbf{P}^\top\mathbf{\Pi}^{1/2}\right)$, $\mathbf{P}_{ij} = w_{ij}/\sum_j w_{ij}$, $\mathbf{\Pi} = \mathrm{diag}\{\pi_i\}$	Directed	$\mathbf{f}^\top\mathbf{L}\mathbf{f} = \frac{1}{2}\sum_{i,j} \boldsymbol{\pi}_i \mathbf{P}_{ij}\left(\frac{\mathbf{f}_i}{\sqrt{\boldsymbol{\pi}_i}} - \frac{\mathbf{f}_j}{\sqrt{\boldsymbol{\pi}_j}}\right)^2$	Symmetric, $\lambda_i \geq 0$, $\mathbf{U}$ orthogonal

the signal $\mathbf{Wf}$ is considered as a shifted version of $\mathbf{f}$ over the graph, analogous to the shift operation defined in digital signal processing. Using this analogy, [63] defines *total variation* of a signal $\mathbf{f}$ on the graph as

$$\text{Var}_{TV}^p(\mathbf{f}) = \left\| \mathbf{f} - \frac{1}{|\mu_{\text{max}}|} \mathbf{Wf} \right\|_p, \tag{2.7}$$

where $p = 1, 2$ and μ_{max} denotes the eigenvalue of $\mathbf{W}$ with the largest magnitude. It can be shown that for two eigenvalues $|\mu_i| < |\mu_j|$ of $\mathbf{W}$, the corresponding eigenvectors $\mathbf{v}_i$ and $\mathbf{v}_j$ satisfy $\text{Var}_{TV}^p(\mathbf{v}^i) < \text{Var}_{TV}^p(\mathbf{v}^j)$. In order to be consistent with our convention, one can define the variation operator as $\mathbf{L} = \mathbf{I} - \mathbf{W}/|\mu_{\text{max}}|$ which has the same eigenvectors as $\mathbf{W}$ with eigenvalues $\lambda_i = 1 - \mu_i/|\mu_{\text{max}}|$. This allows us to have the same ordering for the graph frequencies and the variations in the basis vectors. Note that for directed graphs, where $\mathbf{W}$ is not symmetric, the GFT basis vectors will not be orthogonal. Further, for some adjacency matrices, there may not exist a complete set of linearly independent eigenvectors. In such cases, one can use generalized eigenvectors in the Jordan normal form of $\mathbf{W}$ as stated before [63].

Variation using the hub-authority model   This notion of variation is based on the hub-authority model [39] for specific directed graphs such as a hyperlinked environment (e.g., the web). This model distinguishes between two types of nodes. Hub nodes $\mathcal{H}$ are the subset of nodes which point to other nodes, whereas authority nodes $\mathcal{A}$ are the nodes to which other nodes point. Note that a node can be both a hub and an authority simultaneously. In a directed network, we need to define two kinds of degrees for each node $i \in \mathcal{V}$, namely the in-degree $p_i = \sum_j w_{ji}$ and the out-degree $q_i = \sum_j w_{ij}$. The co-linkage between two authorities $i, j \in \mathcal{A}$ or two hubs $i, j \in \mathcal{H}$ is defined as

$$c_{ij} = \sum_{h \in \mathcal{H}} \frac{w_{hi} w_{hj}}{q_h} \quad \text{and} \quad c_{ij} = \sum_{a \in \mathcal{A}} \frac{w_{ia} w_{ja}}{p_a} \tag{2.8}$$

respectively, and can be thought of as a cumulative link weight between two authorities (or hubs). Based on this, one can define a variation functional for a signal $\mathbf{f}$ on the authority nodes [80] as

$$\text{Var}_{\mathcal{A}}(\mathbf{f}) = \frac{1}{2} \sum_{i,j \in \mathcal{A}} c_{ij} \left(\frac{f_i}{\sqrt{p_i}} - \frac{f_j}{\sqrt{p_j}} \right)^2. \tag{2.9}$$

In order to write the above functional in a matrix form, define $\mathbf{T} = \mathbf{D}_q^{-1/2}\mathbf{W}\mathbf{D}_p^{-1/2}$, where $\mathbf{D}_p^{-1/2}$ and $\mathbf{D}_q^{-1/2}$ are diagonal matrices with

$$(\mathbf{D}_p^{-1/2})_{ii} = \begin{cases} \frac{1}{\sqrt{p_i}} \text{ if } p_i \neq 0 \\ 0 \text{ otherwise,} \end{cases} \qquad (\mathbf{D}_q^{-1/2})_{ii} = \begin{cases} \frac{1}{\sqrt{q_i}} \text{ if } q_i \neq 0 \\ 0 \text{ otherwise.} \end{cases}$$

It is possible to show that $\text{Var}_{\mathcal{A}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{\mathcal{A}}\mathbf{f}$, where $\mathbf{L}_{\mathcal{A}} = \mathbf{I} - \mathbf{T}^\top\mathbf{T}$. A variation functional for a signal $\mathbf{f}$ on the hub nodes can be defined in the same way as (2.9) and can be written in a matrix form as $\text{Var}_{\mathcal{H}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{\mathcal{H}}\mathbf{f}$, where $\mathbf{L}_{\mathcal{H}} = \mathbf{I} - \mathbf{T}\mathbf{T}^\top$. A convex combination $\text{Var}_\gamma(\mathbf{f}) = \gamma\text{Var}_{\mathcal{A}}(\mathbf{f}) + (1-\gamma)\text{Var}_{\mathcal{H}}(\mathbf{f})$, with $\gamma \in [0, 1]$, can be used to define a variation functional for $\mathbf{f}$ on the whole vertex set $\mathcal{V}$. Note that the corresponding variation operator $\mathbf{L}_\gamma = \gamma\mathbf{L}_{\mathcal{A}} + (1-\gamma)\mathbf{L}_{\mathcal{H}}$ is symmetric and positive semi-definite. Hence, eigenvectors and eigenvalues of $\mathbf{L}_\gamma$ can be used to define an orthogonal GFT similar to the undirected case, where the variation in the eigenvector increases as the corresponding eigenvalue increases.

Variation using the random walk model Every directed graph has an associated random walk with a probability transition matrix $\mathbf{P}$ given by

$$\mathbf{P}_{ij} = \frac{w_{ij}}{\sum_j w_{ij}}. \tag{2.10}$$

By the Perron-Frobenius theorem, if $\mathbf{P}$ is irreducible then it has a stationary distribution $\boldsymbol{\pi}$ which satisfies $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$ [36]. One can then define the following variation functional for signals on directed graphs [23, 79]:

$$\text{Var}_{rw}(\mathbf{f}) = \frac{1}{2}\sum_{i,j}\boldsymbol{\pi}_i\mathbf{P}_{ij}\left(\frac{f_i}{\sqrt{\boldsymbol{\pi}_i}} - \frac{f_j}{\sqrt{\boldsymbol{\pi}_j}}\right)^2. \tag{2.11}$$

Note that if the graph is undirected, the above expression reduces to (2.6) since, in that case, $\boldsymbol{\pi}_i = d_i/\sum_j d_j$. Intuitively, $\boldsymbol{\pi}_i\mathbf{P}_{ij}$ can be thought of as the probability of transition from node i to j in the steady state. We expect it to be large if i is similar to j. Thus, a big difference in signal values on nodes similar to each other contributes more to the variation. A justification for the above functional in terms of generalization

of normalized cut to directed graphs is given in [23, 79]. Let $\mathbf{\Pi} = \text{diag}\{\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_n\}$. Then $\text{Var}_{\text{rw}}(\mathbf{f})$ can be written as $\mathbf{f}^\top \mathbf{L} \mathbf{f}$, where

$$\mathbf{L} = \mathbf{I} - \frac{1}{2} \left(\mathbf{\Pi}^{1/2} \mathbf{P} \mathbf{\Pi}^{-1/2} + \mathbf{\Pi}^{-1/2} \mathbf{P}^\top \mathbf{\Pi}^{1/2} \right). \tag{2.12}$$

It is easy to see that the above $\mathbf{L}$ is a symmetric positive semi-definite matrix. Therefore, its eigenvectors can be used to define an orthonormal GFT, where the variation in the eigenvector increases as the corresponding eigenvalue increases.

Table 2.1 summarizes different choices of GFT bases based on the above variation operators – our theory applies to all of these choices.

2.4 Summary

In this chapter, we introduced basic concepts of graph signal processing that our relevant to our work. Specifically, we introduced variation operators that allow us to obtain notions of frequency for graph signals and quantify their smoothness. We mentioned several examples of variation operators from existing literature both for undirected and directed graphs. A definition of the Graph Fourier Transform (GFT) is given using the eigenvalues and eigenvectors of these operators. This framework allows us to quantify smoothness of signals over the graph in terms of bandlimitedness.

Chapter 3

Sampling Theory for Graph Signals

In this chapter, we focus on developing the sampling theorem for graph signals. As mentioned earlier, our signals of interest are bandlimited in the graph spectral domain, or in other words, *smooth* graph signals. Under this assumption, we answer the following questions: (i) What is the maximum bandwidth a signal $\mathbf{f}$ can have so that it can be perfectly recovered from its subset of samples on $\mathcal{S}(\subset \mathcal{V})$? (ii) Given a bandwidth (or dimensionality of the bandlimited space), what is the best set of nodes to sample for unique reconstruction? Additionally, stability is an important issue while choosing sampling sets, since in practice signals are only approximately bandlimited and/or their samples are noisy. A good sampling set leads to robustness of reconstruction in the presence of noise and model mismatch.

Most approaches to the sampling problem involve explicitly computing a portion of the graph Fourier basis, followed by using these basis elements to check if a unique and stable recovery is possible with the given samples or to choose the best subset of nodes for sampling. This approach works well enough for small graphs, where computing and storing a portion of the graph Fourier basis is practically feasible. However, current applications demand the handling of large graphs with thousands or even millions of nodes, and computing multiple eigenvectors of the variation operators can be burdensome in terms of time and space complexity. Therefore, in our approach, we define certain quantities called *graph spectral proxies* based on powers of the variation operator that allow one to estimate the bandwidth of graph signals. These proxies can be computed using repeated application of the variation operator over signals in a localized and distributed fashion with minimal storage cost, thus forming a key contribution of our approach. Using these proxies, we provide an approximate bound on the maximum bandwidth for unique recovery (i.e., the cutoff frequency) given the sampling set. We show that this quantity also appears in an approximate bound on the reconstruction error, and hence we maximize it using a greedy approach in order to select an approximately optimal sampling set of given size. Our algorithm is efficient and scalable,

---

Work in this chapter has been published in [3, 29, 4].

since it does not require explicit computation of the graph Fourier basis, and is shown to achieve comparable performance in comparison to approaches such as [65, 22] with lower computational cost.

The rest of this chapter is organized as follows: Section 3.1 reviews related work in the area of sampling. In Section 3.2, we provide necessary and sufficient conditions for sampling, and consider results in the scenario of known GFT. Section 3.3 introduces graph spectral proxies along with their properties. Section 3.4 and 3.5 employs these quantities to address the questions posed by the sampling theorem. Finally, we conclude this chapter with experimental validation in Section 3.6, followed by a summary in Section 3.7.

3.1 Related work

Sampling theory for graph signals was first studied in [55], where a sufficient condition for unique recovery of signals is stated for a given sampling set. Using this condition, [49] gives a bound on the maximum bandwidth that a signal can have, so that it can be uniquely reconstructed from its samples on a given subset of nodes. The uniqueness conditions in this section have also appeared prior to our work in [28] and subsequently in [65, 22]. However, the specific form in which these conditions have been presented require the explicit computation of the GFT, thereby limiting its practical utility. Using spectral proxies defined later in Section 3.3, our work circumvents the explicit computation of the graph Fourier basis and states conditions that ensure uniqueness and find a good sampling set. Previous methods for sampling set selection in graphs can be classified into two types, namely spectral-domain methods and vertex-domain methods, which are summarized below.

Spectral-domain approaches

Most of the recent work on sampling theory of graph signals assumes that a portion of the graph Fourier basis is explicitly known. We classify these methods as spectral-domain approaches since they involve computing the spectrum of the variation operator. For example, the work of [65] requires computation and processing of the first r eigenvectors of the graph Laplacian to construct a sampling set that guarantees unique (but not necessarily stable) reconstruction for a signal spanned by those eigenvectors. Similarly, a greedy algorithm for selecting stable sampling sets for a given bandlimited space is

proposed in [22]. It considers a spectral-domain criterion, using minimum singular values of submatrices of the graph Fourier transform matrix, to minimize the effect of sample noise in the worst case. The work of [71, 72] creates a link between the uncertainty principle for graph signals and sampling theory to arrive at similar criteria in the presence of sample noise. It is also possible to generalize this approach using ideas from the theory of optimal experiment design [38] and define other spectral-domain optimality criteria for selecting sampling sets that minimize different measures of reconstruction error when the samples are noisy (for example, the mean squared error). Greedy algorithms can then be used to find sets which are approximately optimal with respect to these criteria.

Vertex-domain approaches

There exist alternative approaches to sampling set selection that do not consider graph spectral information and instead rely on vertex-domain characteristics. Examples include [44] and [52], which select sampling sets based on maximum graph cuts and spanning trees, respectively. However, these methods are better suited for designing downsampling operators required in bipartite graph multiresolution transforms [45, 46]. Specifically, they do not consider the issue of optimality of sampling sets in terms of quality of bandlimited reconstruction. Further, it can be shown that the maximum graph-cut based sampling set selection criterion is closely related to a special case of our proposed approach. There exists an alternate vertex-domain sampling approach, described in [43], that involves successively shifting a signal using the adjacency matrix and aggregating the values of these signals on a given node. However, sampling using this strategy requires aggregating the sample values for a neighborhood size equal to the dimension of the bandlimited space, which can cover a large portion of the graph.

The sampling strategies described so far involve deterministic methods of approximating optimal sampling sets. Following our work, [57] proposed a randomized sampling strategy that guarantees a bound on the worst case reconstruction error in the presence of noise by sampling nodes independently based on a judiciously designed distribution over the nodes. However, one needs to sample many more nodes than the dimension of the bandlimited space to achieve the error bound with high probability.

3.2   Necessary and sufficient conditions

In this section, we address the issue of uniqueness and stability of bandlimited graph signal reconstruction and discuss different optimality criteria for sampling set selection assuming that the graph Fourier basis (i.e., the spectrum of the corresponding variation operator) is known[1]. The results in this section are useful when the graphs under consideration are small and thus, computing the spectrum of their variation operators is computationally feasible. They also serve as a guideline for tackling the aforementioned questions when the graphs are large and computation and storage of the graph Fourier basis is impractical.

In order to give a necessary and sufficient condition for unique identifiability of any signal $\mathbf{f} \in PW_\omega(G)$ from its samples $\mathbf{f}_\mathcal{S}$ on the sampling set $\mathcal{S}$, we first state the concept of uniqueness set [55].

Definition 3.1 (Uniqueness set). *A subset of nodes $\mathcal{S}$ is a uniqueness set for the space $PW_\omega(G)$ iff $\mathbf{x}_\mathcal{S} = \mathbf{y}_\mathcal{S}$ implies $\mathbf{x} = \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in PW_\omega(G)$.*

Unique identifiability requires that no two bandlimited signals have the same samples on the sampling set as ensured by the following theorem [3].

Theorem 3.1 (Unique sampling). *$\mathcal{S}$ is a uniqueness set for $PW_\omega(G)$ if and only if $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$.*

Proof. Given $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$, assume that $\mathcal{S}$ is not a uniqueness set. Then, there exist $\mathbf{f}, \mathbf{g} \in PW_\omega(G), \mathbf{f} \neq \mathbf{g}$ such that $\mathbf{f}_\mathcal{S} = \mathbf{g}_\mathcal{S}$. Hence, we have $\mathbf{f} - \mathbf{g} \in L_2(\mathcal{S}^c), \mathbf{f} - \mathbf{g} \neq \mathbf{0}$. Also, $\mathbf{f} - \mathbf{g} \in PW_\omega(G)$ due to closure. But this is a contradiction as $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$. Therefore, $\mathcal{S}$ must be a uniqueness set.

Conversely, we are given that $\mathcal{S}$ is a uniqueness set. Let ϕ be any signal in $PW_\omega(G) \cap L_2(\mathcal{S}^c)$. Then, for any $\mathbf{f} \in PW_\omega(G)$, we have $\mathbf{g} = \mathbf{f} + \phi \in PW_\omega(G)$ and $\mathbf{f}(\mathcal{S}) = \mathbf{g}(\mathcal{S})$. But since $\mathcal{S}$ is a uniqueness set, one must have $\mathbf{f} = \mathbf{g}$, which implies $\phi = \mathbf{0}$. Therefore, $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$. ∎

Let $\mathbf{S}$ be a matrix whose columns are indicator functions for nodes in $\mathcal{S}$. Note that $\mathbf{S}^\top : \mathbb{R}^n \to \mathbb{R}^{|\mathcal{S}|}$ is the sampling operator with $\mathbf{S}^\top \mathbf{f} = \mathbf{f}_\mathcal{S}$. Theorem 3.1 essentially states that no signal in $PW_\omega(G)$ is in the null space $\mathcal{N}(\mathbf{S}^\top)$ of the sampling operator. Any

---

[1]Parts of this chapter focusing on stability of reconstruction and other optimality criteria, have been done in collaboration with Akshay Gadde and appear in our joint paper [4].

$\mathbf{f} \in PW_\omega(G)$ can be written as $\mathbf{f} = \mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{c}$. Thus, for unique sampling of any signal in $PW_\omega(G)$ on $\mathcal{S}$, we need $\mathbf{S}^\top\mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{c} = \mathbf{U}_{\mathcal{S}\mathcal{R}}\mathbf{c} \neq \mathbf{0} \ \ \forall \, \mathbf{c} \neq \mathbf{0}$. This observation leads to the following corollary (which is also stated in [21]).

Corollary 3.1. *Let $\mathcal{R} = \{1,\ldots,r\}$, where λ_r is the largest graph frequency less than ω. Then $\mathcal{S}$ is a uniqueness set for $PW_\omega(G)$ if and only if $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has full column rank.*

If $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has a full column rank, then a unique reconstruction $\hat{\mathbf{f}} \in PW_\omega(G)$ can be obtained by finding the unique least squares solution to $\mathbf{f}_{\mathcal{S}} = \mathbf{U}_{\mathcal{S}\mathcal{R}}\mathbf{c}$:

$$\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{U}_{\mathcal{S}\mathcal{R}}^+\mathbf{f}_{\mathcal{S}}, \tag{3.1}$$

where $\mathbf{U}_{\mathcal{S}\mathcal{R}}^+ = (\mathbf{U}_{\mathcal{S}\mathcal{R}}^\top\mathbf{U}_{\mathcal{S}\mathcal{R}})^{-1}\mathbf{U}_{\mathcal{S}\mathcal{R}}^\top$ is the pseudo-inverse of $\mathbf{U}_{\mathcal{S}\mathcal{R}}$. The above reconstruction formula is also known as consistent reconstruction [28] since it keeps the observed samples unchanged[2], i.e., $\hat{\mathbf{f}}_{\mathcal{S}} = \mathbf{f}_{\mathcal{S}}$. Moreover, it is easy to see that if the original signal $\mathbf{f} \in PW_\omega(G)$, then $\hat{\mathbf{f}} = \mathbf{f}$.

3.2.1 Issue of stability and choice of sampling set

Note that selecting a sampling set $\mathcal{S}$ for $PW_\omega(G)$ amounts to selecting a set of rows of $\mathbf{U}_{\mathcal{V}\mathcal{R}}$. It is always possible to find a sampling set of size $r = \dim PW_\omega(G)$ that uniquely determines signals in $PW_\omega(G)$ as proven below.

Proposition 3.1. *For any $PW_\omega(G)$, there always exists a uniqueness set $\mathcal{S}$ of size $|\mathcal{S}| = r$.*

Proof. Since $\{\mathbf{u}^i\}_{i=1}^r$ are linearly independent, the matrix $\mathbf{U}_{\mathcal{V}\mathcal{R}}$ has full column rank equal to r. Further, since the row rank of a matrix equals its column rank, we can always find a linearly independent set $\mathcal{S}$ of r rows such that $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has full rank that equals r, thus proving our claim. ∎

In most cases picking r nodes randomly gives a full rank $\mathbf{U}_{\mathcal{S}\mathcal{R}}$. However, all sampling sets of given size are not equally good. A bad choice of $\mathcal{S}$ can give an ill-conditioned $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ which in turn leads to an unstable reconstruction $\hat{\mathbf{f}}$. Stability of reconstruction is important when the true signal $\mathbf{f}$ is only approximately bandlimited (which is the case

---

[2]Existence of a sample consistent reconstruction in $PW_\omega(G)$ requires that $PW_\omega(G) \oplus L_2(\mathcal{S}^c) = \mathbb{R}^N$ [28].

for most signals in practice) or when the samples are noisy. The reconstruction error in this case depends not only on noise and model mismatch but also on the choice of sampling set. The best sampling set achieves the smallest reconstruction error.

Effect of noise

We first consider the case when the observed samples are noisy. Let $\mathbf{f} \in PW_\omega(G)$ be the true signal and $\mathbf{n} \in \mathbb{R}^{|\mathcal{S}|}$ be the noise introduced during sampling. The observed samples are then given by $\mathbf{y}_\mathcal{S} = \mathbf{f}_\mathcal{S} + \mathbf{n}$. Using (3.1), we get the following reconstruction

$$\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{VR}}\mathbf{U}_{\mathcal{SR}}^+\mathbf{f}_\mathcal{S} + \mathbf{U}_{\mathcal{VR}}\mathbf{U}_{\mathcal{SR}}^+\mathbf{n}. \tag{3.2}$$

Since $\mathbf{f} \in PW_\omega(G)$, $\mathbf{U}_{\mathcal{VR}}\mathbf{U}_{\mathcal{SR}}^+\mathbf{f}_\mathcal{S} = \mathbf{f}$. The reconstruction error equals $\mathbf{e} = \hat{\mathbf{f}} - \mathbf{f} = \mathbf{U}_{\mathcal{VR}}\mathbf{U}_{\mathcal{SR}}^+\mathbf{n}$. If we assume that the entries of $\mathbf{n}$ are iid with zero mean and unit variance, then the covariance matrix of the reconstruction error is given by

$$\mathbf{E} = \mathbb{E}[\mathbf{e}\mathbf{e}^\top] = \mathbf{U}_{\mathcal{VR}}(\mathbf{U}_{\mathcal{SR}}^\top\mathbf{U}_{\mathcal{SR}})^{-1}\mathbf{U}_{\mathcal{VR}}^\top. \tag{3.3}$$

Different costs can be defined to measure the reconstruction error as a function of the error covariance matrix. These cost functions are based on optimal design of experiments [16]. If we define the optimal sampling set $\mathcal{S}^{\text{opt}}$ of size m, as the set which minimizes the mean squared error, then assuming $\mathbf{U}_{\mathcal{VR}}$ has orthonormal columns, we have

$$\mathcal{S}^{\text{A-opt}} = \arg\min_{|\mathcal{S}|=m}\ \text{tr}[\mathbf{E}] = \arg\min_{|\mathcal{S}|=m}\ \text{tr}[(\mathbf{U}_{\mathcal{SR}}^\top\mathbf{U}_{\mathcal{SR}})^{-1}]. \tag{3.4}$$

This is analogous to the so-called A-optimal design. Similarly, minimizing the maximum eigenvalue of the error covariance matrix leads to E-optimal design. For an orthonormal $\mathbf{U}_{\mathcal{VR}}$, the optimal sampling set with this criterion is given by

$$\mathcal{S}^{\text{E-opt}} = \arg\min_{|\mathcal{S}|=m}\ \lambda_{\max}(\mathbf{E}) = \arg\max_{|\mathcal{S}|=m}\ \sigma_{\min}(\mathbf{U}_{\mathcal{SR}}), \tag{3.5}$$

where $\sigma_{\min}(.)$ denotes the smallest singular value of a matrix. It can be thought of as a sampling set that minimizes the worst case reconstruction error. The above criterion is equivalent to the one proposed in [22]. Further, one can show that when $\mathbf{U}_{\mathcal{VR}}$ does not have orthonormal columns, (3.4) and (3.5) produce sampling sets that minimize upper bounds on the mean squared and worst case reconstruction errors respectively. Note

that both A and E-optimality criteria lead to combinatorial problems, but it is possible to develop greedy approximate solutions to these problems.

So far we assumed that the true signal $\mathbf{f} \in PW_\omega(G)$ and hence, $\mathbf{U}_{\mathcal{VR}}\mathbf{U}^+_{\mathcal{SR}}\mathbf{f}_\mathcal{S} = \mathbf{f}$. However, in most applications, the signals are only approximately bandlimited. The reconstruction error in such a case is analyzed next.

Effect of model mismatch

Let $\mathbf{P} = \mathbf{U}_{\mathcal{VR}}\mathbf{U}^\top_{\mathcal{VR}}$ be the projector for $PW_\omega(G)$ and $\mathbf{Q} = \mathbf{SS}^\top$ be the projector for $L_2(\mathcal{S})$. Assume that the true signal is given by $\mathbf{f} = \mathbf{f}^* + \Delta\mathbf{f}$, where $\mathbf{f}^* = \mathbf{Pf}$ is the bandlimited component of the signal and $\Delta\mathbf{f} = \mathbf{P}^\perp\mathbf{f}$ captures the "high-pass component" (i.e., the model mismatch). If we use (3.1) for reconstructing $\mathbf{f}$, then a tight upper bound on the reconstruction error [28] is given by

$$\|\mathbf{f} - \hat{\mathbf{f}}\| \le \frac{1}{\cos(\theta_{\max})}\|\Delta\mathbf{f}\|, \tag{3.6}$$

where $\theta_{\max}$ is the maximum angle between subspaces $PW_\omega(G)$ and $L_2(\mathcal{S})$ defined as

$$\cos(\theta_{\max}) = \inf_{\mathbf{f} \in PW_\omega(G), \|\mathbf{f}\|=1} \|\mathbf{Qf}\|. \tag{3.7}$$

$\cos(\theta_{\max}) > 0$ when the uniqueness condition in Theorem 3.1 is satisfied and the error is bounded. Intuitively, the above equation says that for the worst case error to be minimum, the sampling and reconstruction subspaces should be as aligned as possible.

We define an optimal sampling set $\mathcal{S}^{\mathrm{opt}}$ of size m for $PW_\omega(G)$ as the set which minimizes the worst case reconstruction error. Therefore, $L_2(\mathcal{S}^{\mathrm{opt}})$ makes the smallest maximum angle with $PW_\omega(G)$. It is easy to show that $\cos(\theta_{\max}) = \sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$. Thus, to find this set we need to solve a similar problem as (3.5). As stated before, this problem is combinatorial. It is possible to give a greedy algorithm to get an approximate solution. A simple greedy heuristic to approximate $\mathcal{S}^{\mathrm{opt}}$ is to perform column-wise Gaussian elimination over $\mathbf{U}_{\mathcal{VR}}$ with partial row pivoting. The indices of the pivot rows in that case form a good estimate of $\mathcal{S}^{\mathrm{opt}}$ in practice.

The methods described above require computation of many eigenvectors of the variation operator $\mathbf{L}$. We circumvent this issue in the next section, by defining graph spectral proxies based on powers of $\mathbf{L}$. These spectral proxies do not require eigen-decomposition of $\mathbf{L}$ and still allow us to compute the cut-off frequency that also acts as a measure of

quality for different sampling sets. As we will show, these proxies arise naturally in the expression for the bound on the reconstruction error. Thus, a sampling set optimal with respect to these spectral proxies ensures a small reconstruction error bound.

3.3   Graph Spectral Proxies

As discussed earlier, graphs considered in most real applications are very large. Hence, computing and storing the graph Fourier basis explicitly is often practically infeasible. We now present a tool to approximately compute the bandwidth $\omega(\phi)$ of any given signal ϕ without computing the Fourier coefficients explicitly. These quantities shall allow us to express the condition for unique bandlimited reconstruction, in terms of the cut-off frequency, and methods for sampling set selection via simple operations using the variation operator. The following definition holds for any choice of the variation operator $\mathbf{L}$ in Table 2.1:

Definition 3.2 (Graph Spectral Proxies). *For any signal* $\mathbf{f} \neq \mathbf{0}$*, we define its* k^{th} *spectral proxy* $\omega_k(\mathbf{f})$ *with* $k \in \mathbb{Z}^+$ *as*

$$\omega_k(\mathbf{f}) \triangleq \left(\frac{\|\mathbf{L}^k \mathbf{f}\|}{\|\mathbf{f}\|} \right)^{1/k}. \tag{3.8}$$

For an operator $\mathbf{L}$ with real eigenvalues and eigenvectors, $\omega_k(\mathbf{f})$ can be shown to increase monotonically with k:

$$\forall \mathbf{f}, k_1 < k_2 \Rightarrow \omega_{k_1}(\mathbf{f}) \leq \omega_{k_2}(\mathbf{f}). \tag{3.9}$$

These quantities are bounded from above, as a result, $\lim_{k \to \infty} \omega_k(\mathbf{f})$ exists for all $\mathbf{f}$. Consequently, we can show that if $\omega(\mathbf{f})$ denotes the bandwidth of a signal $\mathbf{f}$, then

$$\forall k > 0, \ \omega_k(\mathbf{f}) \leq \lim_{j \to \infty} \omega_j(\mathbf{f}) = \omega(\mathbf{f}). \tag{3.10}$$

Note that (3.10) also holds for an asymmetric $\mathbf{L}$ that has complex eigenvalues and eigenvectors. The proofs of (3.9) and (3.10) are provided in Lemmas 3.1 and 3.2. These properties give us an important insight: as we increase the value of k, the spectral proxies tend to have a value close to the actual bandwidth of the signal, i.e., they essentially indicate the frequency localization of the signal energy.

Lemma 3.1. *If* $\mathbf{L}$ *has real eigenvalues and eigenvectors, then for any* $k_1 < k_2$*, we have* $\omega_{k_1}(\mathbf{f}) \leq \omega_{k_2}(\mathbf{f}), \forall \mathbf{f}$*.*

Proof. We first expand $\omega_{k_1}(\mathbf{f})$ as follows:

$$(\omega_{k_1}(\mathbf{f}))^{2k_1} = \left(\frac{\|\mathbf{L}^{k_1}\mathbf{f}\|}{\|\mathbf{f}\|}\right)^2$$

$$= \frac{\sum_{i,j}(\lambda_i\lambda_j)^{k_1}\tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j}{\sum_{i,j}\tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j} \tag{3.11}$$

$$= \sum_{i,j}(\lambda_i\lambda_j)^{k_1}c_{ij} \tag{3.12}$$

where $c_{ij} = \tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j/\sum_{i,j}\tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j$. Now, consider the function $f(x) = x^{k_2/k_1}$. Note that since $k_1 < k_2$, $f(x)$ is a convex function. Further, since $\sum_{i,j}c_{ij} = 1$, we can use Jensen's inequality in the above equation to get

$$\left(\sum_{i,j}(\lambda_i\lambda_j)^{k_1}c_{ij}\right)^{k_2/k_1} \leq \sum_{i,j}\left((\lambda_i\lambda_j)^{k_1}\right)^{k_2/k_1}c_{ij} \tag{3.13}$$

$$\Rightarrow \left(\sum_{i,j}(\lambda_i\lambda_j)^{k_1}c_{ij}\right)^{1/2k_1} \leq \left(\sum_{i,j}(\lambda_i\lambda_j)^{k_2}c_{ij}\right)^{1/2k_2}$$

$$\Rightarrow \omega_{k_1}(\mathbf{f}) \leq \omega_{k_2}(\mathbf{f}) \tag{3.14}$$

If $\mathbf{L}$ has real entries, but complex eigenvalues and eigenvectors, then these occur in conjugate pairs, hence, the above summation is real. However, in that case, $\omega_k(\mathbf{f})$ is not guaranteed to increase in a monotonous fashion, since c_{ij}'s are not real and Jensen's inequality breaks down. ∎

Lemma 3.2. *Let $\omega(\mathbf{f})$ be the bandwidth of any signal $\mathbf{f}$. Then, the following holds:*

$$\omega(\mathbf{f}) = \lim_{k\to\infty}\omega_k(\mathbf{f}) = \lim_{k\to\infty}\left(\frac{\|\mathbf{L}^k\mathbf{f}\|}{\|\mathbf{f}\|}\right)^{1/k} \tag{3.15}$$

Proof. We first consider the case when $\mathbf{L}$ has real eigenvalues and eigenvectors. Let $\omega(\mathbf{f}) = \lambda_p$, then we have:

$$\omega_k(\mathbf{f}) = \left(\frac{\sum_{i,j=1}^{p}(\lambda_i\lambda_j)^k\tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j}{\sum_{i,j=1}^{p}\tilde{\mathbf{f}}_i\tilde{\mathbf{f}}_j\mathbf{u}_i^\top\mathbf{u}_j}\right)^{1/2k} \tag{3.16}$$

$$= \lambda_p\left(c_{pp} + \sum_{(i,j)\neq(p,p)}\left(\frac{\lambda_i}{\lambda_p}\frac{\lambda_j}{\lambda_p}\right)^k c_{ij}\right)^{1/2k} \tag{3.17}$$

31

where $c_{ij} = \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j / \sum_{i,j} \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j$. Now, using logarithms, we can show

$$\lim_{k \to \infty} \left(c_{pp} + \sum_{(i,j) \neq (p,p)} \left(\frac{\lambda_i}{\lambda_p} \frac{\lambda_j}{\lambda_p} \right)^k c_{ij} \right)^{1/2k} = 1. \tag{3.18}$$

Substituting (3.18) in (3.17), we get

$$\lim_{k \to \infty} \omega_k(\mathbf{f}) = \lambda_p = \omega(\mathbf{f}). \tag{3.19}$$

Now, if $\mathbf{L}$ has complex eigenvalues and eigenvectors, then these have to occur in conjugate pairs since $\mathbf{L}$ has real entries. Hence, for this case, we do a similar expansion as above and take $|\lambda_p|$ out of the expression. Then, the limit of the remaining term is once again equal to 1. ∎

3.4 Cutoff frequency

In order to obtain a measure of quality for a sampling set $\mathcal{S}$, we first find the cutoff frequency associated with it, which can be defined as the largest frequency ω such that $\mathcal{S}$ is a uniqueness set for $PW_\omega(G)$. It follows from Theorem 3.1 that, for $\mathcal{S}$ to be a uniqueness set of $PW_\omega(G)$, ω needs to be less than the minimum possible bandwidth that a signal in $L_2(\mathcal{S}^c)$ can have. This would ensure that no signal from $L_2(\mathcal{S}^c)$ can be a part of $PW_\omega(G)$. Thus, the cutoff frequency $\omega_c(\mathcal{S})$ for a sampling set $\mathcal{S}$ can be expressed as:

$$\omega_c(\mathcal{S}) \triangleq \min_{\phi \in L_2(\mathcal{S}^c), \, \phi \neq \mathbf{0}} \omega(\phi). \tag{3.20}$$

In order to avoid computation of the GFT basis, we use $\omega_k(\phi)$ as a proxy for $\omega(\phi)$ (i.e. bandwidth of ϕ) and this leads us to define the *cut-off frequency estimate of order* k as

$$\Omega_k(\mathcal{S}) \triangleq \min_{\phi \in L_2(\mathcal{S}^c)} \omega_k(\phi) = \min_{\phi \in L_2(\mathcal{S}^c)} \left(\frac{\|\mathbf{L}^k \phi\|}{\|\phi\|} \right)^{1/k}. \tag{3.21}$$

Using the definitions of $\Omega_k(\mathcal{S})$ and $\omega_c(\mathcal{S})$ along with (3.9) and (3.10), we conclude that for any $k_1 < k_2$:

$$\omega_c(\mathcal{S}) \geq \lim_{k \to \infty} \Omega_k(\mathcal{S}) \geq \Omega_{k_2}(\mathcal{S}) \geq \Omega_{k_1}(\mathcal{S}). \tag{3.22}$$

Using (3.22) and (3.20), we now state the following proposition:

32

Proposition 3.2. *For any k, $\mathcal{S}$ is a uniqueness set for $PW_\omega(G)$ if, $\omega < \Omega_k(\mathcal{S})$. $\Omega_k(\mathcal{S})$ can be computed from (3.21) as*

$$\Omega_k(\mathcal{S}) = \left[\min_\psi \frac{\psi^\top ((\mathbf{L}^\top)^k \mathbf{L}^k)_{\mathcal{S}^c} \psi}{\psi^\top \psi} \right]^{1/2k} = (\sigma_{1,k})^{1/2k}, \tag{3.23}$$

where $\sigma_{1,k}$ denotes the smallest eigenvalue of the reduced matrix $((\mathbf{L}^\top)^k \mathbf{L}^k)_{\mathcal{S}^c}$. Further, if $\psi_{1,k}$ is the corresponding eigenvector, and ϕ_k^* minimizes $\omega_k(\phi)$ in (3.21) (i.e. it approximates the smoothest possible signal in $L_2(\mathcal{S}^c)$), then

$$\phi_k^*(\mathcal{S}^c) = \psi_{1,k}, \quad \phi_k^*(\mathcal{S}) = \mathbf{0}. \tag{3.24}$$

We note from (3.22) that to get a better estimate of the true cut-off frequency, one simply needs a higher k. Therefore, there is a trade-off between accuracy of the estimate on the one hand, and complexity and numerical stability on the other (that arise by taking higher powers of $\mathbf{L}$).

3.5 Sampling set selection

3.5.1 Best sampling set of given size

As shown in Proposition 3.2, $\Omega_k(\mathcal{S})$ is an estimate of the smallest bandwidth that a signal in $L_2(\mathcal{S}^c)$ can have and any signal in $PW_\omega(G)$ is uniquely sampled on $\mathcal{S}$ if $\omega < \Omega_k(\mathcal{S})$. Intuitively, we would like the projection of $L_2(\mathcal{S}^c)$ along $PW_\omega(G)$ to be as small as possible. Based on this intuition, we propose the following optimality criterion for selecting the best sampling set of size m:

$$\mathcal{S}_k^{\mathrm{opt}} = \arg\max_{|\mathcal{S}|=m} \Omega_k(\mathcal{S}). \tag{3.25}$$

To motivate the above criterion more formally, let $\mathbf{P}$ denote the projector for $PW_\omega(G)$. The minimum gap [40] between the two subspaces $L_2(\mathcal{S}^c)$ and $PW_\omega(G)$ is given by:

$$\inf_{\mathbf{f}\in L_2(\mathcal{S}^c), \|\mathbf{f}\|=1} \|\mathbf{f} - \mathbf{P}\mathbf{f}\| = \sqrt{\sum_{i:\, \omega<\lambda_i} |\tilde{\mathbf{f}}_i^*|^2}$$

$$\geq \sqrt{\sum_{i\in\mathcal{I}} |\tilde{\mathbf{f}}_i^*|^2}, \tag{3.26}$$

33

where $\mathcal{I} = \{i : \omega < \lambda_i \leq \Omega_k(\mathcal{S})\}$ and $\tilde{\mathbf{f}}_i^*$ denotes the ith GFT coefficient of the minimizer $\mathbf{f}^*$ for the left hand side. The inequality on the right hand side holds because $\Omega_k(\mathcal{S})$ is the smallest bandwidth that any signal in $L_2(\mathcal{S}^c)$ can have. Eq. (3.26) shows that maximizing $\Omega_k(\mathcal{S})$ increases the lower bound on the minimum gap between $L_2(\mathcal{S}^c)$ and $PW_\omega(G)$. The minimum gap equals $\cos(\theta_{\max})$ as defined in (3.7) [40]. Thus, maximizing $\Omega_k(\mathcal{S})$ increases the lower bound on $\cos(\theta_{\max})$ which, in turn, minimizes the upper bound on the reconstruction error $\|\mathbf{f} - \hat{\mathbf{f}}\|$ given in (3.6), where the original signal $\mathbf{f} \notin PW_\omega(G)$ and $\hat{\mathbf{f}} \in PW_\omega(G)$ is obtained by (3.1).

We now show that $\Omega_k(\mathcal{S})$ also arises in the bound on the reconstruction error when the reconstruction is obtained by variational energy minimization:

$$\hat{\mathbf{f}}_m = \underset{\mathbf{y} \in \mathbb{R}^N}{\arg\min} \|\mathbf{L}^m \mathbf{y}\| \text{ subject to } \mathbf{y}_\mathcal{S} = \mathbf{f}_\mathcal{S}. \tag{3.27}$$

It was shown in [56] that if $\mathbf{f} \in PW_\omega(G)$, then the reconstruction error $\|\hat{\mathbf{f}}_m - \mathbf{f}\|/\|\mathbf{f}\|$, for a given m, is upper-bounded by $2(\omega/\Omega_1(\mathcal{S}))^m$. This bound is suboptimal and can be improved by replacing $\Omega_1(\mathcal{S})$ with $\Omega_k(\mathcal{S})$ (which, from (3.22), is at least as large as $\Omega_1(\mathcal{S})$) for any $k \leq m$, as shown in the following theorem:

Theorem 3.2. *Let $\hat{\mathbf{f}}_m$ be the solution to* (3.27) *for a signal $\mathbf{f} \in PW_\omega(G)$. Then, for any $k \leq m$,*

$$\|\hat{\mathbf{f}}_m - \mathbf{f}\| \leq 2 \left(\frac{\omega}{\Omega_k(\mathcal{S})} \right)^m \|\mathbf{f}\|. \tag{3.28}$$

Proof. Note that $(\hat{\mathbf{f}}_m - \mathbf{f}) \in L_2(\mathcal{S}^c)$. Therefore, from (3.21)

$$\|\hat{\mathbf{f}}_m - \mathbf{f}\| \leq \frac{1}{(\Omega_m(\mathcal{S}))^m} \|\mathbf{L}^m(\hat{\mathbf{f}}_m - \mathbf{f})\|$$

$$\leq \frac{1}{(\Omega_m(\mathcal{S}))^m} (\|\mathbf{L}^m \hat{\mathbf{f}}_m\| + \|\mathbf{L}^m \mathbf{f}\|) \tag{3.29}$$

$$\leq \frac{2}{(\Omega_m(\mathcal{S}))^m} \|\mathbf{L}^m \mathbf{f}\| \tag{3.30}$$

$$\leq 2 \left(\frac{\omega_m(\mathbf{f})}{\Omega_m(\mathcal{S})} \right)^m \|\mathbf{f}\| \tag{3.31}$$

$$\leq 2 \left(\frac{\omega}{\Omega_k(\mathcal{S})} \right)^m \|\mathbf{f}\|.$$

(3.29) follows from triangle inequality. (3.30) holds because $\hat{\mathbf{f}}_m$ minimizes $\|\mathbf{L}^m \hat{\mathbf{f}}_m\|$ over all sample consistent signals. (3.31) follows from the definition of $\omega_m(\mathbf{f})$ and the last step follows from (3.10) and (3.22). ∎

Note that for the error bound in (3.28) to go to zero as $m \to \infty$, ω must be less than $\Omega_k(\mathcal{S})$. Thus, increasing $\Omega_k(\mathcal{S})$ allows us to reconstruct signals in a larger bandlimited space using the variational method. Moreover, for a fixed m and k, a higher value of $\Omega_k(\mathcal{S})$ leads to a lower reconstruction error bound. The optimal sampling set $\mathcal{S}_k^{\text{opt}}$ in (3.25) essentially minimizes this error bound.

3.5.2   Obtaining the best sampling set

The problem posed in (3.25) is a combinatorial problem because we need to compute $\Omega_k(\mathcal{S})$ for every possible subset $\mathcal{S}$ of size m. We therefore formulate a greedy heuristic to get an estimate of the optimal sampling set. Starting with an empty sampling set $\mathcal{S}$ ($\Omega_k(\mathcal{S}) = 0$) we keep adding nodes (from $\mathcal{S}^c$) one-by-one while trying to ensure maximum increase in $\Omega_k(\mathcal{S})$ at each step. To achieve this, we first consider the following quantity:

$$\lambda_k^\alpha(\mathbf{1}_\mathcal{S}) = \min_{\mathbf{x}} \left(\omega_k(\mathbf{x}) + \alpha \frac{\mathbf{x}^\top \text{diag}(\mathbf{1}_\mathcal{S})\mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \right), \tag{3.32}$$

where $\mathbf{1}_\mathcal{S} : \mathcal{V} \to \{0,1\}$ denotes the indicator function for the subset $\mathcal{S}$ (i.e. $\mathbf{1}(\mathcal{S}) = \mathbf{1}$ and $\mathbf{1}(\mathcal{S}^c) = \mathbf{0}$). Note that the right hand side of (3.32) is simply a relaxation of the constraint in (3.21). When $\alpha \gg 1$, the components $\mathbf{x}(\mathcal{S})$ are highly penalized during minimization, hence, forcing values of $\mathbf{x}$ on $\mathcal{S}$ to be vanishingly small. Thus, if $\mathbf{x}_k^\alpha(\mathbf{1}_\mathcal{S})$ is the minimizer in (3.32), then $[\mathbf{x}_k^\alpha(\mathbf{1}_\mathcal{S})](\mathcal{S}) \to \mathbf{0}$. Therefore, for $\alpha \gg 1$,

$$\phi_k^* \approx \mathbf{x}_k^\alpha(\mathbf{1}_\mathcal{S}), \quad \Omega_k(\mathcal{S}) \approx \lambda_k^\alpha(\mathbf{1}_\mathcal{S}). \tag{3.33}$$

Now, to tackle the combinatorial nature of our problem, we allow a *binary relaxation* of the indicator $\mathbf{1}_S$ in (3.32), to define the following quantities

$$\omega_k^\alpha(\mathbf{x}, \mathbf{t}) = \left(\omega_k(\mathbf{x}) + \alpha \frac{\mathbf{x}^\top \text{diag}(\mathbf{t})\mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \right), \tag{3.34}$$

$$\lambda_k^\alpha(\mathbf{t}) = \min_{\mathbf{x}} \omega_k^\alpha(\mathbf{x}, \mathbf{t}), \tag{3.35}$$

where $\mathbf{t} \in \mathbb{R}^N$. These relaxations circumvent the combinatorial nature of our problem and have been used recently to study graph partitioning based on Dirichlet eigenvalues [54, 14]. Note that making the substitution $\mathbf{t} = \mathbf{1}_{\mathcal{S}}$ in (3.35) gives us (3.32) exactly. The effect of adding a node to $\mathcal{S}$ on $\Omega_k(\mathcal{S})$ at each step can now be understood by observing the gradient vector $\nabla_{\mathbf{t}} \lambda_k^\alpha(\mathbf{t})$, at $\mathbf{t} = \mathbf{1}_{\mathcal{S}}$. Note that for any $\mathbf{x}$ and $\mathbf{t}$,

$$\frac{d\omega_k^\alpha(\mathbf{x}, \mathbf{t})}{d\mathbf{t}(i)} = \alpha \left(\frac{\mathbf{x}(i)}{\|\mathbf{x}\|} \right)^2. \tag{3.36}$$

When $\mathbf{t} = \mathbf{1}_{\mathcal{S}}$, we know that the minimizer of (3.35) with respect to $\mathbf{x}$ for large α is ϕ_k^*. Hence,

$$\frac{d\lambda_k^\alpha(\mathbf{t})}{d\mathbf{t}(i)} \bigg|_{\mathbf{t}=\mathbf{1}_{\mathcal{S}}} = \frac{d\omega_k^\alpha(\phi_k^*, \mathbf{t})}{d\mathbf{t}(i)} \bigg|_{\mathbf{t}=\mathbf{1}_{\mathcal{S}}} = \alpha \left(\frac{\phi_k^*(i)}{\|\phi_k^*\|} \right)^2. \tag{3.37}$$

The equation above gives us the desired greedy heuristic - starting with an empty $\mathcal{S}$ (i.e., $\mathbf{1}_{\mathcal{S}} = \mathbf{0}$), if at each step, we include the node on which the smoothest signal $\phi_k^* \in L_2(\mathcal{S}^c)$ has maximum energy (i.e. $\mathbf{1}_{\mathcal{S}}(i) \leftarrow 1, i = \arg\max_j(\phi_k^*(j))^2$), then $\lambda_k^\alpha(\mathbf{t})$ and in effect, the cut-off estimate $\Omega_k(\mathcal{S})$, tend to increase maximally. We summarize the method for estimating $\mathcal{S}_k^{\mathrm{opt}}$ in Algorithm 3.1.

One can show that the cutoff frequency estimate $\Omega_k(\mathcal{S})$ associated with a sampling set can only increase (or remain unchanged) when a node is added to it. This is stated more formally in the following proposition.

Proposition 3.3. *Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be two subsets of nodes of G with $\mathcal{S}_1 \subseteq \mathcal{S}_2$. Then $\Omega_k(\mathcal{S}_1) \leq \Omega_k(\mathcal{S}_2)$.*

This turns out to be a straightforward consequence of the eigenvalue interlacing property for symmetric matrices.

Theorem 3.3 (Eigenvalue interlacing [32])**.** *Let $\mathbf{B}$ be a symmetric $n \times n$ matrix. Let $\mathcal{R} = \{1, 2, \ldots, r\}$, for $1 \leq r \leq n - 1$ and $\mathbf{B}_r = \mathbf{B}_{\mathcal{R}}$. Let $\lambda_k(\mathbf{B}_r)$ be the k-th largest eigenvalue of $\mathbf{B}_r$. Then the following interlacing property holds:*

$$\lambda_{r+1}(\mathbf{B}_{r+1}) \leq \lambda_r(\mathbf{B}_r) \leq \lambda_r(\mathbf{B}_{r+1}) \leq \ldots$$
$$\ldots \leq \lambda_2(\mathbf{B}_{r+1}) \leq \lambda_1(\mathbf{B}_r) \leq \lambda_1(\mathbf{B}_{r+1}).$$

The above theorem implies that if $\mathcal{S}_1 \subseteq \mathcal{S}_2$, then $\mathcal{S}_2^c \subseteq \mathcal{S}_1^c$ and thus, $\lambda_{\min}\left[\left((\mathbf{L}^\top)^k \mathbf{L}^k \right)_{\mathcal{S}_1^c} \right] \leq \lambda_{\min}\left[\left((\mathbf{L}^\top)^k \mathbf{L}^k \right)_{\mathcal{S}_2^c} \right].$

Algorithm 3.1 Greedy heuristic for estimating $\mathcal{S}_k^{\text{opt}}$

Require: $G = \{\mathcal{V}, E\}$, $\mathbf{L}$, sampling set size M, $k \in \mathbb{Z}^+$.
Ensure: $\mathcal{S} = \{\emptyset\}$.
 1: **while** $|\mathcal{S}| < M$ **do**
 2:     For $\mathcal{S}$, compute smoothest signal $\phi_k^* \in L_2(\mathcal{S}^c)$ using Proposition 3.2.
 3:     $v \leftarrow \arg\max_i(\phi_k^*(i))^2$.
 4:     $\mathcal{S} \leftarrow \mathcal{S} \cup v$.
 5: **end while**
 6: $\mathcal{S}_k^{\text{opt}} \leftarrow \mathcal{S}$.

Connection with Gaussian elimination

From Section 3.2, we know that the optimal sampling set can be obtained by maximizing $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$ with respect to $\mathcal{S}$. A heuristic to obtain good sampling sets is to perform a column-wise Gaussian elimination with pivoting on the eigenvector matrix $\mathbf{U}$. Then, a sampling set of size i is given by the indices of zeros in the $(i+1)^{\text{th}}$ column of the echelon form. We now show that the greedy heuristic proposed in Algorithm 3.1 is closely related to this rank-revealing Gaussian elimination procedure through the following observation:

Proposition 3.4. *Let Φ be the matrix whose columns are given by the smoothest signals ϕ_∞^* obtained sequentially after each iteration of Algorithm 3.1 with $k = \infty$, (i.e., $\Phi = \left[\phi_\infty^*|_{|\mathcal{S}|=0}\ \phi_\infty^*|_{|\mathcal{S}|=1},\ \ldots\right]$). Further, let $\mathbf{T}$ be the matrix obtained by performing column-wise Gaussian elimination on $\mathbf{U}$ with partial pivoting. Then, the columns of $\mathbf{T}$ are equal to the columns of Φ_∞^* within a scaling factor.*

Proof. If $\mathcal{S}$ is the smallest sampling set for uniquely representing signals in $PW_\omega(G)$ and $r = \dim PW_\omega(G)$, then we have the following:

1. $|\mathcal{S}| = r$.

2. The smoothest signal $\phi_\infty^* \in L_2(\mathcal{S}^c)$ has bandwidth λ_{r+1}.

Therefore, $\phi_\infty^*|_{|\mathcal{S}|=r}$ is spanned by the first $r+1$ frequency basis elements $\{\mathbf{u}_1, \ldots, \mathbf{u}_{r+1}\}$. Further, since $\phi_\infty^*|_{|\mathcal{S}|=r}$ has zeroes on exactly r locations, it can be obtained by performing Gaussian elimination on $\mathbf{u}_{r+1}$ using $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_r$. Hence the $(r+1)^{\text{th}}$ column of Φ is equal (within a scaling factor) to the $(r+1)^{\text{th}}$ column of $\mathbf{T}$. Pivoting comes from the fact that the $(i+1)^{\text{th}}$ sampled node is given by the index of the element with maximum

Table 3.1: Comparison of complexity of different sampling set selection algorithms.

	Methods in [65, 22] (with GFT)	Proposed Method
Eigen-pair computations	$O((\|E\|\|\mathcal{S}\| + C\|\mathcal{S}\|^3)T_1)$	$O\left(k\|E\|\|S\|T_2(k)\right)$
Sampling set search	$O(N\|\mathcal{S}\|^3)$ [65], $O(N\|\mathcal{S}\|^4)$ [22]	$O(N\|\mathcal{S}\|)$
Space complexity	$O(N\|\mathcal{S}\|)$	$O(N)$

magnitude in $\phi_\infty^*\big|_{\|\mathcal{S}\|=i}$, and is used as the pivot to zeros out elements with same index in subsequent columns. ∎

The above result illustrates that Algorithm 3.1 is an iterative procedure that approximates a rank-revealing Gaussian elimination procedure on $\mathbf{U}_{\mathcal{VR}}$. For the known-spectrum case, this is a good heuristic for maximizing $\sigma_{min}(\mathbf{U}_{\mathcal{SR}})$. In other words, our method directly maximizes $\sigma_{min}(\mathbf{U}_{\mathcal{SR}})$ without going through the intermediate step of computing $\mathbf{U}_{\mathcal{VR}}$. As we shall see in the next subsection, this results in significant savings in both time and space complexity.

Complexity and implementation issues

We note that in the algorithm, computing the first eigen-pair of $((\mathbf{L}^\top)^k\mathbf{L}^k)_{\mathcal{S}^c}$ is the major step for each iteration. There are many efficient iterative methods, such as those based on Rayleigh quotient minimization, for computing the smallest eigen-pair of a matrix [41]. The atomic step in all of these methods consists of matrix-vector products. Specifically, in our case, this step involves evaluating the expression $((\mathbf{L}^\top)^k\mathbf{L}^k)_{\mathcal{S}^c}\mathbf{x}$. Note that we do not actually need to compute the matrix $((\mathbf{L}^\top)^k\mathbf{L}^k)_{\mathcal{S}^c}$ explicitly, since the expression can be implemented as a sequence of matrix-vector products as

$$((\mathbf{L}^\top)^k\mathbf{L}^k)_{\mathcal{S}^c}\mathbf{x} = \mathbf{I}_{\mathcal{S}^c\mathcal{V}}\mathbf{L}^\top \dots \mathbf{L}^\top\mathbf{L} \dots \mathbf{L}\mathbf{I}_{\mathcal{VS}^c}\mathbf{x}. \tag{3.38}$$

Evaluating the expression involves $2k$ matrix-vector products and has a complexity of $O(k\|E\|)$, where $\|E\|$ is the number of edges in the graph. Moreover, a localized and parallel implementation of this step is possible in the case of sparse graphs. The number of iterations required for convergence of the eigen-pair computation iterations is a function of the eigen-value gaps [41] and hence dependent on the graph structure and edge-weights.

For the methods of [65] and [22], one needs to compute a portion of the eigenvector matrix, i.e., $\mathbf{U}_{\mathcal{VS}}$ (assuming $\|\mathcal{R}\| = \|\mathcal{S}\|$). This can be done using block-based Rayleigh

quotient minimization methods [41], block-based Kryolov subspace methods such as Arnoldi/Lanczos iterations or deflation methods in conjunction with single eigen-pair solvers [59]. The complexity of these methods increases considerably as the number of requested eigen-pairs increases, making them impractical. On the other hand, our method requires computing a single eigen-pair at each iteration, making it viable for cases when a large number of samples are required. Moreover, the sample search steps in the methods of [65] and [22] require an SVD solver and a linear system solver, respectively, thus making them much more complex in comparison to our method, where we only require finding the maximum element of a vector. Our algorithm is also efficient in terms of space complexity, since at any point we just need to store $\mathbf{L}$ and one vector. On the other hand, [65, 22] require storage of at least $|\mathcal{S}|$ eigenvectors.

A summary of the complexities of all the methods is given in Table 3.1. The eigen-pair computations for [65, 22] are assumed to be performed using a block version of the Rayleigh quotient minimization method, which has a complexity of $O((|E||\mathcal{S}| + C|\mathcal{S}|^3)T_1)$, where T_1 denotes the number of iterations for convergence, and C is a constant. The complexity of computing one eigen-pair in our method is $O(k|E||S|T_2(k))$, where $T_2(k)$ denotes the average number of iterations required for convergence of a single eigen-pair. T_1 and $T_2(k)$ required to achieve a desired error tolerance are functions of the eigen-gaps of $\mathbf{L}$ and $\mathbf{L}^k$ respectively. In general, $T_2(k) > T_1$, since $\mathbf{L}^k$ has lower eigengaps near the smallest eigenvalue. Increasing the parameter k further flattens the spectrum of $\mathbf{L}^k$ near the smallest eigenvalue leading to an increase in $T_2(k)$, since one has to solve a more ill-conditioned problem. We illustrate this in the next section through experiments that compare the running times of all the methods.

The choice of the parameter k depends on the desired accuracy – a larger value of k gives a better sampling set, but increases the complexity proportionally, thus providing a trade-off. Through experiments, we show in the next section that the quality of the sampling set is more sensitive to choice of k for sparser graphs. This is because increasing k results in the consideration of more *global* information while selecting samples. On the other hand, dense graphs have a lower diameter and there is relatively little information to be gained by increasing k.

3.6 Experiments

We now numerically evaluate the performance of the proposed work[3]. The experiments involve comparing the reconstruction errors and running times of different sampling set selection algorithms in conjunction with consistent bandlimited reconstruction (3.1)[4]. We compare our approach with the following methods:

M1: This method [22] uses a greedy algorithm to approximate the $\mathcal{S}$ that maximizes $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$. Consistent bandlimited reconstruction (3.1) is then used to estimate the unknown samples.

M2: At each iteration i, this method [65] finds the representation of $\mathbf{u}_i$ as $\sum_{j<i} \beta_j \mathbf{u}_j + \sum_{u \notin \mathcal{S}} \alpha_u \mathbf{1}_u$, where $\mathbf{1}_u$ is the delta function on u. The node v with maximum $|\alpha_v|$ is sampled. Reconstruction is done using (3.1).

Both the above methods assume that a portion of the frequency basis is known and the signal to be recovered is exactly bandlimited. As a baseline, we also compare all sampling set selection methods against uniform random sampling.

3.6.1 Examples with artificial data

We first give some simple examples on the following simulated undirected graphs:

$G1$: Erdös-Renyi random graph (unweighted) with 1000 nodes and connection probability 0.01.

$G2$: Small world graph [75] (unweighted) with 1000 nodes. The underlying regular graph with degree 8 is rewired with probability 0.1.

$G3$: Barabási-Albert random network [9] with 1000 nodes. The seed network is a fully connected graph with $m_0 = 4$ vertices, and each new vertex is connected to $m = 4$ existing vertices randomly. This model, as opposed to $G1$ and $G2$, is a scale-free network, *i.e.*, its degrees follow a power law $P(k) \sim k^{-3}$.

—————————————————————

[3]Code available at https://github.com/aamiranis/sampling_theory

[4]Although reconstruction using (3.1) requires explicit computation of $\mathbf{U}_{\mathcal{VR}}$, there exist efficient localized reconstruction algorithms that circumvent this [50, 74]. However, in our work, we restrict our attention to the problem of sampling set selection.

The performance of the sampling methods depends on the assumptions about the true signal and sampling noise. For each of the above graphs, we consider the problem in the following scenarios:

$F1$: The true signal is noise-free and exactly bandlimited with $r = \dim PW_\omega(G) = 50$. The non-zero GFT coefficients are randomly generated from $\mathcal{N}(1, 0.5^2)$.

$F2$: The true signal is exactly bandlimited with $r = 50$ and non-zero GFT coefficients are generated from $\mathcal{N}(1, 0.5^2)$. The samples are noisy with additive iid Gaussian noise such that the SNR equals 20dB.

$F3$: The true signal is approximately bandlimited with an exponentially decaying spectrum. Specifically, the GFT coefficients are generated from $\mathcal{N}(1, 0.5^2)$, followed by rescaling with the following filter (where $r = 50$):

$$
h(\lambda) = \begin{cases} 1, & \lambda < \lambda_r \\ e^{-4(\lambda - \lambda_r)}, & \lambda \geq \lambda_r. \end{cases} \tag{3.39}
$$

We generate 50 signals from each of the three signal models on each of the graphs, use the sampling sets obtained from the all the methods to perform reconstruction and plot the mean of the mean squared error (MSE) for different sizes of sampling sets. For our algorithm, we set the value of k to 2, 8 and 14. The result is illustrated in Figure 3.1. Note that when the size of the sampling set is less than $r = 50$, the results are quite unstable. This is expected, because the uniqueness condition is not satisfied by the sampling set. Beyond $|\mathcal{S}| = r$, we make the following observations:

1. For the noise-free, bandlimited signal model F1, all methods lead to zero reconstruction error as soon as the size of the sampling set exceeds the signal cutoff $r = 50$ (error plots for this signal model are not shown). This is expected from the sampling theorem. It is interesting to note that in most cases, uniform random sampling does equally well, since the signal is noise-free and perfectly bandlimited.

2. For the noisy signal model F2 and the approximately bandlimited model F3, our method has better or comparable performance in most cases. This indicates that our method is fairly robust to noise and model mismatch. Uniform random sampling performs very badly as expected, because of lack of stability considerations.
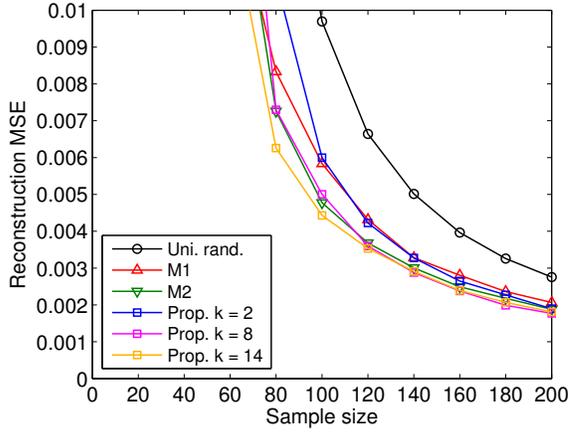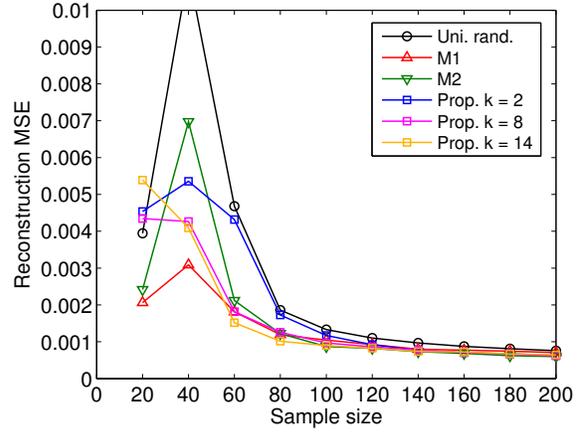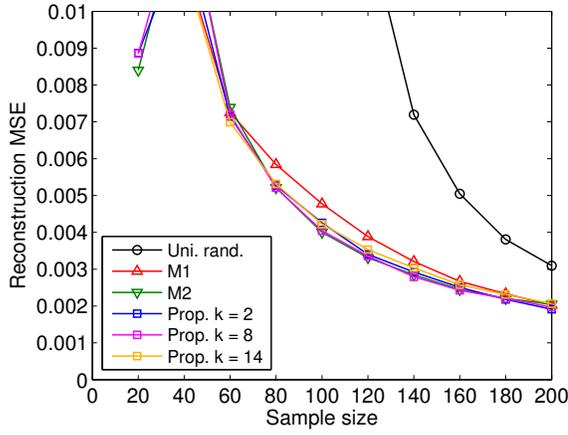
41
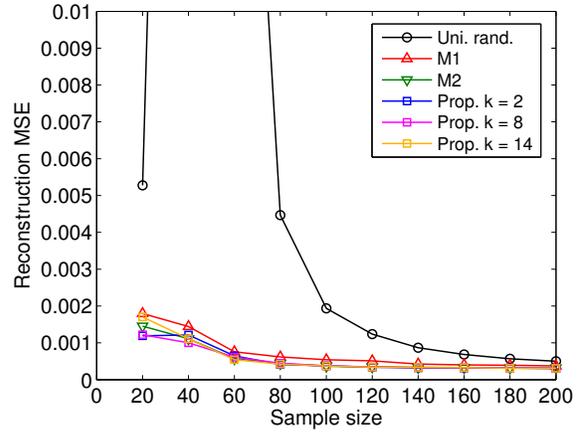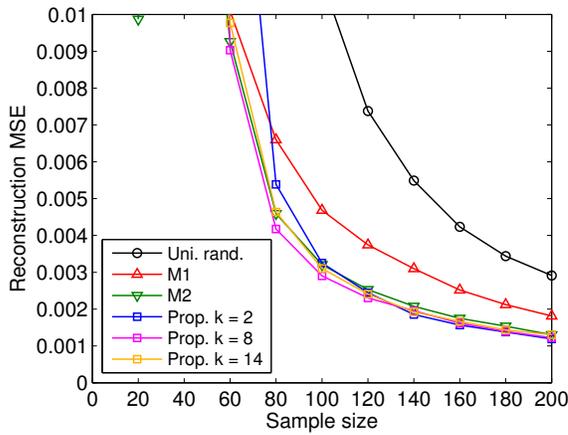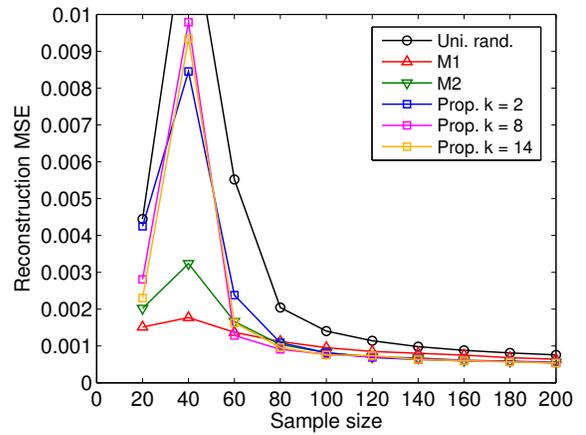
Figure 3.1: Reconstruction results for different graph and signal models. Plots for signal model F1 are not shown since the reconstruction errors are identically zero for all methods when $|\mathcal{S}| \geq \dim PW_\omega(G) = 50$. The large reconstruction errors for $|\mathcal{S}| < 50$ arise due to non-uniqueness of bandlimited reconstruction and hence, are less meaningful.

Effect of parameter k in the spectral proxy

Parameter k in the definition of spectral proxies controls how closely we estimate the bandwidth of any signal $\mathbf{f}$. Spectral proxies with higher values of k give a better approximation of the bandwidth. Our sampling set selection algorithm tries to maximize the smallest bandwidth that a signal in $L_2(\mathcal{S}^c)$ can have. Using higher values of k allows us to estimate this smallest bandwidth more closely, thereby leading to better sampling sets as demonstrated in Figure 3.2. Intuitively, maximizing $\Omega_k(\mathcal{S})$ with $k = 1$ ensures that the sampled nodes are well connected to the unsampled nodes [29] and thus, allows better propagation of the observed signal information. Using $k > 1$ takes into account multi-hop paths while ensuring better connectedness between $\mathcal{S}$ and $\mathcal{S}^c$. This effect is especially important in sparsely connected graphs and the benefit of increasing k becomes less noticeable when the graphs are dense as seen in Figure 3.2. However, this improvement in performance in the case of sparse graphs comes at the cost of increased numerical complexity.

Running time

We also compare the running times of the sampling set selection methods for different sizes of the graph. For our experiments, we generate symmetrized Erdös-Renyi random graphs of different sizes with parameter 0.01, and measure the average running time of selecting 5% of the samples in MATLAB. For computing the eigen-pairs, we use the code for the Locally Optimal Block Prec-conditioned Conjugate Gradient (LOBPCG) method available online [41] (this was observed to be faster than MATLAB's inbuilt sparse eigensolver `eigs`, which is based on Lanczos iterations). The results of the experiments are shown in Table 3.2. We observe that the rate of increase of running time as the graph size increases is slower for our method compared to other methods, thus making it more practical. Note that the increase with respect to k is nonlinear since the eigengaps are a function of k and lead to different number of iterations required for convergence of the eigenvectors.

3.6.2   A real data example

In this example, we apply the proposed method to classification of the USPS hand-written digit dataset [1]. This dataset consists of 1100 images of size 16×16 each corresponding digits 0 to 9. We create 10 different subsets of this dataset randomly,

(a) $p = 0.01$

(b) $p = 0.05$

(c) $p = 0.1$
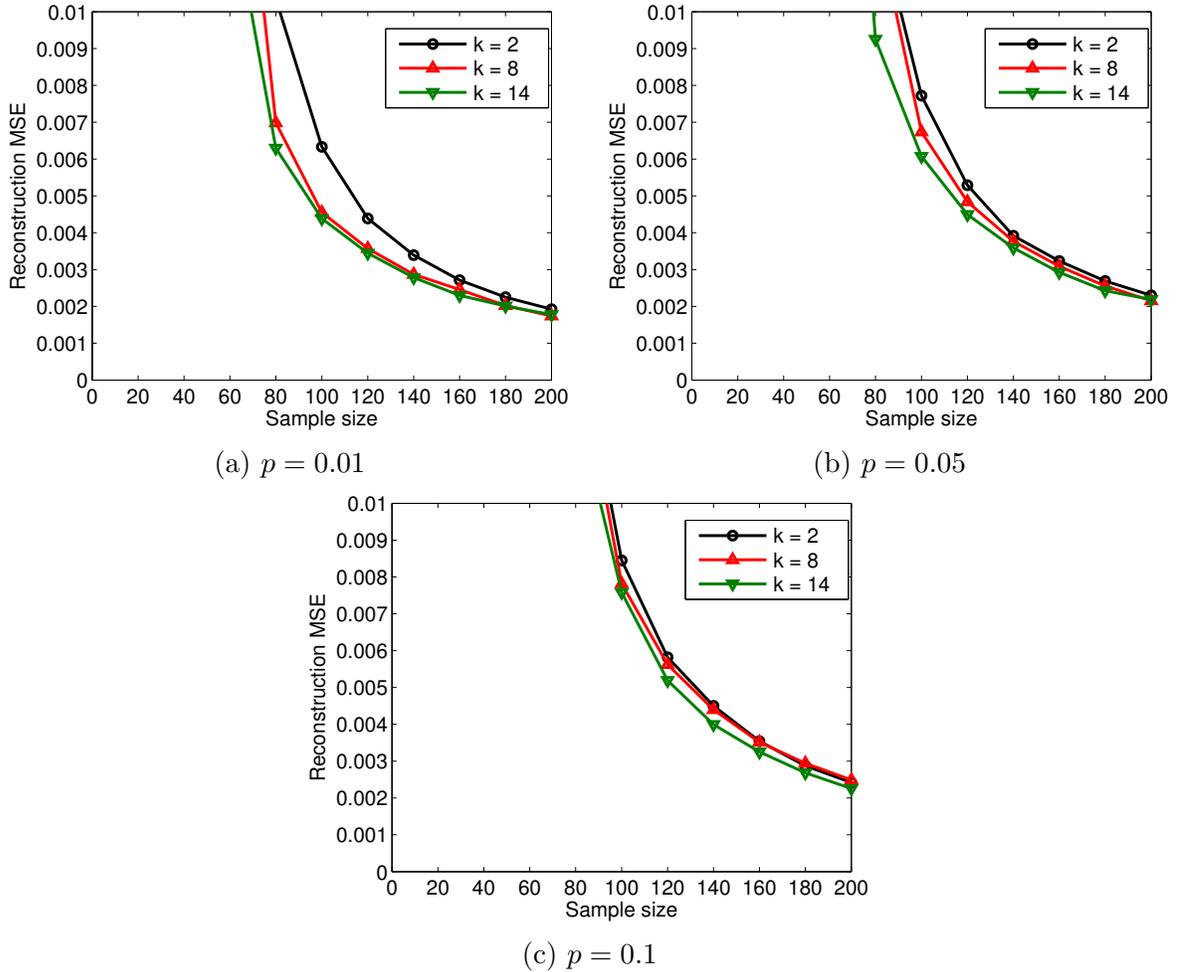
Figure 3.2: Reconstruction performance for noisy signals (model F2) with different values of k in Erdös-Renyi graphs having different connection sparsity levels. Higher connection probability p implies lower sparsity.

consisting of 100 images from each class. The data points can be thought of as points $\{\mathbf{x}_i\}_{i=1}^{1000} \subset \mathbb{R}^{256}$ with labels $\{\mathbf{y}_i\}_{i=1}^{1000}$. For each instance, we construct a symmetrized k-nearest neighbor (k-nn) graph with $k = 10$, where each node corresponds to a data point. We restrict the problem to the largest strongly connected component of the graph for convenience so that a stationary distribution for the resultant random walk exists which allows us to define the random walk based GFT. The graph signal is given by the membership function $\mathbf{f}^c$ of each class c which takes a value 1 on a node which belongs to the class and is 0 otherwise. To solve the multi-class classification task, we use the

Table 3.2: Running time of different methods (in seconds) for selecting 5% samples on graphs of different sizes. The running time for $M1$ increases drastically and is ignored beyond graph size 5k.

	1k	5k	10k	20k
M1	16.76	$12,322.72$	-	-
M2	2.16	57.46	425.92	3004.01
Proposed, $k=4$	2.00	11.13	84.85	566.39
Proposed, $k=6$	13.08	24.46	170.15	1034.21
Proposed, $k=8$	31.16	53.42	316.12	1778.31

one-vs-rest strategy which entails reconstructing the membership function of every class. The final classification for node i is then obtained by

$$\mathbf{y}_i = \arg\max_c \{\mathbf{f}_i^c\}. \tag{3.40}$$

We first compare the performance of the proposed method against M1 and M2 using the normalized adjacency matrix based GFT with the variation operator $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$. The bandwidth parameter r is set to 50. The plot of classification error averaged over the 10 dataset instances vs. number of labels is presented in Figure 3.3a. It shows that the proposed method has comparable performance despite being localized. The performance is also affected by the choice of the variation operators (or, the GFT bases). Figure 3.3b shows that the variation operators based on the hub-authority model and random walk offer higher classification accuracy and thus, are more suited for this particular application. Their superior performance can be explained by looking at the signal representation in the respective GFT domains. Figure 3.3c shows the fraction of signal energy captured in increasing number of GFT coefficients starting from low frequency. Since the hub-authority model based GFT and random walk based GFT offer more energy compaction than adjacency based GFT, the signal reconstruction quality using these bases is naturally better.

3.7 Summary

We studied the problem of selecting an optimal sampling set for reconstruction of bandlimited graph signals. The starting point of our framework is the notion of the Graph Fourier Transform (GFT) which is defined via an appropriate variation operator.
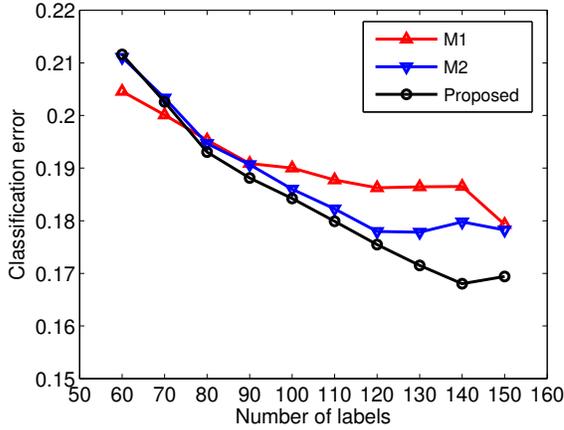
(a) Comparison of different methods using the adjacency based variation operator.
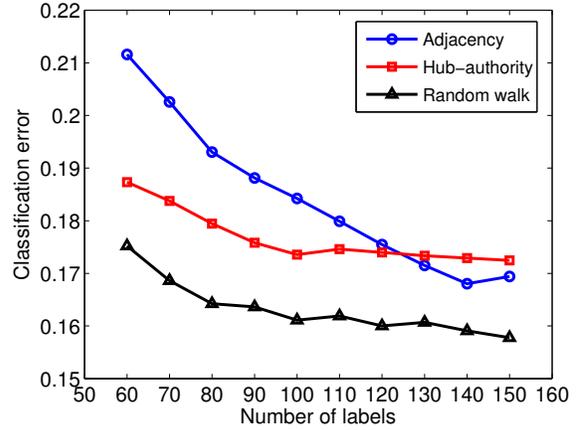
(b) Performance of proposed method with different choices of the variation operator.
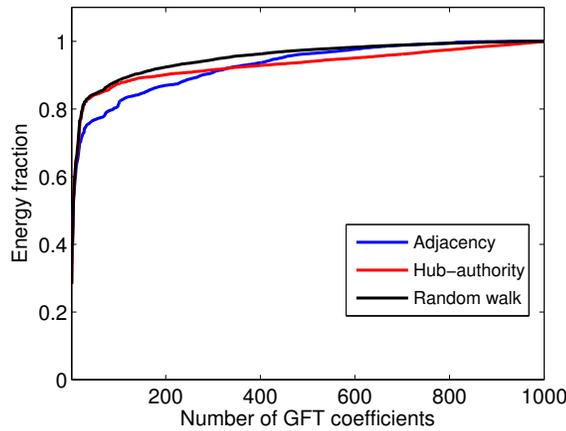
(c) Energy compaction with different GFTs obtained from different variation operators.

Figure 3.3: Classification results for the USPS dataset using different methods and GFTs.

Our goal is to find good sampling sets for reconstructing signals which are bandlimited in the above frequency domain. We showed that when the samples are noisy or the true signal is only approximately bandlimited, the reconstruction error depends not only on the model mismatch but also on the choice of sampling set. We proposed a measure of quality for the sampling sets, namely the cutoff frequency, that can be computed without finding the GFT basis explicitly. A sampling set that maximizes the cutoff frequency is shown to minimize the reconstruction error. We also proposed a greedy algorithm which finds an approximately optimal set. The proposed algorithm

can be efficiently implemented in a distributed and parallel fashion. Together with localized signal reconstruction methods, it gives an effective method for sampling and reconstruction of smooth graph signals on large graphs.

3.7.1 Future work

The present work opens up some new questions for future research. The problem of finding a sampling set with maximum cutoff frequency is combinatorial. The proposed greedy algorithm gives only an approximate solution to this problem. It would be useful to find efficient algorithms with theoretical guarantees on the quality of approximation. The accuracy-complexity trade-off provided by spectral proxies in computing the cutoff frequency indicates the existence of simple heuristics that can be used for fast sampling set selection at the expense of robustness of reconstruction. For example, maximizing the spectral proxy for $k = 1$ removes the need of computing even the smallest eigenpair and can be implemented entirely in the vertex domain. The heuristic in effect ensures that the unsampled nodes are strongly connected to the sampled ones. It would be interesting to understand heuristics arising from higher-order spectral proxies.

Another avenue for research is the consideration of randomized sampling algorithms (along the lines of [57]), that define a distribution over the nodes and provide probabilistic tradeoffs between the sampling set size and bounds on the reconstruction error. This would come however at the expense of increased size of the sampling set and increased reconstruction error. By carefully designing probability distributions for selecting the sampling sets, one can guarantee a probabilistic bound on the error as a function its cardinality using concentration inequalities from probability theory.

Finally, it would also be interesting if one can estimate the bandwidth of any signal using a few observed samples and **L**. This would help us decide "where to stop" while computing the sampling set sequentially, or in other words, understanding the cardinality of the sampling set required for recovering a signal without knowing it *apriori*. A possible approach for achieving this is to randomly observe a few samples of the signal and analyze the convergence of spectral proxies defined using these samples as a function of the cardinality. This can potentially provide a handle on the accuracy of the bandwidth estimate, that one can use to figure out when to stop sampling.

Chapter 4

Wavelet Filterbanks on Graphs

In this chapter, we explore the sampling problem further and consider the design of wavelet filterbanks on graphs. This problem is fundamentally different from sampling bandlimited signals since one needs to find a joint sampling scheme spanning over multiple channels of the filterbank and favoring all signals on the graph, i.e., not only bandlimited signals. These filterbanks are designed taking into account several desirable properties such as compact support, critical sampling, near orthogonality and near perfect reconstruction. Compact support implies using graph spectral filters that are polynomials of the graph adjacency or Laplacian matrix, thus helping keep time and space complexities in check. We show in this chapter that satisfying all these properties simultaneously for graph wavelet filterbanks is only possible under very restrictive conditions, since the structure of the GFT basis is dependent on the structure of the graph.

Recent approaches in graph wavelet filterbank design impose certain structural constraints on the graphs, for example by requiring the graphs to be bipartite or circulant. These assumptions significantly reduce the number of constraints needed to be satisfied for achieving the desirable properties mentioned earlier. Extending these designs to arbitrary graphs involves approximate decomposition into multiple subgraphs that satisfy the structural constraint at the cost of diminished multiresolution performance of the system. In this chapter, we circumvent this issue and design filterbanks directly on arbitrary graphs by decoupling the design of filters and choice of the sampling scheme. Given graph spectral filters that achieve desirable frequency localization, we consider the problem of choosing the best sampling scheme over multiple channels by minimizing a bound on the reconstruction error for the entire filterbank. This objective is optimized by a greedy minimization scheme that allows an efficient implementation, once again without the need of explicitly handling the GFT basis. Experimental results show that our scheme performs well in comparison to other adhoc sampling schemes in the literature. Finally, we also explore an interesting directed extension of bipartite graphs

---

This chapter is based partly on our work in [7, 8].

called M-block cylic graphs [69]. These graphs are useful in modeling periodic finite state machines (FSMs) and Markov decision processes (MDPs). The eigenstructure of these graphs makes them particularly suitable for the design of M-channel filterbanks.

The rest of this chapter is organized as follows: In Section 4.1, we review existing work on the design of graph wavelet filterbanks. Section 4.2 introduces some background and notations relevant for this chapter. Section 4.3 focuses on the general formulation behind two-channel filterbanks and introduces conditions required for attaining desirable properties. In Section 4.4, we design a critical sampling scheme that can be obtained efficiently given predesigned filters. We then turn to the design of M-channel filterbanks on M-block cyclic graphs in Section 4.5. Finally, we conclude the chapter with a summary and possible extensions of our work in Section 4.6.

4.1 Related work

State-of-the-art wavelet filterbanks that satisfy most of the above mentioned properties require imposing certain structural constraints on the underlying graph. For example, the recently proposed two-channel filterbanks in [45, 46] are designed specifically for bipartite graphs. The special structure leads to a natural downsampling-upsampling scheme (on one of the two partitions) in each channel, accompanied by a spectral folding phenomenon that is exploited while designing the filters. In order to extend the design to arbitrary graphs, these works suggest using a multidimensional framework where the input graph is decomposed into multiple bipartite subgraphs over which filterbanks are designed and implemented independently. Various approaches have been proposed to optimize the bipartite subgraph decomposition [52, 77] for designing these multidimensional filterbanks. However, the limitation of this framework is that one is forced to work with simplified graphs that do not contain all the connectivity information. Additionally, there are also works that suggest expanding the input graph to create a bipartite graph thereby leading to an oversampled filterbank [61], which may not be desirable for some applications such as compression. There also exist filterbanks that exploit circulant graph architectures [26, 27]. These works however consider only the analysis filterbank, whereas the synthesis part is assumed to be obtained via least-squares inversion.

Recently, there has also been interest in designing M-channel polynomial filterbanks. Once again, the filterbanks are designed on graphs with a special structure called M-block cyclic graphs [69]. These graphs exhibit an M-fold spectral folding phenomenon (in the GFT basis of the adjacency) upon downsampling-upsampling on any block.

This phenomenon is exploited to state perfect reconstruction conditions for M-channel filterbanks in [70]. However, these conditions are meant for graphs with balanced block sizes and a sampling scheme that involves downsampling-upsampling on the same block in each channel. Moreover, this work does not provide insight into possible solutions satisfying the constraints and suggests using existing filter designs from classical DSP. In our work, we remove the sampling restrictions and provide a possible solution for the perfect reconstruction conditions when M is a power of 2.

4.2   Background and notation

In this chapter, we work with weighted graphs $G = (\mathcal{V}, \mathcal{E})$ consisting of a set of nodes $\mathcal{V} = \{1, 2, \ldots, n\}$ and edges $\mathcal{E} = \{w_{ij}\}, i, j \in \mathcal{V}$, with $w_{ii} = 0$. We denote the adjacency matrix by $\mathbf{A}$ and the degree matrix by $\mathbf{D}$, and assume that $\mathbf{A}$ has been normalized[1] so that $\|\mathbf{A}\|_2 = 1$ (this facilitates the design of graph spectral filters).

When the graph is undirected, we shall work with the symmetric normalized form of the Laplacian defined as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. $\mathbf{L}$ is a symmetric positive semi-definite matrix and has a set of real eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n \leq 2$ and a corresponding orthogonal set of eigenvectors denoted as $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n]$.

We recall that the downsampling operation on a graph signal $\mathbf{f}$ is defined as the restriction of the signal $\mathbf{f}$ to a certain subset of nodes $\mathcal{S} \subset \mathcal{V}$ (known as the downsampling set), and the downsampled signal is a vector of reduced length $|\mathcal{S}|$. The downsampling operator for $\mathcal{S}$ is obtained by sampling the corresponding rows of the identity matrix $\mathbf{I}$, i.e., $\mathbf{S} = \mathbf{I}_{\mathcal{S},\mathcal{V}} \in \{0, 1\}^{|\mathcal{S}| \times n}$. Similarly, the upsampling operation for signals downsampled on $\mathcal{S}$ inserts zeros in place of the missing signal values at appropriate locations and is given by $\mathbf{S}^T$.

Further, while designing filterbanks, we shall employ polynomial graph filters $\mathbf{H} = h(\mathbf{L}) = \sum_{i=0}^{k} h_i \mathbf{L}^i$ (or alternatively, $\mathbf{H} = h(\mathbf{A}) = \sum_{i=0}^{k} h_i \mathbf{A}^i$, when we work with $\mathbf{A}$ in case of directed graphs). These filters are useful because of their efficiency (since a k-degree polynomial filter can be implemented in $O(k|\mathcal{E}|)$ complexity). Note that for undirected graphs, $\mathbf{L}$ is symmetric, and hence $\mathbf{H}$ is symmetric.

---

[1]There can be different ways of normalizing depending on the application at hand, *eg.*, $\frac{1}{|\lambda_{max}|}\mathbf{A}$, or random walk normalization $\mathbf{D}^{-1}\mathbf{A}$.
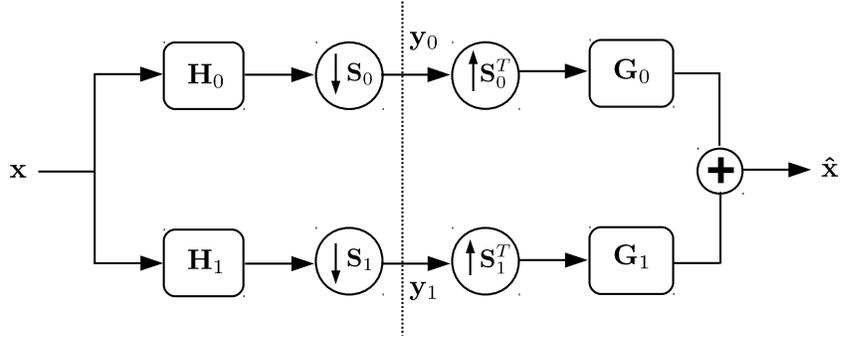
Figure 4.1: A generic two-channel filterbank on graphs.

4.3 Two-channel filterbanks

We now describe the general formulation for two-channel wavelet filterbanks on arbitrary undirected graphs (this can be easily extended to directed graphs). A more detailed description can be found in [45, 46] in the context of bipartite graphs. We make certain changes to notation for compactness.

A generic two-channel wavelet filterbank on a graph decomposes any graph signal $\mathbf{x} \in \mathbb{R}^N$ into a lowpass (smooth) and highpass (detail) component (Figure 4.1). It consists of an analysis filterbank with $\mathbf{H}_0$ and $\mathbf{H}_1$ as lowpass and highpass filters, and a synthesis filterbank with $\mathbf{G}_0$ and $\mathbf{G}_1$ as the lowpass and highpass filters. $\mathbf{S}_0 \in \{0,1\}^{|\mathcal{S}_0| \times N}$ and $\mathbf{S}_1 \in \{0,1\}^{|\mathcal{S}_1| \times N}$ are the downsampling operators for the lowpass and highpass branch, respectively, while $\mathbf{S}_0^T$ and $\mathbf{S}_1^T$ are the corresponding upsampling operators. The outputs of the two branches after the analysis filterbank are $\mathbf{y}_0 \in \mathbb{R}^{|\mathcal{S}_0|}$ and $\mathbf{y}_1 \in \mathbb{R}^{|\mathcal{S}_1|}$. These are given as

$$\begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{S}_0 \mathbf{H}_0 \\ \mathbf{S}_1 \mathbf{H}_1 \end{bmatrix} \mathbf{x} = \mathbf{T}_a \mathbf{x}. \tag{4.1}$$

Similarly, the output of the synthesis filterbank (i.e., the reconstructed signal) is denoted as $\hat{\mathbf{x}} \in \mathbb{R}^N$ and is given by

$$\hat{\mathbf{x}} = \begin{bmatrix} \mathbf{G}_0 \mathbf{S}_0^T & \mathbf{G}_1 \mathbf{S}_1^T \end{bmatrix} \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \end{bmatrix} = \mathbf{T}_s \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \end{bmatrix}, \tag{4.2}$$

with the complete transfer equation for the system given by

$$\hat{\mathbf{x}} = \left(\mathbf{G}_0 \mathbf{S}_0^T \mathbf{S}_0 \mathbf{H}_0 + \mathbf{G}_1 \mathbf{S}_1^T \mathbf{S}_1 \mathbf{H}_1 \right) \mathbf{x}. \tag{4.3}$$

We now state some desirable characteristics of the filterbank along with the conditions needed to satisfy each.

- *Compact support* requires that the filters $\{\mathbf{H}_i, \mathbf{G}_i\}_{i=0,1}$ be expressible as finite polynomials of the graph Laplacian $\mathbf{L}$ (or $\mathbf{A}$ for directed graphs), a notion analogous to FIR filters in classical DSP. A k-degree polynomial filter requires collecting information from a k-degree neighborhood for each node.

- *Critical sampling* requires that the total number of samples after downsampling in both branches should be equal to the dimension of the signal, i.e., $|\mathcal{S}_0| + |\mathcal{S}_1| = N$. If the sampling scheme is constrained to disjoint sets, this can be stated in terms of the sampling operators as

$$\mathbf{S}_0^T \mathbf{S}_0 + \mathbf{S}_1^T \mathbf{S}_1 = \mathbf{I}. \tag{4.4}$$

- *Perfect reconstruction* requires that the transfer function of the entire system be identity, i.e.,

$$\mathbf{G}_0 \mathbf{S}_0^T \mathbf{S}_0 \mathbf{H}_0 + \mathbf{G}_1 \mathbf{S}_1^T \mathbf{S}_1 \mathbf{H}_1 = \mathbf{I}. \tag{4.5}$$

- *Orthogonality* requires the filterbanks to satisfy $\mathbf{T}_s = \mathbf{T}_a^T$ and $\mathbf{T}_a^T \mathbf{T}_a = \mathbf{I}$, which translates to substituting $\mathbf{G}_0 = \mathbf{H}_0$ and $\mathbf{G}_1 = \mathbf{H}_1$ in (4.5).

Note that the perfect reconstruction condition in (4.5) can also be interpreted using the eigendecomposition of $\mathbf{L}$ or $\mathbf{A}$ as $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$ as

$$g_0(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{S}_0^T \mathbf{S}_0 \mathbf{U} h_0(\mathbf{\Lambda}) + g_1(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{S}_1^T \mathbf{S}_1 \mathbf{U} h_1(\mathbf{\Lambda}) = \mathbf{I}, \tag{4.6}$$

For an arbitrary $\mathbf{U}$, it is impossible to satisfy (4.6) using low-degree polynomial filters, since the number of constraints $(= N^2)$ is much larger than the available degrees of freedom. Therefore, one would like to design the system such that $\mathbf{G}_0 \mathbf{S}_0^T \mathbf{S}_0 \mathbf{H}_0 + \mathbf{G}_1 \mathbf{S}_1^T \mathbf{S}_1 \mathbf{H}_1$ is as close as possible to identity. Special structure in the graph results in a structured $\mathbf{U}$ and therefore simplification of (4.6) by elimination of several constraints, as shown next for bipartite graphs.

4.3.1 Special case: bipartite graphs

The special structure of bipartite graphs leads to a *spectral folding* phenomenon that eliminates several constraints in (4.6), thereby allowing two-channel filterbank designs

using low-degree polynomial filters [45, 46]. We now explain this phenomenon in the context of undirected bipartite graphs, however, this analysis can easily extended to directed bipartite graphs, and also M-block cyclic graphs as we shall see in Section 4.5.1.

We begin by noting the following properties of the symmetric normalized Laplacian $\mathbf{L}$ for bipartite graphs:

1. The eigenvectors of $\mathbf{L}$ exhibit a "spectral folding" phenomenon [45], i.e., if $\left\{\lambda, \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \end{pmatrix}\right\}$ is an eigenpair of $\mathbf{L}$ (where $\mathbf{u}_0$ and $\mathbf{u}_1$ are values on partitions $\mathcal{S}_0$ and $\mathcal{S}_1$, respectively), then $\left\{2-\lambda, \begin{pmatrix} \mathbf{u}_0 \\ -\mathbf{u}_1 \end{pmatrix}\right\}$ is also an eigenpair.

2. Using orthogonality of the eigenvectors $\begin{pmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \end{pmatrix}$ and $\begin{pmatrix} \mathbf{u}_0 \\ -\mathbf{u}_1 \end{pmatrix}$, we have $\mathbf{u}_0^T\mathbf{u}_0 - \mathbf{u}_1^T\mathbf{u}_1 = 0 \Rightarrow \mathbf{u}_0^T\mathbf{u}_0 = \mathbf{u}_1^T\mathbf{u}_1$. Further, since the eigenvectors are normalized, we have $\mathbf{u}_0^T\mathbf{u}_0 + \mathbf{u}_1^T\mathbf{u}_1 = 1$, which gives $\mathbf{u}_0^T\mathbf{u}_0 = \mathbf{u}_1^T\mathbf{u}_1 = 1/2$.

3. Further, using orthogonality of the eigenvector $\begin{pmatrix} \mathbf{u}_0^T \\ \mathbf{u}_1 \end{pmatrix}$ to $\begin{pmatrix} \mathbf{u}_0' \\ \mathbf{u}_1' \end{pmatrix}$ and $\begin{pmatrix} \mathbf{u}_0' \\ -\mathbf{u}_1' \end{pmatrix}$, we get $\mathbf{u}_0^T\mathbf{u}_0' + \mathbf{u}_1^T\mathbf{u}_1' = 0$ and $\mathbf{u}_0^T\mathbf{u}_0' - \mathbf{u}_1^T\mathbf{u}_1' = 0$, which gives us $\mathbf{u}_0^T\mathbf{u}_0' = \mathbf{u}_1^T\mathbf{u}_1' = 0$.

We can use the above three properties to simplify (4.6). For simplicity, let us consider balanced bipartite graphs (that have equal-sized partitions) with distinct eigenvalues. The eigenvector matrix for such graphs can be written as

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_0 & \mathbf{U}_0^* \\ \mathbf{U}_1 & -\mathbf{U}_1^* \end{bmatrix}, \tag{4.7}$$

where $\mathbf{U}_0^*$ and $\mathbf{U}_1^*$ are obtained from $\mathbf{U}_0$ and $\mathbf{U}_1$ by reversing or mirroring the column order. Therefore, if $\mathcal{S}_0$, $\mathcal{S}_1$ are two sets of the bipartition, then it can be shown that

$$\mathbf{U}^T\mathbf{S}_0^T\mathbf{S}_0\mathbf{U} = \begin{bmatrix} \mathbf{U}_0^T \\ \mathbf{U}_0^{*T} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 & \mathbf{U}_0^* \end{bmatrix} = \frac{1}{2}(\mathbf{I} + \mathbf{I}^*), \tag{4.8}$$

$$\mathbf{U}^T\mathbf{S}_1^T\mathbf{S}_1\mathbf{U} = \begin{bmatrix} \mathbf{U}_1^T \\ -\mathbf{U}_1^{*T} \end{bmatrix} \begin{bmatrix} \mathbf{U}_1 & -\mathbf{U}_1^* \end{bmatrix} = \frac{1}{2}(\mathbf{I} - \mathbf{I}^*), \tag{4.9}$$

where $\mathbf{I}^*$ is the anti-diagonal identity matrix. Noting that $\mathbf{U}^{-1} = \mathbf{U}^T$ for the undirected case and substituting (4.8) and (4.9) in the left hand side of (4.6), we conclude that (4.6)

is satisfied if the following conditions on the filter responses hold in the spectral domain for $0 \leq \lambda \leq 2$:

$$g_0(\lambda)h_0(\lambda) + g_1(\lambda)h_1(\lambda) = 2, \tag{4.10}$$

$$g_0(\lambda)h_0(2-\lambda) - g_1(\lambda)h_1(2-\lambda) = 0. \tag{4.11}$$

These are exactly identical to the perfect reconstruction conditions stated in [45]. $\mathbf{I}^*$ causes the spectral folding phenomenon and thus generates N additional aliasing constraints besides the N diagonal constraints, resulting in a total of $2N$ constraints that are easier to satisfy with low-degree filters. The analysis can be extended to bipartite graphs with unbalanced partitions and possibly repeated eigenvalues through simple modifications and can be shown to produce the same perfect reconstruction conditions. Note that if we are not restricted to using polynomial filters for synthesis, one can use least-squares inversion for inverting the analysis transfer function $\mathbf{T}_a$, provided it is non-singular.

4.3.2 Characterizing graphs that admit perfect reconstruction filterbanks

We now characterize graphs that admit a critically-sampled, compact support, perfect reconstruction design for the two-channel filterbank depicted in Figure 4.1. For the analysis to hold for both undirected and directed graphs, we work with filters that are polynomial in $\mathbf{A}$ with appropriate normalization[2]. Once again, we begin with the transfer function in the spectral domain as

$$\tilde{\mathbf{T}} = g_0(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{S}_0^T\mathbf{S}_0\mathbf{U}h_0(\mathbf{\Lambda}) + g_1(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{S}_1^T\mathbf{S}_1\mathbf{U}h_1(\mathbf{\Lambda}). \tag{4.12}$$

Recall that for critically-sampled (on disjoint sets), perfect reconstruction polynomial filterbanks, we should satisfy $\tilde{\mathbf{T}} = \mathbf{I}$ with $\mathbf{S}_0^T\mathbf{S}_0 + \mathbf{S}_1^T\mathbf{S}_1 = \mathbf{I}$ and low-degree polynomial

---

[2]Normalization is carried out as $\mathbf{D}^{-1}\mathbf{A}$ or $\frac{1}{\lambda_{\max}(\mathbf{A})}\mathbf{A}$ for directed graphs and $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ for undirected graphs. Note that both these normalizations lead to the following bound on the frequencies: $\lambda \in [-1, 1]$.

filter responses $\{h_k(\lambda), g_k(\lambda)\}$. Similar to [45], let us define a modulation function $\boldsymbol{\beta} \in \{-1, +1\}^n$ and a corresponding modulation operator $\boldsymbol{\Omega}$ as

$$\boldsymbol{\Omega} = \text{diag}(\boldsymbol{\beta}), \text{ where } \boldsymbol{\beta}(v) = \begin{cases} +1 & v \in \mathcal{S}_0 \\ -1 & v \in \mathcal{S}_1 \end{cases}. \tag{4.13}$$

The modulation operator satisfies $\boldsymbol{\Omega}^2 = \mathbf{I}$. With this definition, we have the following relations for critical sampling in the two channels

$$\mathbf{S}_0^T \mathbf{S}_0 = \frac{1}{2}(\mathbf{I} + \boldsymbol{\Omega}), \tag{4.14}$$

$$\mathbf{S}_1^T \mathbf{S}_1 = \frac{1}{2}(\mathbf{I} - \boldsymbol{\Omega}), \tag{4.15}$$

using which (4.12) can be rewritten as

$$\tilde{\mathbf{T}} = \underbrace{\frac{1}{2}\bigg(g_0(\boldsymbol{\Lambda})h_0(\boldsymbol{\Lambda}) + g_1(\boldsymbol{\Lambda})h_1(\boldsymbol{\Lambda})\bigg)}_{\tilde{\mathbf{T}}_{\text{gain}}}$$
$$+ \underbrace{\frac{1}{2}\bigg(g_0(\boldsymbol{\Lambda})\mathbf{U}^{-1}\boldsymbol{\Omega}\mathbf{U}h_0(\boldsymbol{\Lambda}) - g_1(\boldsymbol{\Lambda})\mathbf{U}^{-1}\boldsymbol{\Omega}\mathbf{U}h_1(\boldsymbol{\Lambda})\bigg)}_{\tilde{\mathbf{T}}_{\text{alias}}}. \tag{4.16}$$

The first term on the right hand side of the above equation, denoted by $\tilde{\mathbf{T}}_{\text{gain}}$, is diagonal in the spectral domain and therefore determines the *gain* of the transfer function. The second expression, denoted by $\tilde{\mathbf{T}}_{\text{alias}}$, is termed as the *aliasing* component since $\mathbf{U}^{-1}\boldsymbol{\Omega}\mathbf{U}$ is not diagonal in general ($\boldsymbol{\Omega}$ is not simultaneously diagonalizable with $\mathbf{A}$). As a result, we observe an *input-dependent smearing* in the spectrum which is difficult to reverse. Therefore, for perfect reconstruction, the filter responses and the sampling scheme must be chosen such that $\tilde{\mathbf{T}}_{\text{gain}} = \mathbf{I}$ and $\tilde{\mathbf{T}}_{\text{alias}} = \mathbf{0}$.

In order to characterize aliasing, we expand the modulated basis vectors $\boldsymbol{\Omega}\mathbf{U}$ in the original basis $\mathbf{U}$, by finding a $\mathbf{P}$ such that

$$\boldsymbol{\Omega}\mathbf{U} = \mathbf{U}\mathbf{P}. \tag{4.17}$$

$\mathbf{P}$ contains the coefficients for expressing $\boldsymbol{\Omega}\mathbf{U}$ in $\mathbf{U}$ and since $\mathbf{P} = \mathbf{U}^{-1}\boldsymbol{\Omega}\mathbf{U}$, it also determines the aliasing pattern in the transfer function (illustrated in Figure 4.2 for an

$$\text{Aliasing pattern:} \quad \mathbf{\Omega} \underbrace{\begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}}_{\mathbf{U}} = \underbrace{\begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}}_{\mathbf{U}} \underbrace{\begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}}_{\mathbf{P}}$$

$$\text{Anti-aliasing constraint:} \quad \tilde{\mathbf{T}}_{\text{alias}} = g_0(\mathbf{\Lambda}) \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix} h_0(\mathbf{\Lambda}) - g_1(\mathbf{\Lambda}) \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix} h_1(\mathbf{\Lambda}) = \mathbf{0}$$

(a)

$$\text{Aliasing pattern:} \quad \mathbf{\Omega} \underbrace{\begin{pmatrix} \cdot \\ \cdot \\ \cdot \end{pmatrix}}_{\mathbf{U}} = \underbrace{\begin{pmatrix} \cdot \\ \cdot \\ \cdot \end{pmatrix}}_{\mathbf{U}} \underbrace{\begin{pmatrix} \cdot \\ \cdot \end{pmatrix}}_{\mathbf{P}}$$

$$\text{Anti-aliasing constraint:} \quad \tilde{\mathbf{T}}_{\text{alias}} = g_0(\mathbf{\Lambda}) \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} h_0(\mathbf{\Lambda}) - g_1(\mathbf{\Lambda}) \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} h_1(\mathbf{\Lambda}) = \mathbf{0}$$

$$\Rightarrow \tilde{\mathbf{T}}_{\text{alias}} = g_0(\lambda) h_0(-\lambda) - g_1(\lambda) h_1(-\lambda) = 0$$

(b)

Figure 4.2: An illustration of aliasing patterns for two-channel filterbanks in (a) an arbitrary graph, (b) a bipartite graph. Spectral folding in a bipartite graph results in a concise anti-aliasing constraint in the spectral domain as seen in Section 4.3.1.

arbitrary graph and a bipartite graph). A minimum number of constraints is generated from the condition $\tilde{\mathbf{T}}_{\text{alias}} = \mathbf{0}$ when $\mathbf{P}$ is a permutation matrix. This happens if and only if for all GFT basis vectors $\mathbf{u}$, their modulated versions $\mathbf{\Omega u}$ are also elements of the basis. In other words, if $\{\lambda, \mathbf{u}\}$ is an eigenpair of $\mathbf{A}$, then $\{\mu, \mathbf{\Omega u}\}$ is also an eigenpair.

Observation 4.1. *Minimum number of anti-aliasing constraints is generated if there exists a one-to-one mapping between GFT vectors and their modulated versions. In other words, modulating the GFT matrix is equivalent to applying a column permutation.*

To ensure that polynomial filters can be designed independent of the graph (i.e., without knowing its size or spectrum), the frequencies associated with $\mathbf{u}$ and $\mathbf{\Omega u}$, λ and μ respectively, must be related in a simple fashion. Specifically, we must have $\mu = p(\lambda)$, where $p(\lambda)$ is a continuous function. Continuity of $p(\lambda)$ is required since piecewise functions cannot be expressed as polynomials, and hence cannot be implemented as polynomial filters in the vertex domain. Moreover, since $\mu = p(\lambda)$ and $\lambda = p(\mu)$, p must be an involutory function (i.e., $p(p(\lambda)) = \lambda$) in $[-1, 1]$. Using this relationship in the

aliasing term in (4.16), we conclude that the following condition must be satisfied to eliminate all aliasing:

$$g_0(\lambda)h_0(p(\lambda)) - g_1(\lambda)h_1(p(\lambda)) = 0. \tag{4.18}$$

Note that (4.18) is satisfied if $g_0(\lambda) = h_1(p(\lambda))$ and $g_1(\lambda) = h_0(p(\lambda))$. Further, in order to ensure $g_0(\lambda)$ and $g_1(\lambda)$ are polynomials, the only choice of $p(\lambda)$ is $p(\lambda) = c - \lambda$ (since higher order polynomials are not involutory in $[-1, 1]$). Note that $\text{Tr}(\mathbf{A}) = \sum_{i=1}^{N} \lambda_i = 0$. But we also have $\text{Tr}(\mathbf{A}) = \sum_{i=1}^{N} (c - \lambda_i) = 0$. Therefore, $c = 0$ is the only choice satisfying all our design criteria. This means both $\{\lambda, \mathbf{u}\}$ and $\{-\lambda, \mathbf{\Omega u}\}$ are eigenpairs of $\mathbf{A}$, and this can be true if and only if the graph is bipartite [24]. We summarize our analysis in the following observation:

Observation 4.2. *Two-channel perfect reconstruction filterbanks satisfying the following design criteria:*

1. *Polynomial filters,*

2. *Disjoint sampling sets,*

3. *Independent from graph size N (i.e., constraints expressible as a function of λ),*

4. *Minimum number of anti-aliasing constraint equations (equal to two),*

can be designed if and only if the graph is bipartite.

The criteria in the observation above reduce the problem to the design of four polynomials $h_0(\lambda), h_1(\lambda), g_0(\lambda), g_1(\lambda)$ in the spectral domain satisfying two concise constraint equations. Although this leads to a significant simplification of the design process, we must remark that these criteria can be somewhat restrictive as the design is viable only on one candidate graph, i.e., bipartite. We cannot deny the possibility that filterbanks can be realized on other kinds of graphs if one or more of these criteria is relaxed. For example, an alternative sampling approach has recently been proposed in [67], where spectral folding is embedded in the sampling strategy itself through spectral domain sampling. Such a sampling scheme allows for the design for filterbanks on any graph, at the expense of increased complexity in the sampling step.

In the next section, we relax the perfect reconstruction requirement and explore critically-sampled polynomial designs on arbitrary graphs.

4.4 Critical sampling for filterbanks on arbitrary graphs

4.4.1 Approximately optimal sampling scheme

For a critically sampled design, we must choose $\mathcal{S}_0$ and $\mathcal{S}_1$ such that $|\mathcal{S}_0| + |\mathcal{S}_1| = N$ and the filterbank is as close to perfect reconstruction as possible. One way to achieve this is by minimizing the deviation of the overall transfer function of the system from identity in terms of Frobenius form, i.e., $\|\mathbf{G}_0\mathbf{S}_0^T\mathbf{S}_0\mathbf{H}_0 + \mathbf{G}_1\mathbf{S}_1^T\mathbf{S}_1\mathbf{H}_1 - \mathbf{I}\|_F^2$, which is in fact an upper bound on the squared relative error for all signals on the graph. Further, in our design, we assume that we have already designed filters $\mathbf{H}_0, \mathbf{H}_1, \mathbf{G}_0, \mathbf{G}_1$ to satisfy $\mathbf{G}_0\mathbf{H}_0 + \mathbf{G}_1\mathbf{H}_1 = 2\mathbf{I}$ which is the overcomplete case. The filters can be designed, for example, using the methods of [45, 46]. In order to minimize the reconstruction error over the choice of sampling sets $\mathcal{S}_0$ and $\mathcal{S}_1$, we first introduce a concatenated setting (of $2N$ dimensions) by defining

$$\mathbf{H} = \begin{bmatrix}\mathbf{H}_0 \\ \mathbf{H}_1\end{bmatrix} \in \mathbb{R}^{2N \times N}, \ \mathbf{G} = \begin{bmatrix}\mathbf{G}_0 \\ \mathbf{G}_1\end{bmatrix} \in \mathbb{R}^{2N \times N},$$

$$\mathbf{y} = \begin{bmatrix}\mathbf{y}_0 \\ \mathbf{y}_1\end{bmatrix} \in \mathbb{R}^{|\mathcal{S}_0|+|\mathcal{S}_1|}, \ \mathbf{S} = \begin{bmatrix}\mathbf{S}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_1\end{bmatrix} \in \{0,1\}^{(|\mathcal{S}_0|+|\mathcal{S}_1|)\times 2N}. \tag{4.19}$$

Note that the concatenated downsampling operator $\mathbf{S}$ can be obtained by sampling rows of the $2N$-dimensional identity corresponding to indices in a concatenated sampling set $\mathcal{S} \subset \{1, \ldots, 2N\}$ that contains sampled nodes for both the channels such that $|\mathcal{S}| = |\mathcal{S}_0| + |\mathcal{S}_1|$. Further, $\mathcal{S}^c = \{1, \ldots, 2N\} \setminus \mathcal{S}$ and $\mathcal{S}_0$ and $\mathcal{S}_1$ can be recovered from $\mathcal{S}$ as $\mathcal{S}_0 = \{v|v \in \mathcal{S}, 1 \leq v \leq N\}$ and $\mathcal{S}_1 = \{v - N|v \in \mathcal{S}, n + 1 \leq v \leq 2N\}$. With these definitions, the transfer function of the system can be written as $\mathbf{G}^T\mathbf{S}^T\mathbf{S}\mathbf{H}$ and finding a critical sampling scheme requires solving

$$\min_{\mathcal{S}:|\mathcal{S}|=N} \left\|\mathbf{G}^T\mathbf{S}^T\mathbf{S}\mathbf{H} - \mathbf{I}\right\|_F^2. \tag{4.20}$$

Algorithm 4.1 Basic greedy minimization

Ensure: $\mathcal{S} = \{\emptyset\}$.
1: **while** $|\mathcal{S}| < N$ **do**
2: $\quad \mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$, where $u = \operatorname{argmin}_{v \in \mathcal{S}^c} \phi(\mathcal{S} \cup \{v\})$.
3: **end while**

Since we choose the filters such that $\mathbf{G}^T \mathbf{H} = 2\mathbf{I}$, we can rewrite the objective as

$$\phi(\mathcal{S}) = \left\| \mathbf{G}^T \mathbf{S}^T \mathbf{S} \mathbf{H} - \frac{1}{2} \mathbf{G}^T \mathbf{H} \right\|_F^2$$

$$= \left\| \frac{1}{2} \sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \frac{1}{2} \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right\|_F^2, \tag{4.21}$$

where $\mathbf{g}_i$ and $\mathbf{h}_i$ denote the ith columns of $\mathbf{G}^T$ and $\mathbf{H}^T$ respectively. In order to minimize $\phi(\mathcal{S})$, we propose to use a simple greedy procedure (Algorithm 4.1) that begins with an empty $\mathcal{S}$ and keeps adding nodes one-by-one that minimize $\phi(\mathcal{S})$ at each step. However, this algorithm requires $O(N^2)$ evaluations of the objective $\phi(\mathcal{S})$ which can be quite expensive. Explicitly storing the matrices $\mathbf{G}$ and $\mathbf{H}$ requires $O(N^2)$ space. We now show how one can efficiently implement the algorithm in $O(N)$ graph filtering operations and $O(N)$ space. Using (4.21), the change in the objective $\phi(\mathcal{S})$ when a node $v \subset \{1, \ldots, 2N\}$ is added to $\mathcal{S}$ is given by:

$$\phi(\mathcal{S} \cup \{v\}) = \left\| \left(\frac{1}{2} \sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \frac{1}{2} \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right) + \mathbf{g}_v \mathbf{h}_v^T \right\|_F^2$$

$$= \phi(\mathcal{S}) + p_v(\mathcal{S}) + q_v, \tag{4.22}$$

where we defined

$$p_v(\mathcal{S}) = \operatorname{Tr} \left[\mathbf{h}_v \mathbf{g}_v^T \left(\sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right) \right], \tag{4.23}$$

$$q_v = \|\mathbf{g}_v\|^2 \|\mathbf{h}_v\|^2. \tag{4.24}$$

Thus, we have

$$\operatorname*{argmin}_{v \in \mathcal{S}^c} \phi(\mathcal{S} \cup \{v\}) = \operatorname*{argmin}_{v \in \mathcal{S}^c} \left(p_v(\mathcal{S}) + q_v \right). \tag{4.25}$$

59

Algorithm 4.2 Efficient algorithm for critical sampling
***

Require: Graph $G = \{\mathcal{V}, E\}$, concatenated filters $\mathbf{H}$, $\mathbf{G}$.
Ensure: $\mathcal{S} = \{\emptyset\}$, $\mathbf{p}, \mathbf{q} \in \mathbb{R}^{2N}$ such that $p_v = -2 \langle \mathbf{g}_v, \mathbf{h}_v \rangle$, $q_v = \|\mathbf{g}_v\|^2 \|\mathbf{h}_v\|^2$.
 1: **while** $|\mathcal{S}| < N$ **do**
 2:     $\mathcal{S} \leftarrow \mathcal{S} \cup \{u\}$, where $u = \text{argmin}_{v \in \mathcal{S}^c} (p_v + q_v)$.
 3:     $\mathbf{p} \leftarrow \mathbf{p} + 2(\mathbf{Gg}_u) \circ (\mathbf{Hh}_u)$.
 4: **end while**
***

In order to compute $p_v(\mathcal{S})$ for each $\mathcal{S}$, we first note that

$$p_v(\emptyset) = \text{Tr}\left[\mathbf{h}_v \mathbf{g}_v^T (-\mathbf{G}^T \mathbf{H})\right] = -2 \langle \mathbf{g}_v, \mathbf{h}_v \rangle. \tag{4.26}$$

Further, for a node u, $p_v(\mathcal{S} \cup \{u\})$ can be computed as

$$p_v(\mathcal{S} \cup \{u\}) = \text{Tr}\left[\mathbf{h}_v \mathbf{g}_v^T \left(\sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T + 2\mathbf{g}_u \mathbf{h}_u^T\right)\right]$$
$$= p_v(\mathcal{S}) + 2 \langle \mathbf{g}_v, \mathbf{g}_u \rangle \langle \mathbf{h}_v, \mathbf{h}_u \rangle. \tag{4.27}$$

To make the notation compact, we introduce the vectors $\mathbf{p}(\mathcal{S}), \mathbf{q} \in \mathbb{R}^{2N}$, whose v^{th} elements are $p_v(\mathcal{S})$ and q_v. Therefore, using "$\circ$" to denote element-wise vector product (Hadamard product), we have

$$\mathbf{p}(\mathcal{S} \cup \{u\}) = \mathbf{p}(\mathcal{S}) + 2(\mathbf{Gg}_u) \circ (\mathbf{Hh}_u). \tag{4.28}$$

We summarize the efficient method for choosing $\mathcal{S}$ in Algorithm 4.2. Note that the algorithm is allowed to produce sampling sets $\mathcal{S}_0$ and $\mathcal{S}_1$ that are not disjoint. In order to avoid this, a simple heuristic is to constrain the algorithm in each iteration to disregard "images" (created in the concatenated setting) of already selected indices (*eg.*, selection of $i \le N$ rules out $i + N$ from list of candidate indices in subsequent iterations).

Complexity: The vectors $\mathbf{h}_v$ and $\mathbf{g}_v$ can be computed using two filtering operations each as $\mathbf{H}^T \delta_v$ and $\mathbf{G}^T \delta_v$ respectively, where δ_v is the graph delta signal on node v. Therefore, in terms of time complexity, computing $\mathbf{p}(\emptyset)$ and $\mathbf{q}$ require $4N$ one-time graph filtering operations in total. Further, each greedy iteration requires performing 8 filtering operations. Therefore, Algorithm 4.2 requires $O(N)$ graph filtering operations. The space complexity of the algorithm is $O(N)$ since it is matrix-free, i.e., $\mathbf{A}$ is the only matrix that needs to be stored.

4.4.2 Theoretical guarantees

We now show that it is possible to obtain some theoretical insight into the performance of (a randomized variant of) our greedy algorithm when $\mathbf{G} = \mathbf{H}$. Note that for $\mathcal{S}_1 \subseteq \mathcal{S}_2$ and a $v \notin \mathcal{S}_1, \mathcal{S}_2$,

$$
\begin{aligned}
p_v(\mathcal{S}_1) &= \sum_{i \in \mathcal{S}_1} \langle \mathbf{h}_v, \mathbf{h}_i \rangle^2 - \sum_{j \in \mathcal{S}_1^c} \langle \mathbf{h}_v, \mathbf{h}_j \rangle^2 \\
&\leq \sum_{i \in \mathcal{S}_2} \langle \mathbf{h}_v, \mathbf{h}_i \rangle^2 - \sum_{j \in \mathcal{S}_2^c} \langle \mathbf{h}_v, \mathbf{h}_j \rangle^2 = p_v(\mathcal{S}_2).
\end{aligned} \tag{4.29}
$$

Using this in (4.22), we obtain

$$
\phi(\mathcal{S}_1 \cup \{v\}) - \phi(\mathcal{S}_1) \leq \phi(\mathcal{S}_2 \cup \{v\}) - \phi(\mathcal{S}_2), \tag{4.30}
$$

which implies $\phi(\mathcal{S})$ is supermodular in $\mathcal{S}$. Therefore, the function $\psi(\mathcal{S}) = \phi(\emptyset) - \phi(\mathcal{S})$ is submodular, non-monotone and normalized ($\psi(\emptyset) = 0$). As a result, the set $\mathcal{S}^*$ obtained by the greedy maximization of $\psi(\mathcal{S})$ (or equivalently greedy minimization of $\phi(\mathcal{S})$) with a randomized version of Algorithm 4.1, that selects one node uniformly at random from the best N nodes at each iteration, is at least a 0.3-approximation of the optimal set $\mathcal{S}_{\mathrm{OPT}}$ [17]. To be precise, we have the following guarantees for $\mathcal{S}^*$ obtained from the randomized greedy algorithm

$$
\psi(\mathcal{S}_{\mathrm{OPT}}) \geq \psi(\mathcal{S}^*) \geq 0.3\psi(\mathcal{S}_{\mathrm{OPT}}) \tag{4.31}
$$

$$
\Rightarrow \phi(\mathcal{S}_{\mathrm{OPT}}) \leq \phi(\mathcal{S}^*) \leq 0.3\phi(\mathcal{S}_{\mathrm{OPT}}) + 0.7N. \tag{4.32}
$$

Although, guarantees for the deterministic version of the greedy algorithm are part of ongoing research, we observe empirically that its performance is competitive. Note that for the biorthogonal design when $\mathbf{G} \neq \mathbf{H}$, $\phi(\mathcal{S})$ is no longer supermodular, hence we cannot state guarantees on the performance of the greedy algorithm in this case. However, experiments show that the algorithm performs well in this case as well.

4.4.3 Multi-channel extension

In order to extend our formulation to M-channel filterbanks with analysis/synthesis filter pairs $\{\mathbf{H}_k, \mathbf{G}_k\}_{k=0,\dots,M-1}$ and sampling sets $\{\mathcal{S}_k\}_{k=0,\dots,M-1}$ ($\mathcal{S}_k \subset \{1, \dots, N\}$), one

can create the concatenated filters $\mathbf{H}, \mathbf{G} \in \mathbb{R}^{MN \times N}$ and the concatenated sampling set $\mathcal{S} \subset \{1, \ldots, MN\}$ in a manner similar to that of the two-channel case:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_{M-1} \end{bmatrix} \in \mathbb{R}^{MN \times N}, \ \mathbf{G} = \begin{bmatrix} \mathbf{G}_0 \\ \mathbf{G}_1 \\ \vdots \\ \mathbf{G}_{M-1} \end{bmatrix} \in \mathbb{R}^{MN \times N},$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_{M-1} \end{bmatrix} \in \mathbb{R}^{\sum_{k=1}^{}|\mathcal{S}_k|}, \ \mathbf{S} = \begin{bmatrix} \mathbf{S}_0 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_1 & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{S}_{M-1} \end{bmatrix} \in \{0, 1\}^{(\sum_{k=1}^{}|\mathcal{S}_k|) \times MN}. \quad (4.33)$$

Note that each $\mathcal{S}_k$ can be then be recovered from $\mathcal{S}$ as $\mathcal{S}_k = \{v - kN | v \in \mathcal{S}, kN + 1 \le v \le kN + N\}$. Further, in this case, we require predesigned filters such that $\mathbf{G}^T\mathbf{H} = M\mathbf{I}$, resulting in the objective

$$\phi(\mathcal{S}) = \left\| \left(1 - \frac{1}{M}\right) \sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \frac{1}{M} \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right\|_F^2, \quad (4.34)$$

where $\mathbf{g}_i = \mathbf{G}^T \delta_i$ and $\mathbf{h}_i = \mathbf{H}^T \delta_i$. The objective can be optimized under the constraint $|\mathcal{S}| = N$ using the same technique as that of the two-channel case by computing the change with respect to incremental node additions. In this case,

$$\phi(\mathcal{S} \cup \{v\}) = \left\| \left(\left(1 - \frac{1}{M}\right) \sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \frac{1}{M} \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right) + \mathbf{g}_v \mathbf{h}_v^T \right\|_F^2$$
$$= \phi(\mathcal{S}) + p_v(\mathcal{S}) + q_v, \quad (4.35)$$

where we defined

$$p_v(\mathcal{S}) = 2\text{Tr}\left[\mathbf{h}_v \mathbf{g}_v^T \left(\left(1 - \frac{1}{M}\right) \sum_{i \in \mathcal{S}} \mathbf{g}_i \mathbf{h}_i^T - \frac{1}{M} \sum_{j \in \mathcal{S}^c} \mathbf{g}_j \mathbf{h}_j^T \right) \right], \quad (4.36)$$

$$q_v = \|\mathbf{g}_v\|^2 \|\mathbf{h}_v\|^2. \quad (4.37)$$

To compute $p_v(\mathcal{S})$ for each $\mathcal{S}$, we note that

$$p_v(\emptyset) = 2\text{Tr}\left[\mathbf{h}_v \mathbf{g}_v^T(-\frac{1}{M}\mathbf{G}^T\mathbf{H})\right] = -2\langle \mathbf{g}_v, \mathbf{h}_v \rangle, \tag{4.38}$$

where we used $\mathbf{G}^T\mathbf{H} = M\mathbf{I}$. Further, for a node u, the update $p_v(\mathcal{S} \cup \{u\})$ can be computed as

$$p_v(\mathcal{S} \cup \{u\}) = 2\text{Tr}\left[\mathbf{h}_v \mathbf{g}_v^T\left(\left(1 - \frac{1}{M}\right)\sum_{i \in \mathcal{S}} \mathbf{g}_i\mathbf{h}_i^T - \frac{1}{M}\sum_{j \in \mathcal{S}^c} \mathbf{g}_j\mathbf{h}_j^T + \mathbf{g}_u\mathbf{h}_u^T\right)\right]$$
$$= p_v(\mathcal{S}) + 2\langle \mathbf{g}_v, \mathbf{g}_u \rangle \langle \mathbf{h}_v, \mathbf{h}_u \rangle. \tag{4.39}$$

Note that the computations required to initialize q_v and update $p_v(\mathcal{S})$ are identical to that of the two-channel case. Thus, one can use Algorithm 4.2 to obtain the optimal sampling set with one change – the vectors $\mathbf{p}(\mathcal{S}), \mathbf{q}$ are now MN-dimensional, with v^{th} elements $p_v(\mathcal{S})$ and q_v. In order to force the algorithm to produce a disjoint sampling scheme, one can once again disregard images of selected indices (*eg.*, choice of index $i \leq N$ rules out indices $i + kN, k = 1, \ldots, M - 1$ in subsequent iterations). In terms of complexity, computing $\mathbf{p}(\emptyset)$ and $\mathbf{q}$ require $2MN$ one-time graph filtering operations and each greedy iteration requires performing $4M$ filtering operations. Therefore, similar to the two-channel case, computing an approximately optimal sampling scheme for M-channel filterbanks requires $O(N)$ graph filtering operations, with a space complexity of $O(N)$. Note that the theoretical guarantees discussed in Section 4.4.2 hold in this case as well.

4.4.4 Experiments

In this section, we present simple experiments[3] to demonstrate the effectiveness of our critical sampling scheme for two-channel filterbanks. In our first experiment, we test its performance on three simple bipartite graphs (Figure 4.3) with filters obtained using the *Graph-QMF* design [45] (that approximates the Meyer kernel with a polynomial filter of chosen length 4), and *GraphBior(6,6)* [46]. We observe that the output of Algorithm 4.2, in most cases, matches exactly with that of the optimal sampling scheme for bipartite graphs, that is to downsample the filtered signal in each channel on either partition. However, due to the greedy nature of the algorithm, we occasionally observe that the

---

[3]Code available at https://github.com/aamiranis/cs_fb_arbitrary
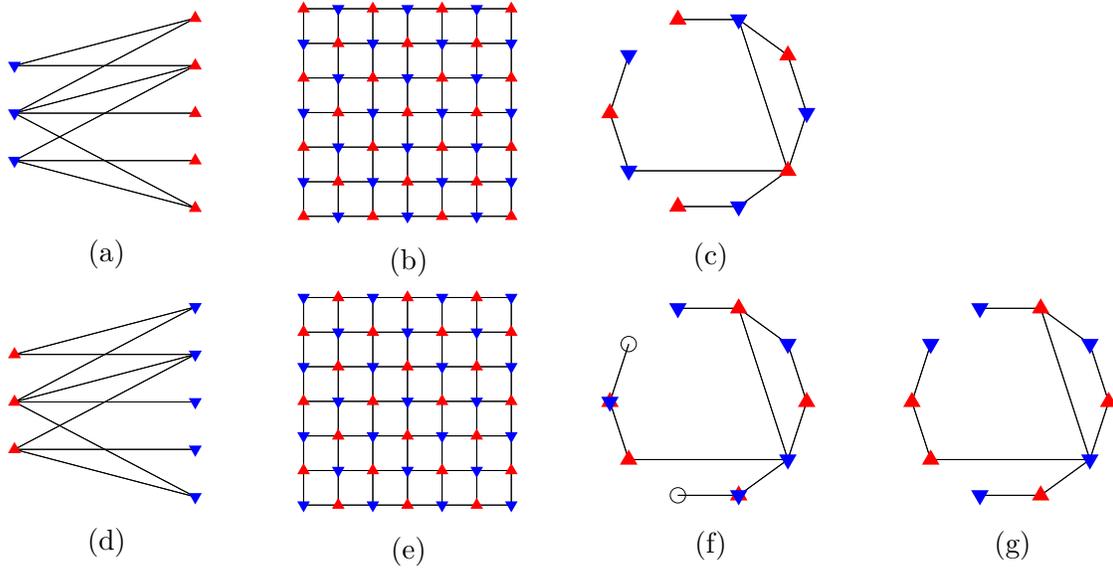
Figure 4.3: Sampling scheme obtained using Algorithm 4.2 for bipartite graphs with length 4 *Graph-QMF* filters: (a)-(c), and *GraphBior*(6,6) filters: (d)-(g). Red and blue colors indicate nodes in low-pass and high-pass channels. The sets are heuristically forced to be disjoint in (g).

obtained sampling scheme differs slightly from the optimal one, as seen in Figure 4.3f for the biorthogonal filters case. Notice that the sampling scheme is not disjoint, we heuristically force the algorithm to obtain disjoint sets in Figure 4.3g. The sampling pattern obtained in this case is nearly perfect with one wrongly assigned pair.

For our second experiment, we design a critically-sampled two-channel filterbank on the Minnesota road network graph using two configurations of analysis/synthesis filters: (i) *Graph-QMF* [45] with 8-degree polynomial approximations of the Meyer kernel, and (ii) *GraphBior*(6,6) [46]. The sampling scheme obtained for each of these configurations is plotted in Figures 4.4a and 4.4b. We observe that the sampling pattern for each channel colors nodes in a predominantly alternating fashion indicating a propensity towards bipartition. The response of the filterbank after determining the sampling set is plotted in Figures 4.4c and 4.4d for unit magnitude delta functions in the spectral domain. We observe that it is close to 1 for all frequencies. Since the transfer function is not diagonalizable in the GFT basis $\mathbf{U}$, there is an associated aliasing effect with the filterbank. We characterize this by plotting the maximum aliasing coefficient in terms of magnitude for each frequency component in Figures 4.4e and 4.4f. We also compare the recontruction performance (in terms of ratio of energies of error signal and original) of

(a)   (b)

(c)   (d)

(e)   (f)
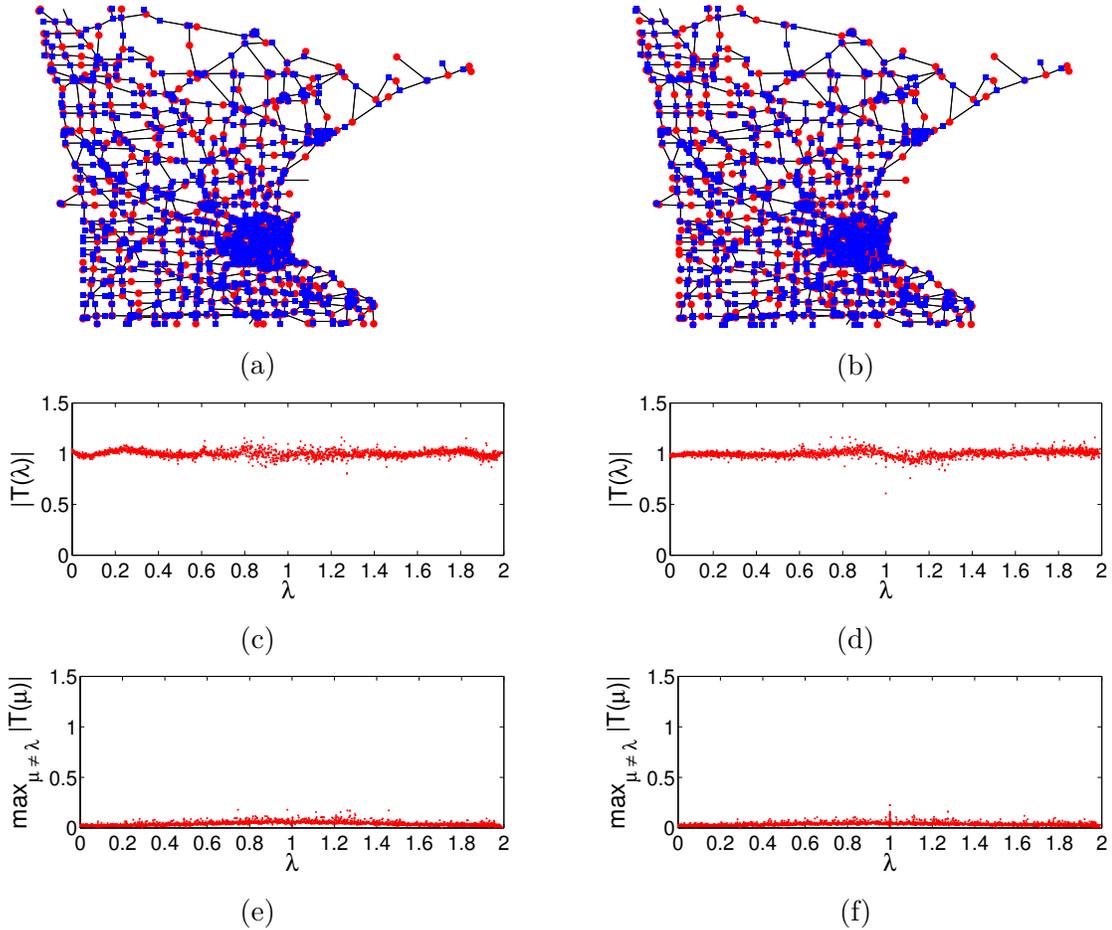
Figure 4.4: Performance of our critical sampling scheme (Algorithm 4.2) on the Minnesota road network graph. (a), (c) and (e) denote the sampling scheme obtained, spectral response, and maximum aliasing component for *Graph-QMF* design. (b), (d) and (f) illustrate corresponding results for *GraphBior*(6,6).

our method against one instance of a randomly selected sampling scheme, and a spectral approximation of MaxCut for 1000 random signals. The average squared relative errors along with the standard deviations are listed in Table 4.1. Observe that our method has superior performance.

In the final experiment, we compare the sampling scheme obtained from our method against random sampling schemes and the optimal sampling scheme that minimizes the objective: $\mathcal{S}_{\text{opt}} = \arg\min_{\mathcal{S}} \phi(\mathcal{S})$. Since determining the optimal scheme requires an exhaustive search, we limit ourselves to small graphs. Specifically, we test the methods on a ring graph with $N = 10$ nodes that also contains cross-links with a probability p.

Table 4.1: Recontruction error results for random signals on Minnesota road graph.

	Graph-QMF (poly 8)	*GraphBior*(6,6)
Random	0.4842 ± 0.0113	0.4629 ± 0.0108
MaxCut	0.1125 ± 0.0069	0.0972 ± 0.0061
Proposed	$\mathbf{0.0779 \pm 0.0049}$	$\mathbf{0.0664 \pm 0.0045}$

Table 4.2: Comparison of our method against an optimal sampling scheme obtained through exhaustive search. The experiment is performed for ring graphs of size $N = 10$ that have randomly added cross-links with probability p. Reconstruction error is averaged over 100 signals, and also 100 sampling schemes for the random sampling case.

Graph	Random	**Proposed**	Optimal
$p = 0.1$	0.410 ± 0.168	$\mathbf{0.100 \pm 0.098}$	0.100 ± 0.098
$p = 0.2$	0.375 ± 0.142	$\mathbf{0.199 \pm 0.093}$	0.190 ± 0.086
$p = 0.3$	0.375 ± 0.133	$\mathbf{0.204 \pm 0.091}$	0.205 ± 0.098
$p = 0.4$	0.364 ± 0.131	$\mathbf{0.223 \pm 0.102}$	0.183 ± 0.080
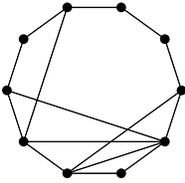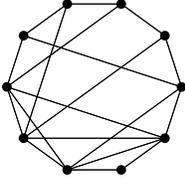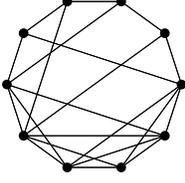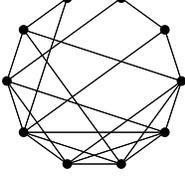
Note that when $p = 0$, such a graph is bipartite, and increasing p results in a deviation from the bipartite property. We also compare the mean error for reconstructing 100 random signals in each case. For the random sampling scheme, the mean is computed over all combinations of 100 randomly generated sets and 100 random signals. The results are illustrated in Table 4.2. We make two significant observations:

1. Despite making use of a greedy approximation for efficiency, our method's performance is comparable to the optimal sampling scheme.
2. As we deviate from the bipartite property, the structure of the graph limits the performance of the filterbank significantly.

The results in this experiment, along with the analyses in Sections 4.3.1 and 4.3.2, suggest that there is little wiggle-room for designing filterbanks on arbitrary graphs in terms of reconstruction error. Specifically, one needs to exploit special structures in the graph, or special sampling strategies (such as the one suggested in [67]) in order to achieve perfect reconstruction. In the next section, we explore a certain class of graphs where the special structure makes them suitable for the design of perfect reconstruction M-channel filterbanks.

4.5 Filterbanks on block-cyclic graphs

We saw in Section 4.3.1 that the special structure in bipartite graphs greatly simplifies the design of two-channel filterbanks with low-degree polynomials filters. In this section, we explore the design of filterbanks in another such class of graphs called M-block cyclic graphs [69]. These graphs are directed with M components, $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_{M-1}$, connected in a cyclic fashion, with no edges within each partition (for $M = 2$, it is a directed bipartite graph). The adjacency of this graph has a block-cyclic structure given by

$$
\mathbf{A} = \begin{bmatrix}
\mathbf{0} & \mathbf{A}_1 & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{A}_2 & \ldots & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \vdots & \mathbf{0} \\
\vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{A}_{M-1} \\
\mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0}
\end{bmatrix}, \tag{4.40}
$$

where each $\mathbf{A}_j$ has arbitrary but appropriate size to make $\mathbf{A}$ square. The adjacency
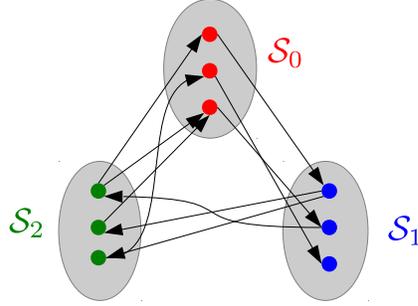
Figure 4.5: A 3-block cyclic graph.

matrix of an M-block cyclic graph has a special eigen-structure suitable for designing filterbanks, as described by the following theorem from [69]:

Theorem 4.1. *For any M-block cyclic graph, if $(\lambda, \mathbf{v})$ is an eigenpair of its adjacency matrix $\mathbf{A}$, then $(\omega_M \lambda, \boldsymbol{\Omega}_M \mathbf{v})$, $(\omega_M^2 \lambda, \boldsymbol{\Omega}_M^2 \mathbf{v})$, $\ldots (\omega_M^{M-1} \lambda, \boldsymbol{\Omega}_M^{M-1} \mathbf{v})$ are also eigenpairs of $\mathbf{A}$, where*

$$\omega_M = e^{-i\frac{2\pi}{M}}, \quad \boldsymbol{\Omega}_M = \text{diag}(\boldsymbol{\beta}), \quad \boldsymbol{\beta}(v) = \begin{cases} 1 & v \in \mathcal{S}_0 \\ \omega_M^1 & v \in \mathcal{S}_1 \\ \vdots \\ \omega_M^{(M-1)} & v \in \mathcal{S}_{M-1} \end{cases}. \quad (4.41)$$

The special form of the eigenstructure of M-block cylic graphs, i.e., the existence of M modulated copies of the eigenvalues and eigenvectors, naturally facilitates the design of M-channel filterbanks. Specifically, modulating an eigenvector with $\boldsymbol{\Omega}_M$ produces another eigenvector whose eigenvalue is also a modulated version of the original eigenvalue. In other words, modulating a signal results in a structured remapping (or folding) of its Fourier coefficients in the spectral domain – akin to the well-known spectral folding phenomenon in traditional sampling theory. Using the fact that downsampling-upsampling operations can be expressed as weighted sums of modulated versions of the original signal, one can succinctly express the perfect reconstruction conditions for an M-channel filterbank on M-block cyclic graphs in the spectral domain, as illustrated next.

4.5.1 Perfect reconstruction conditions for M-channel filterbanks on M-block cyclic graphs

Let us assume for now that the adjacency $\mathbf{A}$ of the M-block cyclic graph is diagonalizable and has distinct eigenvalues[4]. Therefore, we have $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$. The transfer function of an M-channel filterbank is given by

$$\mathbf{T} = \sum_{k=0}^{M-1} g_k(\mathbf{A})\mathbf{S}_k^T\mathbf{S}_k h_k(\mathbf{A}). \tag{4.42}$$

where $\mathbf{S}_k \in \{0,1\}^{|\mathcal{S}_k|\times N}$ is the downsampling operator for the block $\mathcal{S}_k$, and $\mathbf{S}_k^T$ is the corresponding upsampling operator. $\{h_k(\mathbf{A})\}_{k=0,...,M-1}$ and $\{g_k(\mathbf{A})\}_{k=0,...,M-1}$ are the set of analysis and synthesis filters respectively, these are polynomial functions of the adjacency matrix[5]. The transfer function can be expressed in the spectral domain as $\tilde{\mathbf{T}} = \mathbf{U}^{-1}\mathbf{T}\mathbf{U}$

$$\tilde{\mathbf{T}} = \sum_{k=0}^{M-1} g_k(\Lambda)\mathbf{U}^{-1}\mathbf{S}_k^T\mathbf{S}_k\mathbf{U}h_k(\Lambda). \tag{4.43}$$

Note that using properties of the discrete Fourier basis, the downsampling-upsampling operation in each channel can be expressed as

$$\mathbf{S}_k^T\mathbf{S}_k = \frac{1}{M}\sum_{j=0}^{M-1} \omega_M^{-jk}\mathbf{\Omega}_M^j. \tag{4.44}$$

Substituting this in the transfer function, we get

$$\tilde{\mathbf{T}} = \sum_{k=0}^{M-1} g_k(\Lambda)\mathbf{U}^{-1}\left(\frac{1}{M}\sum_{j=0}^{M-1} \omega_M^{-jk}\mathbf{\Omega}_M^j\right)\mathbf{U}h_k(\Lambda) \tag{4.45}$$

$$= \frac{1}{M}\left[\sum_{k=0}^{M-1} g_k(\Lambda)h_k(\Lambda) + \underbrace{\sum_{j=1}^{M-1}\sum_{k=0}^{M-1} \omega_M^{-jk}g_k(\Lambda)\mathbf{U}^{-1}\mathbf{\Omega}_M^j\mathbf{U}h_k(\mathbf{\Lambda})}_{\text{aliasing components}}\right]. \tag{4.46}$$

---

[4]With simple modifications, the analysis can be extended to the case of repeated eigenvalues. The frequency interpretation for the non-diagonalizable case is more complex since one might need to work with the Jordan form [64]. We leave this aspect for future work.

[5]It is easier to work with the adjacency in this case since M-block cyclic graphs are directed.

Using Theorem 4.1 in the aliasing terms, we obtain the following set of perfect reconstruction constraints on polynomials $\{g_k(\lambda), h_k(\lambda)\}$ for $|\lambda| \leq 1$ (assuming $\mathbf{A}$ is appropriately normalized):

$$\sum_{k=0}^{M-1} g_k(\lambda) h_k(\lambda) = M, \tag{4.47}$$

$$\sum_{k=0}^{M-1} \omega_M^{-jk} g_k(\lambda) h_k(\omega_M^{-j}\lambda) = 0, \quad \text{for all } 1 \leq j \leq M-1. \tag{4.48}$$

Similar perfect reconstruction conditions in the spectral domain have been proposed in [70], where filtered signals in each channel are downsampled on the same subset of nodes (i.e., $\mathcal{S}_0$) leading to the absence of the summation weights "ω_M^{-jk}" in (4.48). Ssampling each channel on a different subset of nodes is more amenable to the polyphase implementation carried out in [68] for the two-channel case.

The analyses in this subsection and in [70] do not provide insight about possible solutions for the perfect reconstruction conditions. The problem can be particularly challenging for the M-channel case since one needs to design M pairs of filters satisfying the constraint equations (4.47), (4.48). Further, since the spectrum of M-block cyclic graphs encompasses the entire complex unit disc and not just the complex unit circle (as in classical DSP), the choice of sub-bands for these filterbanks also remains unclear.

In order to gain some insights regarding these questions, we consider the problem under a simpler setting in the following subsection. Our solution works for M-block cylic graphs when M is a power of 2. Specifically, when $M = 2^L$, we propose to use an L-stage hierarchical tree-structured design for the filterbank that naturally extends the two-channel design for bipartite graphs. Our design also suggests a sub-band decomposition of the frequency domain (enclosed in the unit-disc), that in turn provides a frequency interpretation for the $M > 2$ case.

4.5.2 Tree-structured filterbank design

When $M = 2^L$, we propose a hierarchical tree-structured filterbank design consisting of L stages that can be succinctly explained in the following two points:

- In each stage, we group even-numbered and odd-numbered blocks together into two partitions and treat the original graph as a directed bipartite graph over which a 2-channel filterbank can be implemented. Existing solutions meant for undirected
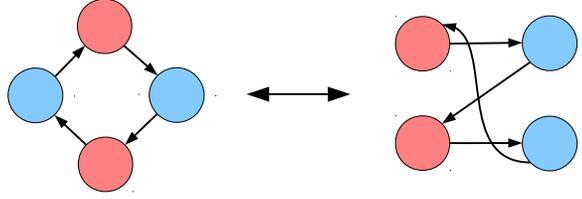
Figure 4.6: A 4-block cyclic graph visualized as a directed bipartite graph by grouping even and odd blocks.

bipartite graphs in [45, 46, 68] are extended to this directed setting via simple modifications.

- We create a new subgraph for each channel that is closely related to the original graph in the spectral domain. The 2-channel design from the previous stage is replicated on these subgraphs, leading to the hierarchical design.

This approach simplifies the design of the filterbank significantly since the problem is reduced to designing two prototype half-band filters (for the biorthogonal case) or one prototype half-band filter (for the orthogonal case), which can be repeatedly used in every stage. In contrast, for the general M-channel design, one would have to design M and $2M$ filters for the orthogonal and biorthogonal cases respectively, with appropriate responses for each sub-band.

Filterbank architecture

By definition, it is easy to see that block cyclic graphs with $M = 2^L$ blocks are directed bipartite, where the two partitions are given by the set of even and odd blocks respectively. This is easily seen, for example in $M = 4$, where one can observe the bipartite structure by applying a block permutation on $\mathbf{A}$ in order to group even and odd blocks (also illustrated graphically in Figure 4.6):

$$\begin{bmatrix} \mathbf{0} & \mathbf{A}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_3 \\ \mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \longrightarrow \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_3 \\ \mathbf{0} & \mathbf{A}_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

To make our notation concise, let us denote the set of all even blocks by $\mathcal{S}_e$ and the set of all odd blocks by $\mathcal{S}_o$ Therefore, we can design a two-channel filterbank for these

graphs, where the filtered signal in one channel is sampled on the even set $\mathcal{S}_e$ and on the odd set $\mathcal{S}_o$ in the other channel. After downsampling, we create new subgraphs for each channel with adjacencies $\mathbf{A}_e$ and $\mathbf{A}_o$ based on the idea proposed in [68]:

$$
\begin{aligned}
\mathbf{A}_e &= \mathbf{S}_e \mathbf{A}^2 \mathbf{S}_e^T, \\
\mathbf{A}_o &= \mathbf{S}_o \mathbf{A}^2 \mathbf{S}_o^T.
\end{aligned}
\tag{4.49}
$$

We now highlight some useful properties of these subgraphs that are crucial to our design.

Lemma 4.1. *If $\mathbf{A}$ is the adjacency matrix of an M-block cylic graph, then $\mathbf{A}_e$ and $\mathbf{A}_o$ are adjacency matrices of $M/2$-block cyclic graphs.*

Proof. Using the structure of $\mathbf{A}$ in (4.40), we have

$$
\mathbf{A}^2 = \begin{bmatrix}
\mathbf{0} & \mathbf{0} & \mathbf{A}_1\mathbf{A}_2 & \ldots & \mathbf{0} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{A}_{M-2}\mathbf{A}_{M-1} \\
\mathbf{A}_{M-1}\mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\
\mathbf{0} & \mathbf{A}_0\mathbf{A}_1 & \mathbf{0} & \ldots & \mathbf{0}
\end{bmatrix}.
\tag{4.50}
$$

Using the defintions of $\mathbf{S}_e$ and $\mathbf{S}_o$ that are meant to sample even and odd block indices respectively, we get

$$
\mathbf{A}_e = \mathbf{S}_e \mathbf{A}^2 \mathbf{S}_e^T = \begin{bmatrix}
\mathbf{0} & \mathbf{A}_1\mathbf{A}_2 & \mathbf{0} & \ldots & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{A}_3\mathbf{A}_4 & \ldots & \mathbf{0} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{A}_{M-3}\mathbf{A}_{M-2} \\
\mathbf{A}_{M-1}\mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0}
\end{bmatrix},
\tag{4.51}
$$

$$
\mathbf{A}_o = \mathbf{S}_o \mathbf{A}^2 \mathbf{S}_o^T = \begin{bmatrix}
\mathbf{0} & \mathbf{A}_2\mathbf{A}_3 & \mathbf{0} & \ldots & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{A}_4\mathbf{A}_5 & \ldots & \mathbf{0} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{A}_{M-2}\mathbf{A}_{M-1} \\
\mathbf{A}_0\mathbf{A}_1 & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0}
\end{bmatrix}.
\tag{4.52}
$$
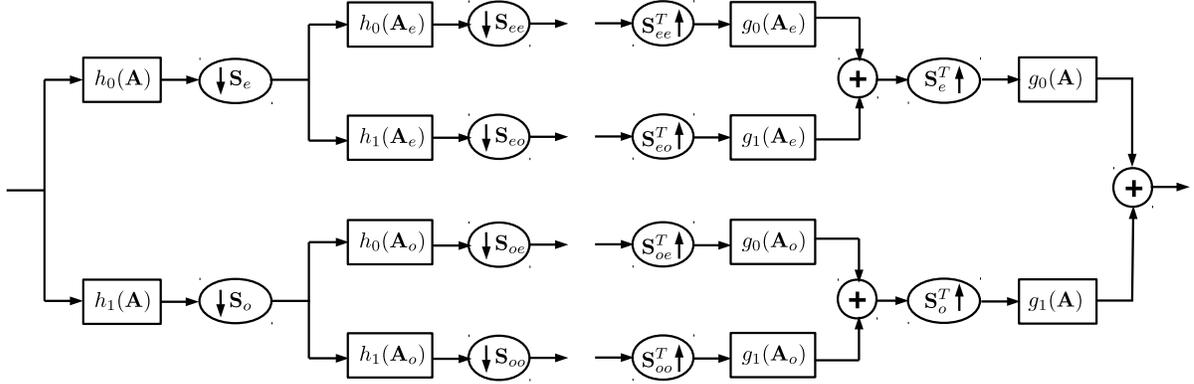
Figure 4.7: A tree-structured filterbank for block cyclic graphs consisting of two stages.

Clearly, $\mathbf{A}_e$ and $\mathbf{A}_o$ are adjacency matrices of $M/2$-block cyclic graphs. ∎

Lemma 4.1 allows us to reuse the same two-channel design on the new subgraphs in the next stage. This reduces the design complexity since we are required to design only one perfect reconstruction two-channel filterbank for a directed bipartite graph. By hierarchically implementing this two-channel design on a 2^L-block cylic graph, we end up with a perfect reconstruction tree-structured filterbank of depth L that effectively has 2^L channels. Note that in every stage, we keep splitting the blocks into even and odd in order to sample the two channels until the last stage where we are left with only one block for each channel. An example of the tree-structured filterbank containing two stages (i.e., up to depth $L = 2$) is shown in Figure 4.7. Since every stage involves splitting the even blocks from the odd blocks, the labels of the blocks in the last stage with respect to the original parent graph can be obtained by the bit-reversal permutation sequence (similar to decimation-in-frequency for the radix-2 FFT algorithm). For example, after two stages of decimation in a 4-block cylic graph, the sampling sets for each channel comprise of one of the four blocks – these are $\mathcal{S}_{ee} = \mathcal{S}_0$, $\mathcal{S}_{eo} = \mathcal{S}_2$, $\mathcal{S}_{oe} = \mathcal{S}_1$, and $\mathcal{S}_{oo} = \mathcal{S}_3$.

The next Lemma gives a relation between the spectrum of the two new subgraphs and the parent graph:

Lemma 4.2. *If $\{\lambda, \mathbf{v}\}$ is an eigen-pair of $\mathbf{A}$, then $\{\lambda^2, \mathbf{S}_e\mathbf{v}\}$ is an eigenpair of $\mathbf{A}_e$ and $\{\lambda^2, \mathbf{S}_o\mathbf{v}\}$ is an eigenpair of $\mathbf{A}_o$.*

Proof. Since $\mathbf{A}$ is bipartite with partitions $\mathcal{S}_e$ and $\mathcal{S}_o$, the following relations are true by definition:

$$\mathbf{S}_e\mathbf{A}\mathbf{S}_e^T = \mathbf{S}_o\mathbf{A}\mathbf{S}_o^T = \mathbf{0}, \quad \mathbf{S}_e\mathbf{A}^2\mathbf{S}_o^T = \mathbf{S}_o\mathbf{A}^2\mathbf{S}_e^T = \mathbf{0}. \tag{4.53}$$

Let $\{\lambda, \mathbf{v}\}$ be an eigenpair of $\mathbf{A}$, then we have $\mathbf{A}^2\mathbf{v} = \lambda^2\mathbf{v}$. Further, let us define a permutation matrix $\mathbf{P} = \begin{bmatrix} \mathbf{S}_e \\ \mathbf{S}_o \end{bmatrix}$. Note that $\mathbf{P}^T\mathbf{P} = \mathbf{P}\mathbf{P}^T = \mathbf{I}$, and therefore, we have

$$\mathbf{P}\mathbf{A}^2(\mathbf{P}^T\mathbf{P})\mathbf{v} = \lambda^2\mathbf{P}\mathbf{v}$$

$$\Rightarrow \left(\begin{bmatrix} \mathbf{S}_e \\ \mathbf{S}_o \end{bmatrix} \mathbf{A}^2 \begin{bmatrix} \mathbf{S}_e^T & \mathbf{S}_o^T \end{bmatrix} \right) \begin{bmatrix} \mathbf{S}_e \\ \mathbf{S}_o \end{bmatrix} \mathbf{v} = \lambda^2 \begin{bmatrix} \mathbf{S}_e \\ \mathbf{S}_o \end{bmatrix} \mathbf{v}$$

$$\Rightarrow \begin{bmatrix} \mathbf{S}_e\mathbf{A}^2\mathbf{S}_e^T & \overbrace{\mathbf{S}_e\mathbf{A}^2\mathbf{S}_o^T}^{\mathbf{0}} \\ \underbrace{\mathbf{S}_o\mathbf{A}^2\mathbf{S}_e^T}_{\mathbf{0}} & \mathbf{S}_o\mathbf{A}^2\mathbf{S}_o^T \end{bmatrix} \begin{bmatrix} \mathbf{S}_e\mathbf{v} \\ \mathbf{S}_o\mathbf{v} \end{bmatrix} = \lambda^2 \begin{bmatrix} \mathbf{S}_e\mathbf{v} \\ \mathbf{S}_o\mathbf{v} \end{bmatrix} \qquad \text{(using (4.53))}$$

$$\Rightarrow \begin{bmatrix} \mathbf{A}_e & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_o \end{bmatrix} \begin{bmatrix} \mathbf{S}_e\mathbf{v} \\ \mathbf{S}_o\mathbf{v} \end{bmatrix} = \lambda^2 \begin{bmatrix} \mathbf{S}_e\mathbf{v} \\ \mathbf{S}_o\mathbf{v} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \mathbf{A}_e\mathbf{S}_e\mathbf{v} \\ \mathbf{A}_o\mathbf{S}_o\mathbf{v} \end{bmatrix} = \lambda^2 \begin{bmatrix} \mathbf{S}_e\mathbf{v} \\ \mathbf{S}_o\mathbf{v} \end{bmatrix}.$$

Therefore, $\mathbf{A}_e(\mathbf{S}_e\mathbf{v}) = \lambda^2(\mathbf{S}_e\mathbf{v})$ and $\mathbf{A}_o(\mathbf{S}_o\mathbf{v}) = \lambda^2(\mathbf{S}_o\mathbf{v})$, and this completes our proof. ∎

Lemma 4.2 has a significant implication: the eigenvectors of $\mathbf{A}_e$ and $\mathbf{A}_o$ are obtained by downsampling the eigenvectors of $\mathbf{A}$. Intuitively, this relationship is quite important for multi-resolution analysis, since it allows us to "zoom in" on a particular portion of the GFT matrix of the parent graph in order to attain finer resolution.

Although the definition of these subgraphs seems arbitrary at first sight, there are many reasons for why we would like to work with this specific choice. Lemmas 4.1 and 4.2 allow us to conclude that these subgraphs not only preserve the block cyclic structure in the vertex domain, but also maintain a close association with the parent graph in the spectral domain. Yet another justification for using these definitions is that they arise naturally while working with polyphase analysis and structures of two-channel filterbanks over bipartite graphs [68]. However, the most important reason in our opinion is that these definitions allow us to apply the noble identities stated in [69] in order to provide a spectral interpretation for the filterbank as explained next.
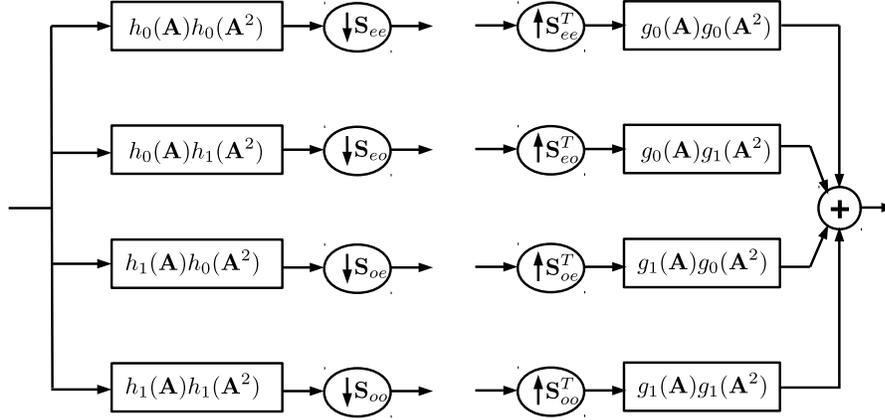
Figure 4.8: Two-stage tree-structured filterbank simplified using the noble identities for bipartite graphs.

Filterbank spectral interpretation

We now analyze the spectral behavior of the proposed tree-structured design. The key idea here is to use the noble identities [69] for the filters in the second stage. Specifically, for any polynomial filter $p(\mathbf{A})$ on a bipartite graph with $\mathcal{S}_e$ and $\mathcal{S}_e$ denoting its odd and even partitions, one has

$$p(\mathbf{A}_e)\mathbf{S}_e = p(\mathbf{S}_e\mathbf{A}^2\mathbf{S}_e^T)\mathbf{S}_e = \mathbf{S}_ep(\mathbf{A}^2), \tag{4.54}$$

$$p(\mathbf{A}_o)\mathbf{S}_o = p(\mathbf{S}_o\mathbf{A}^2\mathbf{S}_o^T)\mathbf{S}_o = \mathbf{S}_op(\mathbf{A}^2), \tag{4.55}$$

$$\mathbf{S}_e^Tp(\mathbf{A}_e) = \mathbf{S}_e^Tp(\mathbf{S}_e\mathbf{A}^2\mathbf{S}_e^T) = p(\mathbf{A}^2)\mathbf{S}_e^T, \tag{4.56}$$

$$\mathbf{S}_o^Tp(\mathbf{A}_o) = \mathbf{S}_o^Tp(\mathbf{S}_o\mathbf{A}^2\mathbf{S}_o^T) = p(\mathbf{A}^2)\mathbf{S}_o^T. \tag{4.57}$$

Equations (4.54) and (4.55) are the first noble identities, whereas (4.56) and (4.57) are the second noble identities for bipartite graphs. Note how these identities serve as a justification for the choice of subgraphs in each channel. We reduce the depth of the filterbank using these identities on the filters of the second stage to obtain a single stage filterbank with filters as indicated in Figure 4.8. This simplification approach can easily be extended to a tree-structured design of depth L by applying the noble identities in a cascaded fashion starting from the last stage and moving backwards until only stage

remains. The resultant analysis and synthesis filters of any particular channel can then be written as

$$h_k(\mathbf{A}) = h_{i_1}(\mathbf{A})h_{i_2}(\mathbf{A}^2)h_{i_3}(\mathbf{A}^4)\dots h_{i_L}(\mathbf{A}^{2^{L-1}}) = \prod_{l=1}^{L} h_{i_l}(\mathbf{A}^{2^{l-1}}), \;\; i_l \in \{0,1\}, \qquad (4.58)$$

$$g_k(\mathbf{A}) = g_{i_1}(\mathbf{A})g_{i_2}(\mathbf{A}^2)g_{i_3}(\mathbf{A}^4)\dots g_{i_L}(\mathbf{A}^{2^{L-1}}) = \prod_{l=1}^{L} g_{i_l}(\mathbf{A}^{2^{l-1}}), \;\; i_l \in \{0,1\}. \qquad (4.59)$$

It is interesting to observe the behavior of a tree-structured filterbank constructed with ideal filters. For a two-channel filterbank on a directed bipartite graph, the ideal lowpass filter $h_0(\mathbf{A})$ has a half-disc response (in the spectral domain of $\mathbf{A}$) with a passband on the positive real-axis. Similarly, the ideal highpass filter $h_1(\mathbf{A})$ has a passband on the negative real-axis. These ideal kernels are plotted in Figure 4.9a. The responses of $h_0(\mathbf{A}^2)$, $h_1(\mathbf{A}^2)$ are given in Figure 4.9b and the spectral responses of the four channels of a two-stage tree-structured filterbank are given in Figure 4.9c.

Polynomial filter design

The perfect reconstruction conditions for a two-channel filterbank on bipartite graphs can be obtained by plugging $M = 2$ in (4.47) and (4.48). Specifically, we need to design filters $h_0(\lambda)$, $h_1(\lambda)$, $g_0(\lambda)$ and $g_1(\lambda)$ that satisfy the following conditions for $|\lambda| \leq 1$ (λ is complex):

$$h_0(\lambda)g_0(\lambda) + h_1(\lambda)g_1(\lambda) = 2, \qquad (4.60)$$
$$h_0(-\lambda)g_0(\lambda) - h_1(-\lambda)g_1(\lambda) = 0. \qquad (4.61)$$

When the graph is undirected, λ is restricted to the real axis. Orthogonal solutions for these equations cannot be obtained using polynomial filters, thus an approximate solution is proposed in [45]. However, perfect reconstruction using polynomial filters is possible for the biorthogonal design, and a solution based on the CDF maximally-flat filter design approach is presented in [46].

Since M-block cyclic graphs are directed, our problem requires designing polynomial filter responses in the complex unit disc. In the following, we focus on the biorthogonal design since it allows perfect reconstruction. Similar to the approach in [46], we choose

(a) Ideal lowpass and highpass kernel responses $h_0(\mathbf{A})$ and $h_1(\mathbf{A})$.

(b) Second stage filter responses $h_0(\mathbf{A}^2)$ and $h_1(\mathbf{A}^2)$.
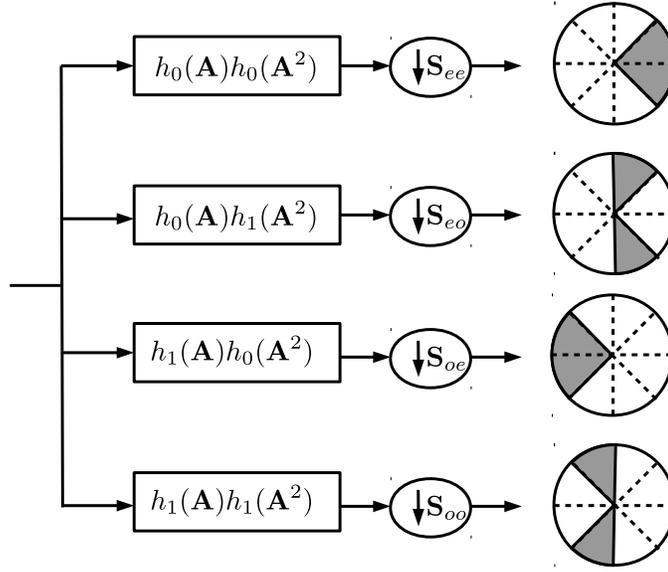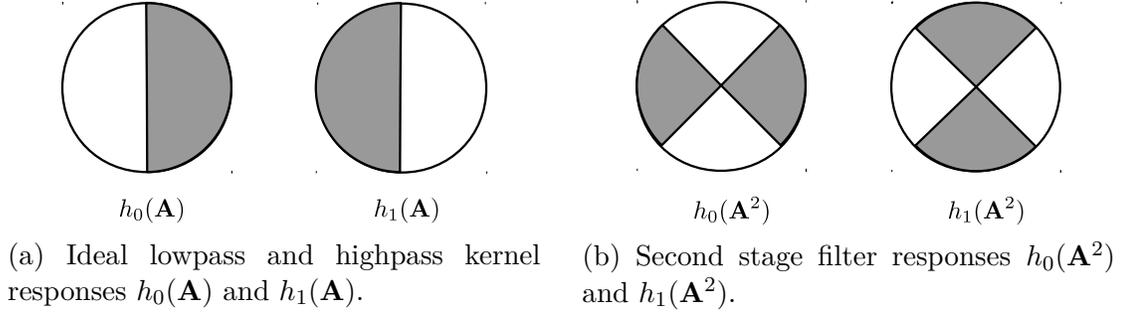
(c) Responses of the four channels of a two-stage tree-structured filterbank. The rightmost and left-most pie slices correspond to the lowest and highest frequency sub-bands respectively.

Figure 4.9: Spectral characterization of a two-stage tree-structured filterbank with ideal filters. Gray-shaded areas indicate passbands.

$h_1(\lambda) = g_0(-\lambda)$ and $g_1(\lambda) = h_0(-\lambda)$ so that (4.61) is automatically satisfied leaving us with the design criterion:

$$h_0(\lambda)g_0(\lambda) + h_0(-\lambda)g_0(-\lambda) = 2, \quad \forall\{\lambda : |\lambda| \leq 1\}. \tag{4.62}$$

We then define $p(\lambda) = h_0(\lambda)g_0(\lambda)$, so that (4.62) can be rewritten as

$$p(\lambda) + p(-\lambda) = 2, \quad \forall\{\lambda : |\lambda| \leq 1\}. \tag{4.63}$$

Since $p(\lambda)$ is the product of two lowpass kernels, it is also a lowpass kernel. Therefore, our objective is to design a polynomial half-band kernel $p(\lambda)$ that satisfies the complementarity condition (4.63), followed by its spectral factorization to obtain $h_0(\lambda)$ and $g_0(\lambda)$. It is immediately clear from (4.63) that $p(\lambda)$ must have the following the form:

$$p(\lambda) = 1 + \sum_{k=0}^{D} c_k \lambda^{2k+1}. \tag{4.64}$$

We now describe an approach for obtaining $p(\lambda)$ by generalizing the maximally-flat design presented in [46] for undirected bipartite graphs. The key idea is to force $p(\lambda)$ to have K_1 roots at $\lambda = -1$. However, since we are working in the complex unit disc, this does not guarantee a flat response as we move on the imaginary axis away from the real axis (i.e. the top and bottom of the complex unit disc). In order to have a better transition band, we also place K_2 roots at $-1+i$ and $-1-i$ (note that they must be equal in number to have a polynomial with real coefficients). Therefore, the design approach involves finding a polynomial $r(\lambda) = \sum_{m=0}^{R} r_m \lambda^m$ such that

$$(\lambda + 1)^{K_1} (\lambda + 1 + i)^{K_2} (\lambda + 1 - i)^{K_2} r(\lambda) = p(\lambda) \tag{4.65}$$

$$\Rightarrow (\lambda + 1)^{K_1} (\lambda^2 + 2\lambda + 2)^{K_2} \left(\sum_{m=0}^{R} r_m \lambda^m \right) = 1 + \sum_{k=0}^{D} c_k \lambda^{2k+1}. \tag{4.66}$$

Comparing highest powers on both sides of (4.66), we have $R + K_1 + 2K_2 = 2D + 1$. The left hand side of (4.66) has $R+1$ unknowns and the right hand side has $D+1$ constraints. Therefore, to have a unique polynomial $p(\lambda)$ that satisfies (4.63), we should have the same number of constraints as unknowns, which implies $R = D = (K_1 + 2K_2 - 1)$. We can thus rewrite (4.66) as

$$(\lambda + 1)^{K_1} (\lambda^2 + 2\lambda + 2)^{K_2} \left(\sum_{m=0}^{K_1 + 2K_2 - 1} r_m \lambda^m \right) = 1 + \sum_{k=0}^{K_1 + 2K_2 - 1} c_k \lambda^{2k+1}. \tag{4.67}$$

The $K_1 + 2K_2$ unknowns $\{r_m\}$ can be found by solving a linear system of $K_1 + 2K_2$ equations. Once $r(\lambda)$ is found, we obtain $p(\lambda)$ using (4.65). Note that we can also place roots at other locations to shape the response accordingly and modify (4.66) to obtain the appropriate $r(\lambda)$ and the corresponding $p(\lambda)$.
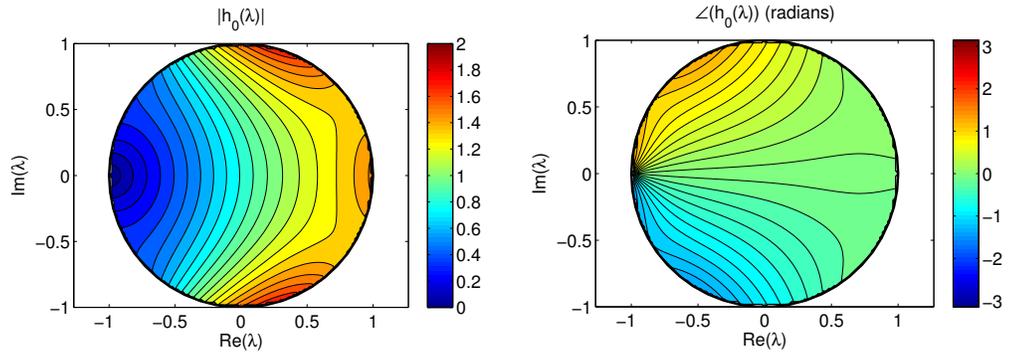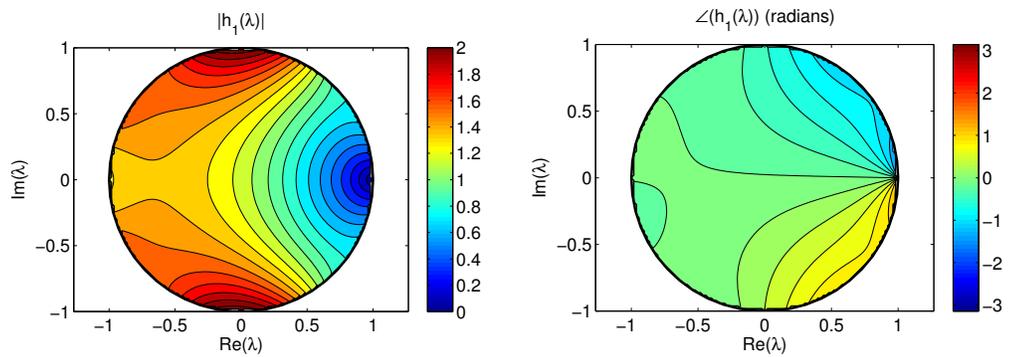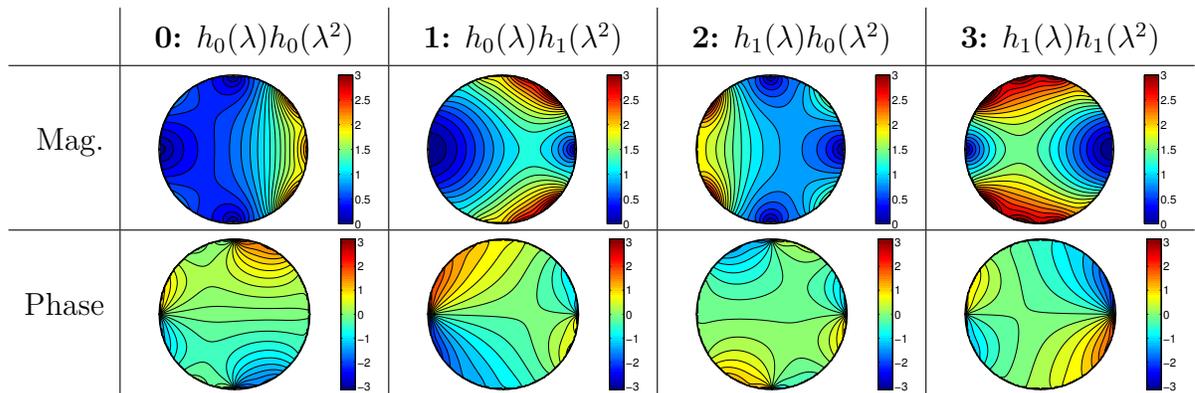
Figure 4.10: Spectral responses (magnitude and phase) of polynomial filters designed using the maximally-flat approach with $K1 = K2 = 2$: (a) lowpass kernel $h_0(\lambda)$ (length = 6), (b) highpass kernel $h_1(\lambda)$ (length = 7). (c) Channel responses of a two-stage tree-structured filterbank built using the polynomial kernels $h_0(\lambda)$ and $h_1(\lambda)$.

Once we have $p(\lambda)$, we factor it to obtain the kernels $h_0(\lambda)$ and $g_0(\lambda)$ using the approach presented in [46]. This approach produces maximally-balanced kernels by splitting roots of $p(\lambda)$ between $h_0(\lambda)$ and $g_0(\lambda)$ as evenly as possible while ensuring that the filterbank is as close to orthogonal as possible. Examples of a lowpass filter kernel $h_0(\lambda)$ and $h_1(\lambda)(= g_0(-\lambda))$ designed using this approach are plotted in Figures 4.10a and 4.10b. The spectral responses of the four channels of a two-stage tree-structured polynomial filterbank built using these kernels is illustrated in Figure 4.10c.

4.5.3 Preliminary experiment

In this section, we perform a simple filtering experiment[6] on a synthetically generated 4-block cyclic graph to evaluate our tree-structured filterbank design. The graph consists of 100 nodes and is generated in the following way:

1. We first create 25 disjoint directed cycles, each consisting of 4 nodes and oriented in the same direction with edges of weight 1. Note that these cycles collectively form a disconnected 4-block cylic graph of 100 nodes.

2. Next, we add directed edges of weight 1 randomly with probability $p = 0.2$ while preserving the block-cyclic property (i.e., by only connecting adjacent blocks with edges of consistent directionality).

3. Finally, we normalize the edge weights to ensure that the rows of the adjacency $\mathbf{A}$ sum to 1 (random-walk normalization). Such a normalization makes $\mathbf{A}$ a stochastic matrix with a cyclic structure, that is commonly used to model periodic Finite State Machines (FSMs) and Markov Decision Processes (MDPs) [37, 58].

Before performing the experiment, we ensure that the generated graph instance is connected. In order to evaluate its performance, we consider a two-stage tree-structured filterbank and compute the filtered channel outputs for a given input signal that is zero on all the blocks except one where it has *i.i.d.* values chosen uniformly from [0,1]. The graph chosen for our experiment is illustrated in Figure 4.11a, along with its eigenvalues in Figure 4.11b, and the input signal in Figure 4.11c.

We first perform the filtering experiment using ideal filters (illustrated in Figure 4.9c), followed by resorting to the polynomial kernels $h_0(\lambda)$ and $h_1(\lambda)$ obtained from the maximally-flat design for $K1 = K2 = 2$ (these filters have length 6 and 7 respectively, the spectral response of each channel in this case is illustrated in Figure 4.10c).

---

[6]Code available at https://github.com/aamiranis/tree_structured_fb
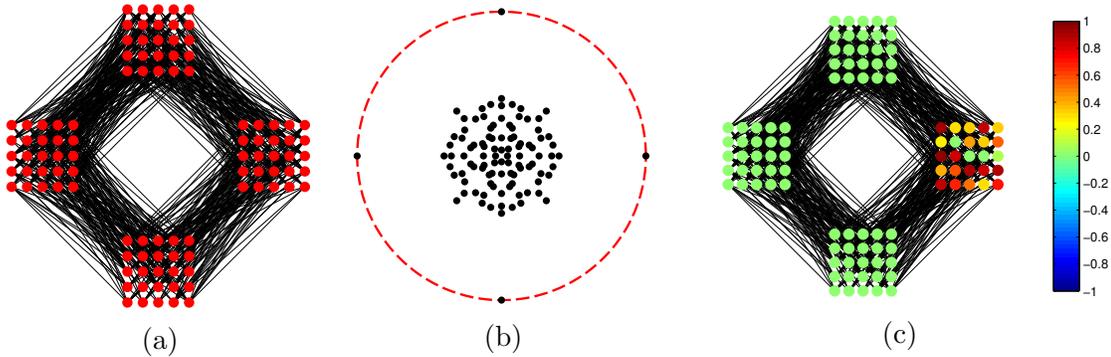
Figure 4.11: (a) A 4-block cyclic graph considered in the filtering experiment (the edges are oriented in counter-clockwise fashion). (b) The spectrum of its adjacency in the complex unit disc. (c) An input signal on the graph used for filtering experiments with the two-stage tree-structured filterbank.

The output of each channel for both cases is illustrated in Figure 4.12. We observe that the filtered output for the first channel (filtered by $h_0(\mathbf{A})h_0(\mathbf{A}^2)$) in both cases has relatively little variation across the blocks, thus confirming our intuition that it corresponds to the lowest frequency sub-band. Similarly, the output of the third channel (filtered by $h_0(\mathbf{A})h_0(\mathbf{A}^2)$) has the highest variation with significant sign changes across blocks, thus making it the highest frequency sub-band. The other two channels have moderate variation and therefore correspond to the intermediate frequency sub-bands.

4.6 Summary and future work

In this chapter, we considered the sampling problem in the context of designing graph wavelet filterbanks. This is fundamentally different from the earlier problem of sampling bandlimited graph signals since one is required to choose a sampling scheme over multiple channels of the filterbank while satisfying useful properties such as perfect reconstruction, critical sampling and polynomial filter responses.

We began by spelling out design criteria for two-channel polynomial filterbanks that are critically sampled, and satisfy perfect reconstruction. We showed that it is not possible in general to attain perfect reconstruction with low-degree polynomial graph filters using any critical sampling scheme unless the graphs have a special eigen-structure. Specifically, downsampling-upsampling or modulating signals over such graphs should lead to a spectral folding phenomenon. Bipartite graphs, for example, are particularly suited for designing perfect reconstruction two-channel filterbanks.

Channel	Ideal kernel	Polynomial kernel
0: $h_0(\lambda)h_0(\lambda^2)$		
1: $h_0(\lambda)h_1(\lambda^2)$		
2: $h_1(\lambda)h_0(\lambda^2)$		
3: $h_1(\lambda)h_1(\lambda^2)$		
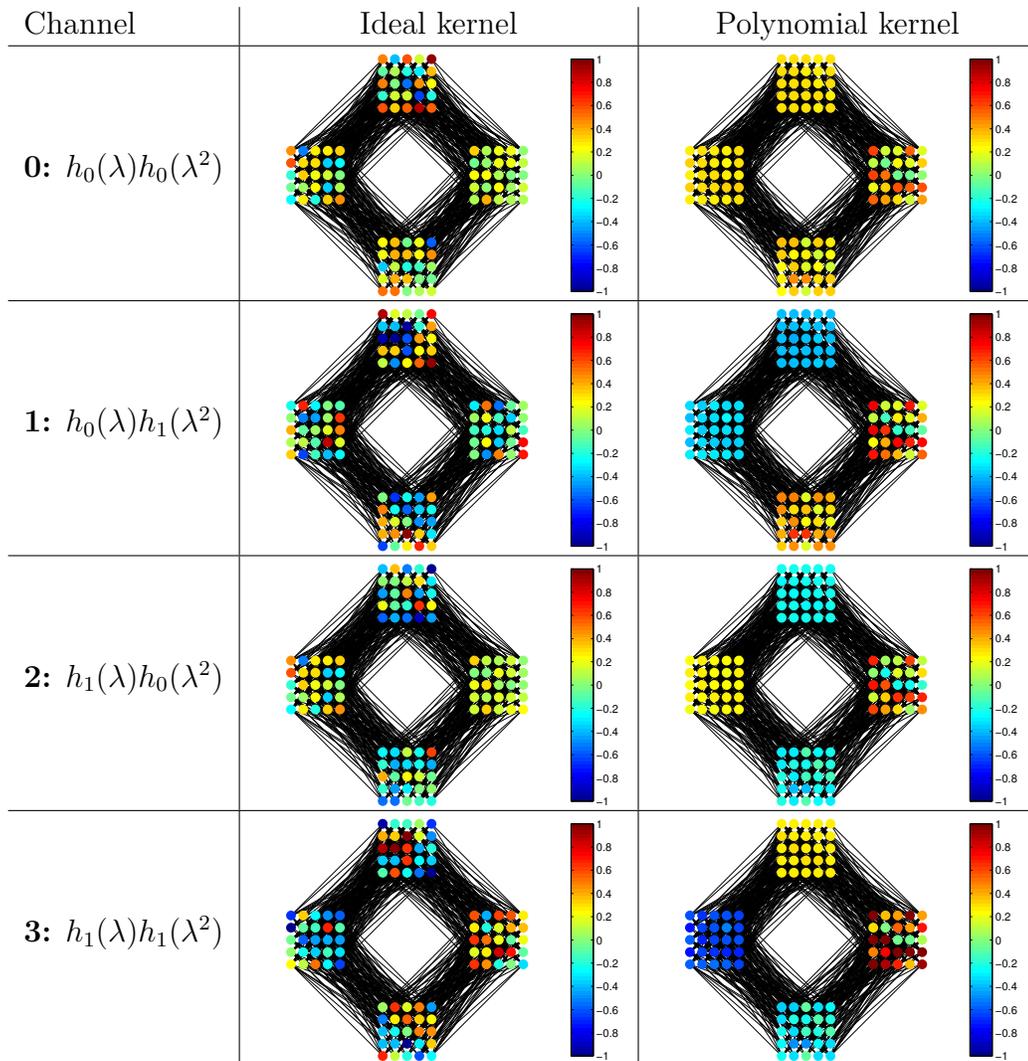
Figure 4.12: Output obtained in each channel of a two-stage tree-structured filterbank after filtering the input signal from Figure 4.11c using ideal filters (Figure 4.9c) and polynomial filters (Figure 4.10c).

We then shifted our focus to the design of critically-sampled near-perfect reconstruction wavelet filterbanks on arbitrary graphs. This problem has two intertwined aspects – designing polynomial filters and designing a critical sampling scheme. Our formulation decouples the two design problems and focuses only on choosing the best possible sampling scheme. Specifically, given a predesigned set of analysis/synthesis filters, our algorithm efficiently chooses the best sampling set for each channel in order to minimize a bound on the overall reconstruction error associated with the filterbank. Experiments

show that the sampling scheme produced by our method outperforms existing heuristics, however, the reconstruction error is indeed significantly limited by the graph structure. Searching for graphs and sampling schemes that are amenable to perfect reconstruction filterbanks would be an interesting direction of future research. An immediate example of such graphs are Kronecker graphs where the Kronecker product structure of the adjacency matrix leads to a regular eigen-structure akin to bipartite graphs. Further, the spectral domain sampling scheme presented in [67] allows for the design of perfect reconstruction filterbanks on any arbitrary graph, at the cost of increased complexity of sampling.

Finally, we turned our attention to the design of M-channel perfect reconstruction filterbanks on M-block cylic graphs. In this case, we simplified the problem by assuming that M is a power of 2 and proposed a tree-structured design where the original graph is decomposed hierarchically into smaller graphs over which one can implement a two-channel filterbank. This formulation significantly reduces the design complexity, since the problem boils down to designing and reusing only one two-channel filterbank for a directed bipartite graph. We proposed a maximally-flat polynomial design in this case that extends the approach presented in [46] for undirected bipartite graphs. Our design produces sub-bands that have a simple spectral interpretation, which we validate in simple filtering experiments. For future work, we would like to extend the design to cases when M is not a power of 2. One possibility is to design p-channel perfect reconstruction filterbanks, where p is a small prime number, and use it to hierarchically construct a larger filterbank when M can be factored as a product of small primes. This approach is similar to the mixed-radix implementations of FFT in traditional DSP.

Chapter 5

Sampling Theory Perspective of Semi-supervised Learning

In this chapter, we explore the semi-supervised learning problem from a graph sampling theory perspective. Graph-based methods have been shown to be quite effective for this problem because they provide an easy means to exploit the underlying geometry of the dataset. These methods involve the construction of a distance-based similarity graph from the feature vectors, whose vertices represent the data points and edge weights capture the similarity between them. The key assumption here is that class labels vary *smoothly* over the graph, i.e., there is little variation between labels corresponding to vertices connected by high-weight edges.

There are numerous ways of quantitatively imposing smoothness constraints over label functions defined on vertices of a similarity graph. Most graph-based semi-supervised classification algorithms incorporate one of these criteria as a penalty against the fitting error in a regularization problem, or as a constraint term while minimizing the fitting error in an optimization problem. Examples of commonly used measures of smoothness include the graph Laplacian regularizer $\mathbf{f}^T \mathbf{L} \mathbf{f}$ [83, 78], the iterated graph Laplacian regularizers $\mathbf{f}^T \mathbf{L}^m \mathbf{f}$ [81], *etc.*, and many algorithms involve minimizing these functions while ensuring that the label function on the vertices $\mathbf{f}$ satisfies the known set of labels. On the other hand, a spectral theory based classification algorithm restricts $\mathbf{f}$ to be spanned by the first few eigenvectors of the graph Laplacian [10, 11], that are known to form a representation basis for smooth functions on the graph. Similarly, a more recent approach, derived from sampling theory considers class indicator signals as smooth signals over the similarity graph. This assumption is incorporated via bandwidth in the graph Fourier domain. The classification algorithm then involves estimating a label function that minimizes prediction error on the known set under a bandwidth constraint and can be carried out without explicit eigendecomposition of the Laplacian, as illustrated in our recent work [50, 29]. In each of the examples, the criterion enforces

---

This chapter is based on our work in [5, 6].

smoothness of the labels over the graph – a lower value of the regularizer $\mathbf{f}^T \mathbf{L} \mathbf{f}$, a smaller number of leading eigenvectors and a smaller bandwidth of $\mathbf{f}$ imply that the labels vary less across similar vertices.

The focus of this chapter is to provide a formal justification for using bandwidth as a regularizer, thereby providing a sampling theory perspective of graph-based learning methods. An interpretation of this smoothness measure would help complete our theoretical understanding of graph-based semi-supervised classification approaches and strengthen their link with the semi-supervised smoothness assumption and its variants. Under a generic statistical model of the data and a graph construction scheme, we provide a geometric interpretation of the bandwidth (estimated via spectral proxies) of class indicator signals in the asymptotic limit of infinite data points. We show that this quantity is closely connected to the supremum of the data distribution at the class boundaries. This result helps us justify sampling theory based learning:

1. A lower bandwidth of class indicator signals indicates that the class boundary passes through regions of low data density.

2. Given enough labeled data, bandlimited reconstruction of the class indicator learns a decision boundary that respects the labels and over which the maximum density of the data points is as low as possible, similar to other graph-based methods.

3. And finally, we also show that sampling theory based learning attains the theoretical label complexity, i.e., the minimum number of labels required for perfectly predicting the unknown labels from the known ones.

In summary, our results reinforce the smoothness assumption on class indicator signals in graph-based learning methods. From previous analyses of spectral clustering we observe that asymptotically, there is a strong link between the value of a cut and the bandwidth of its associated indicator signal. Thus, the geometric properties desired of "minimal cuts" in clustering translate to those of "minimal bandwidth" indicator signals for classification in the presence of labels.

The rest of this chapter is organized as follows: In Section 5.1, we introduce the statistical models of the data, the graph construction scheme and relevant concepts from sampling theory, particularly bandlimited interpolation. Section 5.2 reviews existing work in the literature and their relation to our work. In Section 5.3 we state and discuss our main results (without proofs), followed by their numerical validation in 5.4. We

provide a brief discussion along with directions in Section 5.5. Finally, the proofs of our results are laid out in Section 5.6.

5.1 Preliminaries

5.1.1 Data models

The separable model

In this model, we assume that the dataset consists of a pool of n random, d-dimensional feature vectors $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ drawn *independently* from some probability density function $p(\mathbf{x})$ supported on $\mathbb{R}^d$ (this is assumed for simplicity, the analysis can be extended to arbitrary manifolds $\mathcal{M} \subset \mathbb{R}^d$, but would more technically involved). To simplify our analysis, we also assume that $p(\mathbf{x})$ is bounded from above, Lipschitz continuous and twice differentiable. We assume that a smooth hypersurface ∂S, with radius of curvature lower bounded by a constant τ, splits $\mathbb{R}^d$ into two disjoint classes S and S^c, with indicator functions $1_S(\mathbf{x}) : \mathbb{R}^d \to \{0, 1\}$ and $1_{S^c}(\mathbf{x}) : \mathbb{R}^d \to \{0, 1\}$. This is illustrated in Figure 5.1a. Thus, the n-dimensional class indicator signal for class S is given by $\mathbf{1}_S \in \{0, 1\}^n$ such that $\mathbf{1}_S(i) = 1_S(\mathbf{X}_i)$, i.e., the ith entry of $\mathbf{1}_S$ is 1 if $\mathbf{X}_i \in S$ and 0 otherwise.

The nonseparable model

In this model, we assume that each class has its own conditional distribution supported on $\mathbb{R}^d$ (that may or may not overlap with other distributions of other classes). The data set consists of a pool of n random and independent d-dimensional feature
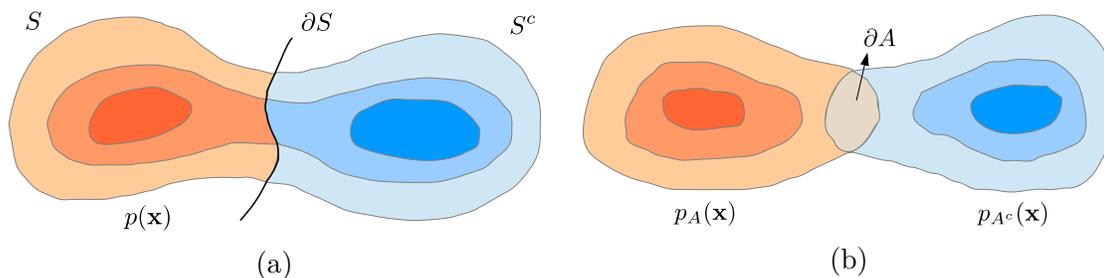
Figure 5.1: Statistical models of data considered in this work: (a) the separable model, (b) the nonseparable model.

86

vectors $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ drawn independently from any of the distributions $p_i(\mathbf{x})$ with probabilities α_i, such that $\sum_i \alpha_i = 1$. For our analysis, we consider a class denoted by an index A with selection probability α_A, class conditional distribution $p_A(\mathbf{x})$ and an n-dimensional indicator vector $\mathbf{1}_A$ whose i^{th} component takes value 1 if $\mathbf{X}_i$ is drawn from class A. This model is illustrated in Figure 5.1b. Further, we denote by $\alpha_{A^c} = 1 - \alpha_A$ the probability that a point does not belong to A and by $p_{A^c}(\mathbf{x}) = \sum_{i \neq A} \alpha_i p_i(\mathbf{x})/\alpha_{A^c}$ the density of all such points. The marginal distribution of data points is then given by the mixture density

$$p(\mathbf{x}) = \alpha_A p_A(\mathbf{x}) + \alpha_{A^c} p_{A^c}(\mathbf{x}). \tag{5.1}$$

Once again, to simplify our analysis, we assume that all distributions are Lipschitz continuous, bounded from above and twice differentiable in $\mathbb{R}^d$. Next, we introduce the notion of a "boundary" for classes in the nonseparable model as follows: for class A, we define its *overlap* region ∂A as

$$\partial A = \{\mathbf{x} \in \mathbb{R}^d \mid p_A(\mathbf{x})p_{A^c}(\mathbf{x}) > 0\}. \tag{5.2}$$

Intuitively, ∂A can be considered as the region of ambiguity, where both points belonging and not belonging to A co-exist. In other words, ∂A can be thought of as a "boundary" that separates the region where points can only belong to A from the region where points can never belong to A. Since class indicator signals on graphs will change values only within the overlap region, one would expect that the indicators will be smoother if there are fewer data points within this region. We shall show later that this is indeed the case, both theoretically and experimentally. Note that the definition of the boundary is not very meaningful for class conditional distributions with decaying tails, such as the Gaussian, since the boundary encompasses the entire feature space. However, in such cases, one can approximate the boundary with appropriate thresholds in the definition and this approximation can also be formalized for distributions with exponentially decaying tails.

5.1.2 Graph model

Using the n feature vectors, we construct an undirected distance-based similarity graph where nodes represent the data points and edge weights are proportional to their similarity, given by the Gaussian kernel:

$$w_{ij} = K_{\sigma^2}(\mathbf{X}_i, \mathbf{X}_j) = \frac{1}{(2\pi\sigma^2)^{d/2}} e^{-\|\mathbf{X}_i - \mathbf{X}_j\|^2 / 2\sigma^2}, \tag{5.3}$$

where σ is the variance (bandwidth) of the Gaussian kernel. Further, we assume $w_{ii} = 0$, i.e., the graph does not have self-loops. The adjacency matrix of the graph $\mathbf{W}$ is an $n \times n$ symmetric matrix with elements w_{ij}, while the degree matrix is a diagonal matrix with elements $\mathbf{D}_{ii} = \sum_j w_{ij}$. We define the graph Laplacian as $\mathbf{L} = \frac{1}{n}(\mathbf{D} - \mathbf{W})$. Normalization by n ensures that the norm of $\mathbf{L}$ is stochastically bounded as n grows. Since the graph is undirected, $\mathbf{L}$ is a symmetric matrix with non-negative eigenvalues $0 \le \lambda_1 \le \cdots \le \lambda_n$ and an orthogonal set of corresponding eigenvectors $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$. It is known that for a larger eigenvalue λ, the corresponding eigenvector $\mathbf{u}$ exhibits greater variation when plotted over the nodes of the graph [66].

5.1.3 Estimating bandwidth

Recall that the bandwidth $\omega(\mathbf{f})$ of any signal $\mathbf{f}$ on the graph as the largest eigenvalue for which the projection of the signal on the corresponding eigenvector is non-zero, i.e.,

$$\omega(\mathbf{f}) = \max_i \{\lambda_i \mid \mathbf{u}_i^T \mathbf{f} > 0\}. \tag{5.4}$$

Ideally, computing the bandwidth $\omega(\mathbf{f})$ of a graph signal $\mathbf{f}$ requires computing the eigenvectors $\{\mathbf{u}_i\}$ and the corresponding projections $\tilde{\mathbf{f}}_i = \mathbf{u}_i^T \mathbf{f}$. However, analyzing the convergence of these coefficients is technically challenging. Therefore, we resort to *Graph Spectral Proxies* introduced in Section 3.3 in order to estimate the bandwidth. Since, in this chapter, we deal with a symmetric $\mathbf{L}$, we slightly modify the definition for simplicity:

$$\omega_m(\mathbf{f}) = \left(\frac{\mathbf{f}^T \mathbf{L}^m \mathbf{f}}{\mathbf{f}^T \mathbf{f}} \right)^{1/m}, \tag{5.5}$$

where $\omega_m(\mathbf{f})$ is the mth-order spectral proxy. Recall that the bandwidth estimates satisfy the property: for all m_1, m_2 such that $0 < m_1 < m_2$, $\omega_{m_1}(\mathbf{f}) \leq \omega_{m_2}(\mathbf{f}) \leq \omega(\mathbf{f})$. Therefore, we have:

$$\forall \mathbf{f}, \ \omega(\mathbf{f}) = \lim_{m \to \infty} \omega_m(\mathbf{f}), \tag{5.6}$$

Analyzing the convergence of $\omega_m(\mathbf{1}_S)$ and $\omega_m(\mathbf{1}_A)$ as $n \to \infty$, $\sigma \to 0$ and $m \to \infty$ constitutes the main subject for the rest of this chapter. Specifically, we relate these quantities to the underlying data distribution $p(\mathbf{x})$ and class boundaries (the hypersurface ∂S in the separable case and the overlap region ∂A in the nonseparable case).

Note that the limit in (5.6) holds in a point-wise sense. This means that analyzing the convergence of the bandwidth estimates $\omega_m(\mathbf{1}_S)$ and $\omega_m(\mathbf{1}_A)$ as $n \to \infty$ and then applying the limit $m \to \infty$ gives only an idea about the convergence of actual bandwidths $\omega(\mathbf{1}_S)$ and $\omega(\mathbf{1}_A)$ as $n \to \infty$. Specifically, it does not imply convergence of $\omega(\mathbf{1}_S)$ and $\omega(\mathbf{1}_A)$ to the same values as $\omega_m(\mathbf{1}_S)$ and $\omega_m(\mathbf{1}_A)$, since the limits are not interchangeable unless (5.6) holds in a uniform sense. However, based on our experiments and results on label complexity, we believe that our intuition is accurate, i.e., the convergence results hold for the actual bandwidths, not only their estimates. We leave the analysis of this intricacy for future work.

5.1.4 Bandlimited interpolation for classification

Bandwidth plays an important role in the spectral approach for semi-supervised learning. In this approach, one finds a label assignment by minimizing the error over the known set, while ensuring that the resulting class indicator vector is bandlimited over the similarity graph, i.e,

$$\text{Minimize} \quad \|\mathbf{f}(L) - \mathbf{y}(L)\|^2 \quad \text{subject to} \quad \omega(\mathbf{f}) < \omega_L, \tag{5.7}$$

where L denotes the set of known labels, $\mathbf{y}$ denotes the true class labels and $\mathbf{f}(L)$ and $\mathbf{y}(L)$ denote the values of $\mathbf{f}$ and $\mathbf{y}$ on the set L respectively. ω_L restricts the hypothesis space by constraining it to a set of bandlimited signals which is equivalent to enforcing smoothness of the labels over the graph. Therefore, it is important to understand its connection to the geometry of the dataset. A good choice for ω_L is the cutoff frequency associated with the labeled set that can be estimated using results in Section 3.4.

Note that the bandwidth-based approach for semi-supervised learning differs from the Fourier eigenvector approach suggested in [10, 11] since it can be implemented without

89

explicitly computing the eigenvectors of $\mathbf{L}$. The method is based on iteratively and alternately projecting onto convex sets and can be implemented in an efficient manner via graph filtering operations [50].

Note that if the original indicators $\mathbf{1}_S$ (or $\mathbf{1}_A$) are bandlimited with respect to the labeled set, i.e., $\omega(\mathbf{1}_S) < \omega_L$ (or $\omega(\mathbf{1}_A) < \omega_L$), then the estimate $\mathbf{f}_{\mathrm{LS}}$ in (5.7) is guaranteed to be equal to $\mathbf{1}_S$ (or $\mathbf{1}_A$) as a consequence of the sampling theorem. Moreover, in this case, $\mathbf{1}_S$ and $\mathbf{1}_A$ can also be perfectly estimated by the solution of the following "dual" problem:

$$\mathbf{f}_{\min} = \arg \min_{\mathbf{f}} \ \omega(\mathbf{f}) \quad \text{s.t.} \quad \mathbf{f}(L) = \mathbf{1}_S(L) \ \ (\text{or } \mathbf{f}(L) = \mathbf{1}_A(L)). \tag{5.8}$$

These facts leads to the following insight regarding bandlimited interpolation for classification:

Observation 5.1. *If $\omega(\mathbf{1}_S) < \omega_L$ and $\omega(\mathbf{1}_A) < \omega_L$, then*

1. $\mathbf{1}_S$ and $\mathbf{1}_A$ can be perfectly recovered using either (5.7) and (5.8).

2. $\mathbf{1}_S$ and $\mathbf{1}_A$ are guaranteed to have minimum bandwidth among all indicator vectors satisfying the label constraints.

The observations above have significant implications: Given enough and appropriately chosen labeled data, bandlimited interpolation effectively recovers an indicator vector with minimum bandwidth, that respects the label constraints. Note that by labeling enough data appropriately, we mean to ensure that the cut-off frequency ω_L of the labeled set is greater than the bandwidths $\omega(\mathbf{1}_S)$ and $\omega(\mathbf{1}_A)$ of the indicator functions of interest. If this condition is not satisfied, both observations break down, i.e., the solutions of (5.7) and (5.8) would be different and serve only as approximations for $\mathbf{1}_S$ and $\mathbf{1}_A$. Moreover, the minimum bandwidth signal $\mathbf{f}_{\min}$ satisfying the label constraints, would differ from $\mathbf{1}_S$ and $\mathbf{1}_A$ and may not even be an indicator vector. To help ensure that the condition is satisfied, one can use the efficient algorithm in Section 3.5. We note that in practice, (5.7) can be solved via efficient iterative techniques [50].

5.2 Related work and connections

Existing graph-based semi-supervised learning and spectral clustering methods have been justified by analyzing the convergence of graph-based smoothness measures (such as the graph cut and the Laplacian regularizer) for various graph construction schemes in

Table 5.1: Related convergence results in the literature under different data models and graph construction schemes. All models assume that the distributions are smooth (at least twice-differentiable). Further, the graph Laplacian is defined as $\mathbf{L} = \frac{1}{n}(\mathbf{D} - \mathbf{W})$ in all cases. [42] also studies convergence of graph cuts for weighted k-nearest neighbor and r-neighborhood graphs which we do not include for brevity.

Work	Data model	Graph model	Quantity	Convergence regime	Limit (within constant scaling factor)
Narayanan et al [51]	$p(\mathbf{x})$ supported on manifold $\mathcal{M} \subset \mathbb{R}^d$, separated into S and S^c by smooth hypersurface ∂S	Normalized Gaussian weights $w'_{ij} = \frac{w_{ij}}{\sqrt{d_i d_j}}$	$\frac{1}{n\sigma} \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S$	$n \to \infty,\ \sigma \to 0$	$\int_{\partial S} p(\mathbf{s}) d\mathbf{s}$
Maier et al [42]	$p(\mathbf{x})$ supported on $\mathcal{M} \subset \mathbb{R}^d$, separated into S and S^c by hyperplane ∂S	r-neighborhood, unweighted	$\frac{1}{nr^{d+1}} \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S$	$n \to \infty,\ r \to 0$	$\int_{\partial S} p^2(\mathbf{s}) d\mathbf{s}$
		k-nn, unweighted, $t = (k/n)^{1/d}$	$\frac{1}{nt^{d+1}} \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S$	$n \to \infty,\ t \to 0$	$\int_{\partial S} p^{1-1/d}(\mathbf{s}) d\mathbf{s}$
		fully-connected, Gaussian weights	$\frac{1}{n\sigma} \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S$	$n \to \infty,\ \sigma \to 0$	$\int_{\partial S} p^2(\mathbf{s}) d\mathbf{s}$
Bousquet et al [15], Hein [34]	$p(\mathbf{x})$ and $f(\mathbf{x})$ supported on $\mathbb{R}^d$	fully-connected, weights $w_{ij} = \frac{1}{n\sigma^d} K\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}\right)$, where $K(.)$ is a smooth decaying kernel	$\frac{1}{n\sigma^2} \mathbf{f}^T \mathbf{L} \mathbf{f}$	$n \to \infty,\ \sigma \to 0$	$\int \|\nabla f(\mathbf{x})\|^2 p^2(\mathbf{x}) dx$
Zhou et al [81]	Uniformly distributed on d-dim. submanifold $\mathcal{M}$	fully-connected, Gaussian weights	$\frac{1}{n\sigma^m} \mathbf{f}^T \mathbf{L}^m \mathbf{f}$	$n \to \infty,\ \sigma \to 0$	$\int f(\mathbf{x}) \Delta^m f(\mathbf{x}) dx$
El Alaoui et al [2]	$p(\mathbf{x})$ supported on $[0,1]^d$	fully-connected, weights $w_{ij} = K\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\sigma}\right)$, where $K(.)$ is a smooth decaying kernel	$\frac{1}{n^2 \sigma^{p+d}} J_p(\mathbf{f})$	$n \to \infty,\ \sigma \to 0$	$\int \|\nabla f(\mathbf{x})\|^p p^2(\mathbf{x}) dx$
Current work	$p(\mathbf{x})$ supported on $\mathbb{R}^d$, separated into S and S^c by smooth hypersurface ∂S	fully-connected, Gaussian weights	$\omega_m(\mathbf{1}_S)$	$n \to \infty,\ \sigma \to 0,$ $m \to \infty$	$\sup_{\mathbf{s} \in \partial S} p(\mathbf{s})$
	Drawn from $p_A(\mathbf{x})$ and $p_{A^c}(\mathbf{x})$ supported on $\mathbb{R}^d$ with probabilities α_A and α_{A^c}	fully-connected, Gaussian weights	$\omega_m(\mathbf{1}_A)$	$n \to \infty,\ \sigma \to 0,$ $m \to \infty$	$\sup_{\mathbf{x} \in \partial A} p(\mathbf{x})$

two different settings – *classification* and *regression.* The classification setting assumes that labels indicate class memberships and are discrete, typically with $1/0$ values. Note that both the separable and nonseparable data models considered in our work are in the classification setting. On the other hand, in the regression setting, one allows the class label signal $\mathbf{f}$ to be smooth and continuous with soft values, i.e, $\mathbf{f} \in \mathbb{R}^n$ and later applies some thresholding mechanism to infer class memberships. For example, in the two class problem, one can assign $+1$ and -1 to the two classes and threshold $\mathbf{f}$ at 0. Convergence analysis of smoothness measures in this setting requires different scaling conditions than the classification setting, and leads to fundamentally different limiting quantities that require differentiability of the label functions. A summary of convergence results in the literature for both settings is presented in Table 5.1. Although these results do not focus on analyzing the bandwidth of class indicator signals, the proof techniques used in this paper are inspired by some of these works. We review them in this section and discuss their connections to our work.

5.2.1   Classification setting

Prior work under this setting assumes the separable data model where the feature space is partitioned by smooth decision boundaries into different classes. When $m = 1$, the bandwidth estimate $\omega_m(\mathbf{1}_S)$ for the separable model in our work reduces (within a scaling factor) to the empirical graph cut for the partitions S and S^c of the feature space, i.e.,

$$Cut(S, S^c) = \sum_{\mathbf{X}_i \in S, \mathbf{X}_j \in S^c} w_{ij} = \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S \ . \tag{5.9}$$

Convergence of this quantity has been studied before in the context of spectral clustering, where one tries to minimize it across the two partitions of the nodes. It has been shown in [42] that the cut formed by a hyperplane ∂S in $\mathbb{R}^d$ converges with some scaling under the rate conditions $\sigma \to 0$ and $n\sigma^{d+1} \to \infty$ as

$$\frac{1}{n\sigma} \mathbf{1}_S^T \mathbf{L} \mathbf{1}_S \xrightarrow{p.} \frac{1}{\sqrt{2\pi}} \int_{\partial S} p^2(\mathbf{s}) d\mathbf{s}, \tag{5.10}$$

where $d\mathbf{s}$ ranges over all $(d-1)$-dimensional volume elements tangent to the hyperplane ∂S, and $p.$ denotes convergence in probability. The analysis is also extended to other graph construction schemes such as the k-nearest neighbor graph and the r-neighborhood graph, both weighted and unweighted. The condition $\sigma \to 0$ in (5.10) is required to have

a clear and well-defined limit on the right hand side. We borrow this convergence regime in our work, since it allows a succinct interpretation of the bandwidth of class indicator signals. Intuitively, it enforces sparsity in the similarity matrix $\mathbf{W}$ by shrinking the neighborhood volume as the number of data points increases. As a result, one can ensure that the graph remains sparse even as the number of points goes to infinity. The analysis is also extended to other graph construction schemes such as the k-nearest neighbor graph and the r-neighborhood graph, both weighted and unweighted. The condition $\sigma \to 0$ in (5.10) is required to have a clear and well-defined limit on the right hand side. We borrow this convergence regime in our work, since it allows a succinct interpretation of the bandwidth of class indicator signals. Intuitively, it enforces sparsity in the similarity matrix $\mathbf{W}$ by shrinking the neighborhood volume as the number of data points increases. As a result, one can ensure that the graph remains sparse even as the number of points goes to infinity. A similar result for a similarity graph constructed with normalized weights $w'_{ij} = w_{ij}/\sqrt{d_i d_j}$ was shown earlier for an arbitrary hypersurface ∂S in [51], where d_i denotes the degree of node i. In this case, under the condition $n \to \infty$, and for a vanishing sequence $\{\sigma_n\}$ that satisfies $\sigma_n > 1/\left(n^{\frac{1}{d+1}}\right)$, one has

$$\frac{1}{n\sigma_n}\mathbf{1}_S^T\mathbf{L}'\mathbf{1}_S \xrightarrow{p.} \frac{1}{\sqrt{2\pi}}\int_{\partial S} p(\mathbf{s})d\mathbf{s}, \tag{5.11}$$

where $\mathbf{L}'$ denotes the Laplacian with normalized weights. Normalization leads to a different weighting factor on the right hand side than (5.10). The results in [51, 42] aim to provide an interpretation for spectral clustering – up to some scaling, the empirical cut value converges to a weighted volume of the boundary. Thus, spectral clustering is a means of performing low density separation on a finite sample drawn from a distribution in feature space.

Although these works provide inspiration for the proof techniques used for analyzing the separable model in this paper, they cannot be directly used in the convergence analysis of $\omega_m(\mathbf{1}_S)$ for $m > 1$, which is the main focus of our paper. Additionally, our work is the first to propose and analyze the nonseparable model in the classification setting, i.e., convergence results for $\omega_m(\mathbf{1}_A)$.

5.2.2 Regression setting

To predict the labels of unknown samples in the regression setting, one generally minimizes the graph Laplacian regularizer $\mathbf{f}^T\mathbf{L}\mathbf{f}$ subject to the known label constraints [83]:

$$\min_{\mathbf{f}} \; \mathbf{f}^T\mathbf{L}\mathbf{f} \text{ such that } \mathbf{f}(L) = \mathbf{y}(L). \tag{5.12}$$

One particular convergence result in this setting assumes that n data points are drawn *i.i.d.* from $p(\mathbf{x})$ and are labeled by sampling a smooth function $f(\mathbf{x})$ on $\mathbb{R}^d$. Here, the graph Laplacian regularizer $\mathbf{f}^T\mathbf{L}\mathbf{f}$ can be shown to converge in the asymptotic limit under the conditions $\sigma \to 0$ and $n\sigma^d \to \infty$ as in [15, 34]:

$$\frac{1}{n\sigma^2}\mathbf{f}^T\mathbf{L}\mathbf{f} \xrightarrow{p.} C \int_{\mathbb{R}^d} \|\nabla f(\mathbf{x})\|^2 p^2(\mathbf{x})d\mathbf{x}, \tag{5.13}$$

where for each n, $\mathbf{f}$ is the n-dimensional label vector representing the values of $f(\mathbf{x})$ at the n sample points, ∇ is the gradient operator and C is a constant factor independent of n and σ. The right hand side of the result above is a weighted Dirichlet energy functional that penalizes variation in the label function weighted by the data distribution. Similar to the justification of spectral clustering, this result justifies the formulation in (5.12) for semi-supervised classification: given label constraints, the predicted label function must vary little in regions of high density. The work of [34, 33] also generalizes the result for arbitrary kernel functions used in defining graph weights, and data distributions defined over arbitrary manifolds in $\mathbb{R}^d$. Similar convergence results have also been derived for the higher-order Laplacian regularizer $\mathbf{f}^T\mathbf{L}^m\mathbf{f}$ obtained from uniformly distributed data [81]. In this case, it was shown that for data points obtained from a uniform distribution on a d-dimensional submanifold $\mathcal{M} \subset \mathbb{R}^N$ such that $\text{Vol}(\mathcal{M}) = 1$ and $2m$-differentiable functions $f(\mathbf{x})$, one has as $n \to \infty$, $\sigma \to 0$,

$$\frac{1}{n\sigma_n^m}\mathbf{f}^T\mathbf{L}^m\mathbf{f} \xrightarrow{p.} C \int_{\mathcal{M}} f(\mathbf{x})\Delta^m f(\mathbf{x})d\mathbf{x}, \tag{5.14}$$

where Δ is the Laplace operator and $\sigma_n = n^{-1/(2d+4+\alpha)}$ is a vanishing sequence with $\alpha > 0$. Extensions for non-uniform probability distributions $p(\mathbf{x})$ over the manifold can be obtained using the weighted Laplace-Beltrami operator [12, 82]. More recently, an ℓ_p-based Laplacian regularization has been proposed for imposing smoothness constraints in semi-supervised learning problems [2]. This is similar to higher-order regularization but is defined as $J_p(\mathbf{f}) = \sum_{i,j \in E} w_{ij}^p |f_i - f_j|^p$, where $w_{ij} = K(\|\mathbf{X}_i - \mathbf{X}_j\|/\sigma)$ and $K(.)$

is a smoothly decaying Kernel function. It has been shown for a bounded density $p(\mathbf{x})$ defined on $[0, 1]^d$ that for every $p \geq 2$, as $n \to \infty$, $\sigma \to 0$,

$$\frac{1}{n^2 \sigma^{p+d}} J_p(\mathbf{f}) \xrightarrow{p.} C \int_{[0,1]^d} \|\nabla f(\mathbf{x})\|^p p^2(\mathbf{x}) d\mathbf{x}. \tag{5.15}$$

Although our work also uses higher powers of $\mathbf{L}$ in the expressions for $\omega_m(\mathbf{1}_S)$ and $\omega_m(\mathbf{1}_A)$, we cannot use the convergence results in (5.14) and (5.15) since they are only applicable for smooth functions (i.e., differentiable upto certain order) on $\mathbb{R}^d$. Specifically, these results cannot be applied for the bandwidth of discrete $0/1$ class indicator functions.

To summarize, the results in the literature mostly pertain to convergence analysis of variants of the graph cut or the graph Laplacian regularizer for different models of data and graph construction schemes, and do not provide insight into the convergence of bandwidths of discrete $0/1$ class indicator signals. In contrast, we analyze bandwidth expressions involving these class indicator signals and higher powers of $\mathbf{L}$, and for the first time, extend it to a nonseparable data model. As opposed to other smoothness measures considered earlier, analyzing the bandwidth allows us to interpret graph-based semi-supervised learning using the sampling theorem [4] and provide a quantitative insight into label complexity based on data geometry.

5.3 Main results and discussion

5.3.1 Interpretation of bandwidth and bandlimited reconstruction

We first show that under certain conditions, the bandwidth estimates of class indicator signals, over the distance-based similarity graph described earlier, converge to quantities that are functions of the underlying distribution and the class boundary for both data models. This convergence is achieved under the following asymptotic regime:

1. Increasing size of dataset: $n \to \infty$.
2. Shrinking neighborhood volume: $\sigma \to 0$.
3. Improving bandwidth estimates: $m \to \infty$.

Note that an increasing size of the dataset (Condition 1) is required for the stochastic convergence of the bandwidth estimate. Condition 2 ensures that the limiting values

95

are concise and have a simple interpretation in terms of the data geometry. Intuitively, Condition 2 ensures that as the number of data points increases, one looks at a smaller neighborhood around each data point, as a result, the degree of each node in the graph does not blow up. Finally, Condition 3 leads to improving values of the bandwidth estimate. The convergence results are precisely stated in the theorems below:

Theorem 5.1. *If $n \to \infty$, $\sigma \to 0$ and $m \to \infty$ while satisfying the following rate conditions*

1. $(n\sigma^{md})/(mC^m) \to \infty$, where $C = 2/(2\pi)^{d/2}$,

2. $m/n\sigma \to 0$,

3. $m\sigma^2 \to 0$,

4. $\sigma^{1/m} \to 1$,

then for the separable model, one has

$$\omega_m(\mathbf{1}_S) \xrightarrow{p.} \sup_{\mathbf{s} \in \partial S} p(\mathbf{s}), \tag{5.16}$$

where "p." denotes convergence in probability.

Theorem 5.2. *If $n \to \infty$, $\sigma \to 0$ and $m \to \infty$ while satisfying the following rate conditions*

1. $(n\sigma^{md})/(mC^m) \to \infty$, where $C = 2/(2\pi)^{d/2}$,

2. $m/n \to 0$,

3. $m\sigma^2 \to 0$,

then for the non-separable model, one has

$$\omega_m(\mathbf{1}_A) \xrightarrow{p.} \sup_{\mathbf{x} \in \partial A} p(\mathbf{x}). \tag{5.17}$$

The dependence of the results on the rate conditions will be explained later in the proofs section. An example of parameter choices that allow all the scaling laws to hold simultaneously is illustrated in the following corollary:

Corollary 5.1. *Equations* (5.16) *and* (5.17) *hold if for each value of n, we choose m and σ as follows:*

$$m = [m_0 \ (\log n)^y], \tag{5.18}$$

$$\sigma = \sigma_0 \ n^{-x/md}, \tag{5.19}$$

for constants $m_0, \sigma_0 > 0$, $1/2 < y < 1$ *and* $0 < x < 1$. $[\,.\,]$ *indicates taking the nearest integer value.*

Theorems 5.1 and 5.2 give an explicit connection between bandwidth estimates of class indicator signals and the underlying geometry of the dataset. This interpretation forms the basis of justifying the choice of bandwidth as a smoothness constraint in graph-based learning algorithms. Theorem 5.1 suggests that for the separable model, if the boundary ∂S passes through regions of low probability density, then the bandwidth of the corresponding class indicator vector $\omega(\mathbf{1}_S)$ is low. A similar conclusion is suggested for the nonseparable model from Theorem 5.2, i.e., if the density of data points in the overlap region ∂A is low, then the bandwidth $\omega(\mathbf{1}_A)$ is low. In other words, low density of data in the boundary regions leads to smooth indicator functions.

From our results, we also get an intuition behind the smoothness constraint imposed in the bandlimited reconstruction approach (5.7) for semi-supervised learning. Basically, enforcing smoothness on classes in terms of indicator bandwidth ensures that the learning algorithm chooses a boundary passing through regions of low data density in the separable case. Similarly, in the nonseparable case, it ensures that variations in labels occur in regions of low density. Further, the bandwidth cutoff ω_L effectively imposes a constraint on the complexity of the hypothesis space – a larger value increases the size of the hypothesis space and results in choosing more complex boundaries.

As a special case of our analysis, we also get a convergence result for the graph cut in the nonseparable model analogous to the results of [42] for the separable model. Note that the cut in this case equals the sum of weights of edges connecting points that belong to class A to points that do not belong to class A, i.e.,

$$Cut(A, A^c) = \sum_{\mathbf{X}_i \in A, \mathbf{X}_j \in A^c} w_{ij} = \mathbf{1}_A^t \mathbf{L} \mathbf{1}_A. \tag{5.20}$$

With this definition, we have the following result:
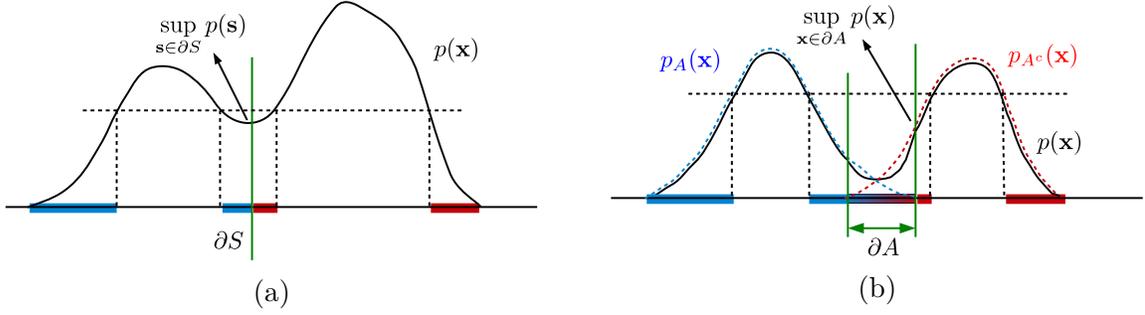
Figure 5.2: 1-D example illustrating the theoretical label complexity for (a) the separable model, (b) the nonseparable model. Note that labeling all points where density is lower than supremum density over the boundary resolves all ambiguity and results in perfect prediction.

Theorem 5.3. *If $n \to \infty$, $\sigma \to 0$ and $n\sigma^{d+1} \to \infty$, then*

$$\frac{1}{n} Cut(A, A^c) \xrightarrow{p.} \int \alpha_A \alpha_{A^c} p_A(\mathbf{x}) p_{A^c}(\mathbf{x}) d\mathbf{x} \tag{5.21}$$

The result above indicates that if the overlap between the conditional distributions of a particular class and its compliment is low, then the value of the graph cut is lower. This justifies the use of spectral clustering in the context of nonseparable models.

5.3.2 Label complexity of SSL

In the context of semi-supervised learning, we define the label complexity as the minimum fraction of labeled examples required for perfectly predicting the labels of the unlabeled data points. This quantity essentially is an indicator of how "good" the semi-supervised problem is, i.e., how much help do we get from geometry while learning. A low label complexity is indicative of a favorable situation, where one is able to learn from only a few known labels by exploiting data geometry. In the following discussion, we first estimate the theoretical label complexities of the data models we consider, and then show that the expected label complexity of the sampling theoretic approach to learning exactly matches these values in the asymptotic limit.

Theoretical label complexities

A simple way to compute the label complexity, for the data models we consider, is to find the fraction of points belonging to a region that fully encompasses the boundary. To formalize this, let us define the following two regions in $\mathbb{R}^d$:

$$\mathcal{X}_S = \{\mathbf{x} : p(\mathbf{x}) \leq \sup_{\mathbf{s} \in \partial S} p(\mathbf{s})\}, \tag{5.22}$$

$$\mathcal{X}_A = \{\mathbf{x} : p(\mathbf{x}) \leq \sup_{\mathbf{x} \in \partial A} p(\mathbf{x})\}. \tag{5.23}$$

Note that by definition, ∂S is fully contained in $\mathcal{X}_S$ and ∂A is fully contained in $\mathcal{X}_A$ (see Figure 5.2 for an example in $\mathbb{R}^1$). To perfectly reconstruct $\mathbf{1}_S$ and $\mathbf{1}_A$, it is sufficient to know the labels of all points in $\mathcal{X}_S$ and $\mathcal{X}_A$ respectively, as this strategy removes all ambiguity in labeling the two classes. Based on this, we arrive at the following conclusions:

Observation 5.2. *The theoretical label complexity of learning $\mathbf{1}_S$ and $\mathbf{1}_A$ in the asymptotic limit are $P(\mathcal{X}_S)$ and $P(\mathcal{X}_A)$ respectively, where $P(E) = \int_E p(\mathbf{x})d\mathbf{x}$.*

Label complexity of graph-based learning

Using our results, we can show that the same label complexities hold for the graph-based sampling theoretic approach to semi-supervised classification. In this context, label complexity can be seen as the fraction of samples required for perfectly reconstructing a signal on the similarity graph. It is known that the fraction of samples required for perfectly reconstructing a bandlimited signal cannot be more than the fraction of eigenvalues of the Laplacian below the signal's bandwidth [4]. Since our bandwidth convergence results relate the bandwidth of indicators for the two data models with data geoemtry, we only need to asymptotically relate the number of eigenvalues of $\mathbf{L}$ below any constant in terms of data geometry. This is achieved through the following result:

Theorem 5.4. *Let $\mathcal{N}_{\mathbf{L}}(t)$ be the number of eigenvalues of $\mathbf{L}$ below a constant t. Then, as $n \to \infty$ and $\sigma \to 0$, we have*

$$\mathbb{E}\left\{\frac{1}{n}\mathcal{N}_{\mathbf{L}}(t)\right\} \longrightarrow P\left(\{\mathbf{x} : p(\mathbf{x}) \leq t\}\right), \tag{5.24}$$

Proof. See Section 5.6.4. ∎

99

Substituting the bandwidth convergence results from Theorems 5.1 and 5.2 (i.e., $t = \omega_m(\mathbf{1}_S)$ and $t = \omega_m(\mathbf{1}_A)$), we immediately get the desired value of the expected label complexity of graph-based semi-supervised learning:

Theorem 5.5. *If the conditions in Theorems 5.1 and 5.2 hold, then the expected label complexities of bandlimited reconstruction for the separable and nonseparable models are given as*

$$\lim \; \frac{1}{n}\mathbb{E}\left\{\mathcal{N}_{\mathbf{L}}(\omega_m(\mathbf{1}_S))\right\} \to P(\mathcal{X}_S), \; and \tag{5.25}$$

$$\lim \; \frac{1}{n}\mathbb{E}\left\{\mathcal{N}_{\mathbf{L}}(\omega_m(\mathbf{1}_A))\right\} \to P(\mathcal{X}_A). \tag{5.26}$$

The following remarks are in order:

1. Note that Theorem 5.4 and Theorem 5.5 can be strengthened by proving convergence of $\frac{1}{n}\mathcal{N}_{\mathbf{L}}(t)$ rather than its expected value. This requires further analysis, which we leave for future work. The result in Theorem 5.5 also encourages us to conjecture that the convergence results for bandwidth estimates also hold for the convergence of the bandwidth itself.

2. This result further strengthens the connection between graph-based learning methods and the semi-supervised smoothness assumption, since one can conclude that the number of labeled examples required for perfect prediction depends on the geometry of the data around the boundary. A low value of the density at the boundary results in a lower label complexity.

3. One might ask what is the advantage of using graph-based methods for semi-supervised learning, if we can predict the class labels by the simple labeling strategy used to compute label complexities in Observation 5.2. Note that our definition of label complexity is an ideal one which aims for perfect reconstruction. The power of graph-based methods would be more evident for a more practical definition of label complexity, where one tries to find the number of labels required for achieving a certain error tolerance. We leave this issue for future work.

In the following two sections, we first numerically validate our results through experiments and then provide theoretical proofs.

Table 5.2: Illustrative boundaries used in the separable model.

Boundary	Description	$\sup_{\mathbf{s} \in \partial S} p(\mathbf{s})$
∂S_1	$x = 0$	0.0607
∂S_2	$x = -1$	0.2547
∂S_3	$x = y^2 - 1$	0.2547
∂S_4	$y = 0$	0.5969
∂S_5	$x^2 + y^2 = 1$	0.5969

5.4 Numerical validation

We now present simple numerical analyses[1] to validate our results and demonstrate their usefulness in practice. For simulating the separable model, we first consider a data distribution based on a 2D Gaussian Mixture Model (GMM) with two Gaussians: $\mu_1 = [-1\ \ 0], \Sigma_1 = 0.25\mathbf{I}$ and $\mu_2 = [1\ \ 0], \Sigma_2 = 0.16\mathbf{I}$, and mixing proportions $\alpha_1 = 0.4$ and $\alpha_2 = 0.6$ respectively. The probability density function is illustrated in Figure 5.3. Next, we evaluate the claim of Theorem 5.1 on five boundaries, described in Table 5.2. These boundaries are depicted in Figure 5.4 and are illustrative of typical separation assumptions such as linear or non-linear and low or high density.

For simulating the nonseparable model, we first construct the following smooth (twice-differentiable) 2D probability density function

$$q(x, y) = \begin{cases} \frac{3}{\pi} \left[1 - (x^2 + y^2) \right]^2, & x^2 + y^2 \leq 1 \\ 0, & x^2 + y^2 > 1 \end{cases}. \tag{5.27}$$

Note that datapoints (X, Y) can be sampled from this distribution by setting the coordinates $X = \sqrt{1 - U^{1/4}} \cos(2\pi V)$, $Y = \sqrt{1 - U^{1/4}} \sin(2\pi V)$, where $U, V \sim \text{Uniform}(0, 1)$. We then use $q(x, y)$ to define a nonseparable 2D model with mixture density $p(x, y) = \alpha_A p_A(x, y) + \alpha_{A^c} p_{A^c}(x, y)$, where $p_A(x, y) = q(x - 0.75, y)$, $p_{A^c}(x, y) = q(x + 0.75, y)$ and $\alpha_A = \alpha_{A^c} = 0.5$. The probability density function is illustrated in Figure 5.3. The overlap region or boundary ∂A for this model is given by

$$\partial A = \left\{ (x, y) : (x - 0.75)^2 + y^2 < 1 \text{ and } (x + 0.75)^2 + y^2 < 1 \right\}. \tag{5.28}$$

---

[1]Code available at https://github.com/aamiranis/asymptotics_graph_ssl
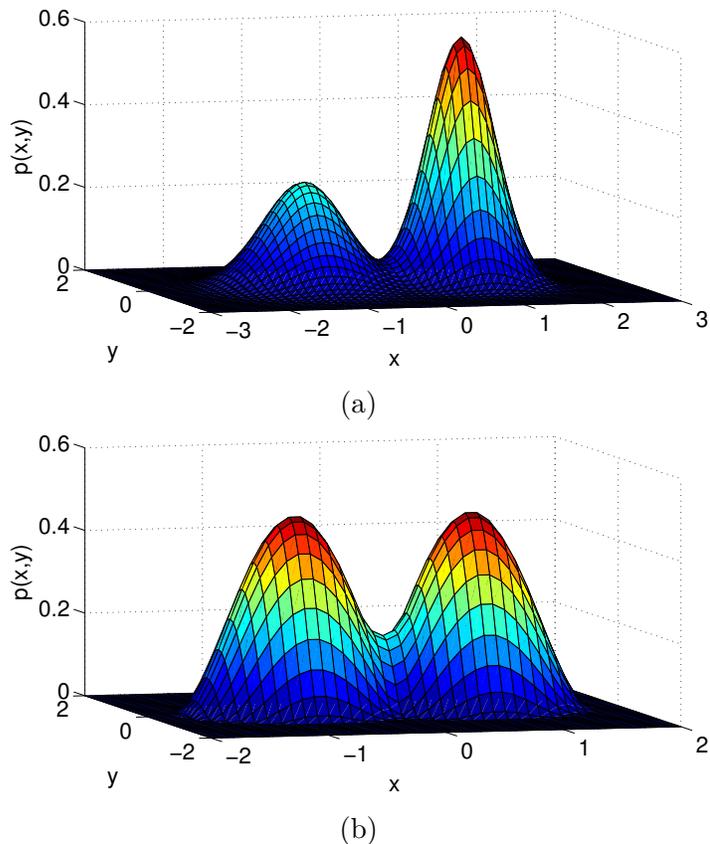
Figure 5.3: Probability density functions to generate data for (a) separable model, (b) nonseparable model.

Further, for this model, we have $\sup_{\partial A} p(\mathbf{x}) = 0.2517$.

In our first experiment, we validate the statements of Theorems 5.1 and 5.2 by comparing the left and right hand sides of (5.16) and (5.17) for corresponding boundaries. This is carried out in the following way: we draw $n = 2500$ points from each model and construct the corresponding similarity graphs using $\sigma = 0.1$. Then, for the boundaries ∂S_i in the separable model and ∂A in the nonseparable model, we carry out the following steps:

1. We first construct the indicator functions $\mathbf{1}_{S_i}$ and $\mathbf{1}_A$ on the corresonding graphs.

2. We then compute the empirical bandwidth $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ in a manner that takes care of numerical error: we first obtain the eigenvectors of the corresponding
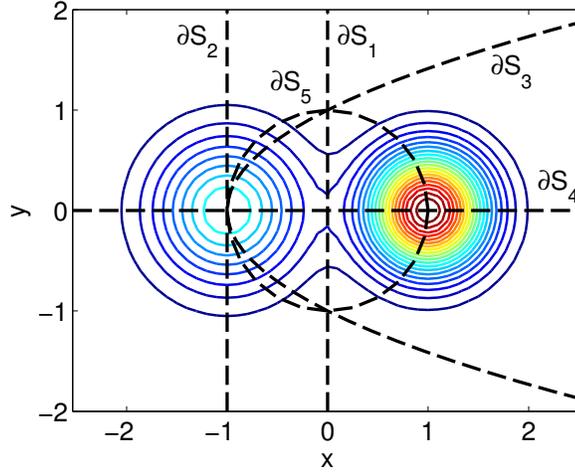
Figure 5.4: Boundaries $\{\partial S_i\}$ considered in the separable model.
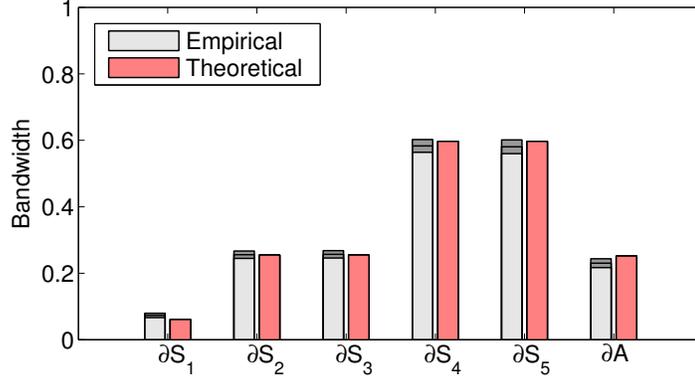
Figure 5.5: Convergence of empirical value of bandwidths $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ for different boundaries $\{\partial S_i\}$ and ∂A on corresponding graphs. Dark shaded regions denote standard deviation over 100 experiments. Red bars indicate theoretical values.

$\mathbf{L}$, then set $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ to be ν for which energy contained in the Fourier coefficients corresponding to eigenvalues $\lambda_j > \nu$ is at most 0.01%, i.e.,

$$\omega(\mathbf{1}_{S_i}) = \min\left\{\nu \,\Big|\, \sum_{j:\lambda_j>\nu} \left(\mathbf{u}_j^T \mathbf{1}_{S_i}\right)^2 \le 10^{-4}\right\} \tag{5.29}$$

$$\omega(\mathbf{1}_A) = \min\left\{\nu \,\Big|\, \sum_{j:\lambda_j>\nu} \left(\mathbf{u}_j^T \mathbf{1}_A\right)^2 \le 10^{-4}\right\}. \tag{5.30}$$
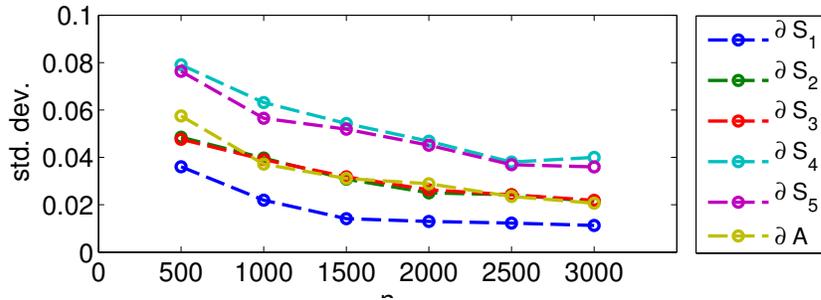
Figure 5.6: Standard deviation of $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ as a function of n.
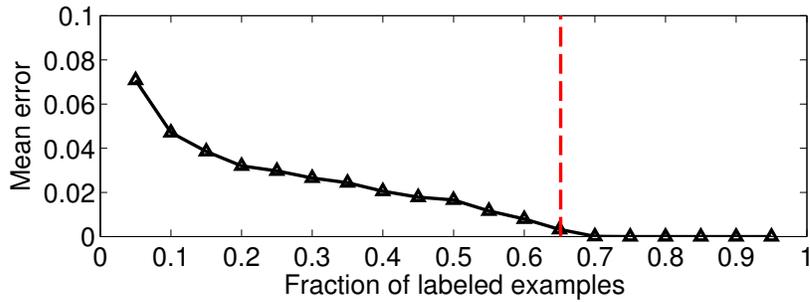
(a)
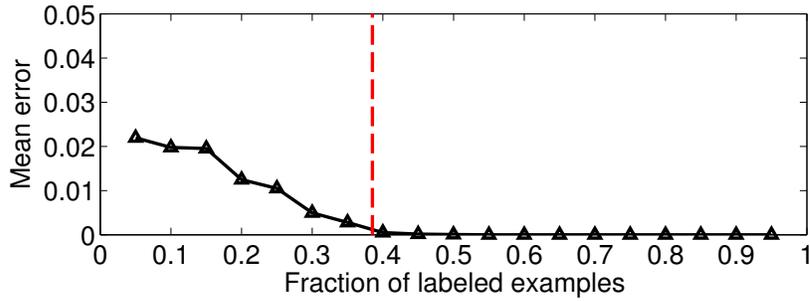
(b)

Figure 5.7: Mean reconstruction errors averaged over 100 experiments for (a) $\mathbf{1}_{S_3}$, and (b) $\mathbf{1}_A$. Red-dashed lines indicate the theoretical label complexities of ∂S_3 and ∂A.

The procedure above is repeated 100 times and the mean of $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ are compared with $\sup_{\mathbf{s}\in\partial S_i} p(\mathbf{s})$ and $\sup_{\mathbf{x}\in\partial A} p(\mathbf{x})$ respectively. The result is plotted in Figure 5.5. We observe that the empirical bandwidth is close to the theoretically predicted value and has a very low standard deviation. This supports our conjecture that stochastic convergence should hold for the bandwidth. To further justify this claim, we study the

behavior of the standard deviation of $\omega(\mathbf{1}_{S_i})$ and $\omega(\mathbf{1}_A)$ as a function of n in Figure 5.6, where we observe a decreasing trend consistent with our result.

For our second experiment, we validate the bound on the label complexity of graph-based SSL in Theorem 5.5 by reconstructing the indicator function corresponding to ∂S_3 and ∂A from a fraction of labeled examples on the corresponding graphs. This is carried out as follows: For a given budget B, we find the set of points to label pivoted column-wise Gaussian elimination on the eigenvector matrix $\mathbf{U}$ of $\mathbf{L}$. This method ensures that the obtained labeled set guarantees perfect recovery for signals spanned by the first B eigenvectors of $\mathbf{L}$ [4]. We then recover the indicator functions from these labeled sets by solving the least squares problem in (5.7) followed by thresholding. The mean reconstruction error is defined as

$$E_{\text{mean}} = \frac{\text{No. of mismatches on unlabeled set}}{\text{Size of unlabeled set}} \tag{5.31}$$

We repeat the experiment 100 times by generating different graphs and plot the averaged E_{mean} against the fraction of labeled examples. The result is illustrated in Figure 5.7. We observe that the error goes to zero as the fraction of labeled points goes beyond the theoretically predicted label complexity as predicted. This reinforces the intuition that the bandwidth of class indicators is closely linked with the inherent geometry of the data.

5.5 Summary

In this chapter, we provided an interpretation of the graph sampling theoretic approach to semi-supervised learning. Our work analyzed the bandwidth of class indicator signals with respect to the Laplacian eigenvector basis and revealed its connection to the underlying geometry of the dataset. This connection is useful in justifying graph-based approaches for semi-supervised and unsupervised learning problems, and provides a geometrical interpretation of the smoothness assumptions imposed in the bandlimited reconstruction approach. Specifically, our results have shown that an estimate of the bandwidth of class indicators converges to the supremum of the probability density on the class boundaries for the separable model, and on the overlap regions for the nonseparable model. This quantifies the connection between the assumptions of smoothness (in terms of bandlimitedness) and low density separation, since boundaries passing through

regions of low data density result in lower bandwidth of the class indicator signals. We numerically validated these results through various experiments.

Our analysis also sheds light on the label complexity of graph-based semi-supervised learning problems. We showed that perfect prediction from a few labeled examples using a graph-based bandlimited interpolation approach requires the same amount of labeling as one would need to completely encompass the boundary or region of ambiguity. This indicates that graph-based approaches achieve the theoretical label complexity as dictated by the underlying geometry of the problem. We believe that the main potential of graph-based methods will be apparent in situations where one can tolerate a certain amount of prediction error, in which case such approaches shall require fewer labeled examples. We plan to investigate this as part of future work.

5.5.1   Future work

There are several directions in which our results can be extended. In this work, we only considered Gaussian-weighted graphs, an immediate extension would be to consider arbitrary kernel functions for computing graph weights, or density dependent edge-connections such as k-nearest neighbors [42]. Another possibility is to consider data defined on a subset of the d-dimensional Euclidean space. It will also be useful to have uniform convergence results over all possible boundaries. Moreover, our definition of label complexity considers perfect label predictions, it would be interesting to study trade-offs between prediction error, the number of labeled examples, and the impact of labeling noise.

Finally, note that our results on label complexity are theoretical at this point – we assume complete knowledge of the indicator signal in order to quantify the amount of labeling required. However, this result can be made useful in a practical sense if one were able to *apriori* estimate the bandwidth of a class indicator as this would give an indication of "when to stop" labeling. One possible approach is to estimate the bandwidth of any signal through randomly (or possibly, cleverly) chosen samples:

$$\omega(\mathbf{f}) \approx \omega_m(\mathbf{f}) = \left(\frac{\mathbf{f}^T \mathbf{L}^m \mathbf{f}}{\mathbf{f}^T \mathbf{f}} \right)^{1/m} \approx \left(\frac{\mathbf{f}^T\, \mathbf{S}^T \mathbf{S}\, \mathbf{L}^m\, \mathbf{S}^T \mathbf{S}\, \mathbf{f}}{\mathbf{f}^T\, \mathbf{S}^T \mathbf{S}\, \mathbf{f}} \right)^{1/m} , \qquad (5.32)$$

where $\mathbf{S}$ is the sampling operator. Analyzing the convergence of this bandwidth estimate along with our prediction error trade-offs will give us a handle on how much labeling is required for achieving a particular error tolerance.

5.6 Proofs

We now present the proofs of Theorems 5.1 and 5.2 through a sequence of lemmas. The main idea is to perform a variance-bias decomposition of the bandwidth estimate and then prove the convergence of each term independently. Specifically, for the indicator vector $\mathbf{1}_R \in \{0,1\}^n$ of any region $R \subset \mathbb{R}^n$, one can consider the random variable:

$$(\omega_m(\mathbf{1}_R))^m = \frac{\mathbf{1}_R^T \mathbf{L}^m \mathbf{1}_R}{\mathbf{1}_R^T \mathbf{1}_R} = \frac{\frac{1}{n}\mathbf{1}_R^T \mathbf{L}^m \mathbf{1}_R}{\frac{1}{n}\mathbf{1}_R^T \mathbf{1}_R}. \tag{5.33}$$

We study the convergence of this quantity by considering the numerator and denominator separately (it is easy to show that the fraction converges if both the numerator and denominator converge). By the strong law of large numbers, we conclude the following for the denominator as $n \to \infty$:

$$\frac{1}{n}\mathbf{1}_R^T \mathbf{1}_R \xrightarrow{a.s.} \int_{\mathbf{x} \in R} p(\mathbf{x})d\mathbf{x}. \tag{5.34}$$

For the numerator, we decompose it into two parts – a variance term for which we show stochastic convergence using a concentration inequality, and a bias term for which we prove deterministic convergence.

5.6.1 Convergence of variance terms

Let $V = \frac{1}{n}\mathbf{1}_R^T \mathbf{L}^m \mathbf{1}_R$, then we have the following concentration result:

Lemma 5.1 (Concentration). *For every $\epsilon > 0$, we have:*

$$\Pr\left(|V - \mathbb{E}\{V\}| > \epsilon\right)$$
$$\leq 2\exp\left(\frac{-[n/(m+1)]\sigma^{md}\epsilon^2}{2C^m \mathbb{E}\{V\} + \frac{2}{3}|C^m - \sigma^{md}\mathbb{E}\{V\}|\epsilon}\right), \tag{5.35}$$

where $C = 2/(2\pi)^{d/2}$.

Proof. Recalling that $w_{i,j} = K_{\sigma^2}(\mathbf{X}_i, \mathbf{X}_j)$, we begin by explicitly expanding $V = \frac{1}{n}\mathbf{1}_R^T(\mathbf{D} - \mathbf{W})^m\mathbf{1}_R$ into the following summation

$$V = \frac{1}{n^{m+1}} \sum_{i_1, i_2, \ldots, i_{m+1}} g\left(\mathbf{X}_{i_1}, \mathbf{X}_{i_2}, \ldots, \mathbf{X}_{i_{m+1}}\right). \tag{5.36}$$

The above expansion has the form of a V-statistic. Details on how to explicitly write the summation are given in Section 5.6.5. Note that g is composed of a sum of 2^m terms, each a product of m kernel functions that are non-negative. Therefore,

$$g \leq 2^m \|K\|_\infty^m = \left(\frac{2}{(2\pi\sigma^2)^{d/2}} \right)^m = \frac{C^m}{\sigma^{md}}. \tag{5.37}$$

In order to apply a concentration inequality for V, we first re-write it in the form of a U-statistic by regrouping terms in the summation in order to remove repeated indices, as given in [35]:

$$V = \frac{1}{n^{(m+1)}} \sum_{(n,m+1)} g^* \left(\mathbf{X}_{i_1}, \mathbf{X}_{i_2}, \ldots, \mathbf{X}_{i_{m+1}} \right), \tag{5.38}$$

where $\sum_{(n,m+1)}$ denotes summation over all ordered (m+1)-tuples of distinct indices taken from the set $\{1, \ldots, n\}$, $n^{(m+1)} = n.(n-1)\ldots(n-m)$ is the number of (m+1)-permutations of n and g^* is a convex combination of specific instances of g that absorbs repeating indices (see supplementary material for a complete expansion):

$$g^* \left(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m+1} \right) = \frac{n^{(m+1)}}{n^{m+1}} g \left(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m+1} \right)$$
$$+ \text{(terms with repeated indices)}. \tag{5.39}$$

Therefore, g^* has the same upper bound as that of g derived in (5.37). Moreover, using the fact that $\mathbb{E}\{V\} = \mathbb{E}\{g^*\}$, we can bound the variance of g^* as

$$\text{Var}\{g^*\} \leq \|g^*\|_\infty \mathbb{E}\{g^*\} = \frac{C^m}{\sigma^{md}} \mathbb{E}\{V\}. \tag{5.40}$$

Finally, plugging in the bound and variance of g^* in Bernstein's inequality for U-statistics as stated in [35, 33], we arrive at the desired result of (5.35). $\blacksquare$

Note that as $n \to 0$ and $\sigma \to 0$ with rates satisfying $(n\sigma^{md})/(mC^m) \to \infty$, we have $P(|V - \mathbb{E}\{V\}| > \epsilon) \to 0$ for all $\epsilon > 0$. The continuous mapping theorem then allows us to conclude that $V^{1/m} \xrightarrow{p.} (\mathbb{E}\{V\})^{1/m}$.

5.6.2 Convergence of the bias term for the separable model

To evaluate the convergence of bias terms, we shall require the following properties of the d-dimensional Gaussian kernel:

Lemma 5.2. *If $p(\mathbf{x})$ is twice differentiable, then*

$$\int K_{\sigma^2}(\mathbf{x}, \mathbf{y}) p(\mathbf{y}) d\mathbf{y} = p(\mathbf{x}) + O\left(\sigma^2\right). \tag{5.41}$$

Proof. Using the substitution $\mathbf{y} = \mathbf{x} + \mathbf{t}$ followed by Taylor expansion about $\mathbf{x}$, we have

$$
\begin{aligned}
\int K_{\sigma^2}&(\mathbf{x}, \mathbf{y}) p(\mathbf{y}) d\mathbf{y} \\
&= \int \frac{1}{(2\pi\sigma^2)^{d/2}} e^{-\|\mathbf{t}\|^2/2\sigma^2} p(\mathbf{x} + \mathbf{t}) d\mathbf{t} \\
&= \int \frac{1}{(2\pi\sigma^2)^{d/2}} e^{-\|\mathbf{t}\|^2/2\sigma^2} \left(p(\mathbf{x}) + \mathbf{t}^T \nabla p(\mathbf{x}) \right. \\
&\qquad \left. + \frac{1}{2} \mathbf{t}^T \nabla^2 p(\mathbf{x}) \mathbf{t} + \dots \right) d\mathbf{t} \\
&= p(\mathbf{x}) + 0 + \frac{\sigma^2}{2} \text{Tr}(\nabla^2 p(\mathbf{x})) + \dots \\
&= p(\mathbf{x}) + O(\sigma^2).
\end{aligned}
$$

where the third step follows from simple component-wise integration. ∎

Lemma 5.3. *If $p(\mathbf{x})$ is twice differentiable, then*

$$
\begin{aligned}
\int K_{a\sigma^2}&(\mathbf{x}, \mathbf{z}) K_{b\sigma^2}(\mathbf{z}, \mathbf{y}) p(\mathbf{z}) d\mathbf{z} \\
&= K_{(a+b)\sigma^2}(\mathbf{x}, \mathbf{y}) \left(p\left(\frac{b\mathbf{x} + a\mathbf{y}}{a+b} \right) + O\left(\sigma^2\right) \right).
\end{aligned} \tag{5.42}
$$

Proof. Note that

$$
\begin{aligned}
K_{a\sigma^2}(\mathbf{x}, \mathbf{z}) K_{b\sigma^2}(\mathbf{z}, \mathbf{y}) &= \frac{1}{(2\pi a\sigma^2)^{\frac{d}{2}}} e^{-\frac{\|\mathbf{x}-\mathbf{z}\|^2}{2a\sigma^2}} \frac{1}{(2\pi b\sigma^2)^{\frac{d}{2}}} e^{-\frac{\|\mathbf{z}-\mathbf{y}\|^2}{2b\sigma^2}} \\
&= \frac{1}{(2\pi(a+b)\sigma^2)^{\frac{d}{2}}} e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2(a+b)\sigma^2}} \frac{1}{(2\pi \frac{ab}{a+b}\sigma^2)^{\frac{d}{2}}} e^{-\frac{\|\mathbf{z}-\frac{b\mathbf{x}+a\mathbf{y}}{a+b}\|^2}{2(\frac{ab}{a+b})\sigma^2}} \\
&= K_{(a+b)\sigma^2}(\mathbf{x}, \mathbf{y}) \; K_{\frac{ab}{a+b}\sigma^2} \left(\frac{b\mathbf{x} + a\mathbf{y}}{a+b}, \mathbf{z} \right).
\end{aligned}
$$

Therefore, we have

$$\int K_{a\sigma^2}(\mathbf{x}, \mathbf{z}) K_{b\sigma^2}(\mathbf{z}, \mathbf{y}) p(\mathbf{z}) d\mathbf{z}$$

$$= K_{(a+b)\sigma^2}(\mathbf{x}, \mathbf{y}) \int K_{\frac{ab}{a+b}\sigma^2}\left(\frac{b\mathbf{x} + a\mathbf{y}}{a+b}, \mathbf{z}\right) p(\mathbf{z}) d\mathbf{z}$$

$$= K_{(a+b)\sigma^2}(\mathbf{x}, \mathbf{y}) \left(p\left(\frac{b\mathbf{x} + a\mathbf{y}}{a+b}\right) + O\left(\sigma^2\right)\right),$$

where the last step follows from Lemma 5.2. ∎

In order to prove convergence for the separable model, we need the following results:

Lemma 5.4. *If $p(\mathbf{x})$ is Lipschitz continuous, then for a smooth hypersurface ∂S that divides $\mathbb{R}^d$ into S_1 and S_2, and whose radius has curvature that is bounded by $\tau > 0$,*

$$\lim_{\sigma \to 0} \frac{1}{\sigma} \int_{S_1} \int_{S_2} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\alpha(\mathbf{x}_1) p^\beta(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2$$

$$= \frac{1}{\sqrt{2\pi}} \int_{\partial S} p^{\alpha+\beta}(\mathbf{s}) d\mathbf{s}, \tag{5.43}$$

where α and β are positive integers. Moreover, for positive integers a, b, and $\alpha, \beta, \alpha', \beta'$ such that $\alpha + \beta = \alpha' + \beta' = \gamma$, we have:

$$\lim_{\sigma \to 0} \frac{1}{\sigma} \int_{S_1} \int_{S_1} \left[K_{a\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\alpha(\mathbf{x}_1) p^\beta(\mathbf{x}_2) \right.$$

$$\left. - K_{b\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^{\alpha'}(\mathbf{x}_1) p^{\beta'}(\mathbf{x}_2) \right] d\mathbf{x}_1 d\mathbf{x}_2$$

$$= \frac{\sqrt{b} - \sqrt{a}}{\sqrt{2\pi}} \int_{\partial S} p^\gamma(\mathbf{s}) d\mathbf{s}. \tag{5.44}$$

The proof of this lemma is given in Appendix 5.6.6. We now prove the deterministic convergence of $\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}$ in the following lemma:

Lemma 5.5. *As $n \to \infty$, $\sigma \to 0$ such that $m/n \to 0, m\sigma^2 \to 0$, we have*

$$\frac{1}{\sigma} \mathbb{E}\left\{\frac{1}{n} \mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\} \to \frac{t(m)}{\sqrt{2\pi}} \int_{\partial S} p^{m+1}(\mathbf{s}) d\mathbf{s}, \tag{5.45}$$

where $t(m) = \sum_{r=1}^{m-1} \binom{m-1}{r} (-1)^r (\sqrt{r+1} - \sqrt{r})$.

110

Proof. We evaluate $\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}$ term by term by expanding $\mathbf{L}^m$ as $(\mathbf{D}-\mathbf{W})^{m-1}(\mathbf{D}-\mathbf{W})$. Details on the intermediate steps of this expansion are given in Section 5.6.5. Using (5.41) repeatedly, we have for the first two terms of the expansion:

$$
\frac{1}{\sigma}\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{D}\ldots\mathbf{D}(\mathbf{D}-\mathbf{W})\mathbf{1}_S\right\}
$$
$$
= \frac{1}{\sigma}\int_S \int_{S^c} K_{\sigma^2}(\mathbf{x},\mathbf{y})p^m(\mathbf{x})p(\mathbf{y})d\mathbf{x}d\mathbf{y}
$$
$$
+ O\left(\sigma\right) + O\left(\frac{m}{n\sigma}\right). \tag{5.46}
$$

For the rest of the terms, we also require the use of (5.42). However, in this case, we encounter several terms of the form $p(\theta\mathbf{x}+(1-\theta)\mathbf{y})$ for some $\theta \in [0,1]$. Since $m\sigma^2 \to 0$ and $p(\mathbf{x})$ is assumed to be Lipschitz continuous, we can approximate such terms by $p(\mathbf{x})$ or $p(\mathbf{y})$. Therefore, for all terms in the expansion of $(\mathbf{D}-\mathbf{W})^{m-1}$ containing $r > 1$ occurrences of $\mathbf{W}$ (there are $\binom{m-1}{r}$ such terms), repeated use of (5.41), (5.42) gives:

$$
\frac{1}{\sigma}\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T[\mathbf{D}^{m-1-r},\mathbf{W}^r](\mathbf{D}-\mathbf{W})\mathbf{1}_S\right\}
$$
$$
= \frac{1}{\sigma}\left[\int_S \int_S K_{r\sigma^2}(\mathbf{x},\mathbf{y})p^\alpha(\mathbf{x})p^\beta(\mathbf{y})d\mathbf{x}d\mathbf{y}\right.
$$
$$
\left. - \int_S \int_S K_{(r+1)\sigma^2}(\mathbf{x},\mathbf{y})p^{\alpha'}(\mathbf{x})p^{\beta'}(\mathbf{y})d\mathbf{x}d\mathbf{y}\right]
$$
$$
+ O(\sigma) + O\left(\frac{m}{n\sigma}\right). \tag{5.47}
$$

where $\alpha + \beta = \alpha' + \beta' = m + 1$ and $[\mathbf{D}^{m-1-r},\mathbf{W}^r]$ denotes an expression containing r occurrences of $\mathbf{W}$ and $m-1-r$ occurrences of $\mathbf{D}$. Now, using Lemma 5.4, we conclude that the right hand sides of (5.46) and (5.47) converge to $\frac{1}{\sqrt{2\pi}}\int_{\partial S} p^{m+1}(\mathbf{s})d\mathbf{s}$ and $\frac{\sqrt{r+1}-\sqrt{r}}{\sqrt{2\pi}}\int_{\partial S} p^{m+1}(\mathbf{s})d\mathbf{s}$, respectively, as $\sigma \to 0$ and $m/n\sigma \to 0$. Putting everything together in the expansion of $\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}$, we get the desired result. ∎

Since $\sigma^{1/m} \to 1$, we have

$$
\left(\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}\right)^{1/m} = \sigma^{1/m}\left(\frac{1}{\sigma}\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}\right)^{1/m}
$$
$$
\to \left(\frac{t(m)}{\sqrt{2\pi}}\int_{\partial S} p^{m+1}(\mathbf{s})d\mathbf{s}\right)^{1/m} \tag{5.48}
$$

111

Finally, we note that as $m \to \infty$, we have

$$\left(\frac{\frac{t(m)}{\sqrt{2\pi}} \int_{\partial S} p^{m+1}(\mathbf{s}) d\mathbf{s}}{\int_S p(\mathbf{x}) d\mathbf{x}} \right)^{1/m} \xrightarrow{\text{s.}} \sup_{\mathbf{s} \in \partial S} p(\mathbf{s}) \tag{5.49}$$

Therefore, we conclude for the separable model

$$\omega_m(\mathbf{1}_S) \to \sup_{\mathbf{s} \in \partial S} p(\mathbf{s}) \tag{5.50}$$

5.6.3 Convergence of bias term for the nonseparable model

For the nonseparable model, we need to prove convergence of $\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_A^T \mathbf{L}^m \mathbf{1}_A\right\}$. This is illustrated in the following lemma:

Lemma 5.6. *As $n \to \infty$, $\sigma \to 0$ such that $m/n \to 0$, $m\sigma^2 \to 0$, we have*

$$\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_A^T \mathbf{L}^m \mathbf{1}_A\right\} \to \int \alpha_A \alpha_{A^c} p_A(\mathbf{x}) p_{A^c}(\mathbf{x}) p^{m-1}(\mathbf{x}) d\mathbf{x}. \tag{5.51}$$

Proof. Similar to the proof of Lemma 5.5, we evaluate $\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_A^T \mathbf{L}^m \mathbf{1}_A\right\}$ term by term by expanding $\mathbf{L}^m$ as $(\mathbf{D} - \mathbf{W})^{m-1}(\mathbf{D} - \mathbf{W})$. Details on the intermediate steps of this expansion are given in Section 5.6.5. Using (5.41) repeatedly, we have for the first two terms of the expansion:

$$\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_A^T \mathbf{D} \dots \mathbf{D}(\mathbf{D} - \mathbf{W}) \mathbf{1}_A\right\}$$
$$= \int \alpha_A \alpha_{A^c} p_A(\mathbf{x}) p_{A^c}(\mathbf{x}) p^{m-1}(\mathbf{x}) d\mathbf{x} + O\left(\sigma^2\right) + O\left(\frac{m}{n}\right). \tag{5.52}$$

Further, for all terms in the expansion of $(\mathbf{D} - \mathbf{W})^{m-1}$ containing $r > 1$ occurrences of $\mathbf{W}$ (there are $\binom{m-1}{r}$ such terms), repeated use of (5.41), (5.42) gives:

$$\mathbb{E}\left\{\frac{1}{n} \mathbf{1}_A^T [\mathbf{D}^{m-1-r}, \mathbf{W}^r](\mathbf{D} - \mathbf{W}) \mathbf{1}_A\right\} = O\left(\sigma^2\right) + O\left(\frac{m}{n}\right). \tag{5.53}$$

Therefore, as $\sigma \to 0$, $m/n \to 0$, we get the desired result. ∎

We finally note that as $m \to \infty$, we have

$$\left(\frac{\int \alpha_A \alpha_{A^c} p_A(\mathbf{x}) p_{A^c}(\mathbf{x}) p^{m-1}(\mathbf{x}) d\mathbf{x}}{\int_A p(\mathbf{x}) d\mathbf{x}} \right)^{1/m} \xrightarrow{\text{s.}} \sup_{\mathbf{x} \in \partial A} p(\mathbf{x}) \tag{5.54}$$

112

Therefore, we conclude for the nonseparable model

$$\omega_m(\mathbf{1}_A) \to \sup_{\mathbf{x} \in \partial A} p(\mathbf{x}) \tag{5.55}$$

Note that Lemma 5.6 for special case for $m = 1$ yields

$$\frac{1}{n}\mathbf{1}_A^T \mathbf{L}\mathbf{1}_A \to \int \alpha_A \alpha_{A^c} p_A(\mathbf{x}) p_{A^c}(\mathbf{x}) d\mathbf{x} \tag{5.56}$$

which proves Theorem 5.3.

5.6.4 Proof of Theorem 5.4

We begin by recalling the definition of the empirical spectral distribution (ESD) of $\mathbf{L}$: $\mu_n(x) = \frac{1}{n} \sum_{i=1}^n \delta(x - \lambda_i)$, where $\{\lambda_i\}$ are the eigenvalues of $\mathbf{L}$. For each x, $\mu_n(x)$ is a function of $\mathbf{X}_1, \ldots, \mathbf{X}_n$, and thus a random variable. Note that the fraction of eigenvalues of $\mathbf{L}$ below a constant t, and its expected value can be computed from the ESD as

$$\frac{1}{n}\mathcal{N}_{\mathbf{L}}(t) = \int_0^t \mu_n(x) dx, \tag{5.57}$$

$$\mathbb{E}\left\{ \frac{1}{n}\mathcal{N}_{\mathbf{L}}(t) \right\} = \int_0^t \mathbb{E}\left\{ \mu_n(x) \right\} dx, \tag{5.58}$$

Therefore, to understand the behavior of the expected fraction of eigenvalues of $\mathbf{L}$ below t, we need to analyze the convergence of the expected ESD in the asymptotic limit. The idea is to show the convergence of the moments of $\mathbb{E}\left\{ \mu_n(x) \right\}$ to the moments of a limiting distribution $\mu(x)$. Then, by a standard convergence result, $\mathbb{E}\left\{ \mu_n(I) \right\} \to \mu(I)$ for intervals I. More precisely, let the $\Rightarrow$ symbol denote weak convergence of measures, then we use the following result that follows from the Weierstrass approximation theorem:

Lemma 5.7. *Let μ_n be a sequence of probability measures and μ be a compactly supported probability measure. If $\int x^m \mu_n(dx) \to \int x^m \mu(dx)$ for all $m \geq 1$, then $\mu_n \Rightarrow \mu$.*

We then use the following result on equivalence of different notions of weak convergence of measures [13, Theorem 25.2] in order to prove our result for cumulative distribution functions.

Lemma 5.8. *$\mu_n \Rightarrow \mu$ if and only if $\mu_n(A) \to \mu(A)$ for every μ-continuity set A.*

We begin by writing the mth moment of $\mathbb{E}\{\mu_n(x)\}$:

$$\int x^m \mathbb{E}\{\mu_n(x)\}\,dx = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\{\lambda_i^m\} = \mathbb{E}\left\{\frac{1}{n}\mathrm{Tr}\,(\mathbf{L}^m)\right\}. \tag{5.59}$$

Now, note that $\mathbf{L}^m = (\mathbf{D} - \mathbf{W})^m = \mathbf{D}^m + \sum_{k=1}^m \binom{m}{k}[\mathbf{D}^{m-k}, \mathbf{W}^k]$, where $[\mathbf{D}^{m-k}, \mathbf{W}^k]$ denotes product terms with $m-k$ occurrences of $\mathbf{D}$ and k occurrences of $\mathbf{W}$. Therefore, we have for the right hand side of (5.59):

$$\begin{aligned}
\mathbb{E}&\left\{\frac{1}{n}\mathrm{Tr}\,(\mathbf{L}^m)\right\} \\
&= \int \left(\int K(\mathbf{x}_{i_1}, \mathbf{x}_{i_2})p(\mathbf{x}_{i_2})d\mathbf{x}_{i_2}\right)\dots \\
&\quad \left(\int K(\mathbf{x}_{i_1}, \mathbf{x}_{i_{m+1}})p(\mathbf{x}_{i_{m+1}})d\mathbf{x}_{i_{m+1}}\right)p(\mathbf{x}_{i_1})d\mathbf{x}_{i_1} \\
&\quad + O\left(\frac{m}{n}\right)\text{(expected value of other terms)}
\end{aligned} \tag{5.60}$$

Using (5.41) repeatedly in the equation above, we get:

$$\mathbb{E}\left\{\frac{1}{n}\mathrm{Tr}\,(\mathbf{L}^m)\right\} = \int p^{m+1}(\mathbf{x})d\mathbf{x} + O\left(\frac{m}{n}\right) + O\left(\sigma^2\right) \tag{5.61}$$

Therefore, as $n \to \infty$ and $\sigma \to 0$, we have:

$$\int x^m \mathbb{E}\{\mu_n(x)\}\,dx = \int p^m(\mathbf{x})p(\mathbf{x})d\mathbf{x} \tag{5.62}$$

From the right hand side of the equation above, we conclude that the mth moment of the expected ESD of $\mathbf{L}$ converges to the mth moment of the distribution of a random variable $Y = p(\mathbf{X})$, where $p(\mathbf{x})$ is the probabilty density function of $\mathbf{X}$. Moreover, since $p_Y(y)$ has compact support, $\mathbb{E}\{\mu_n(x)\}$ converges weakly to the probability density function of $p_Y(y)$. Hence, the following can be said about the expected fraction of eigenvalues of $\mathbf{L}$:

$$\begin{aligned}
\mathbb{E}\left\{\frac{1}{n}\mathcal{N}_{\mathbf{L}}(t)\right\} &= \int_0^t \mathbb{E}\{\mu_n(x)\}\,dx \\
&\xrightarrow{\text{s.}} \int_0^t p_Y(y)dy = \int_{p(\mathbf{x})\leq t} p(\mathbf{x})d\mathbf{x}.
\end{aligned} \tag{5.63}$$

This proves our claim in Theorem 5.4. Note that, to prove the stochastic convergence of the fraction itself rather than its expected value, we would need a condition similar to

114

those in Theorems 5.1 and 5.2 to hold for each moment. In that case, σ will go to 0 in a prohibitively slow fashion. We believe that this is an artifact of the methods we employ for proving the result. Hence, our conjecture is that the convergence result must hold for $\frac{1}{n}\mathcal{N}_{\mathbf{L}}(t)$ itself, and we leave the analysis of this statement for future work.

5.6.5   Expansions of $\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S$ and $\mathbb{E}\left\{\frac{1}{n}\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S\right\}$

To expand $\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S$ in terms of the elements w_{ij} of $\mathbf{W}$, we first write the expression for each product term. Since $\mathbf{L}^m = \frac{1}{n^m}(\mathbf{D} - \mathbf{W})^m$, there are 2^m such terms. Let us first introduce the following notation: $[\mathbf{D}, \mathbf{W}]_m$ denotes a product term containing the matrices $\mathbf{D}$ and $\mathbf{W}$, such that there are m matrices in the product. Note that $\mathbf{L}^m$ is essentially a summation of all possible $[\mathbf{D}, \mathbf{W}]_m$ with appropriate signs.

Now, the explicit expression for $\mathbf{1}_S^T [\mathbf{D}, \mathbf{W}]_m \mathbf{1}_S$ can be obtained using the following procedure:

1. All product terms have a form defined by the following template:

$$
\begin{aligned}
&\mathbf{1}_S^T [\mathbf{D}, \mathbf{W}]_m \mathbf{1}_S \\
&= \sum_{i_1, \ldots, i_{m+1}} (\mathbf{1}_S)_{i_1} w_{i_1 i_2} w_{*i_3} \ldots w_{*i_m} w_{*i_{m+1}} (\mathbf{1}_S)_*
\end{aligned}
\tag{5.64}
$$

   where the locations with $*$ need to be filled with appropriate indices depending on the product term. Note that each w_{ij} is contributed by either a $\mathbf{D}$ or $\mathbf{W}$ depending on its location in the expression.

2. We fill the locations one-by-one from left to right, using the following set of rules. Let w_{ab} be the term preceding w_{*c}, then

   - If w_{ab} is contributed by $\mathbf{D}$, then $* = a$.
   - If w_{ab} is contributed by $\mathbf{W}$, then $* = b$.

3. Let $w_{a i_{m+1}}$ denote the term preceding $(\mathbf{1}_S)_*$. Then, we have the following rule:

   - If $w_{a i_{m+1}}$ is contributed by $\mathbf{D}$, then $* = a$.
   - If $w_{a i_{m+1}}$ is contributed by $\mathbf{W}$, then $* = i_{m+1}$.

The expansion of $\mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S$ can be found by summing up the expansions of the individual product terms $\mathbf{1}_S^T [\mathbf{D}, \mathbf{W}]_m \mathbf{1}_S$. Recalling that $w_{ij} = K(\mathbf{X}_i, \mathbf{X}_j)$, we conclude

$$\frac{1}{n} \mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S = \frac{1}{n^{m+1}} \sum_{i_1, \dots, i_{m+1}} g(\mathbf{X}_{i_1}, \mathbf{X}_{i_2}, \dots, \mathbf{X}_{i_{m+1}}) \tag{5.65}$$

The expression for $\mathbb{E} \left\{ \frac{1}{n} \mathbf{1}_S^T \mathbf{L}^m \mathbf{1}_S \right\}$ can be evaluated in a similar fashion, except that the summations are replaced by integrals. We first evaluate the expected value of individual product terms $\mathbb{E} \left\{ \frac{1}{n} \mathbf{1}_S^T [\mathbf{D}, \mathbf{W}]_m \mathbf{1}_S \right\}$ by the following rules:

1. The template for the expected value of any product term can be expressed through the following template:

$$\mathbb{E} \left\{ \frac{1}{n} \mathbf{1}_S^T [\mathbf{D}, \mathbf{W}]_m \mathbf{1}_S \right\}$$
$$= \int \dots \int \left(\mathbf{1}_S(\mathbf{x}_1) K(\mathbf{x}_1, \mathbf{x}_2) K(\mathbf{x}_*, \mathbf{x}_3) \right.$$
$$\left. \dots K(\mathbf{x}_*, \mathbf{x}_{m+1}) \mathbf{1}_S(\mathbf{x}_*) \right) p(\mathbf{x}_1) d\mathbf{x}_1 \dots p(\mathbf{x}_{m+1}) d\mathbf{x}_{m+1} \tag{5.66}$$

where the locations with $*$ need to be filled with appropriate indices depending on the product term. Once again, each $K(\mathbf{x}_i, \mathbf{x}_j)$ is contributed by either a $\mathbf{D}$ or a $\mathbf{W}$.

2. We fill the locations one-by-one from left to right, using the following set of rules. Let $K(\mathbf{x}_a, \mathbf{x}_b)$ be the term preceding $K(\mathbf{x}_*, \mathbf{x}_c)$. Then

   - If $K(\mathbf{x}_a, \mathbf{x}_b)$ is contributed by $\mathbf{D}$, then $* = a$.
   - If $K(\mathbf{x}_a, \mathbf{x}_b)$ is contributed by $\mathbf{W}$, then $* = b$.

3. Further, let $K(\mathbf{x}_a, \mathbf{x}_{m+1})$ be the term preceding $\mathbf{1}_S(\mathbf{x}_*)$. Then

   - If $K(\mathbf{x}_a, \mathbf{x}_{m+1})$ is contributed by $\mathbf{D}$, then $* = a$.
   - If $K(\mathbf{x}_a, \mathbf{x}_{m+1})$ is contributed by $\mathbf{W}$, then $* = m + 1$.

5.6.6 Proof of Lemma 4

The key ingredient required for evaluating the integrals in Lemma 5.4 involves selecting a radius R $(< \tau)$ as a function of σ that satisfies the following properties as $\sigma \to 0$:

1. $R \to 0$,

2. $R/\sigma \to \infty$,

3. $R^2/\sigma \to 0$,

4. $\epsilon_R/\sigma \to 0$, where $\epsilon_R := \int_{\|\mathbf{z}\|>R} K_{\sigma^2}(\mathbf{0}, \mathbf{z})d\mathbf{z}$.

A particular choice of R is given by $R = \sqrt{d\sigma^2 \log 1/\sigma^2}$. Note that $R \to 0$ as $\sigma \to 0$. Further, substituting this expression in the tail bound for the norm of a d-dimensional Guassian vector gives us:

$$
\begin{aligned}
\frac{\epsilon_R}{\sigma} &= \frac{1}{\sigma} \int_{\|\mathbf{z}\|>R} K_{\sigma^2}(\mathbf{0}, \mathbf{z})d\mathbf{z} \\
&\leq \frac{1}{\sigma} \left(\frac{\sigma^2 d}{R^2}\right)^{-d/2} e^{-\frac{R^2}{2\sigma^2}+\frac{d}{2}} \\
&= \frac{1}{\sigma} \left(e\sigma^2 \log(1/\sigma^2)\right)^{d/2}
\end{aligned}
\tag{5.67}
$$

Therefore, if $d > 1$, then $\epsilon_R/\sigma \to 0$ as $\sigma \to 0$. Further, it is easy to ensure $R < \tau$ for the regime of σ in our proofs.

We now consider the proof of equation (5.43), let

$$
I := \frac{1}{\sigma} \int_{S_1} \int_{S_2} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\alpha(\mathbf{x}_1) p^\beta(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2
\tag{5.68}
$$

Further, let $[S_1]_R$ indicate a tubular region of thickness R adjacent to the boundary ∂S in S_1, i.e., the set of points in S_1 at a distance $\leq R$ from the boundary. Then, we have

$$
\begin{aligned}
I = &\underbrace{\frac{1}{\sigma} \int_{[S_1]_R} p^\alpha(\mathbf{x}_1) \int_{S_2} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \, d\mathbf{x}_1}_{I_1} \\
&+ \underbrace{\frac{1}{\sigma} \int_{[S_1]_R^c} p^\alpha(\mathbf{x}_1) \int_{S_2} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \, d\mathbf{x}_1}_{E_1}
\end{aligned}
\tag{5.69}
$$

E_1 is the error associated with approximating I by I_1 and exhibits the following behavior

Lemma 5.9. $\lim_{\sigma \to 0} E_1 = 0$.

Proof. Note that

$$
\begin{aligned}
E_1 &\leq \frac{1}{\sigma} (p_{\max})^\beta \int_{[S_1]_R^c} p^\alpha(\mathbf{x}_1) \left(\int_{S_2} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_2 \right) d\mathbf{x}_1 \\
&\leq \frac{1}{\sigma} (p_{\max})^\beta \int_{[S_1]_R^c} p^\alpha(\mathbf{x}_1) \left(\int_{\|\mathbf{z}\| > R} K_{\sigma^2}(\mathbf{0}, \mathbf{z}) d\mathbf{z} \right) d\mathbf{x}_1 \\
&= \frac{\epsilon_R}{\sigma} (p_{\max})^\beta \int_{[S_1]_R^c} p^\alpha(\mathbf{x}_1) d\mathbf{x}_1 \\
&\leq \frac{\epsilon_R}{\sigma} (p_{\max})^{\alpha+\beta}
\end{aligned}
\tag{5.70}
$$

Using $\lim_{\sigma \to \infty} \epsilon_R / \sigma = 0$, we get the desired result. ∎

In order to analyze I_1, we need to define certain geometrical constructions (illustrated in Figure 5.8) as follows:

Definition 5.1. *Geometrical constructions for analyzing the integrals*

1. *For each $\mathbf{x}_1 \in [S_1]_R$, we define a transformation of coordinates as:*

$$
\mathbf{x}_1 = \mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1),
\tag{5.71}
$$

   where $\mathbf{s}_1$ is the foot of the perpendicular dropped from $\mathbf{x}_1$ onto ∂S, r_1 is the distance between $\mathbf{s}_1$ and $\mathbf{x}_1$, and $\mathbf{n}(\mathbf{s}_1)$ is the surface normal at $\mathbf{s}_1$ (towards the direction of $\mathbf{x}_1$). Since the minimum radius of curvature of ∂S is τ and $R < \tau$, this mapping is injective.

2. *For each $\mathbf{s}_1 \in \partial S$, let $H_{\mathbf{s}_1}^+$ denote the halfspace created by the plane tangent on $\mathbf{s}_1$ and on the side of S_2. Similarly, let $H_{\mathbf{s}_1}^-$ denote the halfspace on the side of S_1, that is, $H_{\mathbf{s}_1}^- = \mathbb{R}^d \setminus H_{\mathbf{s}_1}^+$.*

3. *Let $W_{\mathbf{s}_1}^+(x)$ denote an infinite slab of thickness x tangent to ∂S at $\mathbf{s}_1$ and towards the side S_2. Let $W_{\mathbf{s}_1}^-(y)$ denote a similar slab of thickness y on the side of S_1.*

4. *Finally, for any $\mathbf{x}$, let $B(\mathbf{x}, R)$ denote the Euclidean ball of radius R centered at $\mathbf{x}$.*
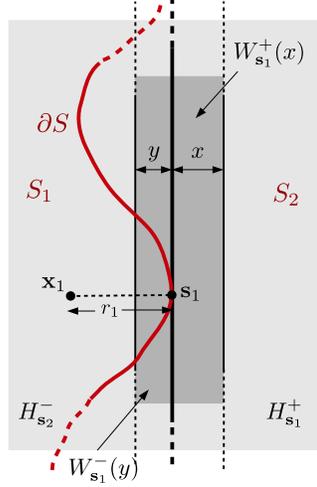
Figure 5.8: Geometrical constructions in Definition 5.1.

We now consider I_1, the main idea here is to approximate the integral over S_2 by an integral over the halfspace $H_{\mathbf{s}_1}^+$. Hence, we have:

$$
I_1 = \underbrace{\frac{1}{\sigma} \int_{[S_1]_R} p^\alpha(\mathbf{x}_1) \int_{H_{\mathbf{s}_1}^+} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 d\mathbf{x}_1}_{I_2}
$$
$$
+ \underbrace{\frac{1}{\sigma} \int_{[S_1]_R} p^\alpha(\mathbf{x}_1) \int_{S_2 - H_{\mathbf{s}_1}^+} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 d\mathbf{x}_1}_{E_2} \tag{5.72}
$$

where E_2 is the error associated with the approximation. Therefore, we have

$$
I = I_2 + E_2 + E_1 \tag{5.73}
$$

We now show that as $\sigma \to 0$, $I_2 \to \frac{1}{\sqrt{2\pi}} \int_{\partial S} p^{\alpha+\beta}(\mathbf{s}) d\mathbf{s}$, and $E_2 \to 0$.

Lemma 5.10. $\lim_{\sigma \to 0} I_2 = \frac{1}{\sqrt{2\pi}} \int_{\partial S} p^{\alpha+\beta}(\mathbf{s}) d\mathbf{s}$.

Proof. Using the change of coordinates $\mathbf{x}_1 = \mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1)$, we have

$$
I_2 = \frac{1}{\sigma} \int_{\partial S} \int_0^R p^\alpha(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1))
$$
$$
\left(\int_{H_{\mathbf{s}_1}^+} K_{\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \right)
$$
$$
|\det J(\mathbf{s}_1, r_1)| d\mathbf{s}_1 dr_1 \tag{5.74}
$$

where $J(\mathbf{s}_1, r_1)$ denotes the Jacobian of the transformation. Now, an arc $\widehat{PQ}$ of length ds at a distance r_1 away from ∂S gets mapped to an arc $\widehat{P'Q'}$ on ∂S whose length lies in the interval $[ds(1 - \frac{r_1}{\tau}), ds(1 + \frac{r_1}{\tau})]$. Therefore, for all points within $[S_1]_R$, we have

$$\left(1 - \frac{R}{\tau}\right)^{d-1} \leq |\det J(\mathbf{s}_1, r_1)| \leq \left(1 + \frac{R}{\tau}\right)^{d-1}. \tag{5.75}$$

Further, since $p(\mathbf{x})$ is Lipschitz continuous with constant L_p, $p^\alpha(\mathbf{x})$ is also Lipschitz continuous with constant $L_{p,\alpha}$. Therefore, for any $\mathbf{x}_1 \in [S_1]_R$, we have $p^\alpha(\mathbf{x}_1) = p^\alpha(\mathbf{s}_1) + L_{p,\alpha}R$. This leads to the following simplification for I_2:

$$I_2 = \left(1 + O(R^{d-1})\right) \int_{\partial S} p^\alpha(\mathbf{s}_1) I_3(\mathbf{s}_1) d\mathbf{s}_1 + O(R^d) \int_{\partial S} I_3(\mathbf{s}_1) d\mathbf{s}_1, \tag{5.76}$$

where we defined

$$I_3(\mathbf{s}_1) := \frac{1}{\sigma} \int_0^R \int_{H_{\mathbf{s}_1}^+} K_{\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 dr_1. \tag{5.77}$$

Note that every $\mathbf{x}_2 \in H_{\mathbf{s}_1}^+$ can be written as $\mathbf{s}_2 + r_2 \mathbf{n}(\mathbf{s}_2)$, where $\mathbf{n}(\mathbf{s}_2) = -\mathbf{n}(\mathbf{s}_1)$. Hence, we get

$$I_3(\mathbf{s}_1) = \int_{\mathbb{R}^{d-1}} \frac{1}{(2\pi\sigma^2)^{\frac{d-1}{2}}} e^{-\frac{\|\mathbf{s}_1 - \mathbf{s}_2\|^2}{2\sigma^2}} p^\beta(\mathbf{s}_2 - r_2 \mathbf{n}(\mathbf{s}_1)) d\mathbf{s}_2$$

$$\times \frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1 + r_2)^2}{2\sigma^2}} dr_1 dr_2$$

$$= \left(\int_{\mathbb{R}^{d-1}} \frac{1}{(2\pi\sigma^2)^{\frac{d-1}{2}}} e^{-\frac{\|\mathbf{s}_1 - \mathbf{s}_2\|^2}{2\sigma^2}} p^\beta(\mathbf{s}_2) d\mathbf{s}_2 + O(R) \right)$$

$$\times \frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1 + r_2)^2}{2\sigma^2}} dr_1 dr_2$$

$$= \left(p^\beta(\mathbf{s}_1) + O(\sigma^2) + O(R) \right)$$

$$\times \frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1 + r_2)^2}{2\sigma^2}} dr_1 dr_2, \tag{5.78}$$

where we used Lipschitz continuity of $p^\beta(\mathbf{x})$ in the second equality and applied Lemma 5.2 to arrive at the last step. Further, using the definition of the Q-function and integration by parts, we note that

$$
\begin{aligned}
\frac{1}{\sigma} \int_0^R \int_0^\infty & \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1+r_2)^2}{2\sigma^2}} dr_1 dr_2 \\
&= \int_0^{R/\sigma} \int_0^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{(x+y)^2}{2}} dx dy \\
&= \int_0^{R/\sigma} Q(y) dy \\
&= yQ(y)\Big|_0^{R/\sigma} - \int_0^{R/\sigma} Q'(y) dy \\
&= \frac{R}{\sigma} Q\left(\frac{R}{\sigma}\right) + \frac{1}{\sqrt{2\pi}} \left(1 - e^{-R^2/2\sigma^2}\right).
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
I_3(\mathbf{s}_1) = \Big(p^\beta(\mathbf{s}_1) + O(\sigma^2) + O(R)\Big) \times \\
\left(\frac{R}{\sigma} Q\left(\frac{R}{\sigma}\right) + \frac{1}{\sqrt{2\pi}} \left(1 - e^{-R^2/2\sigma^2}\right)\right).
\end{aligned} \tag{5.79}
$$

Combining (5.76) and (5.79) and using the fact that $R/\sigma \to \infty$ as $\sigma \to 0$ (from the definition of R), we get

$$
\lim_{\sigma \to \infty} I_2 = \frac{1}{\sqrt{2\pi}} \int_{\partial S} p^{\alpha+\beta}(\mathbf{s}) d\mathbf{s}, \tag{5.80}
$$

which concludes the proof. ∎

We now consider the error term E_2 and prove the following result:

Lemma 5.11. $\lim_{\sigma \to 0} E_2 = 0$.

Proof. Let us first rewrite E_2 as follows:

$$
E_2 = \frac{1}{\sigma} \int_{[S_1]_R} p^\alpha(\mathbf{x}_1) I_4(\mathbf{x}_1) d\mathbf{x}_1 \tag{5.81}
$$

where we defined

$$
I_4(\mathbf{x}_1) := \int_{S_2 - H_{\mathbf{s}_1}^+} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \tag{5.82}
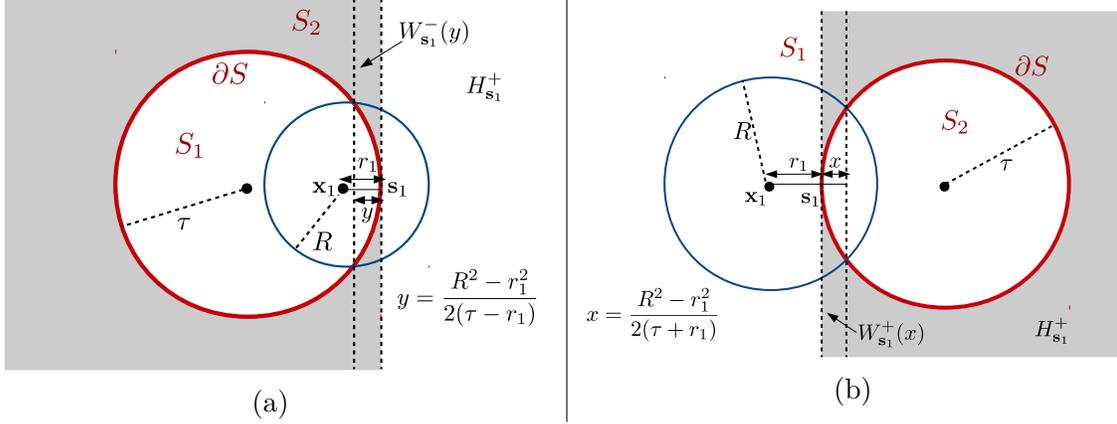$$

Figure 5.9: Worst-case scenarios for the boundary ∂S when (a) S_1 is a ball of radius τ, (b) S_2 is a ball of radius τ.

The key idea is to lower and upper bound $I_4(\mathbf{x}_1)$ for all $\mathbf{x}_1$ using worst case scenarios and evaluate the limits of the bounds. Note that $I_4(\mathbf{x}_1)$ is largest in magnitude when S_1 or S_2 is a sphere of radius τ, as illustrated in Figures 5.9a and 5.9b. We now make certain geometrical observations. For any $\mathbf{x}_1 = \mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1) \in [S_1]_R$, we observe from Figure 5.9b that

$$
\begin{aligned}
I_4(\mathbf{x}_1) &\leq \int_{W_{\mathbf{s}_1}^-\left(\frac{R^2-r_1^2}{2(\tau-r_1)}\right)} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \\
&\quad + \int_{B(\mathbf{x}_1, R)^c} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \\
&\leq \int_{W_{\mathbf{s}_1}^-(R')} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 + p_{\max}^\beta \epsilon_R.
\end{aligned}
\tag{5.83}
$$

where $R' = \frac{R^2}{2(\tau-R)}$. Similarly, from Figure 5.9a, we observe that

$$
\begin{aligned}
I_4(\mathbf{x}_1) &\geq -\left[\int_{W_{\mathbf{s}_1}^+\left(\frac{R^2-r_1^2}{2(\tau+r_1)}\right)} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \right. \\
&\quad \left. + \int_{B(\mathbf{x}_1, R)^c} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 \right] \\
&\geq -\left[\int_{W_{\mathbf{s}_1}^+(R')} K_{\sigma^2}(\mathbf{x}_1, \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 + p_{\max}^\beta \epsilon_R \right].
\end{aligned}
\tag{5.84}
$$

122

Substituting these in (5.81) and using a simplification similar to that of I_2 in (5.76), we get

$$E_2 \leq \left(1 + O(R^{d-1})\right) \int_{\partial S} p^\alpha(\mathbf{s}_1) I_5^-(\mathbf{s}_1) d\mathbf{s}_1$$
$$+ O(R^d) \int_{\partial S} I_5^-(\mathbf{s}_1) d\mathbf{s}_1 + \frac{\epsilon_R}{\sigma} p_{\max}^{\alpha+\beta}, \qquad (5.85)$$

$$E_2 \geq - \left(1 + O(R^{d-1})\right) \int_{\partial S} p^\alpha(\mathbf{s}_1) I_5^+(\mathbf{s}_1) d\mathbf{s}_1$$
$$- O(R^d) \int_{\partial S} I_5^+(\mathbf{s}_1) d\mathbf{s}_1 - \frac{\epsilon_R}{\sigma} p_{\max}^{\alpha+\beta}, \qquad (5.86)$$

where we defined

$$I_5^-(\mathbf{s}_1) := \frac{1}{\sigma} \int_0^R \int_{W_{\mathbf{s}_1}^-(R')} K_{\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 dr_1, \qquad (5.87)$$

$$I_5^+(\mathbf{s}_1) := \frac{1}{\sigma} \int_0^R \int_{W_{\mathbf{s}_1}^+(R')} K_{\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 dr_1. \qquad (5.88)$$

Similar to the evaluation of $I_3(\mathbf{s}_1)$ in (5.78), we have

$$I_5^+(\mathbf{s}_1) = \left(p^\beta(\mathbf{s}_1) + O(\sigma^2) + O(R)\right) \times \frac{1}{\sigma} \int_0^R \int_0^{R'} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1+r_2)^2}{2\sigma^2}} dr_1 dr_2, \qquad (5.89)$$

$$I_5^-(\mathbf{s}_1) = \left(p^\beta(\mathbf{s}_1) + O(\sigma^2) + O(R)\right) \times \frac{1}{\sigma} \int_0^R \int_0^{R'} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2\sigma^2}} dr_1 dr_2. \qquad (5.90)$$

We now evaluate the two 1-D integrals as follows:

$$\frac{1}{\sigma} \int_0^R \int_0^{R'} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1+r_2)^2}{2\sigma^2}} dr_1 dr_2$$
$$= \int_0^{R/\sigma} \int_0^{R'/\sigma} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x+y)^2}{2}} dx dy$$
$$= \int_0^{R/\sigma} \left(Q(y) - Q\left(y + \frac{R'}{\sigma}\right)\right) dy$$
$$= \int_0^{R/\sigma} Q(y) dy + \int_0^{R'/\sigma} Q(y) dy - \int_0^{\frac{R+R'}{\sigma}} Q(y) dy$$
$$= \frac{R}{\sigma} Q\left(\frac{R}{\sigma}\right) + \frac{R}{\sigma} Q\left(\frac{R'}{\sigma}\right) - \frac{R+R'}{\sigma} Q\left(\frac{R+R'}{\sigma}\right)$$
$$\frac{1}{\sqrt{2\pi}} \left(1 - e^{-\frac{R^2}{2\sigma^2}} - e^{-\frac{R'^2}{2\sigma^2}} + e^{-\frac{(R+R')^2}{2\sigma^2}}\right).$$

Similarly,

$$\frac{1}{\sigma}\int_0^R \int_0^{R'} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2\sigma^2}} dr_1 dr_2$$

$$= \int_0^{R/\sigma} \int_0^{R'/\sigma} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-y)^2}{2}} dx dy$$

$$= \int_0^{R/\sigma} \left(Q\left(y - \frac{R'}{\sigma}\right) - Q(y) \right) dy$$

$$= \int_{-R'/\sigma}^0 Q(y)dy + \int_0^{\frac{R-R'}{\sigma}} Q(y)dy - \int_0^{R/\sigma} Q(y)dy$$

$$= \frac{R'}{\sigma}Q\left(-\frac{R'}{\sigma}\right) + \frac{R-R'}{\sigma}Q\left(\frac{R-R'}{\sigma}\right) - \frac{R}{\sigma}Q\left(\frac{R}{\sigma}\right)$$

$$\frac{1}{\sqrt{2\pi}}\left(e^{-\frac{R'^2}{2\sigma^2}} - 1 + e^{-\frac{(R+R')^2}{2\sigma^2}} - e^{-\frac{R^2}{2\sigma^2}} \right)$$

Noting that as $\sigma \to 0$, $R/\sigma \to \infty$ and $R'/\sigma \to 0$, we conclude that $\lim_{\sigma\to 0} E_2 = 0$. ∎

The proof of (5.44) proceeds in a similar fashion by approximating the inner integral using hyperplanes. Specifically, similar to the proof of (5.43), we can show that the integral on the left hand side can be written as $I + E$, where

$$I := \frac{1}{\sigma}\int_{[S_1]_R} \int_{H_{\mathbf{s}_1}^-} \left[K_{a\sigma^2}(\mathbf{x}_1, \mathbf{x}_2)p^\alpha(\mathbf{x}_1)p^\beta(\mathbf{x}_2) \right.$$

$$\left. - K_{b\sigma^2}(\mathbf{x}_1, \mathbf{x}_2)p^{\alpha'}(\mathbf{x}_1)p^{\beta'}(\mathbf{x}_2) \right] d\mathbf{x}_1 d\mathbf{x}_2, \tag{5.91}$$

and E is the residual associated with the approximation that can be shown to go to zero as $\sigma \to 0$ (we skip this proof since it is quite similar to the analysis for (5.43)). In order to evaluate I, we perform a change of coordinates $\mathbf{x}_1 = \mathbf{s}_1 + r_1\mathbf{n}(\mathbf{s}_1)$ as before to obtain

$$I = \frac{1}{\sigma}\int_{\partial S} \int_0^R \left[p^\alpha(\mathbf{s}_1 + r_1\mathbf{n}(\mathbf{s}_1))\left(\int_{H_{\mathbf{s}_1}^-} K_{a\sigma^2}(\mathbf{s}_1 + r_1\mathbf{n}(\mathbf{s}_1), \mathbf{x}_2)p^\beta(\mathbf{x}_2)d\mathbf{x}_2 \right) \right.$$

$$\left. - p^{\alpha'}(\mathbf{s}_1 + r_1\mathbf{n}(\mathbf{s}_1))\left(\int_{H_{\mathbf{s}_1}^-} K_{b\sigma^2}(\mathbf{s}_1 + r_1\mathbf{n}(\mathbf{s}_1), \mathbf{x}_2)p^{\beta'}(\mathbf{x}_2)d\mathbf{x}_2 \right) \right] |\det J(\mathbf{s}_1, r_1)| d\mathbf{s}_1 dr_1$$

$$= \int_{\partial S} p^\alpha(\mathbf{s}_1)I_\beta(\mathbf{s}_1)d\mathbf{s}_1 - \int_{\partial S} p^{\alpha'}(\mathbf{s}_1)I_{\beta'}(\mathbf{s}_1)d\mathbf{s}_1 + O\left(R^d\right), \tag{5.92}$$

where we defined

$$I_\beta(\mathbf{s}_1) := \frac{1}{\sigma} \int_0^R \int_{H_{\mathbf{s}_1}^-} K_{a\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^\beta(\mathbf{x}_2) d\mathbf{x}_2 dr_1,$$

$$I_{\beta'}(\mathbf{s}_1) := \frac{1}{\sigma} \int_0^R \int_{H_{\mathbf{s}_1}^-} K_{b\sigma^2}(\mathbf{s}_1 + r_1 \mathbf{n}(\mathbf{s}_1), \mathbf{x}_2) p^{\beta'}(\mathbf{x}_2) d\mathbf{x}_2 dr_1.$$

By using change of coordinates for $\mathbf{x}_2$ similar to the steps in (5.78), we obtain

$$I_\beta(\mathbf{s}_1) = \left(p^\beta(\mathbf{s}_1) + O(\sigma^2) + O(R) \right) \times$$
$$\frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi a\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2a\sigma^2}} dr_1 dr_2, \tag{5.93}$$

$$I_{\beta'}(\mathbf{s}_1) = \left(p^{\beta'}(\mathbf{s}_1) + O(\sigma^2) + O(R) \right) \times$$
$$\frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi b\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2b\sigma^2}} dr_1 dr_2, \tag{5.94}$$

The 1-D integrals can be evaluated as follows:

$$\frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi a\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2a\sigma^2}} dr_1 dr_2$$
$$= \sqrt{a} \int_0^{R/\sqrt{a}\sigma} \int_0^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-y)^2}{2}} dx dy$$
$$= \sqrt{a} \int_0^{R/\sqrt{a}\sigma} Q(-y) dy$$
$$= \sqrt{a} \int_0^{R/\sqrt{a}\sigma} (1 - Q(y)) dy$$
$$= \frac{R}{\sigma} - \frac{R}{\sigma} Q\left(\frac{R}{\sqrt{a}\sigma} \right) - \frac{\sqrt{a}}{\sqrt{2\pi}} \left(1 - e^{-R^2/2a\sigma^2} \right),$$
$$\frac{1}{\sigma} \int_0^R \int_0^\infty \frac{1}{\sqrt{2\pi b\sigma^2}} e^{-\frac{(r_1-r_2)^2}{2b\sigma^2}} dr_1 dr_2$$
$$= \frac{R}{\sigma} - \frac{R}{\sigma} Q\left(\frac{R}{\sqrt{b}\sigma} \right) - \frac{\sqrt{b}}{\sqrt{2\pi}} \left(1 - e^{-R^2/2b\sigma^2} \right).$$

Using the fact that $\alpha + \beta = \alpha' + \beta' = \gamma$, and taking the limit $\sigma \to 0$ after putting everything together, we conclude

$$\lim_{\sigma \to 0} I = \frac{\sqrt{b} - \sqrt{a}}{\sqrt{2\pi}} \int_{\partial S} p^\gamma(\mathbf{s}) d\mathbf{s}. \tag{5.95}$$

Reference List

[1] USPS handwritten digits data. http://www.cs.nyu.edu/~roweis/data.html.

[2] A. E. Alaoui, X. Cheng, A. Ramdas, M. J. Wainwright, and M. I. Jordan. Asymptotic behavior of ℓ_p-based Laplacian regularization in semi-supervised learning. In V. Feldman, A. Rakhlin, and O. Shamir, editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 879–906, Columbia University, New York, New York, USA, 23–26 Jun 2016. PMLR.

[3] A. Anis, A. Gadde, and A. Ortega. Towards a sampling theorem for signals on arbitrary graphs. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3864–3868, May 2014.

[4] A. Anis, A. Gadde, and A. Ortega. Efficient sampling set selection for bandlimited graph signals using graph spectral proxies. *IEEE Transactions on Signal Processing*, 64(14):3775–3789, July 2016.

[5] A. Anis, A. E. Gamal, S. Avestimehr, and A. Ortega. Asymptotic justification of bandlimited interpolation of graph signals for semi-supervised learning. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5461–5465, April 2015.

[6] A. Anis, A. E. Gamal, S. Avestimehr, and A. Ortega. A sampling theory perspective of graph-based semi-supervised learning. *Submitted to IEEE Transactions on Information Theory*, April 2017.

[7] A. Anis and A. Ortega. Critical sampling for wavelet filterbanks on arbitrary graphs. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3889–3893, March 2017.

[8] A. Anis, D. Tay, and A. Ortega. Tree-structured filterbanks on M-block cyclic graphs. In *Asilomar Conference on Signals, Systems, and Computers*, October 2017.

[9] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[10] M. Belkin and P. Niyogi. Using manifold stucture for partially labeled classification. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 953–960. MIT Press, 2003.

[11] M. Belkin and P. Niyogi. Semi-supervised learning on Riemannian manifolds. *Machine Learning*, 56(1):209–239, 2004.

[12] M. Belkin and P. Niyogi. Towards a theoretical foundation for Laplacian-based manifold methods. *J. Comput. Syst. Sci.*, 74(8):1289–1308, 2008.

[13] P. Billingsley. *Probability and Measure*. Wiley, New York, NY, 3rd edition, 1995.

[14] B. Bourdin, D. Bucur, and E. Oudet. Optimal partitions for eigenvalues. *SIAM Journal on Scientific Computing*, 31(6):4100–4114, 2010.

[15] O. Bousquet, O. Chapelle, and M. Hein. Measure based regularization. In *Advances in Neural Information Processing Systems (NIPS) 16*. MIT Press, 2004.

[16] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2009.

[17] N. Buchbinder, M. Feldman, J. S. Naor, and R. Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1433–1452, 2014.

[18] E. J. Candes and M. B. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, March 2008.

[19] Y. H. Chao, A. Ortega, and S. Yea. Graph-based lifting transform for intra-predicted video coding. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1140–1144, March 2016.

[20] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-Supervised Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2006.

[21] S. Chen, A. Sandryhaila, and J. Kovačević. Sampling theory for graph signals. In *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, April 2015.

[22] S. Chen, R. Varma, A. Sandryhaila, and J. Kovačević. Discrete signal processing on graphs: Sampling theory. *Signal Processing, IEEE Transactions on*, 2015.

[23] F. Chung. Laplacians and the Cheeger inequality for directed graphs. *Annals of Combinatorics*, 9(1):1–19, 2005.

[24] F. R. K. Chung. *Spectral graph theory*, volume 92. CBMS Regional Conference Series in Mathematics, AMS, 1997.

[25] S. Deutsch, A. Ortega, and G. Medioni. Manifold denoising based on spectral graph wavelets. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4673–4677, March 2016.

[26] V. N. Ekambaram, G. Fanti, B. Ayazifar, and K. Ramchandran. Critically-sampled perfect-reconstruction spline-wavelet filterbanks for graph signals. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 475–478, Dec 2013.

[27] V. N. Ekambaram, G. Fanti, B. Ayazifar, and K. Ramchandran. Wavelet-regularized graph semi-supervised learning. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 423–426, Dec 2013.

[28] Y. C. Eldar. Sampling with arbitrary sampling and reconstruction spaces and oblique dual frame vectors. *Journal of Fourier Analysis and Applications*, 9(1):77–96, 2003.

[29] A. Gadde, A. Anis, and A. Ortega. Active semi-supervised learning using sampling theory for graph signals. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 492–501, 2014.

[30] A. Gadde, S. Narang, and A. Ortega. Bilateral filter: Graph spectral interpretation and extensions. In *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pages 1222–1226, Sept 2013.

[31] K. Gröchenig. A discrete theory of irregular sampling. *Linear Algebra and its applications*, 193:129–150, 1993.

[32] W. H. Haemers. Interlacing eigenvalues and graphs. *Linear Algebra and its applications*, 226:593–616, 1995.

[33] M. Hein. *Geometrical aspects of statistical learning theory.* PhD thesis, TU Darmstadt, April 2006.

[34] M. Hein. *Uniform Convergence of Adaptive Graph-Based Regularization*, pages 50–64. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[35] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[36] R. A. Horn and C. R. Johnson. *Matrix analysis.* Cambridge University Press, 2012.

[37] Y. Hu and B. Defourny. Near-optimality bounds for greedy periodic policies with application to grid-level storage. In *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 1–8, Dec 2014.

[38] S. Joshi and S. Boyd. Sensor selection via convex optimization. *Signal Processing, IEEE Transactions on*, 57(2):451–462, 2009.

[39] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[40] A. Knyazev, A. Jujunashvili, and M. Argentati. Angles between infinite dimensional subspaces with applications to the rayleigh–ritz and alternating projectors methods. *Journal of Functional Analysis*, 259(6):1323–1345, 2010.

[41] A. V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM journal on scientific computing*, 23(2):517–541, 2001.

[42] M. Maier, U. von Luxburg, and M. Hein. How the result of graph clustering methods depends on the construction of the graph. *ESAIM: Probability and Statistics*, 17:370–418, 1 2013.

[43] A. G. Marques, S. Segarra, G. Leus, and A. Ribeiro. Sampling of graph signals with successive local aggregations. *IEEE Transactions on Signal Processing*, 64(7):1832–1843, April 2016.

[44] S. Narang and A. Ortega. Local two-channel critically sampled filter-banks on graphs. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 333–336, Sept 2010.

[45] S. Narang and A. Ortega. Perfect reconstruction two-channel wavelet filter banks for graph structured data. *IEEE Transactions on Signal Processing*, 60(6):2786–2799, June 2012.

[46] S. Narang and A. Ortega. Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs. *Signal Processing, IEEE Transactions on*, 61(19):4673–4685, Oct 2013.

[47] S. K. Narang, Y. H. Chao, and A. Ortega. Graph-wavelet filterbanks for edge-aware image processing. In *Statistical Signal Processing Workshop (SSP), IEEE*, pages 141–144, 2012.

[48] S. K. Narang, Y. H. Chao, and A. Ortega. Critically sampled graph-based wavelet transforms for image coding. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, pages 1–4, Oct 2013.

[49] S. K. Narang, A. Gadde, and A. Ortega. Signal processing techniques for interpolation in graph structured data. In *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, pages 5445–5449, 2013.

[50] S. K. Narang, A. Gadde, E. Sanou, and A. Ortega. Localized iterative methods for interpolation in graph structured data. In *Signal and Information Processing (GlobalSIP), IEEE Global Conference on*, 2013.

[51] H. Narayanan, M. Belkin, and P. Niyogi. On the relation between low density separation, spectral clustering and graph cuts. In *Advances in Neural Information Processing Systems (NIPS) 19*, 2006.

[52] H. Nguyen and M. Do. Downsampling of signals on graphs via maximum spanning trees. *Signal Processing, IEEE Transactions on*, 63(1):182–191, Jan 2015.

[53] H. Q. Nguyen, P. A. Chou, and Y. Chen. Compression of human body sequences using graph wavelet filter banks. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6152–6156, May 2014.

[54] B. Osting, C. D. White, and Édouard Oudet. Minimal Dirichlet energy partitions for graphs. *SIAM Journal on Scientific Computing*, 36(4):A1635–A1651, 2014.

[55] I. Pesenson. Sampling in Paley-Wiener spaces on combinatorial graphs. *Transactions of the American Mathematical Society*, 360(10):5603–5627, 2008.

[56] I. Pesenson. Variational splines and Paley–Wiener spaces on combinatorial graphs. *Constructive Approximation*, 2009.

[57] G. Puy, N. Tremblay, R. Gribonval, and P. Vandergheynst. Random sampling of bandlimited signals on graphs. *Applied and Computational Harmonic Analysis*, pages –, 2016.

[58] J. O. Riis. Discounted Markov programming in a periodic process. *Operations Research*, 13(6):920–929, 1965.

[59] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. SIAM, second edition, 2011.

[60] A. Sakiyama and Y. Tanaka. Edge-aware image graph expansion methods for oversampled graph laplacian matrix. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2958–2962, Oct 2014.

[61] A. Sakiyama and Y. Tanaka. Oversampled graph laplacian matrix for graph filter banks. *IEEE Transactions on Signal Processing*, 62(24):6425–6437, Dec 2014.

[62] A. Sandryhaila and J. Moura. Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure. *Signal Processing Magazine, IEEE*, 31(5):80–90, Sept 2014.

[63] A. Sandryhaila and J. Moura. Discrete signal processing on graphs: Frequency analysis. *Signal Processing, IEEE Transactions on*, 62(12):3042–3054, June 2014.

[64] A. Sandryhaila and J. M. F. Moura. Discrete signal processing on graphs. *IEEE Transactions on Signal Processing*, 61(7):1644–1656, April 2013.

[65] I. Shomorony and A. Avestimehr. Sampling large data on graphs. In *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*, pages 933–936, Dec 2014.

[66] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *Signal Processing Magazine, IEEE*, 30(3):83–98, May 2013.

[67] Y. Tanaka. Spectral domain sampling of graph signals. June 2017. arXiv:1706.05147 [cs.IT].

[68] D. B. H. Tay and A. Ortega. Bipartite graph filter banks: Polyphase analysis and generalization. *IEEE Transactions on Signal Processing*, 65(18):4833–4846, Sept 2017.

[69] O. Teke and P. P. Vaidyanathan. Extending classical multirate signal processing theory to graphs – Part i: Fundamentals. *IEEE Transactions on Signal Processing*, 65(2):409–422, Jan 2017.

[70] O. Teke and P. P. Vaidyanathan. Extending classical multirate signal processing theory to graphs – Part ii: M-channel filter banks. *IEEE Transactions on Signal Processing*, 65(2):423–437, Jan 2017.

[71] M. Tsitsvero, S. Barbarossa, and P. D. Lorenzo. Uncertainty principle and sampling of signals defined on graphs. In *2015 49th Asilomar Conference on Signals, Systems and Computers*, pages 1813–1818, Nov 2015.

[72] M. Tsitsvero, S. Barbarossa, and P. D. Lorenzo. Signals on graphs: Uncertainty principle and sampling. *IEEE Transactions on Signal Processing*, 64(18):4845–4860, Sept 2016.

[73] M. Vetterli, P. Marziliano, and T. Blu. Sampling signals with finite rate of innovation. *IEEE Transactions on Signal Processing*, 50(6):1417–1428, Jun 2002.

[74] X. Wang, P. Liu, and Y. Gu. Local-set-based graph signal reconstruction. *IEEE Transactions on Signal Processing*, 63(9):2432–2444, May 2015.

[75] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[76] L. Yang and W. Guo. Greedy local-set based sampling and reconstruction for band-limited graph signals. In *2016 23rd International Conference on Telecommunications (ICT)*, pages 1–5, May 2016.

[77] J. Zeng, G. Cheung, and A. Ortega. Bipartite subgraph decomposition for critically sampled wavelet filterbanks on arbitrary graphs. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6210–6214, March 2016.

[78] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In S. Thrun, L. K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 321–328. MIT Press, 2004.

[79] D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd International conference on Machine learning*, pages 1036–1043, 2005.

[80] D. Zhou, B. Schölkopf, and T. Hofmann. Semi-supervised learning on directed graphs. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1633–1640. MIT Press, Cambridge, MA, 2004.

[81] X. Zhou and M. Belkin. Semi-supervised learning by higher order regularization. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, pages 892–900, 2011.

[82] X. Zhou, M. Belkin, and N. Srebro. An iterated graph laplacian approach for ranking on manifolds. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, 2011*, pages 877–885, 2011.

[83] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning*, pages 912–919, 2003.