Graph Spectral Compressed Sensing

Xiaofan Zhu

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

July 2012

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering.

© 2012 Xiaofan Zhu

Abstract

Consider a signal whose entries are supported on the nodes of a graph. We study the metric to measure the smoothness of signals supported on graphs and provide theoretical explanations for when and why the Laplacian eigenbasis can be regarded as a meaningful "Fourier" transform of such signals. Moreover, we characterize the desired properties of the underlying graph for better compressibility of the signals. For a smooth signal with respect to the graph topology, our work proves that we can gather measurements from a random subset of nodes and then obtain a stable recovery with respect to the graph Laplacian eigenbasis, leveraging ideas from compressed sensing. We also show how such techniques can be used for both temporally and spatially correlated signals sampled by wireless sensor networks. Significant savings are made in terms of energy resources, bandwidth, and query latency by using this approach. All the theoretical analysis and the performance of proposed algorithms are verified using both synthesized data and real world data.

Abrégé

Nous considérons ici un signal dont les éléments sont supportés par les noeuds d'un graphe. Nous étudions les métriques qui mesurent la régularité des signaux supportés par ces graphes and apportons des explications théoriques sur quand et pourquoi les vecteurs propres du Laplacien eigenbasis peuvent être considérés comme une transformation de Fourier significative pour de tels signaux. De plus, nous caractérisons les propriétés souhaitées pour le graphe sous-jacent afin d'obtenir une meilleure compressibilité de ces signaux. Pour un signal régulier par rapport à la topologie du graphe, notre travail prouve que nous pouvons rassembler les mesures d'un sous-ensemble aléatoire de noeuds et obtenir une récupération stable par rapport aux vecteurs propres du Laplacien eigenbasis du graphe. Nous montrons aussi que de telles techniques peuvent être utilisées pour des signaux corrélés à la fois spatialement and temporellement et provenant de réseaux de capteurs. Cette approche apporte des diminutions significatives en terme d'utilisation des ressources énergétiques, de la bande passante et de la latence nécessaire. Toutes les analyses théoriques et les performances des algorithmes proposés sont validées par des simulations et des donnes provenant de systémes existants.

Acknowledgments

First and foremost, I owe my deepest gratitude to Prof. Michael Rabbat, whose sincerity and encouragement I will never forget. His constructive comments and guidance on my research project make this thesis possible while his continuous help and support make my study at McGill University an invaluable experience.

I am also very grateful to all the members in the lab for maintaining an positive working atmosphere. I would like to thank Zhe for reading some of the proofs in this thesis and also his helpful discussion. Moreover, my appreciation also goes to Tao, Deniz, Konstantinos, Santosh, Rizwan, Milad, Rodrigo for their incisive comments and advices on my work. Special thanks go to Benjamin for translating the abstract into French. All the friendships which have developed during this two years at McGill will never be forgotten.

Lastly, I am heartily thankful to my parents for all their support. Without their encouragement and trust, my journey in Canada would never happen. I would also like to thank all my friends I met at McGill for their help in these two years.

Contents

List of Figures

List of Notations

List of Acronyms

i.i.d.	identically and independent distributed
BP	Basis Pursuit
CIMIS	California Irrigation Management Information System
CoSamp	acrostic compressive sampling matching pursuit
CS	Compressed Sensing
CWS	Compressive Wireless Sensing
CWT	Continuous Wavelet Transform
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
FC	Fusion Center
IHT	Iterative Hard Thresholding
IP	Internet Protocol
JL-Lemma	Johnson-Lindenstrauss Lemma
KLT	Karhunen-Loeve Transform
KKT	KarushCKuhnCTucker
KNN	K-Nearest Neighbor
GFT	Graph Fourier Transform
GSCS	Graph Spectral Compressed Sensing
LMMSE	Linear Minimum Mean Square Error
MSE	Minimum Mean Square Error
MP	Matching Pursuit
RAmp	Restricted Amplification Property
RIP	Restricted Isometry Property
SPD	Semi Positive Definite

PCA	Principle Component Analysis
TV	Total Variation
WSN	Wireless Sensor Network

Chapter 1

Introduction

1.1 Motivation

Signals on graphs are now common in various application areas including wireless sensor networks [60], dimension reduction [12] and network monitoring [24]. For example, in field estimation [6, 7], a huge number of wireless sensors are distributed randomly in a field to collect measurements, such as temperature or solar radiation, where the whole sensor network can be modeled as a random geometric graph. In computer graphics, the shape of a 3D object can be approximated by a regular graph, with its nodes containing the coordinate information [15, 39]. In the traditional realm of approximation theory, we are interested in approximating a certain function by a simpler one. So far the approximation theory has focused on 1D signals and 2D images while less work has considered signals on graphs. So a general question one might ask is: how can we approximate signals supported on graphs?

A natural starting point is that of Fourier analysis. It is well known that the Fourier transform plays a core role in approximation theory and the idea that any arbitrary periodic function can be represented as a series of harmonically related sinusoids has a profound impact in mathematical analysis, physics, and engineering. In signal processing, it has been shown that a smooth signal is compressible and can be well approximated by a small portion of its Fourier coefficients because of the compressibility. Conventional approximation theory [41] shows that both the linear approximation error and non-linear approximation error of smooth signals decay fast if we maintain more Fourier coefficients. Moreover, recent developments in Compressed Sensing (CS) [21, 29] also exhibit promising behavior in

approximating smooth signals. Candès et al. [18, 22] and Rudelson & Vershynin [54] show that we can randomly sample the smooth signal with sampling rate far below the Nyquist rate, a stable recovery is still guaranteed, where "stable" means that the signal can be well estimated under small perturbation. If a similar paradigm can be extended to signals supported on graphs, there would be significant improvements in the mentioned applications, especially in Wireless Sensor Networks (WSNs).

1.2 Thesis Problem Statement

Our goal here is to extend the CS paradigm to signals with more general structure, or say, signals supported on graphs. More concretely, our main work can be divided into two very specific questions:

First, can we find a "Fourier transform" for signals supported on graphs and how can we construct such a transform basis? Previous literature has shown that there exist wavelet transforms for signals on graphs while many researchers believe that the graph Laplacian eigenbasis exhibits certain behaviors of the Fourier transform. However, not many theoretical studies have been made to support this belief.

Second, if the first question has a positive answer and since the CS theory tells us that random sampling is an efficient approach for smooth signal approximation, a natural question one would ask next is: is random sampling still an efficient approach for acquiring smooth signals on graphs? The main problem here is whether the sensing matrix generated from such random sampling scheme still satisfies the Restricted Isometry Property (RIP) [29]. If not, can the requirement for RIP be relaxed?

In this thesis, we address both of these questions and provide a direct application of our idea to WSNs.

1.3 Thesis Contribution and Organization

Our main contributions are highlighted as follows:

Regarding the first question, it has been believed for quite a while that the eigenbases of a Laplacian matrix can be deemed as the Fourier basis for its corresponding graph. In this paper, we refer to it as the Graph Fourier Transform (GFT). Moreover, there have already existed certain applications which utilize the GFT in data compression [39, 60],

and signal denoising [57]. However, none of them provides a detailed theoretical analysis on why the graph Laplacian eigenbases can be regarded as the Fourier transform of graphs. Nor do they discuss whether the Laplacian eigenvectors are meaningful basis vectors on all graphs. In this work, we address both these issues. We first generalize the concept of smooth signals and define a metric to measure the smoothness of a graph signal. Later, we derive certain properties of the GFT. Those properties imply that if the eigenvalues of the graph Laplacian roughly maintain an increasing trend, then the smooth signals on that graph are likely to be compressible.

In order to answer the second question, we need first delve into the traditional CS literatures. Candès [22] and Rudelson [54] prove that we can construct a sensing matrix by randomly selecting a small portion of the rows from the Discrete Fourier Transform (DFT) matrix. Actually, we can relax the DFT matrix to any orthogonal matrix whose entries are uniformly bounded, which is called structured random matrix [52]. Our work breaks this constraint by showing that an orthogonal matrix without uniformly bounded entries like certain GFT basis can still guarantee a stable recovery with a simple least square estimator if we construct the underlying graph with its corresponding GFT basis carefully and the signal we are interested in is smooth on that graph. We call this technique "Graph Spectral Compressed Sensing (GSCS)". To distinguish from the technique called spectral compressed sensing [31], it is worth pointing out that our approach is related to the graph spectrum, i.e., the graph Laplacian eigenbasis rather than the eigenbasis of the autocorrelation matrix.

GSCS and the GFT have many applications for networked data processing and gathering. In this thesis, we show that GSCS is a promising technique for Wireless Sensor Networks (WSNs) and the GFT is a suitable orthogonal basis for networked data. Via GSCS, we can gather measurements from a random subset of nodes with irregular sturcture and then interpolate with respect to the GFT basis. We propose algorithms for both temporally and spatially correlated signals, and the performance of these algorithms is verified using both synthesized data and real world data. Significant savings are made in terms of energy resources, bandwidth, and query latency.

1.3.1 Thesis Organization

The rest of the thesis is organized as follows:

In Chapter 2, we provide necessary background to understand how GFT and GSCS work. We first review the approximation theory using the Fourier transform basis and review some properties of the Fourier transform. Second, we review the basic ideas of spectral graph theory and previous applications exploiting the "Fourier" property of the Laplacian eigebasis. Finally, we briefly introduce the main idea of CS and its application on WSNs.

In Chapter 3, the general idea of GFT is introduced. We provide a theoretical analysis of its properties, which are similar to those of the Fourier transform. Discussion on how to obtain a proper GFT basis based on a given signal is also made in this chapter. Moreover, simulations and experiments are made to verify our theoretical analysis.

In Chapter 4, we give the whole idea of GSCS and a performance guarantee has been made for our techniques theoretically. Later on, detailed data gathering algorithms for WSNs with spatially and temporally correlated signals are proposed. Both synthesized and real world data are utilized to verify the theory of GSCS and to evaluate the performance of our approaches for sensor network.

In Chapter 5, a conclusion of our work is made and potential future work is discussed.

1.4 Author's Work

Two papers [67, 68] based on content presented in this thesis will be published in the following international conference proceedings:

- Xiaofan Zhu, Michael Rabbat, "Approximating Signals Supported on Graphs," in *Proc. Intl. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, March 2012.
- Xiaofan Zhu, Michael Rabbat, "Graph Spectral Compressed Sensing for Sensor Networks," in *Proc. Intl. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, March 2012.

Chapter 2

Background and Literature Review

2.1 Approximation Theory on Fourier Basis

In order to study the GFT, we need first delve into the conventional approximation theory and learn the role of the Fourier transform. Fourier expansion was discovered by the study of heat diffusion, which is governed by a linear differential equation [41]. Fourier analysis is the basis for the development of approximation theory and compressed sensing theory. In this section, we make a quick review of the Fourier transform and its corresponding properties for signal approximation. Basics of approximation theory are also included [41].

2.1.1 Approximation Theory Background

In this subsection, we introduce some basic definitions in Fourier analysis. We are interested in a continuous signal $x(t)$, then $\widehat{x(\omega)} = \int_{-\infty}^{+\infty} x(t)e^{-iwt}dt$ denotes its Fourier transform. For the convenience of analysis, it is conventional to model the signal $x(t)$ as square integrable over $[0, 1]$. Then, we can decompose a signal $x(t) = \sum_{m=-\infty}^{+\infty} |\langle x(u), e^{i2\pi mu}\rangle| e^{i2\pi mt}$ with $\langle x(u), e^{i2\pi mu}\rangle = \int_0^1 x(u)e^{-i2\pi mu}du$. $\widehat{x(m)} = \langle x(u), e^{i2\pi mu}\rangle$ is called the Fourier coefficients, which form the "discrete version" of $\widehat{x(\omega)}$. In many applications, the signals do not have fast-varying structure. Such signals are called smooth signals. For smooth signals, it is well known that the coefficients with small m, in other words, the low frequency components tend to dominate the behavior of the whole signal. Hence, we use linear approximation represent the original signal by keeping only those low frequency components: the M-term

Fourier linear approximation is defined as

$$x_M = \sum_{|m|\leq M/2} |\langle x(u), e^{i2\pi mu}\rangle| e^{i2\pi mt}$$

The linear approximation is non-adaptive to signals. Different from the linear approximation, non-linear approximation is adaptive to different signal structure, i.e., it extracts the M largest Fourier coefficients and discards the other, which is shown as follows:

$$\tilde{x}_M = \sum_{m\in\Pi} |\langle x(u), e^{i2\pi mu}\rangle| e^{i2\pi mt},$$

where Π is the set of indices of the M largest Fourier coefficients in magnitude. The distortion of M-term non-linear approximation is less than or equal to that of the M-term linear approximation. However, the downside of this approach is that we need the prior knowledge of the M largest coefficients. Such knowledge might be difficult to obtain in certain applications.

Since "smoothness" is an important concept here, we need to use certain metric to measure it. In conventional approximation theory, total variation is used to describe the overall smoothness of a signal:

Definition 1. For a continuous differentiable function x, the total variation is defined as $\|x\|_V = \int_{-\infty}^{+\infty} |x'(t)|dt$, where $x'(t)$ is the derivative of x. For discrete signals, $\|x\|_V = \sum_n |x(n) - x(n-1)|$. We say that x has a bounded variation if $\|x\|_V < +\infty$.

Total variation measures the overall signal variation. It plays an important role in signal processing since it impacts the decaying behavior of its Fourier coefficients. If a signal $x(t)$ is square integrable over $[0, 1]$, we define the M-term linear approximation error as follows:

Definition 2. M-term Linear Fourier Approximation Error:

$$\epsilon_l(M, x) = \sum_{|m|>M/2} |\langle x(u), e^{i2\pi mu}\rangle|^2$$

.

The linear approximation keeps the M lowest frequency components while discards the rest. It has several important properties related to signal acquisition and compression. Correspondingly, the non-linear approximation is defined as follows:

Definition 3. The *M-term non-linear Fourier Approximation Error* is

$$\epsilon_n(M, x) = \sum_{m \notin \Pi} |\langle x(u), e^{i2\pi mu} \rangle|^2,$$

It is well known [41] that there exist certain relations between the total variation and the behavior of linear approximation error, which are included in the next subsection.

2.1.2 Properties of the Fourier Transform

As has been introduced above, the Fourier transform is a mathematical operation that decomposes a signal into its constituent frequency components and smooth signals are likely to have large lower frequency components while the higher frequency components are small. Hence, we can approximate the original signal using only its low frequency components. This approach is at the heart of many lossy compression techniques. Such properties are expressed in the following theorems in this subsection. We emphasize the importance of understanding them in order to understand the "Fourier" properties of the Graph Fourier Transform(GFT) present in the next Chapter.

Proposition 2.1.1 ([41]). *If $x(t)$ is differentiable and $\widehat{x(\omega)} = \int_{-\infty}^{+\infty} x(t)e^{-iwt}dt$ denotes its Fourier transform, then*

$$|\widehat{x(\omega)}| \leq \frac{\|x\|_V}{|\omega|} \tag{2.1}$$

,

where $\omega \geq 0$. It is worth pointing out that there exist other similar results for the DFT with slightly different definition of the total variation. For example, for the case that $\widehat{x(m)} = \langle x(u), e^{i2\pi mu} \rangle$ and $\|x\|_V = \sup_P \sum_n |x(n) - x(n-1)|$ where P is all the possible partitions over the domain of x, $|\widehat{x(m)}| = O(\frac{\|x\|_V}{|m|})$ [34, 44].

Theorem 2.1.2 ([41]). *If $\|x\|_V < +\infty$, then $\epsilon_l(M, x) = O(\|x\|_V M^{-1})$.*

Theorem 2.1.3 ([41]). *For any $s > 1/2$, there exists some constants $A, B > 0$ such that if $\sum_{m=0}^{+\infty} |m|^{2s} |\langle x, g_m \rangle|^2 < +\infty$ where g_m is the mth vector from an arbitrary orthogonal basis, then*

$$A \sum_{m=0}^{+\infty} |m|^{2s} |\langle x, g_m \rangle|^2 \leq \sum_{N=0}^{+\infty} N^{2s-1} \epsilon_l(M, x) \leq B \sum_{m=0}^{+\infty} |m|^{2s} |\langle x, g_m \rangle|^2$$

and thus $\epsilon_l(M, x) = o(M^{-2s})$.

The theorems above describe the decay rate of Fourier coefficients and the behavior of linear approximation error. It is worth noting that Theorem 2.1.1 is consistent with the fact that a smooth signal is likely to be compressible in the Fourier domain. Theorem 2.1.2 shows that the linear approximation error is upper bounded by total variation and thus signals with small total variation will result in less linear approximation error. Proposition 2.1.3 states that the behavior of the linear approximation error depends on the decay rate of $|\langle x, g_m \rangle|$. In the next chapter, we show that similar statements of all the three theorems above apply to the GFT.

2.1.3 Uncertainty Principle

Time and frequency energy concentrations are restricted by the Heisenberg uncertainty principle. This principle has a particularly important interpretation in quantum mechanics as an uncertainty on the position and momentum of a free particle. Also in the realm of signal processing, uncertainty principle plays an important role in signal sampling and recovery. The idea of compressed sensing is built upon uncertainty principle.

The state of a one-dimensional particle is described by a wave function $x \in L^2(R)$. The probability density of a particle at location t is $\frac{1}{\|x\|^2}|x(t)|^2$, where $\|x\|$ is the 2-norm of x. The probability density of the energy spreading at ω is $\frac{1}{2\pi\|x\|^2}|\widehat{x(\omega)}|^2$. Hence, the average location of the particle is

$$u = \frac{1}{\|x\|^2} \int_{-\infty}^{+\infty} t|x(t)|^2 dt$$

while the average energy spreading is

$$\xi = \frac{1}{2\pi\|x\|^2} \int_{-\infty}^{+\infty} \omega|\widehat{x(\omega)}|^2 d\omega$$

The variances around these average values are

$$\sigma_t^2 = \frac{1}{\|x\|^2} \int_{-\infty}^{+\infty} (t-u)^2|x(t)|^2 dt$$

and

$$\sigma_\omega^2 = \frac{1}{2\pi\|x\|^2} \int_{-\infty}^{+\infty} (\omega-\xi)^2|\widehat{x(\omega)}|^2 d\omega$$

respectively.

Theorem 2.1.4 ([41]). *The temporal variance and the frequency variance of $x \in L^2(R)$ satisfy*

$$\sigma_t^2 \sigma_\omega^2 \geq \frac{1}{4}$$

The inequality is an equality if and only if there exist $(u, \xi, a, b) \in R^2 \times C^2$ such that

$$x(t) = a \exp[i\xi t - b(t-u)^2].$$

In quantum mechanics, this theorem shows that we cannot arbitrarily reduce the uncertainty of the position and the momentum of a free particle simultaneously. In signal processing, this theorem tells us that we cannot determine where the signal locates both in frequency domain and in time domain. In other words, if the signal is concentrated around a certain frequency, it would be spread over the time domain and vice versa. This property plays an important role in signal recovery techniques like sparse signal reconstruction and further contribute to compressed sensing [20, 21]. Interestingly, some researchers [3, 4] recently showed that there also exists an uncertainty principle for signals supported on graphs. Such content will be included in one of the next following sections.

2.2 Compressed Sensing

The development of approximation theory, along with uncertainty principle, contributes to the theory of sparse recovery and finally formed the theoretical basis of compressed sensing. The following subsections introduce the basic theory of Compressed Sensing (CS) and some of its applications.

2.2.1 Compressed Sensing Background

Compressed Sensing, which was first developed by E. Candes, J. Romberg, T. Tao [22] and D. Donoho [29], is a very useful tool to handle sparse or compressible signals. The main motivation for CS is that many real-world signals can be well-approximated by sparse ones, that is, they can be approximated by an expansion in terms of a suitable basis, which has only a few non-vanishing terms. Such idea is the reason why conventional lossy compression techniques such as linear approximation or non-linear approximation perform

so well. However, there exist certain problems in those traditional approaches. First, we spend huge efforts and costs to acquire or say sampling the whole information of the signal and then throw away most of its coefficients to obtain the compressed version. Hence, one might ask if there exists a better way of obtaining the compressed version of the signal directly, which results in a lower sampling rate. Second, it would be difficult for us to sample those large coefficients directly since we do not know the prior knowledge of where the largest ones are. As an alternative, compressed sensing provides a way of obtaining the compressed version of a signal using only a small number of linear and non-adaptive measurements. Even more surprisingly, CS theory proves that recovering the signal from its undersampled measurements can be done with computationally efficient methods, like ℓ_1 programming or greedy methods.

CS theory first considers sparse signals. The sparse signals is defined as follow:

Definition 4. For a signal $x \in \mathbb{R}^N$, we say that x is γ sparse if and only if x has no more than γ non- zero entries, where $\gamma \ll N$

Typically, when we refer to signal x as a γ sparse signal, it means that γ is far less than the dimensionality N and hence "sparse". We call the set of indices corresponding to the nonzero entries the support of x and denote it by $\text{supp}(x)$. The set of all γ sparse signals is the union of the $\binom{N}{\gamma}$ γ-dimensional subspaces aligned with the coordinate axes in $\mathbb{R}^N$.

Suppose that instead of collecting all the coefficients of a vector $x \in \mathbb{R}^N$ where x is a γ-sparse signal, we merely record M inner products (measurements) of x with $M \ll N$ pre-selected vectors which form the rows of a $M \times N$ sensing matrix Φ:

$$y = \Phi x,$$

To recover x from y, one would in fact want to find the sparsest solution of $y = \Phi x$ by solving

$$\min_{x \in \mathbb{R}^N} \|x\|_0 \ s.t. \ y = \Phi x.$$

This is a difficult combinatorial problem. Solving such problem is not realistic for real world applications. Fortunately, it has been proven that if the sensing matrix satisfies the Restricted Isometry Property (RIP) [22, 29], then we can reconstruct the original sparse

signal perfectly by solving the linear program (ℓ_1 decoding):

$$\min_{x\in\mathbb{R}^N}\|x\|_1 \ s.t.\ y=\Phi x.$$

Sometimes the signal x is not sparse directly but have sparse transform coefficients on a certain orthogonal basis Ψ, i.e., $x=\Psi\theta$ where θ is sparse. Then we can still solve the problem via:

$$\min_{\theta}\|\theta\|_1 \ s.t.\ y=\Phi\Psi\theta.$$

and obtain $x=\Psi\theta$. It is worth noting that there exist several other recovery algorithms such as greedy algorithms like Matching Pursuit [30] or CoSamp [45]. The definition of RIP is show as follows:

Definition 5. An $M\times N$ matrix Φ has the $\gamma-$Restricted Isometry Property ($\gamma-$RIP) with constant δ_γ if for all γ sparse signals x, we have

$$(1-\delta_\gamma)\|x\|_2^2\le\|\Phi x\|_2^2\le(1+\delta_\gamma)\|x\|_2^2.$$

The constant δ is called the restricted isometry constant.

The $\gamma-$RIP requires every submatrix of Φ to have a good isometry[1] property and prevents the signal x from lying in the null space of Φ.

Actually, it has been proved [22, 29] that the solutions to the ℓ_1 decoding problem and to ℓ_0 decoding problem are equivalent in the following sense:

1. If $\delta_{2\gamma}<1$, the ℓ_0 problem has a unique sparse solution.
2. If $\delta_{2\gamma}<\sqrt{2}-1$, the solution to the ℓ_1 problem is equivalent to that of the ℓ_0 problem.

Hence, the core idea of CS theory is to determine whether the sensing matrix Φ satisfies the RIP if we are given a signal of sparsity γ.

In other words, perfect recovery of the original sparse signal under the conventional CS paradigm is based on two conditions:

1. The signal should be sparse.

[1] isometry means distance-preserving maps

2. The sensing matrix should satisfy the RIP.

However, signals encountered in nature are not always sparse. Even though many natural and manmade signals are not strictly sparse, but can be approximated as such; Such signals are called compressible signals.

Definition 6. Consider a signal x whose coefficients, when sorted in order of decreasing magnitude, decay according to a power law:

$$|x_{(i)}| \leq Si^{-1/r}$$

for some constant $r > 0$, where $i = 1, 2, \cdots, N$ and $x_{(i)}$ is the ith largest coefficients.

Thanks to the rapid decay of their coefficients, such signals are well-approximated by sparse signals. For compressible signals, the non-linear approximation error can be bounded as:

$$\epsilon_n(\gamma, x) \leq (rs)^{-1/2} S\gamma^{-s}$$

with $s = \frac{1}{r} - \frac{1}{2}$ and the γ term non-linear approximation means we merely maintain the largest γ coefficients of the original signal x. This upper bound implies that for compressible signals, the signal's best approximation error has a power-law decay with exponent s as γ increases. Hence, such signals can be referred as $s-$compressible signals.

One great achievement in the area of CS is proven by E. Candes et al. [19] that CS can be applied to more applicable situations where there is no specific constraint on the sparsity of a signal and the measurements are corrupted by noise. More specifically, we observe:

$$y = \Phi x + z,$$

where z is an unknown noise term and the ℓ_1 decoding problem can be slightly modified as follows:

$$\min_x \|x\|_1 \ s.t. \ \|y - \Phi x\|_2 \leq \epsilon, \tag{2.2}$$

where ϵ is an upper bound which is determined by the impact of noise. E. Candes [18] has proved a stable recovery of the above recovery algorithm:

Theorem 2.2.1. *Assume that $\delta_{2\gamma} < \sqrt{2} - 1$ and $\|z\|_2 \leq \epsilon$. Then the solution to (2.2) obeys*

$$\|x^* - x\|_2 \leq C_0 \gamma^{-1/2} \|x - \tilde{x}_\gamma\|_1 + C_1 \epsilon,$$

where C_0 and C_1 is some constants and x^* is the solution to the ℓ_1 decoding problem.

It is worth noting that if the signal x is γ sparse, then the above theorem reduces to the perfect recovery conclusion of sparse signals. Also, it is straightforward to see that if the original signal x is compressible, then the recovery upper bound $\|x - \tilde{x}_\gamma\|_1$ will be small with an adequately large γ. It is also worth pointing out that the ℓ_1 programming is one standard recovery algorithm but not the only one. There are lots of CS recovery algorithms such as Iterative Hard Thresholding (IHT) [16], subspace pursuit [28], Matching pursuit [30], CoSamp [45], etc. Thanks to RIP, all of them provide a robust and stable recovery of compressible signals. Hence, the remaining problem of CS is which sensing matrices satisfy the RIP.

Random Matrices: Random matrices are commonly utilized sensing matrices in CS. We generate such matrices by drawing each entry of the matrix from a i.i.d. Random Variables such as Gaussian or Bernoulli. Such matrix Φ are proved to satisfy the concentration inequality shown below:

$$Pr(|\|\Phi x\|_2^2 - \|x\|_2^2| \geq \epsilon \|x\|_2^2) \leq 2e^{c_0(\epsilon)},$$

where $0 < \epsilon < 1$ and $c_0(\epsilon)$ is some constant only related to ϵ. For such matrices, it has been shown that if we have an adequate number of rows, the RIP is satisfied with an overwhelming probability. More specifically, the following theorem [52] provides us with a lower bound for the number of measurements for random noisy matrices:

Theorem 2.2.2 ([52])**.** *Let $\Phi \in \mathbb{R}^{M \times N}$ be a Gaussian or Bernoulli random matrix. Let $\epsilon, \delta \in (0, 1)$ and assume that*

$$M \geq C\delta^{-2}(\gamma \ln(N/\gamma) + \ln(\epsilon^{-1}))$$

for a constant $C > 0$. Then, with probability at least $1 - \epsilon$ the restricted isometry constant for matrix Φ satisfies $\delta_\gamma \leq \delta$.

This theorem, combined with earlier introduced results about stable recovery, states that if such noisy random matrices have more than $M = O(\gamma \ln(N/\gamma))$ number of measurements, then a robust and stable recovery of CS is guaranteed with overwhelming probability. This is the statement usually found in the literature. There exist several proofs for the above

theorem. In [9] a particularly nice and simple proof is given, which, however, yields an additional $\ln(\delta^{-1})$ term.

Sparse Random Matrices: Further development of random sensing matrices has been made for certain specific applications like wireless sensor networks. Computing such matrices and gathering the measurements in a distributed setting would be expensive. One solution for such problems is given by sparse random matrices [62]. For sparse random projection, we set the sensing matrix as:

$$\Phi_{ij} = \begin{cases} 1 & : \text{ with prob. } \frac{1}{2s} \\ 0 & : \text{ with prob. } 1 - \frac{1}{s} \\ -1 & : \text{ with prob. } \frac{1}{2s} \end{cases}$$

The parameter s controls the degree of sparsity of the random projections. Thus if $s = 1$, the random matrix has no sparsity; and if $s = \frac{\ln N}{N}$, the expected number of non zeros in each row of the random matrix is $O(\ln N)$. Wang et al. [62] show that $O(\gamma^2 \ln N)$ sparse random projections are sufficient to recover a data approximation which is comparable to the optimal γ-term approximation, with high probability. The expected degree of sparsity, or say, the average number of nonzeros in each random projection vector, is $O(\ln N)$.

Structured Random Matrices: Although the random sensing matrices ensure sparse recovery via ℓ_1 decoding, sometimes they are of limited use in real applications. Often the design of the measurement matrix is subject to physical or other constraints due to the applications, or it is actually given to us without having the freedom to alter its design, and therefore it is quite likely that the matrix does not follow a Gaussian or Bernoulli distribution. Moreover, Gaussian or other unstructured matrices have the disadvantage that no fast matrix multiplication algorithm is available and storing an unstructured matrix may be difficult. Hence, several papers [8, 32, 51] focus on the construction of structure random matrices and their recovery guarantee.

In this thesis, we would like to focus on one type of structured random matrices. This type of random matrix is generated by randomly selecting a portion of the rows of an orthogonal matrix who has a bounded coherence. The coherence of a matrix is a metric for measuring the capability of being a good sensing matrix. A smaller coherence tends to have a better RIP [22, 52, 54].

For some cases that the signal is compressible or sparse on a certain orthogonal domain,

i.e.,

$$y = \Phi\Psi\theta$$

, where $x = \Psi\theta$ and Φ is an orthogonal measurement system, i.e., $\Phi^*\Phi = I$. In order to evaluate the mutual orthogonality of the matrix Φ and the orthogonal basis Ψ, the coherence is defined as:

Definition 7. The mutual coherence of Φ and Ψ is

$$\mu(\Phi, \Psi) = \max_{j,k} |\langle \Phi_j, \Psi_k \rangle|,$$

Coherence is a classical way to measure the quality of a measurement matrix with normalized columns. If the coherence is small, then the columns of the sensing matrix are almost mutually orthogonal. A small coherence is desired in order to have good sparse recovery properties. One direct example for mutual coherence is the partial Fourier ensemble, which is defined as randomly selecting a portion of the rows from the DFT basis. In this case Ψ is the DFT basis and Φ is a random row submatrix of an identity matrix. Hence the mutual coherence of the partial Fourier ensemble is the largest magnitude of the entries of the DFT basis, i.e., $\frac{1}{\sqrt{N}}$. Generally speaking, the smaller the coherence is, the less number of measurements we need for the recovery process [22, 52].

The partial Fourier basis is a special case of structured random matrices. More generally, CS theory is concerned with the matrices with the following properties [54]:

1. $\Phi\Psi$ is orthogonal

2. The magnitude of the entries of matrix $\Phi\Psi$ is bounded with $O(\frac{1}{\sqrt{N}})$

In other words, the mutual coherence for structured random matrices like DFT basis can be represented as its largest magnitude of the entries. This concept is closely related to our work since in contrast to conventional sparse approximation theories, we consider random matrices without a bounded magnitude of entries.

Previous results in CS theory tell us that with random sensing matrices that satisfy the above two conditions, there are two types of recovery approaches: the uniform recovery and the nonuniform recovery. A uniform recovery guarantee means that once the random matrix is chosen, then with high probability all sparse signals can be recovered. A nonuniform recovery guarantee states that only a sparse signal with fixed but arbitrary support can

be recovered with high probability using a random structure matrix. There are several works [22,52,54] discussing about the uniform recovery. And the best result so far has been given in [54] which states that a uniform recovery of structured random matrices requires $M \geq C\gamma \ln^4 N$ number of measurements while it was widely believed that the actual lower bound should be $C\gamma \ln^\alpha N$ with $\alpha = 1$ or 2. For nonuniform recovery, the bound for the number of measurements has been achieved in [18], which says that $M \geq C \cdot \gamma \ln N$ where C is some constant.

This subsection introduced the standard CS theory. However, for many real world applications, there are special situations where we do not require the RIP be satisfied. In other words, RIP is a sufficient condition for CS recovery but not necessary. The next subsection introduces a concept called model based CS that parallels the conventional theory and provides concrete guidelines on how to create model-based recovery algorithms with provable performance guarantees.

2.2.2 Model-based Compressed Sensing

Although many natural and manmade signals can be modeled as compressible or sparse signals, some of them tend to have support of their coefficients with underlying inter-dependencies. For example, block sparsity [17, 59] deals with the scenario that the non zero coefficients of a signal form clusters. The Model-based CS theory take advantage of such prior knowledge and hence outperforms the conventional CS recovery algorithm in two aspects: First, the required number of measurements for recovery is reduced; Second, model based CS recovery algorithms better recovers the original signal from limited signal spaces. We introduce Model-based CS since the theoretical analysis of our proposed method works under the framework of Model-based CS.

Model-based CS relies greatly on the structure of the coefficients support. We denote the support set by T, where T is a subset of $\{1, 2, \cdots, N\}$ with N the signal dimension. Let T^c denote the complement of the set T. In [10], in order to provide a general model to include the structured signal ensemble, the signal model M_γ is defined as:

$$M_\gamma = \bigcup_{\gamma=1}^{m_\gamma} X_m, X_m = \{x : x_{T_{m_\gamma}} \in R^\gamma, x_{T^c_{m_\gamma}} = 0\}$$

where m_γ is the number of the possible support set and γ represents the sparsity. $x_{T_{m_\gamma}}$

corresponds to a γ dimensional signal whose entries are extracted from the support T_{m_γ} on x. Thus the model M_γ is defined by the set of possible supports $\{T_1, \cdots, T_{m_\gamma}\}$. Clearly, $m_\gamma \leq \binom{N}{\gamma}$. Correspondingly, there is a RIP defined for signals desirable by this model:

Definition 8. An $M \times N$ matrix Φ satisfies the $M_\gamma-$Restricted Isometry Property ($M_\gamma-$RIP) with constant δ_{M_γ} if for all $x \in M_\gamma$, we have

$$(1 - \delta_{M_\gamma})\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta_{M_\gamma})\|x\|_2^2.$$

It is straightforward to see that the model based RIP is a weaker condition than the conventional RIP because it only applies to the signals $x \in M_\gamma$, where M_γ is a subset of all possible $\binom{N}{\gamma}$ subspaces. [10] proved that the model sparse signals can be stably recovered with random sensing matrices while it requires fewer measurements compared with conventional CS decoder.

As in the case of conventional CS theory, sparse signals cannot fit a lot of applications in real world while compressible signals are more realistic. In model based CS, Baraniuk et al. [10] define the model compressible signals as below:

Definition 9. The set of $s-$model-compressible signals is defined as:

$$\mathbb{M}_s = \{x \in \mathbb{R}^N : \epsilon_n(\gamma, x) \leq S\gamma^{-s}, 1 \leq \gamma \leq N, S < \infty\},$$

where $\epsilon_n(\gamma, x) = \inf_{\bar{x} \in M_\gamma} \|x - \bar{x}\|_2$.

Positive results show that for exactly γ-model-sparse signals, we can perfectly recover them with the help of model based RIP. Also, due to the smaller range of the possible subspaces, the number of measurements can be significantly reduced, model-sparse concepts and results do not immediately extend to model-compressible signals. This is because the model based RIP merely deals with signals whose non-zero coefficients lie in M_γ while it cannot cope with compressible signals. Hence, it is necessary to develop a generalization of the M_γ-RIP that can be used to quantify the stability of recovery for model compressible signals. Before giving a detailed description about this generalized version of RIP, we first need to define the residual subspaces $R_{j,\gamma}$ as $R_{j,\gamma} = \{u \in \mathbb{R}^N$ such that $u = M(x, j\gamma) - M(x, (j-1)\gamma)\}$ for $j = 1, 2, \cdots, \lceil N/\gamma \rceil$, where $M(x, j\gamma) = \underset{\bar{x} \in M_{j\gamma}}{\operatorname{argmin}} \|x - \bar{x}\|_2$. Then the Restricted Amplification Property is defined as follows:

Definition 10. A matrix Φ has the $(\epsilon_\gamma, r)-$ restricted amplification property(RAmP) for the residual subspaces $R_{j,\gamma}$ if

$$\|\Phi u\|_2^2 \leq (1+\epsilon_\gamma) j^{2r} \|u\|_2^2$$

for any $u \in R_{j,\gamma}$ where $1 \leq j \leq \lceil N/\gamma \rceil$.

It is easy to see that if $r = 0$, then the RAmP is no different with the upper bound of the RIP. RAmP can be utilized to measure the property of the tail bound of x. One way to analyze the stability of compressible signal recovery in conventional CS is to consider the tail of the signal outside its γ-term non-linear approximation as contributing additional "noise" to the measurements of size $\|\Phi(x - \tilde{x}_\gamma)\|_2$ where $\tilde{x}_\gamma$ is the best $\gamma-$ term non linear approximation of x. This technique can also be used to quantify the stability of model-compressible signal recovery. The key quantity that must be controlled is the amplification of the model-based approximation residual through the sensing matrix Φ since the signal energy in residual space is the tail of this signal and can be regard as "noise". In [10], the tail of a model compressible signal outside the γ-term approximation $M(x, \gamma)$, i.e., $\|\Phi(x - M(x,\gamma))\|_2$, is proved to be upper bounded by $C\sqrt{1+\epsilon_\gamma} S \gamma^{-s} \ln\lceil \frac{N}{\gamma} \rceil$ with the help of RAmP and model compressible signal. Since $\|\Phi(x - M(x,\gamma))\|_2$ is small, the robust recovery of the model based algorithm can be easily verified [10].

Model-based CS recovery algorithms are mainly based on conventional CS recovery algorithms like CoSamp or IHT. These algorithms iteratively search for the best support for the signals and recover the magnitudes via a minimum MSE estimator. The way they detect the best support is based on techniques for finding the best γ term non linear approximation while the model based CS recovery algorithm merely replaces the best γ term approximation with the best γ term model-based approximation $M(x, \gamma)$. Since M_γ is far less than $\binom{N}{\gamma}$, fewer measurements will be required for the same degree of robust signal recovery. Alternatively, using the same number of measurements, more accurate recovery can be achieved. Moreover, a performance bound has been provided for model-based CS recovery algorithm [10].

The model based CS is closely related to our work since the theoretical analysis we utilize is based on the model-based CS framework. The scenario we focus on is an extreme condition of model based CS where $m_\gamma = 1$, i.e., we know exactly where the coefficients with largest magnitude are and the results in model based CS can also be applied to our

case.

2.2.3 Compressed Sensing for Sensor Networks

A Wireless Sensor Networks (WSN) [5] contains a number of self-organized wireless sensors that cooperate with each other for conducting the same tasks. WSNs have a promising capability to monitor the physical world via their spatially distributed sensor nodes. Since WSN is an attractive low-cost technology for a wide range of remote sensing and environmental monitoring applications, the development of method to estimate the parameters of the underlying signals has become an exceedingly hot research area [46, 53, 53].

Prolonging the lifetime of a WSN is important for both commercial and tactical applications. This is because wireless sensors contain non-rechargeable batteries, which place stringent energy constraints on the design of all WSN operations. In addition, bandwidth resources are also limited for wireless sensor network. We always want to design a WSN algorithms that consume as less bandwidth resources as possible. All these present requirements create formidable challenges upon the design of communication, networking, and local signal processing algorithms performed by a WSN. Lots of effort has been made for designing energy efficient estimation algorithms for WSNs [56, 63].

Fortunately, the conventional CS theory can form the basis of promising methods for achieving best tradeoffs between energy and bandwidth resources. There are two main advantages of applying CS to distributed estimation tasks:

1. Compressed Sensing requires far fewer observations than the number of sensor nodes.

2. Compressed Sensing is a general idea that can be applied on any data or parameters as long as the measuring signal is sparse or compressible on a certain domain. This means the signal itself does not have to be sparse on space domain.

The first advantage about CS motivates the application of CS on distributed estimation since fewer observations lead to less energy consumption or less bandwidth requirements.The second claim gives rise to a wider area of application of CS on distributed estimation problems: we do not have to utilize CS on sparse parameters estimation but can apply it as long as we can find an orthonormal basis where the data can be sparsely represented. Thus, one core problem for designing the distributed estimation system with CS is how to find such a basis. In the past few years, researchers have developed several

techniques of applying CS to sensor networks [6, 23, 33, 38, 50]. One related work which deals with this problem is Compressive Wireless Sensing (CWS) [6].

Compressive Wireless Sensing

Bajwa et al. [6, 7] proposed a distributed matched source-channel communication scheme for field estimation. Their method, which is based on theory of CS, estimates the sensed data at the Fusion Center (FC) and analyzes the tradeoffs between energy, distortion and latency(bandwidth). Their method is based on a similar philosophy rooted in image processing: They regard each sensor as a single pixel. If we have the prior knowledge of the orthogonal basis where the target signal is sparse and the subspace where the sparse parameter lives, then it is feasible to utilize the conventional image processing scheme like JPEG [61] to encode the signal and reconstruct it at FC. The proposed approach is based on analog scheme but only needs M unit of bandwidth resources, i.e., M different frequency channels are allocated for each measurement. M is far less than the size of the WSN N. However, it is not always practical for us to have such prior knowledge about the optimal subspace where the signal is. In order to deal with this situation, a universal scheme called Compressive Wireless Sensing (CWS) is proposed:

Instead of projecting the sensor network data onto a subset of deterministic orthogonal basis (like JPEG), the FC tries to estimate a parameter x from noisy random projections of the sensor network data. Specifically, let each sensor node multiply its readings with a random variable and gather their sum at the FC. Repeat such process for M times and we will obtain the M dimensional measurement vector y at FC. Such process can be represented as following:

$$y = \Phi(x + w) + \tilde{n}$$

where x is a $M \times 1$ observation vector, w is the sampling noise of i.i.d zero mean Gaussian distributed with variance σ_w^2 and $\tilde{n}$ is the channel noise of i.i.d zero mean Gaussian distributed with variance σ_n^2. Because the entries of the projection matrix(compression matrix, sensing matrix) Φ are generated at random, observations of this form are called random projections of the signal. The above mathematical model can be further simplified as:

$$y = \Phi x + (\Phi w + \tilde{n}) = \Phi x + n$$

where $n = \Phi w + \tilde{n}$. It has been proved [37] that the above model is equivalent to the original model and n can be regarded as a noise term which is independent with Φ. Given a countable collection χ of candidate reconstruction signals, such that $|x_i| \leq B$ for all entries, the estimate of original signal x, $\hat{x}$, is obtained as a solution of

$$\hat{x} = \underset{x \in \chi}{\operatorname{argmin}} \{ \|y - \Phi x\|_2^2 + \frac{c(x) \log 2}{\epsilon} \}$$

where $c(x)$ is a non-negative number assigned to each $x \in \chi$ and $\epsilon > 0$ is a constant that depends on the function bound B and the noise variance. Moreover, if we can find a deterministic basis Ψ where x is compressible or sparse, then we can use Ψ in the estimator and rewrite the estimator as:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \{ \frac{1}{M} \|y - \Phi\Psi\theta\|_2^2 + \frac{2 \log(2) \log \gamma}{\epsilon} \|\theta\|_0 \},$$

and $\hat{x} = \Psi\hat{\theta}$.

CWS manages to reduce the number of measurements, which means that it requires less number of bandwidth resources or query latency (if we utilize a TDMA scheme). It applies similar ideas in image processing, i.e., projecting signal onto proper orthogonal basis, to obtain compressible coefficients and exploit CS for measurements reduction. No prior knowledge about the location of the transformed coefficients θ is required. However, there are still some problems in the CWS framework: First, collecting one CS measurement requires the participation of every sensor and to obtain an M dimensional measurement vector requires each sensor transmit its readings M times to the FC. The total number of transmission is MN , which might not be quite energy efficient. Second, in their experiments, they utilize some conventional orthogonal basis in image processing, e.g., wavelet, Haar transform. Such orthogonal basis only applies to regular structured, or say, a 2D grid. However, the topologies of many WSNs do not have such property. Later in this thesis, we will show how our work deals with these issues.

WSN Monitoring via Compressed Sensing

The key idea in CWS is to make use of the spatial correlation of the parameter vector to reduce required number of projections (observation). Another interesting idea [42, 43] of

utilizing compressed sensing for distributed estimation is motivated by the temporal correlation between desired signals. They exploit the Karhunen-Loeve Transform (KLT) basis to obtain compressible coefficients and propose an online algorithm for signal recovery. The general idea of [42, 43] is introduced as follows:

The estimator at the FC utilizes the recent r estimations to help estimate the current readings for each iteration since the signal is temporally correlated. Hence, this method is an online estimation scheme. Consider that there exist N wireless sensors monitoring some underlying signals (e.g. temperature, humidity) for a spatial area. Let $x_t(i)$ where $i = 1, \cdots, N$ denote the sampled data by sensor i at time t. Accordingly, x_t is a $N \times 1$ vector. Also, $\bar{x} = \frac{1}{r}\sum_{k=t-r}^{t-1} x_k$ is the sample mean vector and $C = \frac{1}{r}\sum_{k=t-r}^{t-1}(x_k - \bar{x})(x_k - \bar{x})^T$ is the sample covariance matrix. Via the basic theory of linear algebra, we can calculate an orthonormal matrix U whose columns are the unitary eigenvectors of the covariance matrix C. It is now possible to project a given measurement x_t onto the vector space spanned by the columns of U. Now, let $\theta_t = U^T(x_t - \bar{x})$ and reorder the entries of θ_t as follows: $\theta_t(1) \geq \theta_t(2) \geq \cdots \geq \theta_t(N)$. Then θ_t is the KLT of the signal x_t. Since they assumed that the signal x is temporally correlated, there exists an $\gamma \ll N$ such that when $i > \gamma$, $\theta_t(i)$ is negligible compared to the largest γ entries. Thus we can say that it is very likely that θ_t is compressible or it is γ sparse.

In this framework, instead of transmitting all N observations to the FC, the wireless sensor network randomly chooses M sensors to send their sampled data to FC. Thus the observations received by FC can be represented as: $y_t = I_\Omega x_t$, where x is a $M \times 1$ observation vector and I_Ω is a random row submatrix of an $N \times N$ identity matrix.

Before delving into the detailed procedure of this online estimation algorithm, we need first to clarify its assumptions: the FC has the perfect knowledge of the past r samples, i.e., the FC knows the signal set $\{x_{t-1}, \cdots, x_{t-r}\}$. The parameter r is chosen according to the temporal correlation of the observed phenomena to validate this assumption. Thus the procedure of this estimator is as follows:

First, the wireless sensor network transmits its sampled version of x_t, i.e., $y_t = I_\Omega x_t$ to the FC. From the equation $\theta_t = U^T(x_t - \bar{x})$, we can see that

$$y_t = I_\Omega(\bar{x} + U\theta_t) = I_\Omega\bar{x} + I_\Omega U\theta_t = I_\Omega\bar{x} + U_\Omega\theta_t$$

, where $U_\Omega = I_\Omega U$ is the sensing matrix in Compressed sensing framework.

Second, when the FC obtains the observation y_t, it can obtain the sensing matrix Φ by distinguishing which sensors have been activated. And since the previous r readings is known, $\bar{x}$ is known to the FC. Then the FC can obtain

$$Y_t = y_t - I_\Omega \bar{x} = I_\Omega \bar{x} + U_\Omega \theta_t - I_\Omega \bar{x} = U_\Omega \theta_t$$

. where $U_\Omega = I_\Omega U$ and we calculate U via the previous r readings as discussed.

Since we have the prior knowledge that θ_t is a sparse or compressible signal, we can obtain the estimation according to the framework of compressed sensing:

$$\hat{\theta}_t = \text{argmin}_{\theta_t} \|\theta_t\|_1 \quad s.t. \quad Y_t = U_\Omega \theta_t$$

Finally, applying the following calculation: $\hat{x_t} = \bar{x} + U\hat{\theta}_t$, we can get the final estimate of the underlying parameter and update the stored previous r readings.

In this work, Masiero et al. [42, 43] use experiments to illustrate the performance of their algorithms. However, still certain issues are unclear in their papers. They did not provide the proof that U_Ω can be a valid sensing matrix, neither do they discuss the required number of measurements. Actually, our work adopts similar online algorithm on temporally correlated signals but ours utilized the Laplacian eigenbasis rather than KLT basis and hence is able to provide more detailed theoretical discussion.

2.3 Spectral Analysis on Graphs

2.3.1 Spectral Graph Theory Basics

In mathematics, spectral graph theory [58] is the study of properties of a graph based on the characteristic polynomial, eigenvalues, and eigenvectors of matrices associated to the graph, such as its adjacency matrix or Laplacian matrix.

In spectral graph theory or other graph theories, a graph $G = (V, E, w)$ can be well specified by its vertex set, V , edge set E and the weight set defined on edges. For unweighted graphs, the definition can be reduced to $G = (V, E)$. In an undirected graph, the edge set $E = \{(i, j) : i \sim j\}$ is a set of unordered pairs of vertices while in an directed graph, the set of pairs of vertices is ordered. In this thesis, we focus on undirected graphs.

Unless otherwise specified, all graphs will be undirected, and finite.

Typically, and without loss of generality, we will assume that $V = \{1, \cdots, N\}$. One natural matrix to associate with a graph G is its adjacency matrix, A, since A is able to contain all the topology information. For simplicity, we will just use A to denote the adjacency matrix. The weighted adjacency matrix A of G is the $N \times N$ matrix with entries

$$A_{i,j} = \begin{cases} w_{i,j} & : \text{ if there is an edge between vertix } i,\ j \\ 0 & : \text{ otherwise} \end{cases}$$

and $N = |V|$ is the number of nodes. If $w_{i,j} \in \{0, 1\}$, then A reduces to an unweighted adjacency matrix. Another related matrix is the Laplacian matrix. To construct this, let D be the diagonal matrix in which $D(i, i)$ is the degree of vertex i. The degree of vertex i is defined as the number of edges which is connected to i for undirected graphs while for directed graphs, we only count the outgoing edges. We have:

$$D(i, i) = \sum_j A_{i,j}$$

The quadratic form associated with a graph is defined in terms of its Laplacian matrix:

$$L = D - A$$

Many elementary properties of the Laplacian follow from this definition. In particular, it is immediate that for all x whose entries are supported on the nodes of a graph

$$x^T L x = \sum_{(i,j) \in E} w_{i,j}(x(i) - x(j))^2 \geq 0.$$

From the above equation, we can see that L is SPD. If we let u denote one eigenvector of the Laplacian matrix L and λ as its corresponding eigenvalue, then we have:

$$u_i^T L u_i = \lambda_i u_i^T u_i$$

Since L is a symmetric matrix for undirected graphs, λ is real and non negative because L is Semi Positive Definite (SPD). There are certain basic properties of spectrum of the Laplacian matrix:

1. Observe that $Lx = 0$ for $x(i) = c$ with $i = 1, \cdots, N$ is a constant vector. Hence we can see that the smallest eigenvalue of a Laplacian matrix is 0.

2. We say that a graph G is connected if for any pair of nodes in the graph, there always exists a path between them. Let $0 = \lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{N-1}$ be the eigenvalues of the Laplacian matrix. Then $\lambda_1 > 0$ if and only if G is connected.

3. $\lambda_0 = \min_x \frac{x^T L x}{x^T x}$ and $\lambda_{N-1} = \max_x \frac{x^T L x}{x^T x}$. The ratio $\frac{x^T L x}{x^T x}$ is called the Rayleigh quotient.

Accordingly, from the first two properties we can see that the eigenvalues of the Laplacian matrix maintain a non-decreasing trend starting from 0. And for a connected graph, the multiplicity of the 0 eigenvalue is 1 and its corresponding eigenvector is a constant vector. In Chapter 3, we will see that the Rayleigh quotient of the Laplacian matrix is closely related to the smoothness of the signals supported on graphs.

2.3.2 Graph Laplacian Eigenbasis

If we let $U = [u_0, \cdots, u_{N-1}]$, where u_i is the eigenvector of a Laplacian matrix, which corresponds to λ_i. We call U the Laplacian eigenbasis. The graph Laplacian eigenbasis has long been exploited by the computer science society for machine learning problems such as regression, classification, clustering [14, 55], and especially for semi-supervised learning problems [11, 65]. It has also be utilized for dimension reduction techniques in Laplacian eigenmaps [12]. Moreover, some of the researchers in the area of computer graphics utilized the methodology of signal processing and utilized graph Laplacian eigenbasis as compression techniques for 3D objects [39]. This subsection will provide brief introductions on some of the related works.

Spectral Compression

One idea in the area of computer science refers to the idea of image compression techniques in signal processing is called spectral compression. Karni and Gotsman [39] show how spectral methods may be applied to 3D mesh data to obtain compact representations. This is achieved by projecting the mesh geometry onto an orthonormal basis derived from the mesh topology.

More specifically, in image compression techniques like JPEG [61], we deal with 2D images and project the image signal onto the DCT domain and only maintain the low

frequency components. Correspondingly, in spectral compression, we deal with 3D mesh data. The core idea of spectral compression is quite simple: it utilizes an efficient algorithm to compute the $N \times N$ Laplacian eigenbasis and the authors claim that the eigenvectors u_i can be regarded as "low frequency" if $i \ll N$. Then, they project the coordinate data onto the Laplacian eigenbasis while only remain the "low frequency" components of the coefficients. In their work, they claim that the the graph Laplacian eigenbasis has certain "Fourier" properties but they did not provide adequate analysis. Later on, the work [15] shows a theoretical proof on the optimality of spectral compression. However, this conclusion is restricted to the following two conditions:

1. The coordinate data of each node conforms to a strictly sorted order, i.e., along the x-axis, the x coordinates of the nodes always keep increasing and the same happens for y, z-axis.

2. The degree of each node is equal to 4.

Such strict requirements make the conclusion [15] difficult to be extended to more general situations. In [39], it is mentioned that the Laplacian eigenbasis can be regarded as having certain "Fourier" properties although the authors did not delve into this topic any further. One natural question would be: Does such behavior exist for more general topology structures? This question has been studied later in our thesis.

Manifold Embedding with Eigenmaps

In machine learning, dimension reduction is the process of reducing the number of random variables under consideration, and can be divided into feature selection and feature extraction. Principle Component Analysis (PCA) or random projection are common dimension reduction techniques. Moreover, there exists other prominent nonlinear techniques that include manifold learning techniques such as Eigenmaps [12]. In [12], Belkin and Niyogi present a new algorithm and a methodology of theoretical analysis for their geometrically motivated dimensionality reduction.

The general process of the algorithm can be described as following:

1. Construct the adjacency graph based on the data points via an $\epsilon-$graph or KNN graph, i.e., connecting those nodes who are close to each other

2. Choose the weights on the edges. The author suggests the use of a gaussian kernel or unweighted graphs.

3. Compute eigenvalues and eigenvectors for the generalized eigenvector problem: $Lu = \lambda Du$, where D is the diagonal matrix corresponds to the node degrees and obtain the M-dimensional embedded data as $(u_1, \cdots, u_M)$

The solution reflects the intrinsic geometric structure of the manifold. The justification comes from the role of the Laplacian operator in providing an optimal embedding.

It has been shown that the Laplacian of the graph obtained from the data points may be viewed as an approximation to the Laplacian Beltrami operator defined on the manifold and the Laplacian Beltrami operator is suitable to preserve the locality by trying to find

$$\underset{\|x\|=1}{\operatorname{argmin}} \int_{\mathscr{M}} \|\Delta x\|^2,$$

where Δ denotes the Laplacian Beltrami operator. The Laplacian Beltrami operator, like the Laplacian, is the divergence of the gradient for the underlying manifold. Hence, this optimization problem corresponds directly to minimizing $\sum_{(i,j)\in E} w_{ij}(x(i) - x(j))^2$ on a graph, which tries to maintain the smoothness of the low dimensional signal. In our work, we show that the "smoothness" with regard to the graph here has further meanings under the framework of signal processing and is worthy deeper study. The algorithm of Eigenmaps is simple and easy to implement. M. Belkin [11] further shows that the core idea of the eigenmaps can be applied to semi-supervised learning.

Manifold Structure for Semi-Supervised Classification

In computer science, semi-supervised learning is a class of machine learning techniques that make use of both labeled and unlabeled data for training a small amount of labeled data with a large amount of unlabeled data. Based on similar ideas developed in eigenmaps, the classification techniques are developed [11] under the assumption that the data resides on a low dimensional manifold within a high dimensional representation space. The technique utilizes both the labeled data and unlabeled data for better performance on classification.

The procedure of the classification is quite similar to that of the eigenmaps: Consider N points x_i where $i = 1, 2, \cdots, N$ with only γ points with binary labels c_i, where $c_i \in \{1, -1\}$

1. Constructing an adjacency graph based the data points via an $\epsilon-$graph or KNN graph.

2. Compute the eigenbasis U of the Laplacian matrix of the graph.

3. Build the classifier and obtain the M dimensional classifier parameter θ by minimizing the error function: $E(\theta) = \sum_i^{\gamma}(c_i - \sum_{j=0}^{M-1}\theta_j u_j(i))^2$, where u_i is the ith eigenvector of the Laplacian matrix and M is some constant smaller than N.

4. Classify the unlabeled data by:

$$c_i = \begin{cases} 1 & : \text{ if } \sum_{i=0}^{M-1}\theta_j u_j(i) > 0 \\ -1 & : \text{ otherwise} \end{cases}$$

Their main theoretical support for the method is that the Laplacian can be regarded as a smoothness function. If we denote $\mathscr{M}$ as a manifold and a smoothness function is defined as $S(x) = \int_{\mathscr{M}} \|\Delta x\|^2$. Hence, it is easy to see that for eigenfunction u_i, its smoothness function is λ_i and by keeping the first M eigenfunction components, the smoothness of the approximation is well maintained. Moreover, the authors argue in their paper that the Laplacian matrix of the graph can be regarded as the discrete version of Beltrami operator.

Overall, such method makes use of the smoothness of the manifold function, which can be inferred by the unlabeled data, and the information of the labeled data in order to improve the performance of the classification task. It is worth pointing out that for semi-supervised learning, an alternative classifier is proposed by following similar ideas:

$$\min_{x(i)\in R} \sum_i^N w_{ij}(x(i) - x(j))^2 \ s.t.\ x(i) = c_i \text{ for all labels}$$

The above optimization problem also preserves the smoothness of the function x since $x^T Lx$ is the discrete version of the smoothness function $S(x)$.

On the other hand, there are still certain unresolved issues in [11]. The authors do not provide further discussion about the relationship between error rate and the choice of M. Moreover, they build the classifier via minimizing $E(\theta) = \sum_i^{\gamma}(c_i - \sum_{j=0}^{M-1}\theta_j u_j(i))$, which is merely based on the intuition of keeping the smoothness without the discussion about how to pick the parameter M. Actually, the solution of the optimization problem

$\min_{\theta} \sum_{i}^{\gamma}(c_i - \sum_{j=0}^{M-1} \theta_j u_j(i))$ is the conventional least square estimator in estimation theory [49]. Such estimator is widely utilized in regression and signal processing tasks and the choice of the first M coefficients during classifier building is closely related to the linear approximation concept in classical approximation theory. Also it is worth pointing out that such technique is quite closely related to ours since our proposed techniques utilize similar procedure for signal estimation rather than classification, which makes this problem more complex. But following the philosophy of signal processing, we are able to shed light on the uncleared issues, which are difficult to be solved via the "computer science" methodology.

2.3.3 Signal Processing on Graphs

While graph theories are widely used in computer science techniques, there have been a lot of efforts of applying graph theories to signal processing problems, especially to network applications. Common IP networks, ad hoc networks or wireless sensor networks can be modeled as graphs and sometimes we are interested in extracting the information from the networks. Such scenarios motivate the developments of signal processing techniques on graphs.

Wavelets on Graphs

The classical Continuous Wavelet Transform (CWT) [40] may be considered as a form of time-frequency representation for continuous-time (analog) signals while more and more efforts have been put on the development of wavelets for signals supported on graphs [26, 27, 36]. The recent work of [36] construct the "wavelet" transform for signals on graphs via spectral graph theory. In this paper, the authors call the eigenbasis of the Laplacian matrix as the graph Fourier Transform. They deduct the graph wavelet functions:

$$\psi_{t,n}(m) = \sum_{l=0}^{N-1} g(t\lambda_l) u_l^*(n) u_l(m)$$

where g is defined as the spectral graph wavelet kernel and u_i is the ith eigenfunction of the Laplacian. It is easy to see that t is the scaling factor while n is the location factor. Formally, the wavelet coefficients of a given function x are produced by taking the inner

product with these wavelets, as

$$W_x(t,n) = \langle \psi_{t,n}, x \rangle$$

Their work shows that scaling may be implemented in the spectral domain of the graph Laplacian.

Uncertainty Principle for Signals Supported on Graphs

As has been introduced in Subsection 2.1.2, the uncertainty principle plays an important role in the area of conventional signal processing. Recently, Agaskar and Lu [3, 4] extended this classical result to functions defined on graphs. They first justify the use of the graph Laplacian eigenbasis as a surrogate for the Fourier basis for graphs, and define the notions of spread in the graph and spectral domains and establish an analogous uncertainty principle for signals on graphs.

In their work, they first claim that the Laplacian eigenvalue λ_i corresponds to the square of frequency ω^2 and then define the spectral spread of signal x:

$$\sigma_s^2 = \frac{1}{\|x\|^2} \sum_{i=0}^{N-1} \lambda_i |\langle x, u_i \rangle|^2 = \frac{1}{\|x\|^2} x^T L x$$

Then, the graph spread of a vector $x \in \mathbb{R}^N$ is defined as:

$$\sigma_g^2 = \frac{1}{\|x\|^2} \min_{v_0 \in V} \sum_{v \in V} d(v, v_0) |x(v)|^2$$

where $d(v_1, v_2)$ is the distance between vertex v_1 and vertex v_2, be the smallest number of edges that need to be traversed to get from one to the other and v_0 is defined as the center node of this graph. With the help of such definitions, it can be further proved that if the center point $x(v_0)$ is smaller than its neighboring points and $x(v) = 0$ if the degree of vertex v is 1, the following applies to any connected and acyclic graphs (graphs without cycle):

$$\sigma_s^2 \sigma_g^2 \geq \frac{1}{32}$$

The theoretical analysis on the uncertainty principle sheds light on the "Fourier" properties of the graph Laplacian. It is worth pointing out that there are actually more related

techniques which utilize the philosophy of "Fourier" properties of graph Laplacian. In parallel, Pesenson [47,48] studied sampling theorems for "bandlimited" functions on graphs. Here "bandlimited" means that the functions x only contains "low frequency" components, i.e., $\langle x, u_i \rangle$ for large i. Moreover, there are certain lines of research which focus on practical applications. In [57], a method to efficiently distribute the application of graph Fourier multipliers to the high-dimensional signals collected by sensor networks is proposed. Such method features approximations of the graph Fourier multipliers by shifted Chebyshev polynomials. Their method is also based on the belief on the "Fourier" property of the graph Laplacian eigenbasis.

2.4 Discussion

In this chapter, we have introduced three categories of research lines. We have introduced the basics of approximation theory and several properties of the Fourier transform. We also included compressed sensing, which is based on approximation theory. Moreover, we discussed several techniques based on graph spectral analysis. Interestingly, all the researchers in the area of computer science and those in signal processing have made outstanding achievements in graph-based techniques respectively. In Subsection 2.3, we have seen that the scholars from computer graphics and manifold learning have developed many graph-based techniques by adopting the philosophy of signal processing while the signal processing community also contributes to this topic but with more bias on theoretical analysis. Our work applies the graph-based techniques on signal processing tasks, leveraging the idea from manifold learning. Moreover, the theoretical analysis of our work is built upon the basis of signal processing. In the following chapters, we will show that our work contribute to connecting manifold learning and signal processing. We show that the theoretical tool developed in signal processing can be exploited to analyze the graph based techniques developed in manifold learning and semi-supervised learning and we also show that we can generalize such techniques and apply to signal processing tasks.

Chapter 3

The Graph Fourier Transform

In the previous chapter, we have seen that there have already been several works considering the "Fourier" properties of the signals supported on graphs while a detailed theoretical analysis has not been made yet. Hence, our work tries to fill the gap problem. This chapter extends the conventional approximation theory to signals on graphs and provides a theoretical analysis about why and when the graph Laplacian eigenbasis can be regarded as a Fourier transform for signals supported on graphs.

3.1 Towards Properties of the Graph Fourier Transform

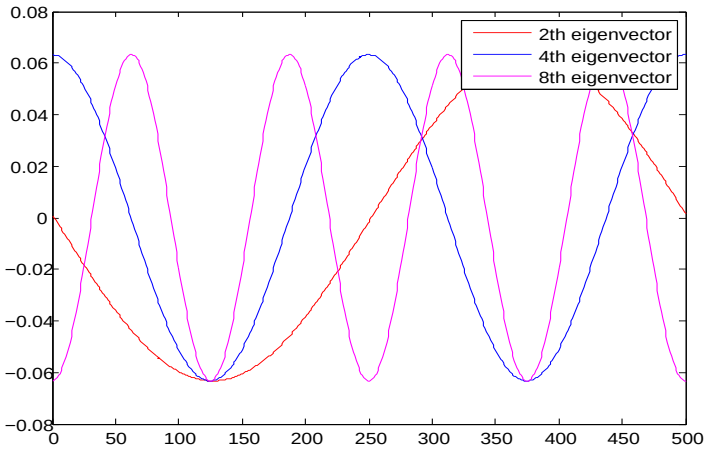


Fig. 3.1 Illustration of some eigenvectors of a ring with 500 nodes

Signals supported on graphs are fairly common in real applications. For a given graph

$G = (V, E)$, we write $x \in \mathbb{R}^V$ to mean that x is supported on the vertices of G. We are most interested in the situation that the distribution of the signal is closely related to its underlying graph topology. For example, consider the data flow readings from the routers in a network. It is reasonable to assume that the data flow is highly correlated to the underlying topology. Or consider the readings from a group of sensor nodes for field estimation. If we construct an $\epsilon-$graph of the network by its location information, then it is also reasonable to assume that the neighbor nodes share similar readings. In other words, the desired "Fourier" transform of signals supported on graphs should be able to capture the topology information. Spectral graph theory provides us with powerful tools to analyze the graph topology such as the study about Laplacian matrix.

An interesting fact which has been noted many times is that the 1-D ring and the 2-D grid are examples of circulant graphs, and it is well known that the Discrete Fourier Transform (DFT) is an eigenbasis for all circulant matrices [35]; i.e., the Laplacian matrix of any circulant graph is diagonalized by the DFT basis. This has been a starting point for researchers to adopt the Laplacian eigenbasis (i.e., the GFT) as a "Fourier" transform of graphs. Fig. 3.1 shows the 2nd, 4th and 8th eigenvector of a ring with 500 nodes. It is clear that they exhibits certain "Fourier" properties. Hence, a natural question one might ask: Is it possible for graphs with more general structures to have similar properties of the Fourier transform? The following subsection considers this issue.

3.2 Properties of the Graph Fourier Transform

One vital concept closely related to the Fourier transform is the smoothness of signals, since smooth signals have compressible Fourier coefficients; i.e., the sorted magnitudes of their Fourier coefficients exhibit a power law decay. Hence, we can keep a small portion of the large ones to approximate the signal while discarding all the others. Similarly, in the graph setting we need a notion of the smoothness of signals on graphs. In this work, we care about more general graphs and signals than certain previous work [39]. Accordingly, we extend this notion to "the value associated with a vertex is very close to that of its neighbors". More concretely, the following definition of 2-norm graph total variation describes the overall smoothness of a signal.

Definition 11. 2-norm Graph Total Variation: Given a signal $x \in \mathbb{R}^V$, $\|x\|_G = (x^T L x)^{1/2} = (\sum_{i\sim j}(x(i) - x(j))^2)^{1/2}$, where $i \sim j$ means there exists an edge between node i and node

j.

The 2-norm graph total variation quantifies the smoothness of a signal defined on the vertices of a graph. The smaller the graph total variation a signal has, the smoother the signal is on the graph. Zhu et al. [66] also mention that $x^T Lx$ measures the smoothness of x on the graph.

Definition 12. We say that $x \in \mathbb{R}^V$ has bounded variation if $\|x\|_G < +\infty$.

Remark 3.2.1.

1. In an asymptotic sense, if the number of graph nodes $N \to +\infty$, the bounded variation condition implies that $\sum_{i=0}^{+\infty} \lambda_i |\widehat{x(\lambda_i)}|^2 < +\infty$, which gives $\lim_{i\to\infty} \lambda_i |\widehat{x(\lambda_i)}|^2 = 0$. Hence, the GFT coefficients of a signal with bounded graph variation are closely related to the Laplacian eigenvalues λ_i, and thus the graph structure. For example, if we consider a signal with bounded variation on a complete graph[1], $|\widehat{x(\lambda_i)}| \to 0$ since $\lambda_i \to +\infty$, where $i = 1, 2, \cdots$, i.e., only signals containing a DC component can be considered smooth for complete graphs. It is worth pointing out that this definition is consistent with the total variation of continuous signal in conventional approximation theory.

2. However, for graphs with finite number of nodes, the bounded variation cannot guarantee any strong conclusions for the decay of the GFT coefficients. Hence, for finite graphs, we say a signal x has a small total variation if its $\|x\|_G^2 \ll \lambda_{N-1}\|x\|_2^2$. This is straightforward since $\|x\|_G^2$ ranges from 0 to $\lambda_{N-1}\|x\|_2^2$. Again, consider a complete graph with finite number of nodes. The bounded variation condition cannot imply much here. However, if the signal has a small total variation, the DC component should dominate the signal, i.e., other coefficients are small. This is because $\|x\|_G^2 = \lambda_0 |\widehat{x(\lambda_0)}|^2 + \lambda_{N-1} \sum_{i=1}^{N-1} |\widehat{x(\lambda_i)}|^2$ is far smaller than $\lambda_{N-1}\|x\|_2^2 = \lambda_{N-1} \sum_{i=0}^{N-1} |\widehat{x(\lambda_i)}|^2$.

Now that we have the concept of total variation for signals on graphs, next let us define the linear and non-linear approximation error for the GFT. They are similar to those of the Fourier transform.

Definition 13. The M-term linear approximation error is

$$\epsilon_l(M, x) = \sum_{i=M}^{N-1} |\widehat{x(\lambda_i)}|^2,$$

[1] Where every node is neighbors with all other nodes

where $\widehat{x(\lambda_i)} = \langle x, u_i \rangle$ denotes the ith GFT coefficient of signal x, and where u_i is the ith eigenvector of the Laplacian matrix of graph G.

Definition 14. The M-term non-linear approximation error is

$$\epsilon_n(M, x) = \sum_{i \notin \Omega} |\widehat{x(\lambda_i)}|^2,$$

where Ω is the set of indices of the M largest graph Fourier coefficients in magnitude.

The following theorems describe the properties of the GFT.

Theorem 3.2.1. *Given a signal $x \in \mathbb{R}^V$ on vertices of a graph $G = (V, E)$, let λ_i denote the ith eigenvalue of the Laplacian matrix L and $\widehat{x(\lambda_i)} = \langle x, u_i \rangle$ denotes the ith GFT coefficient of the signal x. Then,*

$$|\widehat{x(\lambda_i)}| \leq \frac{\|x\|_G}{\sqrt{\lambda_i}}.$$

Proof. By Definition 11,

$$\|x\|_G^2 = \sum_{i \sim j} w_{ij}(x(i) - x(j))^2 \tag{3.1}$$

$$= x^T L x \tag{3.2}$$

$$= x^T (\sum_{i=0}^{N-1} \lambda_i u_i u_i^T) x, \tag{3.3}$$

where u_i is the ith eigenvector of the Laplacian matrix L. Put x^T and x inside the sum,

$$\|x\|_G^2 = \sum_{i=0}^{N-1} \lambda_i |\langle u_i, x \rangle|^2 \tag{3.4}$$

$$= \sum_{i=0}^{N-1} \lambda_i |\widehat{x(\lambda_i)}|^2. \tag{3.5}$$

It is straightforward to see that $\lambda_i|\widehat{x(\lambda_i)}|^2 \leq \sum_{i=0}^{N-1} \lambda_i|\widehat{x(\lambda_i)}|^2 = \|x\|_G^2$, thus

$$|\widehat{x(\lambda_i)}| \leq \frac{\|x\|_G}{\sqrt{\lambda_i}}. \tag{3.6}$$

□

Compared with Proposition 2.1.1, Theorem 3.2.1 implies that eigenvalues of the graph Laplacian play the same role as "frequencies" in traditional signal processing; i.e., $\lambda_0, \cdots, \lambda_{N-1}$ index the GFT coefficients from low to high "frequencies". Accordingly, the eigenvectors of the Laplacian are actually the "frequency" components of a graph. The next theorem discusses the bound for linear approximation error.

Theorem 3.2.2. *Consider a signal $x \in \mathbb{R}^V$ on the graph $G = (V, E)$. If x has a bounded variation, then:*

$$\epsilon_l(M, x) \leq \|x\|_G^2 \lambda_M^{-1}$$

Proof. Due to the proof of Theorem 3.2.1, we know that:

$$\|x\|_G^2 = \sum_{i=0}^{N-1} \lambda_i|\widehat{x(\lambda_i)}|^2. \tag{3.7}$$

Also since $\lambda_i \geq 0$, it is straightforward to see that

$$\sum_{i=M}^{N-1} \lambda_i|\widehat{x(\lambda_i)}|^2 \leq \sum_{i=0}^{N-1} \lambda_i|\widehat{x(\lambda_i)}|^2 \tag{3.8}$$

$$= \|x\|_G^2 \tag{3.9}$$

The first inequality holds if x has bounded variation and the last equality is due to Eq. 3.7.

By the definition of linear approximation error, we have:

$$\epsilon_l(M,x) = \sum_{i=M}^{N-1} |\widehat{x(\lambda_i)}|^2 = \frac{\lambda_M}{\lambda_M} \sum_{i=M}^{N-1} |\widehat{x(\lambda_i)}|^2 \tag{3.10}$$

$$= \frac{1}{\lambda_M} \sum_{i=M}^{N-1} \lambda_M |\widehat{x(\lambda_i)}|^2 \tag{3.11}$$

$$\leq \frac{1}{\lambda_M} \sum_{i=M}^{N-1} \lambda_i |\widehat{x(\lambda_i)}|^2 \tag{3.12}$$

The last inequality is due to the fact that $\lambda_i \leq \lambda_{i+1}$. By adopting the inequality $\sum_{i=M}^{N-1} \lambda_i |\widehat{x(\lambda_i)}|^2 \leq \|x\|_G^2$, we finally have $\epsilon_l(M,x) \leq \frac{1}{\lambda_M}\|x\|_G^2$. □

Remark 3.2.2.

1. For the case where $N \to +\infty$, this statement is analogous to Theorem 2.1.2 for the classical Fourier transform. It shows that the decaying rate of the linear approximation error is $O(\frac{1}{\lambda_M})$. The difference from Fourier transform is that the upper bound of the linear approximation error is related to both the Laplacian eigenvalues and the graph total variation. It implies that if the eigenvalues keep increasing, the linear approximation error decays.

2. For graphs with finite number of nodes, the asymptotic explanation about the decaying rate of the GFT coefficients can no longer stand. Actually, since $\|x\|_G^2 = \alpha\lambda_{N-1}\|x\|_2^2$ where α range from 0 to 1, then it is straightforward to see that $\epsilon_l(M,x) \leq \alpha\frac{\lambda_{N-1}}{\lambda_M}\|x\|_2^2$. If $\alpha \geq \frac{\lambda_M}{\lambda_{N-1}}$, then the upper bound $\frac{\|x\|_G^2}{\lambda_M}$ does not imply any thing about the linear approximation error. On the other hand, in order to let this upper bound dominate the behavior of its linear approximation error, we need x to have an adequately small total variation, i.e., $\|x\|_G^2 \ll \lambda_{N-1}\|x\|_2^2$. For example, if $\|x\|_G = 0$, then $\epsilon_l(M,x) = 0$ for $M \geq 1$, which means that this signal only has DC component.

The above two theorems describe upper bounds for the GFT coefficients and the linear approximation error . The next theorem, which is similar to Theorem 2.1.3, gives the relation between the decaying rate of the GFT coefficients and that of the linear approximation error. It is worth noting that Theorem 2.1.3 gives the simple fact that fast decaying coefficients leads to fast decaying linear approximation error. This result can also be applied to the case of GFT since it applies to any orthogonal basis. However, the theorem does

not take the distribution of eigenvalues into account while the decaying rate of the GFT coefficients is highly correlated to the distribution of eigenvalues. In order to address this issue, we derive the following lemma and theorem by extending Theorem 2.1.3:

Lemma 3.2.3. *Consider a signal $x \in \mathbb{R}^V$ on a connected finite graph, for any $s \geq 1$:*

$$\sum_{i=0}^{N-1} i^{s-1}\lambda_i^s|\widehat{x(\lambda_i)}|^2 \leq \sum_{M=0}^{N-1} M^{s-1}\lambda_M^s\epsilon_l(M,x) \leq C_s\sum_{i=0}^{N-1}(i\lambda_i)^s|\widehat{x(\lambda_i)}|^2$$

If we consider a graph G with an infinite number of nodes, then for any $s \geq 1$

$$\sum_{i=0}^{+\infty} i^{s-1}\lambda_i^s|\widehat{x(\lambda_i)}|^2 \leq \sum_{M=0}^{+\infty} M^{s-1}\lambda_M^s\epsilon_l(M,x) \leq C_s\sum_{i=0}^{+\infty}(i\lambda_i)^s|\widehat{x(\lambda_i)}|^2$$

where C_s is some constant larger than $1/s$.

Proof. First we prove the case when $s = 1$. Notice the fact that $\sum_{M=0}^{N-1}\lambda_M^s\sum_{i=M}^{N-1}|\widehat{x(\lambda_i)}|^2 = \sum_{i=0}^{N-1}|\widehat{x(\lambda_i)}|^2(\sum_{M=0}^{i}\lambda_M)$, which immediately gives the lower bound. Moreover, since $\lambda_n \leq \lambda_m$ for all $n \leq m$, we obtain the upper bound.

Next consider the case $s > 1$. Still $\sum_{M=0}^{N-1} M^{s-1}\lambda_M^s\sum_{i=M}^{N-1}|\widehat{x(\lambda_i)}|^2 \geq \sum_{M=0}^{N-1} M^{s-1}\lambda_M^s|\widehat{x(\lambda_M)}|^2$ gives lower bound. On the other hand,

$$\begin{aligned}
\sum_{M=0}^{N-1} M^{s-1}\lambda_M^s\sum_{i=M}^{N-1}|\widehat{x(\lambda_i)}|^2 &= \sum_{i=0}^{N-1}|\widehat{x(\lambda_i)}|^2(\sum_{M=0}^{i} M^{s-1}\lambda_M^s) && (3.13)\\
&\leq \sum_{i=0}^{N-1}|\widehat{x(\lambda_i)}|^2\lambda_i^s(\sum_{M=0}^{i} M^{s-1}) && (3.14)\\
&\leq \sum_{i=0}^{N-1}|\widehat{x(\lambda_i)}|^2\lambda_i^s\int_0^i t^{s-1}dt && (3.15)\\
&= \frac{1}{s}\sum_{i=0}^{N-1}|\widehat{x(\lambda_i)}|^2(\lambda_i i)^s && (3.16)
\end{aligned}$$

□

Theorem 3.2.4. *Given a graph G with infinite number of nodes, if $\sum_{i=0}^{+\infty}(i\lambda_i)^s|\widehat{x(\lambda_i)}|^2 <$*

$+\infty$ *for some* $s \geq 1$*, then the* M *term linear approximation error obeys*

$$\epsilon_l(M, x) = o(\frac{1}{M^s \lambda_{M/2}^s}).$$

Proof. From the second statement of Lemma 3.2.3, we notice that

$$\epsilon_l(M, x) \sum_{m=M/2}^{M-1} m^{s-1} \lambda_m^s \leq \sum_{m=M/2}^{M-1} m^{s-1} \lambda_m^s \epsilon_l(m, x) \tag{3.17}$$

$$\leq \sum_{m=M/2}^{+\infty} m^{s-1} \lambda_m^s \epsilon_l(m, x) \tag{3.18}$$

$$\leq \sum_{i=0}^{+\infty} (i\lambda_i)^s |\widehat{x(\lambda_i)}|^2. \tag{3.19}$$

The first inequality holds due to the fact $\epsilon_l(M, x) \leq \epsilon_l(m, x)$ for all $m \leq M$. Since $\sum_{i=0}^{+\infty} (i\lambda_i)^s |\widehat{x(\lambda_i)}|^2 < +\infty$, we have $\sum_{m=M/2}^{+\infty} m^{s-1} \lambda_m^s \epsilon_l(m, x) < +\infty$. Thus,

$$\lim_{M \to \infty} \sum_{m=M/2}^{+\infty} m^{s-1} \lambda_m^s \epsilon_l(m, x) = 0. \tag{3.20}$$

Moreover, it is clear that there exists a constant $C > 0$ such that $C \cdot M^s \leq \sum_{m=M/2}^{M-1} m^{s-1}$. Accordingly, Eq.3.17, Eq.3.18, along with Eq.3.20 implies that

$$\lim_{M \to \infty} (M\lambda_{M/2})^s \epsilon_l(x, M) = 0.$$

□

Remark 3.2.3. Theorem 3.2.4 along with Lemma 3.2.3 describe the behavior of the linear approximation error of graphs with an infinite number of nodes when its eigenvalues are strictly increasing. The condition $\sum_{i=0}^{+\infty} (i\lambda_i)^s |\widehat{x(\lambda_i)}|^2 < +\infty$ implies $|\widehat{x(\lambda_i)}|^2 = o(\frac{1}{i^s \lambda_i^s})$, which is stronger than the bounded variation condition. Then, a similar decay rate of $o(\frac{1}{M^s \lambda_{M/2}^s})$ is guaranteed for the linear approximation error. The above theorem does not require any constraints on the distribution of the Laplacian eigenvalues. However, if we impose certain stronger assumption about the eigenvalues, we can obtain better result: if we assume that $\lambda_M = \Theta(M^s)$ for and $s > 0$, we can obtain $o(\frac{1}{\lambda_M^s})$ as the decay rate of

the linear approximation error. It is worth noting that the condition $\lambda_M = \Theta(M^s)$ for and $s > 0$ rules out the case of complete graph and implies that fast increasing eigenvalues lead to fast decaying linear approximation error.

The above theorems provide us with some implications about which signals on which graphs are likely to be compressible in the corresponding graph Fourier domain. To summarize, there are two main principles: First, from the perspective of signals, we need a smooth signal on the underlying graph, i.e., $\|x\|_G$ is small, since it controls the upper bound of linear approximation error. Second, from the perspective of the underlying graphs, the Laplacian eigenvalue of the graph must keep an increasing trend in order to ensure the graph Fourier coefficients have a decaying upper bound.

3.2.1 Robustness of the Graph Fourier Transform

Since the graph Fourier transform is entirely dependent on the structure of the underlying graph, it is worth discussing how the structural perturbation of a graph affects the decaying rate of the Fourier coefficients. The "perturbation" here refers to adding or removing edges without changing the signal. Zhu et al. [64] discusses the effect of structural perturbations on graph Laplacian eigenvectors. They claim that for regular or small-world networks, eigenvectors corresponding to small eigenvalues usually have small oscillation, which are sensitive to perturbation on a global scale while eigenvectors corresponding to large eigenvalues are mostly sensitive to localized perturbations within a small set of nodes. Moreover, for complex networks that do not possess a regular backbone, they observe that the eigenvectors do not exhibit any periodic wave structure but the above statement still holds.

Zhu's discussion is consistent with our intuition that the eigenvectors with larger eigenvalues correspond to higher frequency basis vectors in the graph Fourier domain. Accordingly, it is easy to conclude that small perturbations of the graph structure will not significantly change the behavior of the GFT coefficients of signals supported on the graph. This is due to the fact that the eigenvectors of higher "frequencies" are more sensitive to small perturbations while their corresponding graph Fourier coefficients are likely to be very small due to Theorem.3.2.1, which means localized perturbations only change those GFT coefficients with small magnitude. Thus, we conclude that the graph Fourier transform of smooth signals are robust to localized perturbations of the underlying graph.

3.2.2 Constructing Graphs for Signal Compression

Given a signal $x \in \mathbb{R}^{N\times D}$, i.e., there exist N graph nodes each with a D dimensional vector value on it. What graph leads to a GFT basis of best compression for x? The properties of the GFT provide us with certain implications of this question. First, each entry in x can be regarded as a node allocated D dimensional data. From our theoretical analysis in last section, we want x to be smooth on the graph, i.e., $\|x\|_G$ should be kept small to let the upper bound of linear approximation error to dominate its decaying behavior. Second, the eigenvalues should keep an increasing tendency, i.e., without too many eigenvalues close to each other. One possible solution to this problem is to use neighborhood graphs. More concretely, let $x(i)$ stands for the $D \times 1$ vector value associated with the ith node. We construct the graph by putting an edge between the nodes which are likely to share similar values so that $(\sum_{i\sim j} \|x(i) - x(j)\|^2)^{1/2}$ is kept small. We therefore provide three methods for constructing such graphs:

1. $\epsilon-$graph: Choose the parameter $\epsilon \in \mathbb{R}$ and then connect the node i and node j if $\|x(i)-x(j)\|_2 \leq \epsilon$. The $\epsilon-$graph is geometrically motivated but it is difficult to choose the parameter ϵ. With a different distribution of the signal x, we need different ϵ to ensure the graph is connected.

2. KNN graph: Node i and node j are connected if i is among the K nearest neighbors of j or j is among the K nearest neighbors of i. Such KNN graph can also be referred to as the "symmetric KNN graph". The degree of each node will be at least K. The choice of K is easier than ϵ since the connectivity of the graph is not significantly affected by the distribution of x if K is determined.

3. Least-weighting graph: The least weighting graph is built in a greedy manner. For each iteration, connect the pair of nodes with least difference, i.e., $\|x(i) - x(j)\|_2$ is smallest. Repeat connecting nodes with least difference until the graph is connected. The least-weighting graph is less geometrically intuitive but one of its main advantage is that it does not require any parameters to be determined in advance.

For neighborhood graphs like the $\epsilon-$graph and KNN graph, we pick a parameter K or ϵ to obtain a desired distribution of the Laplacian eigenvalues. From the perspective of building compressible GFT coefficients, we need the eigenvalues to maintain an increasing

trend. Following this principle, we should avoid constructing graphs like a complete graph. A large choice of K or ϵ will result in the situation where the graph become dense and approximate the behavior of complete graph, ie., the eigenvalues corresponding to high frequencies become close to the largest eigenvalue and hence increase too slowly, which violates the compressibility of the GFT coefficients. Hence, the parameters shouldn't be too large. On the other hand, if the value of K or ϵ is too small, the connectivity of the graph will be weak and the eigenvalues corresponding to low frequencies might be equal to or close to 0. Such behavior also contradicts the increasing trend of eigenvalues that we desire. Thus, the graph we construct should at least be a connected one. The $\epsilon-$graph and KNN graph are common techniques in dimension reduction or semi-supervised learning. However, they are not the only methods for obtaining smooth signals on a graph. Any graph construction approach is desirable if it can result in small graph total variation and increasing eigenvalues. For example, the least-weighting graph is a greedy method which connects the closest pair of nodes at each iteration so that the term $(\sum_{i\sim j} \|x(i)-x(j)\|^2)^{1/2}$ is small and we believe there are more techniques to be developed.

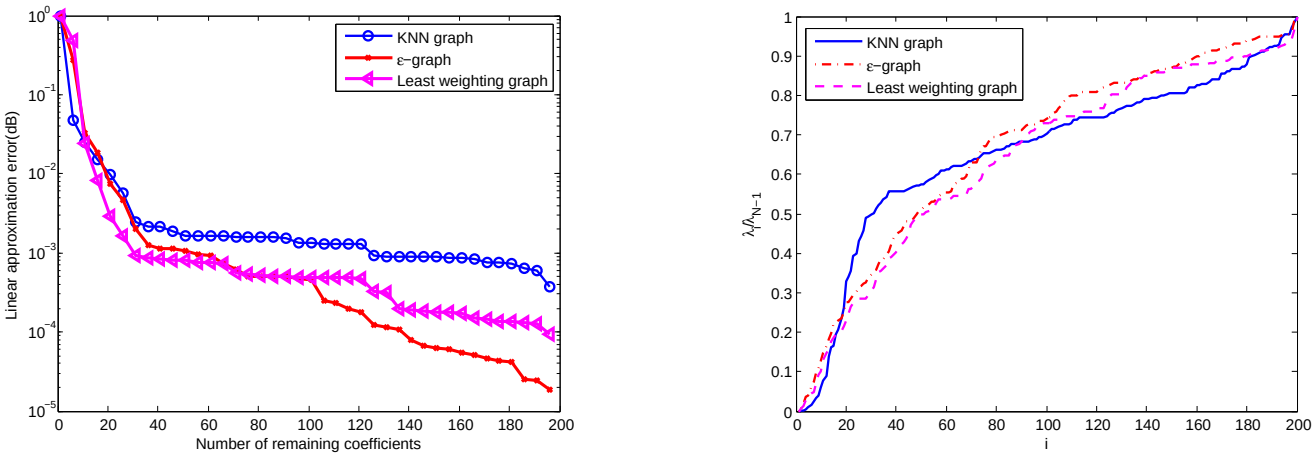
3.3 Simulations and Experiments

In this section, we utilize experiments and simulations to verify the theories we introduced above. First, we will use both synthesized data and real world data to demonstrate how GFT basis works and the distribution of the eigenvalues significantly affects the behavior of the compressibility of a certain signal. The performance of the linear, non-linear approximation and conventional compressed sensing will be used to evaluate the impact of different GFT basis but generated based on a same signal.

3.3.1 Simulated Data

First we utilize the above three methods to generate graphs and check the compressibility of the synthesized data. Fig. 3.2, Fig. 3.3 and Fig. 3.4 show the linear approximation error and the normalized eigenvalues λ_i/λ_{N-1} based on different underlying signals. The signal x is a 200×1 random signal drawn from a (0,1) i.i.d. Gaussian distribution, i.i.d. Uniform distribution and i.i.d. Pareto distribution. The Pareto distribution is a classic "heavily-tailed" distribution, which coincides with social, scientific, and many other types of observable phenomena. In the experiment repeated here, we choose $\alpha = 1.2$ and $b = 3$ for

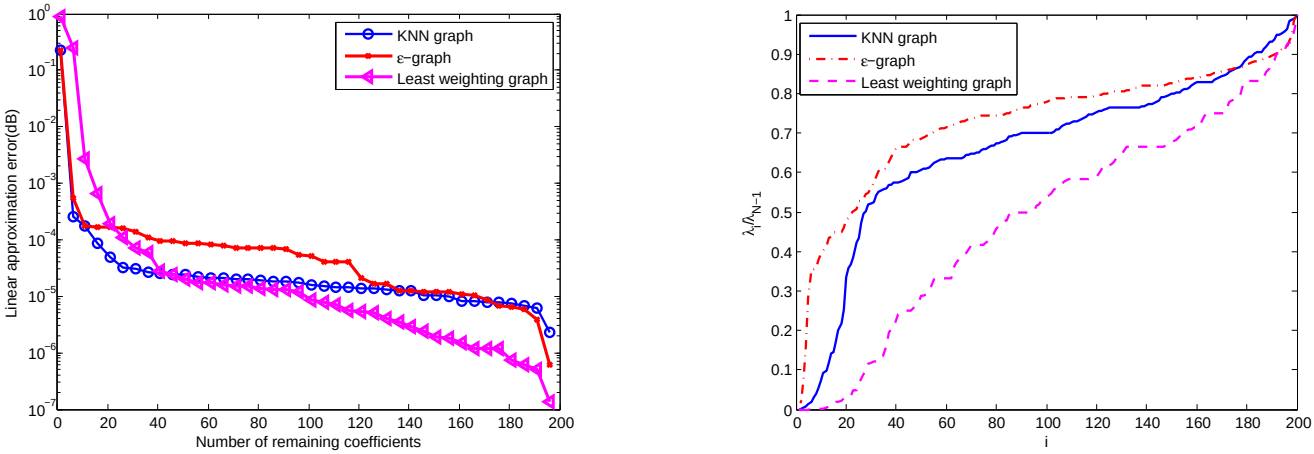
its density function $f(x) = \frac{\alpha}{b}(\frac{b}{x})^{(\alpha+1)}$, for $x >= b$. We select the parameter $K = 7$ for KNN graph and $\epsilon = \sqrt{\frac{C \cdot \log N}{\pi N}} * D$ for $\epsilon-$graph, where $C = 2$ and D is the maximum Euclidean distance among the pair of signal entries x_i and x_j. By picking those two parameters, the underlying graphs are likely to be connected.



(a) Linear approximation error (b) Laplacian Eigenvalues of each graph

Fig. 3.2 The linear approximation error and distribution of Laplacian eigenvalues of $\epsilon-$graph, KNN grapn and least weighting graph. $x(i)$ is drawn from an i.i.d. Gaussian distribution.

The fast decay of the linear approximation error implies that the compressibility of the original signal x, i.e., the GFT coefficients of x decays fast. It is worth noting that the performance varies with different choice of the parameter K or ϵ and the distribution of signal x. From Fig. 3.2, we can see that those 3 methods generate graphs with similar eigenvalue distributions and their linear approximation error are also very close. Fig. 3.3 is based on uniformly distributed signal and its corresponding performance is better than that of Fig. 3.2. In such case, the least weighting graph show certain different behavior when compared with the other two methods: its eigenvalues increases slowly at first (several eigenvalues are quite close to 0) and the linear approximation error decreases slowly at the very beginning correspondingly. From Fig. 3.4, it is straightforward to see that for the "heavy-tailed" distribution, the least weighting graph and the $\epsilon-$graph do not perform well since they construct graphs with very strong connectivity, which are close to the complete graph. The experiments results show that all the above methods can generate very good graphs when given a certain signal. But KNN graph is generally the best for various types of



(a) Linear approximation error

(b) Laplacian Eigenvalues of each graph

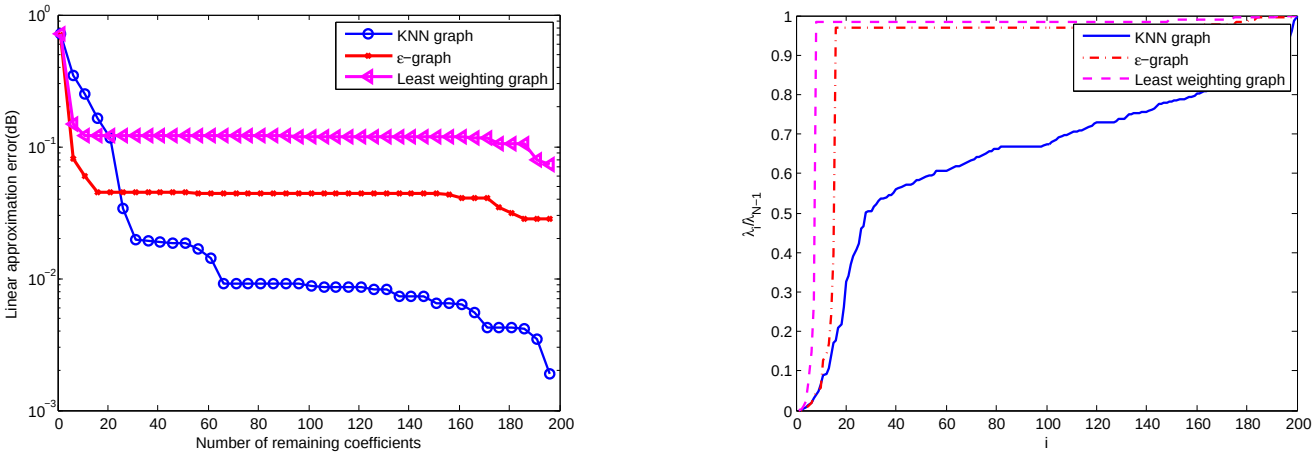
Fig. 3.3 The linear approximation error and distribution of Laplacian eigenvalues of $\epsilon-$graph, KNN grapn and least weighting graph. $x(i)$ is drawn from an uniform distribution.

signals. Hence we recommend to use the KNN graph when dealing with signals of unknown distribution.

It is worth pointing out that in many applications, we may not have prior information about the exact distribution of the signal x, but we can construct the graph based on other information. For example, for field estimation in a wireless sensor network, it is fairly reasonable to assume the values measured at each node are highly correlated to its location, and thus nodes that are geographically close to each other are likely to have similar readings. Hence, we can build the graph based on the location information.

Although the above simulations have already shed light on the relation between linear approximation error and distribution of eigenvalues, it would be much clearer if we utilize the same graph construction techniques but with different choice of parameter to control the pattern of eigenvalues. In the simulation here, we utilize the comparison among KNN graphs with different choice of K. The underlying signal x is an i.i.d. Gaussian distributed random signal and we build a KNN graph based on the node values. We plot their linear approximation error and the distribution of normalized eigenvalue respectively. The result is illustrated in Fig. 3.5.

From Fig. 3.5, we see that when K is set to 30, the eigenvalues corresponding to low frequencies increase sharply and meanwhile the linear approximation error drops significantly.



(a) Linear approximation error

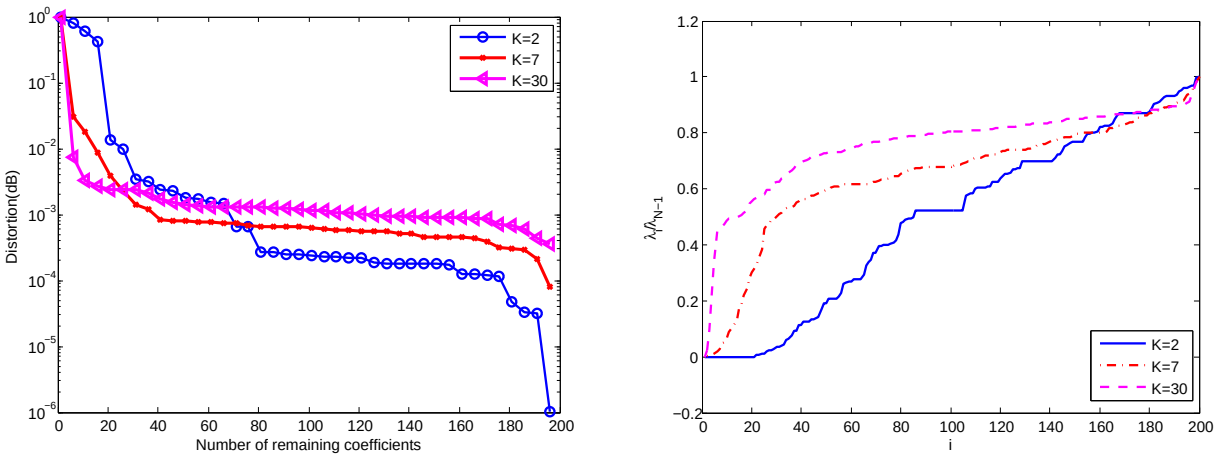
(b) Laplacian Eigenvalues of each graph

Fig. 3.4 The linear approximation error and distribution of Laplacian eigenvalues of $\epsilon-$graph, KNN grapn and least weighting graph. $x(i)$ is drawn from an i.i.d. Pareto distribution.

After the rapid increasing, the eigenvalues corresponding to high frequencies maintain a slow increasing rate and the linear approximation error decay slowly. The situation when $K = 2$ is the opposite: the eigenvalues corresponding to low frequencies increase slowly while the remaining eigenvalues maintain a steady increasing rate. Consequently, the linear approximation error decays very slowly at first but catches up quickly later. Fig. 3.5 clearly illustrates different choice of K is suitable for different number of remaining coefficients for linear approximation. However, neither of the above two cases provide satisfying compressibility: one fails to provide quick decaying behavior for low frequency components while the other for high frequency components. For better compressibility, we often desire a tradeoff between the above two cases. The curve for $K = 7$ illustrates such scenario: we can utilize a small portion of the coefficients to represent the original signal while maintaining the loss acceptable. The results shown in Fig. 3.5 to some extent verify our theoretical analysis, i.e., the decaying rate of the eigenvalues affect the decaying rate of linear approximation error.

3.3.2 Environmental Data

In the following experiments, we investigate the performance of GSCS on data from the California Irrigation Management Information System (CIMIS) [2]. This dataset is gener-



(a) (b)

Fig. 3.5 Fig. 3.5 illustrates the relation between linear approximation error and the distribution of eigenvalues. The signal x is an i.i.d. Gaussian distributed random signal and we utilize KNN graph to generate its corresponding GFT basis. Fig. 3.5(a) shows the linear approximation error with regard to different choice of K. Fig. 3.5(b) plots their corresponding distribution of eigenvalues.

ated by the weather stations across the state of California, which are equipped with sensors that measure solar radiation, temperature, and wind speed, among other variables.

We use the solar radiation data for one day which contains 135 readings from different weather stations to verify our theory about the GFT. We show that the techniques we discussed in Subsection 3.2.2 can be exploited to generate linear compressible signals on real world data. We utilize KNN graphs based on the geological information of weather station to build its GFT basis.

We will compare the performance of linear approximation, non-linear approximation and compressed sensing [21,29] on this dataset. We know that compressed sensing works well for compressible signals and thus its performance can be exploited to imply the compressibility of a signal. For compressed sensing, we use ℓ_1 programming in the Graph Fourier basis as the decoding algorithm. All the experiments are repeated 50 times and the average values are reported. Moreover, we will show which GFT basis is best for approximating signals via CS by changing the parameter K to construct different graphs.

Fig. 3.6 illustrates the performance of CS, linear approximation and non-linear approximation with increasing compression rate. The compression ratio is defined as $\frac{M}{N}$, where M

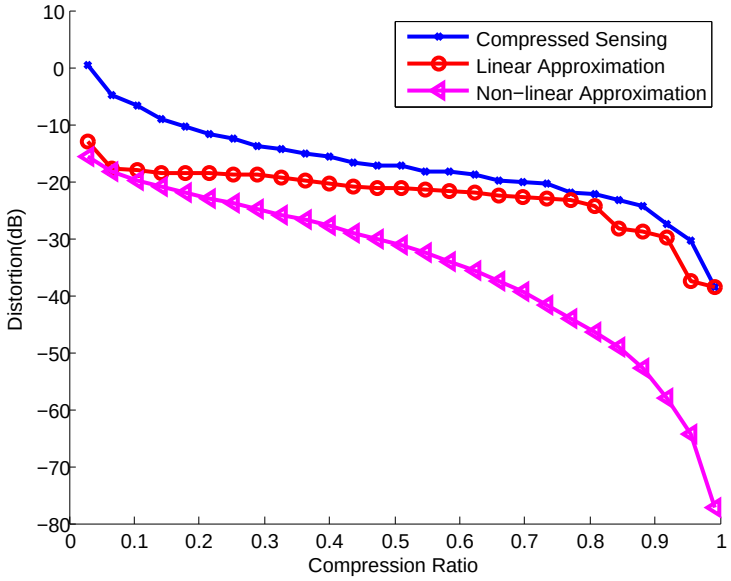
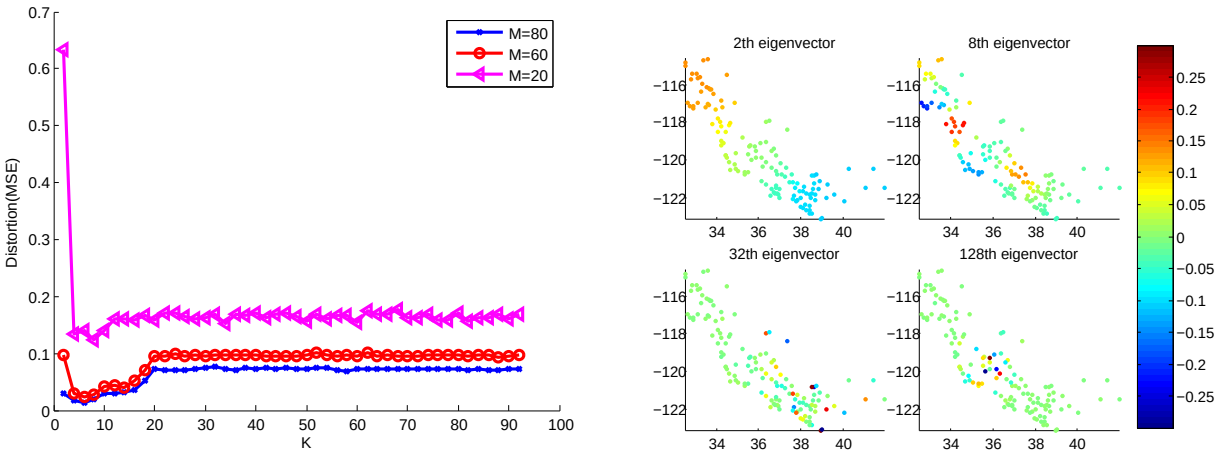


Fig. 3.6 The Performance of Compressed Sensing, linear approximation and non-linear approximation.

is the number of measurements and N is the dimension of signal. Distortion is calculated with Mean Square Error(MSE). The non-linear approximation outperforms the other two methods, while linear approximation performs a little bit better than Compressed Sensing. This result further verifies the conclusion we made in this section: by utilizing prior information for graph construction in real applications, we are able to obtain compressible signals.

Fig. 3.7(a) describes explicitly how the connectivity of a graph affects the performance of compressed sensing. The result agrees with our earlier discussion about the choice of parameter K. Given a constant compression rate, the best performance of Compressed Sensing appears when K is in the range of $5-10$. When K is smaller than 5, the graph is unconnected with high probability. In this case, we have multiple zero eigenvalues. When K become larger than 30, the graph approximate the complete graph, which also gives a poor compressibility. Fig. 3.7(b) gives the behavior of the eigenvectors when K is set to 6. We can see that the low frequency eigenvector entries are close to their neighbors, i.e., change smoothly while the high frequency eigenvector entries change drastically in a local area.



(a) (b)

Fig. 3.7 (a) The performance of Compressed Sensing with different graph Fourier basis. M is the number of measurements. X axis shows the number of neighbors we use to formulate a symmetric KNN graph. (b) plots the behavior of the 2rd, 8th, 32th 128th eigenvectors when we set $K = 6$

3.4 Discussion

In the realm of signal processing, not much emphasis is laid on the graph Laplacian and its properties while a great amount of works have focused on it in the area of computer science. As have been introduced in Chapter 2, [39] utilizes the technique called spectral compression for 3D object compression. However, in their work, Karni and Gotsman merely claim the graph Laplacian eigenbasis has the "Fourier" properties instead of giving a strict theoretical proof. Later on, [15] provides theoretical guarantee by showing the Laplacian matrix is equivalent to the inverse of covariance matrix. Consequently, the graph Laplacian eigenbasis is intrinsically the same as KLT and hence optimal. However, this conclusion is restricted to coordinates on mesh and cannot be extended to generalized situations. Meanwhile, one dimensionality reduction technique called Eigenmaps was developed by Belkin [12]. The procedure of their algorithm is similar as described in subsection 3.2.2 by utilizing KNN or $\epsilon-$graph. Different from our analysis, they justify the method by showing that Laplace Beltrami operator provides an optimal embedding for the manifold and the graph Laplacian converges to Laplace Beltrami operator when the number of nodes $N \to +\infty$ and $\epsilon \to 0$ [13]. Although Eigenmaps applies for more general scenarios than spectral compression, their theory neither provides any instructions on how to choose the

parameter K or ϵ nor shows how good is the embedding.

Different from the above methodologies, our work stems from the approximation theory and deal with not only KNN or $\epsilon-$graph but also graphs with more generalized features (the eigenvalue distribution). It is worth emphasizing that we merely put constraints on the distribution of eigenvalues while no specific graph structures are required in our analysis. Hence, our analysis implies that there might be more types of graphs feasible for manifold embedding. Moreover, the theoretical justification for Eigenmap is based on the asymptotic behavior of the graph Laplacian for uniformly distributed data points. Accordingly, their analysis does not show how to choose the parameters K or ϵ for finite number of nodes with arbitrary distribution. Our work, on the other hand, relates the linear approximation error to the distribution of eigenvalues and take one step further on how to choose those parameters. Although some literatures in semi-supervised learning [11] mention that $x^T Lx$ can represent the smoothness of a signal, they have not analyzed the impact of the graph topology on smoothness while our work relates the smoothness of signals supported on graphs to the conventional concept called total variation in approximation theory and show the Laplacian eigenvalues plays an important role in characterizing the signal as smooth with regard to the underlying graph.

Chapter 4

Graph Spectral Compressed Sensing

The previous chapter discussed how smooth signals supported on graphs can be decomposed into decaying GFT coefficients. If we have a signal that decays fast in the GFT domain and can find a power law decay upper bound, we can refer to such signals as "compressible". In classical approximation theory, it is common to use linear or non-linear approximation to code such "compressible" signals. However, in certain applications like wireless sensor networks, obtaining the linear or non-linear approximation in a distributed manner requires significant overhead. In this section, we provide an alternative which compresses such signals by random sampling and is energy efficient. More concretely, if the signal on graph is adequately smooth with respect to the graph, then we can randomly sample a small portion of the nodes and recover the original signal by a simple least square estimator. The experiments also shows that ℓ_1 decoding still works well for such random sampling scheme. The idea here is leveraged from Compressed Sensing.

4.1 Linear Compressible Signals

The conventional CS theory deals with general sparse or compressible signals while in real application there are more realistic signal models that go beyond simple sparsity and compressibility by including dependencies between values and locations of the signal coefficients. The model-based compressive sensing [10] deals with such situation. They have shown that for the subspaces where the magnitude of the signal is small, we allow a generalized version of RIP and can utilize a simplified version of CoSamp called model based recovery algorithm.

Leveraging the idea from model based CS, we focus on signals and sensing matrices with more special properties and an upper bound is still provided for the recovery error of a least square estimator under such circumstances. Before we delve into the recovery process, we need to first understand the property of the signals supported on graphs and that of the sensing matrices.

Within the scope of this section, we focus on the smooth signals supported on graphs and we are interested in the graphs whose eigenvalues have an increasing trend. From the discussions in last section, we know that such signals exhibit similar behaviors described in conventional approximation theory. More concretely, the GFT coefficients of smooth signals on graphs present a linearly decaying behavior. In order to model such signals, we assume that $\frac{1}{\lambda_i}$ satisfy a power law decay property. For simplicity, we use θ to denote the GFT of x, i.e., $\theta = U^T x$ and thus the corresponding GFT coefficients satisfy $|\theta(i)| = |\widehat{x(\lambda_i)}| \leq Gi^{-1/r}$ for some $r > 0$. We call such signals linearly compressible. As discussed in [10], compressible signals can be defined by their decaying behavior of non-linear approximation error. Since we are talking about linearly compressible signals, we can adapt the definition a little to fit our case:

Definition 15. The set of s-linear-compressible signals is defined as

$$\mathbb{L}_s = \{x \in \mathbb{R}^N : \epsilon_l(\gamma, x) \leq S\gamma^{-s}, 1 \leq \gamma \leq N, S < \infty\},$$

where $\epsilon_l(\gamma, x)$ is the γ-term Linear Graph Fourier Approximation Error.

In conventional CS literatures [19,25], the performance of CS is proved to be comparable to the non-linear approximation. The techniques utilized to prove this conclusion need to divide the space of the compressible signal into roughly $\lceil N/\gamma \rceil$ residual subspaces if we want to relate it to a $\gamma-$term non-linear approximation. We can exploit similar techniques in our scenario while focusing on the linear compressible signals. We want to show that the performance of GSCS is comparable to a $\gamma-$term linear approximation instead of a non-linear approximation. Also, it is straightforward to see that the difference of the linear compressible signal between the $j\gamma$ term linear approximation error and $j(\gamma-1)$ term linear approximation error lies in a deterministic subspaces, which is captured by the following:

Definition 16. Given a signal θ, its jth set of the linear residual subspaces of size γ is defined as: $L_{j,\gamma} = \{u \in \mathbb{R}^N \text{ such that } u = \theta_{j\gamma} - \theta_{(j-1)\gamma}\}$ for $j = 1, 2, \cdots, \lceil N/\gamma \rceil$. We let

$\theta_0 = 0$ here and $\theta_{j\gamma}$ is the $j\gamma$ term linear approximation of x, i.e., $\theta_{j\gamma}$ maintain the first $j\gamma$ entries while set the rest of the others to 0. Also we denote the corresponding support of u as T_j. The last set may have less than γ non-zero entries.

According to the definition, we can split a linear compressible signal into $\lceil N/\gamma \rceil$ sets of the linear residual subspaces. For a linear compressible signal x, $\|\theta_{T_j}\|^2$ decays fast as j becomes larger, where $\theta_{T_j} = \{\theta_{T_j}(i) = \theta(i) \text{ if } i \in T_j \text{ otherwise } \theta_{T_j}(i) = 0\}$. Moreover, if T and Ω are subsets of $\{1, 2, \cdots, N\}$, we denote Φ_T as the submatrix of selecting the corresponding columns from matrix Φ while Φ_Ω as the submatrix of selecting the corresponding rows from Φ. It is worth noting that the linear compressible signals are special case of the model-based compressible signals.

4.2 Coherence of the Graph Fourier Transform Basis

Candès and Tao [22], along with Rudelson and Vershynin [54] discuss conditions that the structured random matrices should satisfy to be valid CS sensing matrices:

1. The matrix should be orthogonal.
2. For a $N \times N$ matrix, if we normalize each column such that its 2-norm is 1, the magnitude of the entries should be upper bounded by $O(\frac{1}{\sqrt{N}})$, i.e., the coherence of the sensing matrix $\mu = O(\frac{1}{\sqrt{N}})$, where $\mu = \max_{i,j} |\Phi_{i,j}|$.

By randomly selecting $M = O(\gamma \ln^4 N))$ rows of such matrices, we can generate valid sensing matrices for CS. The traditional Fourier basis is clearly a candidate fit for such criteria. If F is the Discrete Fourier Transform (DFT) basis and let $\theta = F^T x$. If Ω is a random subset of $\{1, 2, \cdots, N\}$ with dimension $|\Omega| = O(\gamma \ln^4 N))$, where γ is the sparsity of x in the basis F, or say, the number of non-zero coefficients of θ, then we can reconstruct x by solving

$$\min_\theta \|\theta\|_1 \ s.t. \ y = F_\Omega \theta,$$

where F_Ω is a submatrix of F obtained by selecting the rows corresponding to Ω and we can get $x = F\theta$. F_Ω is the so-called "partial Fourier ensemble". Analogously, if U is the GFT basis, then we call U_Ω the partial Graph Fourier ensemble. One direct question one might ask is: as the GFT is considered the "Fourier" basis for signals supported on graphs,

can the partial Graph Fourier ensemble be similarly treated as a CS sensing matrix? In our scenario, it is straightforward to see that the GFT basis satisfies the first condition. So the main problem remaining here is whether the coherence of the GFT matrix is uniformly bounded?

Unfortunately, it is not always guaranteed. For example, circulant graphs will generate eigenbasis with uniformly bounded entries while more general graphs like KNN graphs or $\epsilon-$graphs fail with largest entries close to 1. However, due to the discussion in GFT, we know that KNN graphs or $\epsilon-$graphs are good structures for graph signals compression. Hence, we are very interested in the distribution of the entries in their GFT basis, which implies the coherence of the matrices.

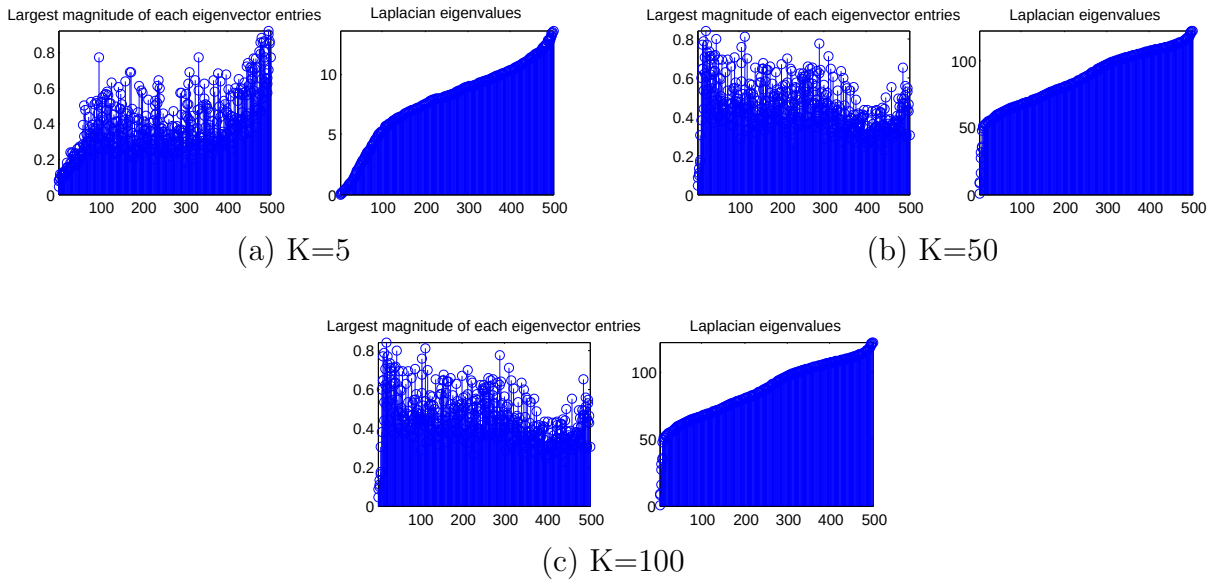


Fig. 4.1 This figure plots the entry with largest magnitude of the entries in each eigenvector. The Graph Fourier basis is generated by extracting the eigenbasis of a symmetric KNN graph. We denote k as the number of neighbors for a KNN graph.

Fig. 4.1 plots the largest components of each column for a GFT basis and its corresponding eigenvalue distribution. X-axis represents the index of the eigenvalues with sorted order and y-axis stands for the largest magnitude of the eigenvector entry for the left side figures. For the right side figures, the y-axis show the magnitude of the eigenvalues. The GFT basis is formulated by obtaining the Laplacian eigenvector of a KNN graph

with $K = 5$, $K = 50$, $K = 100$. The KNN graph is constructed based on 500 uniformly distributed nodes. Clearly, the conventional coherence is close to 1 since this matrix is not uniformly bounded. In order to delve into more details about how the entries of the GFT basis are distributed, we generalize the definition of coherence as follows:

Definition 17. Define $\mu_\Phi(T) = \max_{i,j} |[\Phi_T]_{i,j}|$ to be the coherence of the matrix Φ_T, where T is a subset of $\{1, 2, \cdots, N\}$ and Φ_T is the submatrix obtained by selecting the columns of Φ corresponding to T. If $T = \{1, 2, \cdots, N\}$, then $\mu_\Phi(T)$ is equivalent to μ. In some part of this paper, we abbreviate $\mu_\Phi(T)$ as $\mu(T)$.

It is worth noting that in this specific example, $\mu_U(T)$ is bounded when U_T corresponds to the eigenvectors whose associated eigenvalues are small, even if the coherence of the whole matrix is not bounded by $O(\frac{1}{\sqrt{N}})$. A natural question is: does such a trend exist for all the graphs? Or say, is $\mu_U(T)$ bounded by $O(\frac{1}{\sqrt{N}})$ for all kinds of graphs?

Fig. 4.1 already provides with a negative answer. However, such phenomenon exists for some graphs. Fig. 4.1 also provides us with an observation that the largest magnitude of each eigenvector entries is correlated to its eigenvalue distribution. Such phenomenon can be explained by the anslysis in GFT: we regard $\|x\|_G = x^T L x = \sum_{(i,j)\in E}(x(i) - x(j))^2$ as the smoothness of signal x on a graph. If we also consider the eigenvector u_i as a graph signal and replace x with u_i, it is easy to check that $\|u_i\|_G = \lambda_i$, where λ_i is the eigenvalue corresponds to u_i, i.e. we can say that λ_i describes the smoothness of eigenvector u_i. If λ_i is far smaller than λ_{N-1}, then we say that u_i has a small total variation. Intuitively, a signal with small total variation tends to be smooth, i.e., each entry supported on the graph node tends to be close to other neighboring entries and hence its entries have small upper bound for the magnitude since their total energy are normalized.

One explicit example would be the first eigenvector: since $\lambda_0 = 0$, u_0 is perfectly smooth, or say, is the DC component of the Graph Fourier basis. Hence, u_0 has a small total variation no matter what graphs. In Fig. 4.1(a), we can also see that the the largest entry of each eigenvector keeps increasing until i is around 100 and λ_i is large enough and can no longer affect the distribution of the eigenvector entries. Based on the above discussion and observation, we know that the graphs with adequately small eigenvalues in low frequencies may provide bounded $\mu_U(T)$. An opposite example is the dense graph shown by Fig. 4.1(c) since the eigenvalues except λ_i where $i < 5$ are equally very large, the largest magnitude of the entry for the eigenvectors is still large when $i > 5$. The study

in Chapter 3 provides us with the intuition why small Laplacian eigenvalue corresponds to eigenvector with small magnitude of entries but does not provide us with a accurate upper bound for the magnitude of each entry. However, in academia, there are plenty of researchers focusing on the behavior of the Laplacian eigenvalues while less efforts have been put on that of the eigenvector. The discussion of this issue is without the scope of this paper, but will be our future research lines.

4.3 Compressed Sensing via Graph Fourier Transform Basis

The last section show that using a KNN graph with a proper choice of K, we can have an underlying graph whose first few eigenvalues $\lambda_i \ll \lambda_{N-1}$ and their corresponding eigenvector entries are small. We use $T_1 = \{1, 2, \cdots, \gamma\}$ to denote the set of the index of such eigenvectors. In this section, we shows that for such graphs, as long as the signal to be reconstructed is linearly compressible, i.e., most of its energy lies in the low frequency eigenvectors, a stable recovery is still guaranteed although the overall coherence of U is unbounded. To give an intuition about this result, we can first consider a sparse signal here. If the nonzero entries of the original signal have a fixed support T and Φ is the sensing matrix, then the behavior of submatrix Φ_{T^c} will not affect the recovery process; i.e., we merely require $\mu_\Phi(T) = O(\frac{1}{\sqrt{N}})$. The same conclusion can be generalized to smooth signals supported on graphs by following the same reason: Since most of the energy of the smooth signal is supported on the set T_1, the coherence of the matrix $\Phi_{T_1{}^c}$ is no longer important.

In order to control the isometry property of a fixed set, we first define a special case of model-based RIP:

Definition 18. A matrix Φ has the L_γ-Restricted Isometry Property ($L_\gamma-$RIP) with constant δ_γ if for all θ_γ, we have

$$(1 - \delta_\gamma)\|\theta_\gamma\|_2^2 \leq \|\Phi\theta_\gamma\|_2^2 \leq (1 + \delta_\gamma)\|\theta_\gamma\|_2^2,$$

where θ_γ is a γ sparse signal whose entries are zero except the first γ ones.

The $L_\gamma-$RIP is much weaker than the conventional RIP since conventional RIP requires the inequality holds for γ sparse signal with all possible $\binom{N}{\gamma}$ supports while the $L_\gamma-$RIP only requires it for one support. Hence, the $L_\gamma-$RIP is utilized to deal with linear compressible

signals since the first γ coefficients are likely to contain most of the signal energy. Moreover, if $\mu(T_1) = O(\frac{1}{\sqrt{N}})$, then we can exploit the $L_\gamma-$RIP to guarantee a perfect signal recovery for all the signals whose non-zero entries are supported on the low frequency eigenvectors. The $L_\gamma-$RIP is a special case of model based RIP since we know exactly how the energy of the signal decays and where the residual space is located. Likewise, we also need a tool to deal with how will the small non-zero entries outside $L_{1,\gamma}$ behaves. Due to the fact that the coefficients outside $L_{1,\gamma}$ have very small magnitudes, we can relax the conventional RIP to control their non-isometry property. The following definition of RAmP [10] is a remedy counterpart of RIP in this case.

Definition 19. Given a signal x, a matrix Φ has the $(\epsilon_\gamma, r)-$ restricted amplification property(RAmP) for the linear residual subspaces $L_{j,\gamma}$ of x if

$$\|\Phi u\|_2^2 \leq (1+\epsilon_\gamma) j^{2r} \|u\|_2^2$$

for any $u \in L_{j,\gamma}$ for each $1 \leq j \leq \lceil N/\gamma \rceil$.

The $(\epsilon_\gamma, r)-$RAmP can be regarded as a generalized version of RIP. When $j = 1$, then it is just the upper bound in RIP. Since $L_\gamma-$RIP cannot deal with the signals which have non-zero coefficients outside $L_{1,\gamma}$. For linear compressible signals, when j becomes larger, $\|u\|_2$ for $u \in L_{j,\gamma}$ will become smaller and thus we allow a higher upper bound for $\|\Phi u\|_2^2$.

The recovery algorithms for model based compressive sensing can be built based on CoSamp [45]. The CoSamp algorithm iteratively seeks to find the optimal supports where the residual signal lies in among all $\binom{N}{\gamma}$ subspaces of a N dimensional signal with γ sparsity. On the other hand, the model based compressive sensing takes advantage of the prior knowledge of the signal structure and reduces the number of possible subspaces significantly. While for our cases, the signal we assume is linearly compressible, which is a very special case of model based compressed sensing. It means that the signal can be well approximated by the linear approximation, i.e., the first γ GFT coefficients will contain most of the energy of the signal if γ is adequately large. Thus, instead of detecting the $\binom{N}{\gamma}$ possible subspaces, we can rely on such prior knowledge and utilize a simple least square estimator, which deterministically estimates the first γ GFT coefficients while discards the other small ones.

More specifically, if Φ is the sensing matrix by randomly selecting a subset of the rows from U, i.e., $\Phi = U_\Omega$, let Φ_γ denote the sub-matrix of Φ containing the first γ columns.

For the case that $\Omega > \gamma$, then this is an over-determined system. We can now give the least square estimate that reconstruct the first γ coefficients as $\Phi_\gamma^\dagger y$ where $y = \Phi\theta$ is the measurement and $\Phi_\gamma^\dagger$ is the Moore-Penrose pseudo inverse [49] of Φ_γ. Since $\Phi_\gamma^\dagger y$ is a γ dimensional vector, we need to fill the other $N - \gamma$ coefficients with zero. Accordingly, the formal definition of the least square estimator is shown below:

$$\hat{\theta}(i) = \begin{cases} (\Phi_\gamma^\dagger y)(i) & : \quad i = 1, 2, \cdots, \gamma \\ 0 & : \quad \text{otherwise} \end{cases}$$

And $\hat{x} = U\hat{\theta}$ is the estimate of x. It is worth pointing out that $\Phi_\gamma^\dagger y$ is actually the least square estimates for the first γ entries of θ. And since $x = U\theta$ and $\Phi = U_\Omega$, $y(i) = x(j_i)$ where $\{j_1, j_2, \cdots, j_M\}$ is the index set Ω. With such estimator, we can achieve the following performance guarantee:

Theorem 4.3.1. *Let $x \in \mathbb{L}_s$ be an s-linear compressible signal and $\theta = U^T x$. Also let T_j be defined in Definition 16. If $\mu(T_j) \leq C \cdot j^{s-1}$ for all $j = 1, \cdots, \lceil N/\gamma \rceil$ and some $C > 0$, and if the number of measurements M obeys $M \geq Const \cdot \gamma \cdot \ln(\frac{\gamma}{\delta})$ for some $\delta > 0$, then with probability $1 - \delta$, the estimate $\hat{\theta}$ obtained from the least square estimator satisfies*

$$\frac{1}{\sqrt{2}}\|\theta - \theta_\gamma\|_2 \leq \|\theta - \hat{\theta}\|_2^2 \leq \|\theta - \theta_\gamma\|_2 + C \cdot S\gamma^{-s} ln\lceil\frac{N}{\gamma}\rceil$$

where $C = \frac{C_s\sqrt{1+\epsilon_\gamma}}{\sqrt{1-\delta_\gamma}}$.

The theorem claims that if the entries of the original signal decay quickly, we can still guarantee a stable recovery when the coherence $\mu(T_j)$ keeps increasing for larger j. Actually, we allow $\mu(T_j)$ to become unbounded if the entries of the original signals supported on T_j are small. The above theorem explains that why the partial Graph Fourier ensemble works as a sensing matrix for smooth signals supported on graphs. This is because smooth signals supported on graphs are linear compressible, i.e., most of the large Graph Fourier coefficients are located in the "low frequency component" while those components have relatively low coherence. The full proof of this result is given below. The methodology of the proof is mainly based on [18], [10] and [19].

In order to prove the above result, we first need to determine the property of the sensing matrix such that the $(\epsilon_\gamma, r)-$RAmP is satisfied.

Theorem 4.3.2. *Let $\Phi = \frac{\sqrt{N}}{\sqrt{M}} U_\Omega$ be an $M \times N$ sensing matrix by selecting rows from U, where Ω is the subset of the measurement domain of size $|\Omega| = M$. Fix a subset T of the signal domain. Suppose that the number of measurements M obeys:*

$$M \geq C_3 |T| \ln(\frac{|T|}{\delta})(\frac{\mu_\Phi(T)}{j^r})^2 \tag{4.1}$$

for some positive constant C_3 and with proper choice of ϵ_γ and r. Then, with probability $1-\delta$, the matrix Φ satisfies

$$\|\Phi u\|_2^2 \leq (1+\epsilon_\gamma) j^{2r} \|u\|_2^2. \tag{4.2}$$

for any $u \in L_{j,|T|}$ for each $1 \leq j \leq \lceil N/|T| \rceil$.

The proof of this theorem is mainly based on the techniques in [18] and is included in Appendix. This theorem immediately gives the following corollary:

Corollary 4.3.3. *Let Φ be the same setting described in Theorem 4.3.2. If $\mu_U(T_j) \leq C \cdot \frac{j^r}{\sqrt{N}}$ for all $j = 1, \cdots, \lceil N/\gamma \rceil$ and if the number of measurements M obeys*

$$M \geq Const \cdot \gamma \cdot \ln(\frac{\gamma}{\delta})$$

Then with probability $1-\delta$, the measurement matrix Φ satisfies the $(\epsilon_\gamma, r)-RAmP$ for the linear residual subspaces $L_{j,\gamma}$, where $\gamma = |T_1|$.

Proof. Let the dimension of T in Theorem 4.3.2 be equal to γ. Since $\mu_\Phi(T) = \sqrt{N}\mu_U(T)$ and $\frac{\sqrt{N}\mu_\Phi(T)}{j^r}$ is upper bounded by some constant C, by letting $M \geq C_3 C^2 |T| \ln(\frac{|T|}{\delta})$ gives the corollary. □

Corollary 4.3.4. *Let Φ be the same setting described in Theorem 4.3.2. If $\mu_U(T_1) \leq \frac{C}{\sqrt{N}}$ and the number of measurements M obeys*

$$M \geq Const \cdot \gamma \cdot \ln(\frac{\gamma}{\delta})$$

Then with probability $1-\delta$, the measurement matrix Φ satisfies the $L_\gamma-RIP$ where $\gamma = |T_1|$.

Proof. When we consider the situation $j = 1$ in Theorem 4.3.2, the proof will directly give Corollary 4.3.4. □

Theorem 4.3.2 along with Corollary 4.3.3 provide us with the implication that the coherence of matrix Φ does not have to be uniformly bounded as required in the conventional compressed sensing literatures [18, 22, 54]. More specifically, if the coefficients which are supported on a certain residual space $L_{j,\gamma}$ are quite small, we allow the corresponding coherence $\mu(T_j)$ to be larger. The next theorem shows that with $(\epsilon_\gamma, r)-$RAmP, $\|\Phi(x-x_\gamma)\|_2$ is upper bounded. Since linear compressible signals are just one special case of model based compressible signal, the following theorem stems from [10] directly.

Theorem 4.3.5. *Let $x \in \mathbb{L}_s$ be an s-linear compressible signal and θ its GFT. If Φ has the $(\epsilon_\gamma, r)-$RAmP for the linear residual subspaces $L_{j,\gamma}$ and $r = s-1$, then we have*

$$\|\Phi(\theta - \theta_\gamma)\|_2 \leq C_s\sqrt{1+\epsilon_\gamma} S\gamma^{-s}\ln\lceil\frac{N}{\gamma}\rceil. \tag{4.3}$$

where $C_s = 2^s + 1$.

The detailed proof of this theorem is basically the same to Theorem 3 in [10]. The upper bound we have here from this theorem can be utilized to derive the following upper bound of the least square estimator.

Theorem 4.3.6. *Let $x \in \mathbb{L}_s$ be an s-linear compressible signal. If Φ has the $L_\gamma-$RIP and the $(\epsilon_\gamma, s-1)-$RAmP, then the solution $\hat{\theta}$ obtained from least square estimator satisfies*

$$\frac{1}{\sqrt{2}}\|\theta - \theta_\gamma\|_2 \leq \|\theta - \hat{\theta}\|_2 \leq \|\theta - \theta_\gamma\|_2 + C\cdot S\gamma^{-s}\ln\lceil\frac{N}{\gamma}\rceil$$

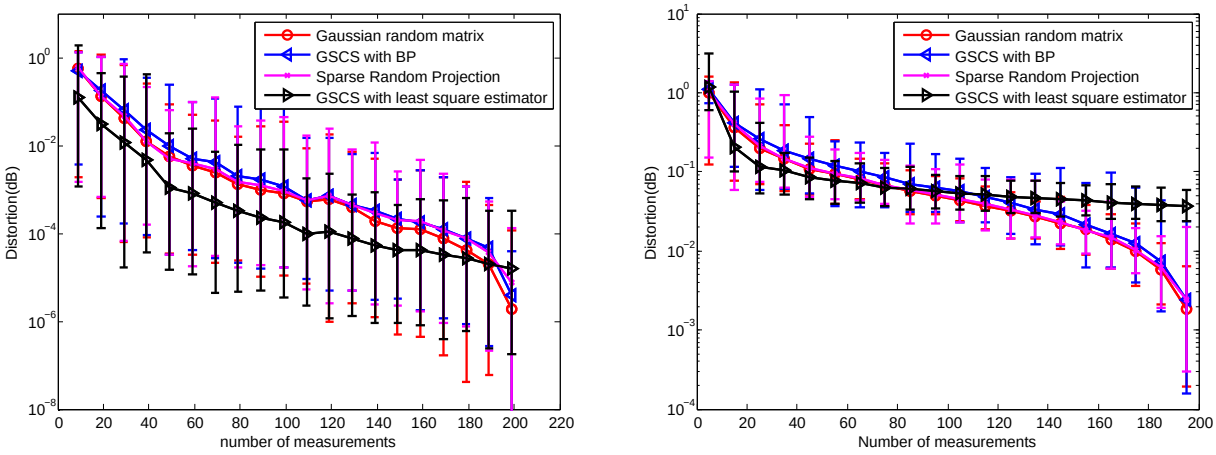
where $C = \frac{C_s\sqrt{1+\epsilon_\gamma}}{\sqrt{1-\delta_\gamma}}$.

The proof of above theorem, which is included in the Appendix, is mainly based on Theorem 4.3.5 and certain elementary properties of matrix norms. With the random sampling scheme and $\mu(T_j) \leq C\cdot j^r$, we can achieve Corollary 4.3.3 and Corollary 4.3.4 easily. Combining the Corollary 4.3.3 and Corollary 4.3.4, we know that $L_{1,\gamma}-$RIP and the $(\epsilon_\gamma, s-1)-$RAmP is satisfied when $M \geq Const\cdot\gamma\cdot\ln(\frac{\gamma}{\delta})$ for some small δ. Then, for a linear compressible signal x and a sensing matrix Φ with $L_{1,\gamma}-$RIP and the $(\epsilon_\gamma, s-1)-$RAmP, we can obtain Theorem 4.3.1 by applying those conditions in Theorem 4.3.6.

4.4 Simulations

In this section, we use synthesized data to verify our theoretical analysis for GSCS with least square estimator. Fig.3.7 shows the performance of GSCS with Basis Pursuit(BP) and GSCS with least square estimator as compared to CS using an i.i.d. Gaussian sensing matrix and sparse random projection. For sparse random projection [62], we set the sensing matrix as:

$$\Phi_{ij} = \begin{cases} 1 & : \text{ with prob. } \frac{\ln N}{2N} \\ 0 & : \text{ with prob. } 1 - \frac{\ln N}{N} \\ -1 & : \text{ with prob. } \frac{\ln N}{2N} \end{cases}$$



(a) (b)

Fig. 4.2 This figure illustrates the performance of GSCS with BP and with least square estimator, conventional CS via i.i.d. Gaussian random matrix and sparse random projection on two different synthesized data sets. (a) utilize the data which is strictly linear compressible on GFT domain while (b) get the GFT coefficients by projecting the signal on the GFT basis constructed on the noisy version of the original signal. In both of the two figures, the averaged distortion is plotted while the best and worst performance is denoted by the error bar.

We use two different kinds of synthesized data here. The signal is generated by two methods:

(1) For Fig.4.2(a), we first generate a 200×1 Gaussian random vector x and then scale its nth entry by a factor $\frac{1}{n^s}$. It is easy to see that the larger s is, the more compressible

the signal will be. In this experiment, we set $s = 2$. We use the BPsolver routine of SparseLab2.1 [1] to solve the ℓ_1 recovery problem. For least square estimator, we set the parameter $\gamma = round(\frac{M}{7})$ in all the experiments here. The algorithm is run for 200 trials to get the best, worst and average performance. Such synthesized data set conforms strictly to our signal model of linear compressibility. The sensing matrix is generated randomly selecting the rows of a GFT basis from a KNN graph, which is constructed based on the nodes with uniform distribution.

(2) Fig.4.2(a) shows the performance of an ideal signal in order to verify our theory while for Fig.4.2(b), we use the data set which is not ideally compressible. We first generate a 200×1 (0,1) Gaussian distributed random vector x. From the experiments in Chapter 3, we can see that the GFT coefficients of Gaussian distributed signals do not decay very fast, which can describe certain situations in real world data sets. Moreover, we don't construct the KNN graph directly on x. Instead, we construct the graph based on $x + n$, where n is a 200×1 i.i.d. $(0, 0.04)$ Gaussian random noise and hence obtain its sensing matrix by random selecting the rows. Such data set is exploited to simulate the common case in real application that we don't have the direct information of x in prior, which means we might not be able to obtain an optimal underlying graph. The other settings are the same as those of the previous ones. The algorithm is run for 200 trials to get the best, worst and average performance.

In the simulation, we merely keep the signals fixed in each iteration. The sensing matrices are randomly generated for different number of measurements in each trial. From Figure 4.2(a), we can see that for a linear 2-compressible signal, the GSCS with least square estimator outperforms all the other methods when $M \ll N$. Its performance is only worse than that of others when $M \to N$. This is easy to understand since the recovery error for least square estimator has an lower bound. It's worth noting that the GSCS with BP performs essentially as well as the Gaussian sensing matrix, on average. The worst case performs slightly worse than that of the Gaussian matrix. None theoretical analysis has been made to prove why ℓ_1 decoding works for non-uniformly bounded orthogonal matrices and this will be one of our future lines of research. The performance of GSCS with least square estimator in Fig. 4.2(b) is worse than that of Fig. 4.2(a). The least square estimator only outperforms other method when the number of measurements is less than 80. And due to the poorly compressible signal and noise disturbance, the distortion for all the four methods decays fairly slow with the increasing number of measurements.

The above two data sets are used to simulate the ideal and unexpected cases while the following experiments will make use of some real world data sets.

4.5 Discussion

The idea of GSCS is originated from the ideas of signal processing, especially from CS. we point out connections to recent related works: Pesenson [47, 48] studies sampling theorems for "bandlimited functions on graphs, results which may be useful in constructing critically sampled transforms. They also proves that if certain conditions are satisfied, "bandlimited functions on graphs can be uniquely determined with the knowledge of a portion of the nodes. In their work, "bandlimited" actually means that the functions only contains "low frequency" components. Compared with their work, we merely assume "linear compressible" signals, which is more general than the "bandlimited functions. Consequently, we provide an upper bound for reconstruction error other than perfect recovery as theirs.

In parallel, M. Belkin [11] developed similar techniques for classification under the assumption that the data resides on a low dimensional manifold within a high dimensional representation space. Their approach is highly correlated with ours. Accordingly, it is worth pointing out the differences and contributions of our work with regard to theirs. First, in our work, the problem we consider here is an estimation problem while classification for theirs. Clearly, our problem is much more complex here. Second, in their work, they provide us with certain theoretical justification for the methods while lack a thorough analysis while we provide detailed analysis about the performance bound. We believe that our results can easily be extended to cover their scenarios. Third, to our best knowledge, GSCS is the first to utilize such idea for signal estimation.

Chapter 5

Graph Spectral Compressed Sensing for Wireless Sensor Networks

GSCS turns out to be a very useful data gathering technique especially for Wireless Sensor Network (WSN). For a lot of WSN applications, the signals measured are likely to be correlated either spatially or temporally, i.e., we can find an appropriate transform domain where the signals are compressible. In order to reduce power consumption and bandwidth resources (or query latency), we want to pre-process the data so that only $\gamma \ll N$ number of measurements are collected, where N is the total number of sensor nodes. Computing a deterministic transform domain and locating the K largest transform coefficient is very difficult to accomplish efficiency in a distributed manner.

GSCS provides an alternative solution to the above issue. In this chapter, we propose two algorithms respectively to deal with both spatially and temporally correlated signals sampled by WSN. We show that if the sampled signals are correlated spatially or temporally, we can construct an underlying graph where the supported signal is smooth. Moreover, if we project the signals onto the corresponding Graph Fourier Transform (GFT) basis, the coefficients are linearly compressible. In this setting, only a small random portion of the sensor nodes need to be activated to sample and transmit the measurements. Both the power consumption and bandwidth resources (or query latency) are reduced.

5.1 Wireless Sensor Networks

As have been introduced in Chapter 2, WSN has a promising capability to monitor the physical world via a spatially distributed network of small and inexpensive wireless sensors. For many WSN applications, especially field monitoring, the signals measured are likely to be correlated either spatially or temporally; i.e., we can find an appropriate transform domain where the signals are compressible. WSNs are characterized by having simple battery-powered wireless nodes with limited energy and communication resources. In order to reduce power consumption and conserve bandwidth (or query latency), it is desirable to apply the philosophy of compressed sensing since we can directly gather a reduced number of informative measurements rather than gathering a large number of redundant measurements.

CS theory shows that, when our signal is sparse or compressible in the transform domain, we can utilize $M = O(\gamma \ln N)$ random projections of the data to estimate the original signal with an error very close to that of the optimal approximation using the γ largest transform coefficients. Many efforts [6, 7] have been made along this line of research. However, the conventional CS sensing matrices like i.i.d. Gaussian or Bernoulli are expensive to compute and each random measurement requires cooperation and communications among all N sensors. Hence, the overall number of transmission via conventional CS will be MN, which results in high power consumption and a complicated design of the communication architecture. Wang et al. [62] solve this problem by proposing sparse random sensing matrices, which significantly reduces the communication overhead. Different from their approach, we utilize the technique called GSCS, which has been introduced earlier, for data gathering via WSNs.

In contrast to previous work, we focus on the particular case of estimating signals which are smooth with respect to a graph. We show by experiments on real world data that if the sampled signals are correlated spatially or temporally, we can construct an underlying graph such that the signal is compressible in a corresponding transform domain. More specifically, if we project signals onto the corresponding GFT basis, the coefficients are likely to be linearly compressible. According to the theory of GSCS, only a small random portion of the sensor nodes need to be activated to sample and transmit measurements, and the original signal can be recovered via least square estimator with small distortion. Consequently, both power consumption, bandwidth usage, and latency are reduced. It

is worth noting that we also try ℓ_1 programming during those experiments, which gives surprisingly positive results. The main contributions of applying GSCS to sensor networks are twofold:

First, to our best knowledge, most of the previous literatures [6, 7, 62] considering data compression or field estimation assume that the signals sampled are compressible in certain orthogonal domains (e.g., 2-d wavelets). These methods are inspired by image processing and treat each sensor node as a single pixel in an image. Accordingly, they assume the sensor nodes are in a regular structure, e.g., 2-d grid. However, in real world applications, sensor nodes may not always exhibit such a rigid structure. The proposed method overcomes this problem by exploiting the GFT, which is suitable for networks with general topology.

Second, much of the existing literature [6, 7, 37] consider Gaussian or Bernoulli distributed random matrices as the sensing matrix. As mentioned above, those matrices have two main disadvantages. Not only does every node have to randomly generate the entries of the sensing matrix, but also the implementation of noisy projections requires more cooperations and communications among sensors. The method we propose successfully solves such a dilemma between bandwidth resources (or query latency) and energy consumption. Both of them can be significantly reduced in our scheme.

The next two sections introduce the two algorithms for spatially correlated signals and temporally correlated signals respectively.

5.2 Spatially Correlated Signals

Spatial correlation describes the correlation between signals at different points in space. Such concept is very common in image processing and also in environment monitoring. In our case, when we distribute a number of sensor nodes in a certain field to acquire its field information like temperature, pressure, or solar radiation, the signal is likely to be spatially correlated since the reading of each sensor is highly correlated with its location and such signals can be regarded as smooth signals since the neighbor nodes tend to share similar values. We focus on such a scenario and propose a simple algorithms for data gathering with lossy compression.

Let $x \in \mathbb{R}^N$ be the data vector for a WSN with N nodes; i.e., each entry x_i is the data reading from the corresponding sensor node, i. Here we wish to sample $M \ll N$ nodes to recover the original signal x. Assume we have perfect knowledge about where each sensor

node is located. We can utilize the location information to generate a symmetric KNN graph of the WSN. According to the analysis in Section 3, we have to select the parameter K carefully, where K here is the number of neighbors each node should be connected to. K should be chosen as small as possible while still keeping the graph well-connected. After obtaining the underlying graph, we can get its Laplacian eigenbasis U. We randomly select $M \ll N$ nodes to report their data to the sink while the other $N - M$ sensors remain in a sleep mode. Denote the set of awakened sensors as Ω and $y \in \mathbb{R}^M$ as the transmitted measurement vector. Then, we have the sensing matrix U_Ω and the measurements y. After the fusion center obtains the measurement y, we can estimate the original signal x by exploiting the least square estimator described in the last chapter to first recover the GFT θ:

$$\hat{\theta}(i) = \begin{cases} (U_{\Omega\gamma}^{\dagger} y)(i) & : \quad i = 1, 2, \cdots, \gamma \\ 0 & : \quad \text{otherwise} \end{cases}$$

where $U_{\Omega\gamma}$ is the sub matrix of U_γ by selecting rows corresponding to the index set Ω. We can obtain the final estimate of x by $\hat{x} = U\hat{\theta}$. Moreover, in the experiments, we also try to estimate the original signal by solving the ℓ_1 optimization problem:

$$\hat{\theta} = \arg\min_{\theta} \|\theta\|_1 \ s.t. \ y = U_\Omega \theta$$

and similarly obtain $\hat{x} = U\hat{\theta}$.

5.3 Temporally Correlated Signals

Temporal correlation describes the predictable relationship between signals observed at different moments in time. In applications of speech or environment monitoring, temporally correlated signals are very common. In our scenario, we distribute a number of sensor nodes in a certain field for data gathering. Since the location of each sensor node is fixed and the readings of each nodes do not change very fast from the previous readings, i.e., each signal is highly correlated with its previous states. If we construct an underlying graph based on the information of its previous states, it is very likely that the current signal is smooth with regard to the graph because of the temporal correlation. Accordingly, a simple online estimation algorithm is proposed for such scenario:

Let $x_t \in \mathbb{R}^N$ be the data samples from a WSN at time instant t, where the network

consists of N sensor nodes. The data is collected via a certain sampling rate at discrete times $t = 1, 2, \cdots$. Here we propose an online estimation algorithm to iteratively estimate the readings x_t based on previous estimates of $x_{t-1}, \ldots, x_1$. We show that merely sampling a small portion of the sensor nodes at each iteration, we can still maintain a stable recovery. The general idea of the algorithm is described as follows:

(1) Assume the central station has already obtained all the estimates $\hat{x}_{t-1}, \ldots, \hat{x}_1$ of the previous readings. We calculate the mean of the r most recent estimates: $\bar{x}_t = \frac{1}{r}\sum_{k=t-r}^{t-1} \hat{x_k}$.

(2) Next we generate a KNN graph G based on $\bar{x}$ by following the principles in [**?**] and obtain its Laplacian matrix U by taking the eigenvalue decomposition Laplacian matrix L corresponding to G.

(3) At the time t, the WSN randomly collects data from a random subset Ω_t of $|\Omega_t| = M \ll N$ sensor nodes. At the fusion center, the received measurements are collected in the M-dimensional vector $y_t = U_{\Omega_t}\theta_t$, where Ω_t is the random sampling subset at time t.

(4) When the fusion center obtains the current measurement vector y_t, it recovers the current estimates $\hat{x}_t$ by using the least square estimator:

$$\hat{\theta}_t(i) = \begin{cases} (U^{\dagger}_{\Omega_t\gamma} y)(i) & : \quad i = 1, 2, \cdots, \gamma \\ 0 & : \quad \text{otherwise} \end{cases}$$

and reconstruct x_t with $\hat{x}_t = U\hat{\theta}_t$. The definition of $U_{\Omega_t\gamma}$ is similar to the least square estimator for spatially correlated signals. Likewise, we also try solving the ℓ_1 optimization problem in this case:

$$\hat{\theta}_t = \arg\min_{\theta} \|\theta\|_1 \ s.t. \ y_t = U_{\Omega_t}\theta_t$$

and obtain $\hat{x_t} = U\hat{\theta}_t$

(5) Set $t = t + 1$ and start a new iteration from step 1.

5.4 Power, Latency and Distortion

For a linear compressible signal, the upper bound shows that $\|\theta - \theta_\gamma\|_2 \leq Const \cdot S\gamma^{-s}$. Combining this with $\ln\lceil\frac{N}{\gamma}\rceil \leq \ln N$, we can see that the MSE $D = \|x - \hat{x}\|_2 \leq Const \cdot \ln N \cdot \gamma^{-s}$. If the signals decays fast, i.e., s is large, then the distortion will have a small upper bound. Moreover, if we increase the number of measurement M, a larger γ could be found to satisfy the condition $M \geq Const \cdot \gamma \cdot \ln\gamma$ and consequently, the distortion will

be reduced. Since the fusion center has to first receive all M measurements and then start recovery process, it will cost the WSN M units of bandwidth and latency.

Different from the conventional CS paradigm, GSCS is able to reduce the number of communications for data gathering significantly. If we adopt the architecture described in [6], for a WSN with N nodes, each sensor have to transmit M_1 times in order to generate the measurement vector y, i.e., the total number of transmissions in the WSN is $M_1 N$. However, by exploiting GSCS, we merely require M_2 nodes to transmit their readings where M_2 is the number of measurements for GSCS to achieve the same reconstruction error, i.e., the total number of transmissions in the WSN is M_2. For a large scale WSN, the reduction of the energy consumption is huge since $M_2 \ll NM_1$. In the next section, we will show by experiment that to achieve the same distortion, M_2 for GSCS is quite close to M_1 for certain real world data sets, which also implies that conventional CS will consume N times more number of transmissions than GSCS does and thus our method is energy efficient.

5.5 Experiments

In this section, we still utilize the CIMIS data sets. We run GSCS on solar radiation data across multiple sensors and multiple time points. We use our proposed algorithms for WSNs to check how GSCS works for WSNs. We choose KNN graph for the underlying graphs. The reason for such choice is listed: First, KNN graphs are more robust to the distribution of the signals while $\epsilon-$graphs are sensitive to nonuniform distribution. We have to adjust the parameter ϵ to fit different signals. Second, KNN graphs are more likely to generate small coherence of low frequency components when compared to least weighting graphs. From the above experience, we know that the best choice of K is in the range of 5 to 10 for constructing a KNN graph. Hence, in the following experiments, we set $K = 7$ all the time. We will compare the performance of GSCS, conventional CS sensing matrix and the sparse random projection [62] method.

It is easy to see if we maintain the same number of measurements for the above three methods, then the number of transmission required for each node of GSCS, sparse random projection and Gaussian random matrix will be $\Theta(1)$, $\Theta(\ln N)$, $\Theta(N)$ respectively with same number of measurements. Thus, the remaining question is for the same number of measurements, how good is the recovery accuracy of GSCS when compared with the other

two methods.

5.5.1 Spatially Correlated Signals

First we use the solar radiation data of one day which contains 135 readings from different weather stations. Since we know the exact coordinates of all those weather stations, we can generate a KNN graph based on the geological information and obtain its GFT basis.

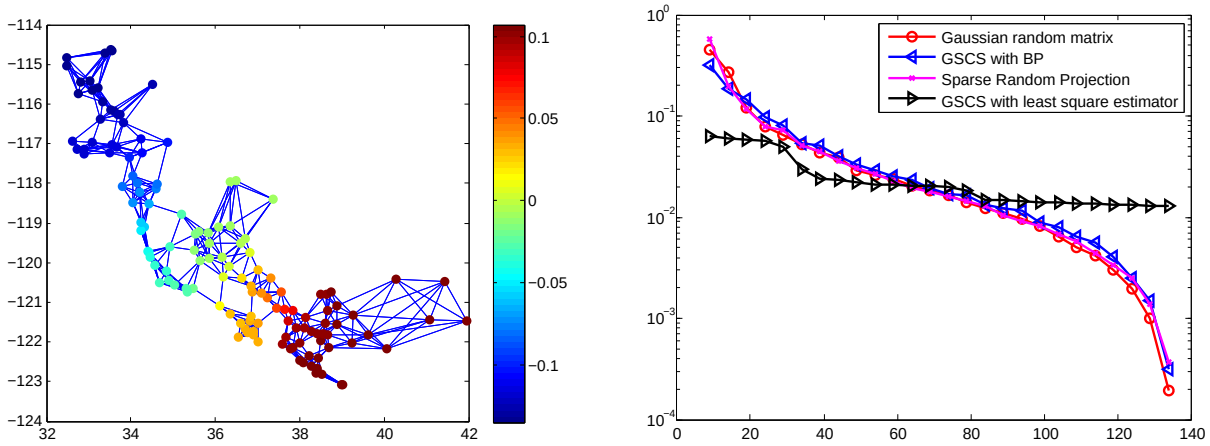


Fig. 5.1 (a) The K-Nearest-Neighbor graph generated using the locations of weather stations in California. We set the number of neighbors for this graph $K = 7$. (b) Performance comparison of GSCS with BP, GSCS with least square estimator, conventional CS with an i.i.d. Gaussian sensing matrix and sparse random projection. The figure plots distortion (mean squared error) as a function of the number of measurements, M.

The resulting network is shown in Fig. 5.1(a), and Fig. 5.1(b) illustrates that the performance of GSCS with BP is comparable with that of the conventional Gaussian random matrix and sparse random projection while the least square estimator works clearly better than all the other methods when $M \ll N$. The distortion is computed for 200 different times and the average distortion is presented.

5.5.2 Temporally Correlated Signals

Next we test the GSCS algorithm on temporally correlated signals. The data set is also from CIMIS. We use 92 daily readings from each of 117 sensor nodes, corresponding to a period

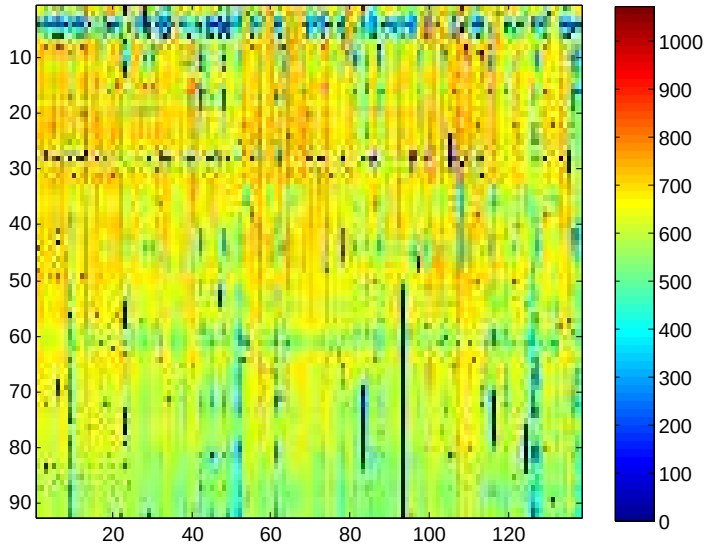
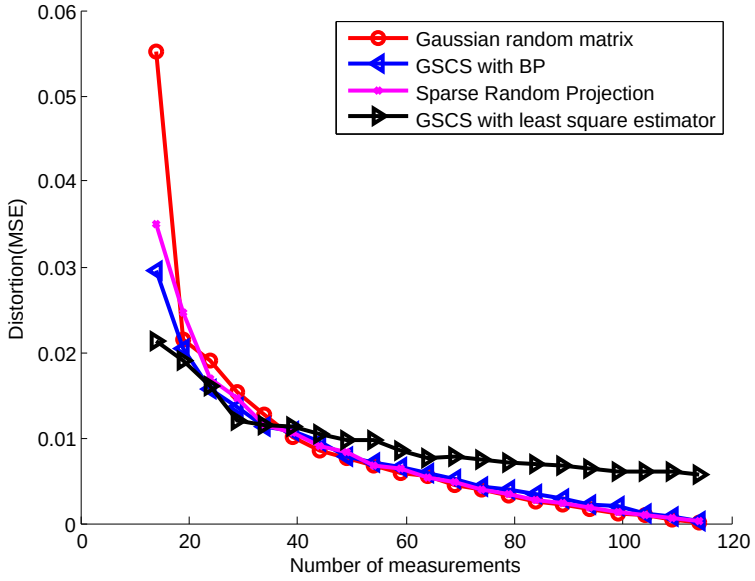


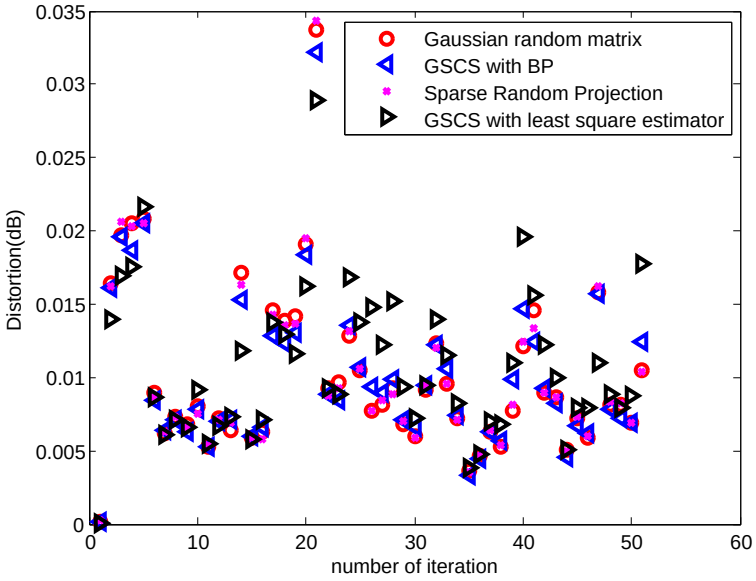
Fig. 5.2 Temporally correlated data set. The horizontal line represents the time of 92 days while the vertical line represents 117 sensor nodes. The color represents the solar radiation readings from each sensor nodes.

of three months. Figure 5.2 illustrates the temporally correlated signal. The horizontal line represents the time of 92 days while the vertical line represents 117 sensor nodes. We can see that this signal is not always well temporally correlated since for some days, there happen certain changes in the weather, which leads to uncorrelated solar radiation readings.

First we set $r = 40$ and let the sensor data of the first 10 days to be fully transmitted to formulate the initial estimated data and obtain its mean of $\bar{x}$ to generate the corresponding KNN graph. For the remaining 52 days we exploit the procedure described in Subsection 5.3 to estimate the original signals. Figure 5.3(a) shows how the number of measurements affects the performance of GSCS. The averaged MSE is around 0.025 when the number of measurements exceeds 20. Figure 5.3(b) gives the MSE for each iteration when we randomly activate 40 nodes to transmit the data. This experiment is run for 100 trials and the average is plotted. By comparing with the original signals shown in Fig. 5.2, we find that the large spikes of the error usually correspond to signals that deviate from the the day before. Compared with other methods, the GSCS with least square estimator does not outperform significantly when $M \ll N$. One main reason here is that some daily readings might change quickly from the past and such signals does not exhibits strict linear compressibility.



(a) Performance comparison of GSCS with BP and with least square estimator, conventional CS sensing matrix and sparse random projecting on temporally correlated data as a function of number of measurements per day.



(b) Mean square error of each iteration for GSCS with BP and with least square estimator, conventional CS sensing matrix and sparse random projecting. The number of measurement M is set to 40..

Fig. 5.3 Performance comparison of GSCS with BP and with least square estimator, conventional CS sensing matrix and sparse random projecting on temporally correlated signals. The parameter K is set to 7.

This is one main disadvantage of the least square estimator: it requires the signals to conform to the linear compressible model strictly since it recovers the signal on a fixed support. In such cases, BP solver might be a better choice. It is worth noting that the least square estimator is computationally much faster than all the other methods while the BP solver is the slowest. This makes least square estimator suitable for some specific online estimation tasks, which requires fast recovery process.

5.6 Discussion

In this chapter, we introduce two algorithms based on GSCS for WSNs to deal with temporally or spatially correlated signals. For spatially correlated signals, GSCS is a general approach for regular or irregularly structured WSNs. For temporally correlated signals, GSCS provides an online estimation technique which iteratively learns the underlying transform domain where the signal is compressible. Both algorithms exhibit great improvement in saving both the energy consumption and bandwidth resources (or latency) since GSCS merely requires a small portion of the whole sensor nodes to sample and transmit the data. Moreover, we use real world data to verify that the GFT basis is suitable for irregular structured sensor topology. Also, the experiment results show that both least square estimator and ℓ_1 decoding work for signal recovery algorithm.

Chapter 6

Conclusion

6.1 Summary and Discussion

Our work analyzes a concept called the Graph Fourier Transform (GFT). To the best of our knowledge, this is the first work to address (i) when we can compress signals supported on graphs using the graph Laplacian eigenbasis, and (ii) on what conditions the graph and signals should satisfy for approximation. We define the smoothness of signals supported on graphs and extend the concept of bounded variation to signals supported on graphs. We also analyze the impact of the distribution of the Laplacian eigenvalues of the underlying graph. It has been shown that in order to obtain the best compressibility of a certain signal, we require two conditions: First, the signal should be smooth with regard to the underlying graph. Second, the underlying graph should have different eigenvalues to represent different frequencies and different distribution of the eigenvalues will result in different behavior of the linear approximation error. In addition to the theoretical discussion about the properties of the GFT, we also provide simulations and experiments for further study. We suggest different approaches of constructing the underlying graph to generate a smooth signal and proper distribution of the eigenvalues. It is worth noting that our work on GFT is not only related to the area of approximation theory, but also highly correlated with manifold learning and semi-supervised learning.

The GFT extends the conventional approximation theory to signals on graphs. On the other hand, we also show that the GFT has further applications as being the sensing matrix of compressed sensing. We have proved that although the entries of the GFT basis are not uniformly bounded, we can still guarantee a stable recovery through an simple least square

estimator for the smooth signals supported on graphs. Different from the conventional partial Fourier ensemble, our approach deals with more specific cases: the smooth signals supported on certain graphs. Such method is called Graph Spectral Compressed Sensing (GSCS). GSCS is very suitable for applications in WSNs since it only need to sample a small portion of the sensor nodes randomly and provides a lossy compressed version of the original signal.

Accordingly, we also introduce two algorithms based on GSCS for WSNs to deal with temporally or spatially correlated signals. For spatially correlated signals, GSCS is a general approach for regular or irregularly structured WSNs. For temporally correlated signals, GSCS provides an online estimation technique which iteratively learns the underlying transform domain where the signal is compressible. Both algorithms exhibit great improvement in saving both the energy consumption and bandwidth resources (or latency) since GSCS merely requires a small portion of the whole sensor nodes to sample and transmit the data,

6.2 Future Work

Since GFT and GSCS are relatively a new realm of study in approximation theory, there still exists several uncleared issues for further development:

From the perspective of approximation theory, we would like to consider a question: Given a signal x, does there exist an optimal GFT basis for least approximation error? Based on the properties of the GFT, we have already known that we desire the graph with increasing eigenvalues. However, is there an optimal distribution of eigenvalues for better approximation? More specifically, when we construct a KNN graph, we know that a very small choice of K and a large one will both lead to larger approximation error. However, what is the optimal choice of K? The result of experiments shows when $K = 5 \sim 10$, we have the smallest approximation error. But is such choice universal? Or under what conditions, such choice is optimal?

From the perspective of graph theory, we are interested in if there are any other graph construction techniques for better compression. One possible way of research is related to chromatic number. The intuition comes from the fact that graphs like ring or 2D grid maintain small chromatic number of 2. Does a graph with small chromatic number have a GFT basis for good compression? If so, graph topologies such as the planar graph might be a good choice since it maintains small chromatic number. Moreover, Nodal's theorem [58]

in graph theory describes the behavior of the Laplacian eigenvectors. Is it related to the "Fourier" properties of the eigenvector? Last but not least, another question concerns about the Laplacian matrix. In our work, we utilized the unnormalized Laplacian matrix, but how about the normalized Laplacian? The normalized Laplacian has some desired properties such as it eliminates the bias on nodes of large degree and its eigenvalues range from 0 to 2. All the above questions need further theoretical study and thus provide one line of future research.

From the applicational perspective, we want to apply GFT and GSCS to more scenarios. In this paper, we concentrate on the case that we construct the underlying graphs given a certain signal. It is shown that such method can be well applied to WSNs and provide the partial Graph Fourier ensemble which GSCS requires. However, sometimes we are interested in the converse situation, i.e., we want to approximate a smooth signal which is supported on a given graph. For example, each router in a certain network records its total amount of data flows and we consider the underlying network as the graph with the readings from routers as the signal. Since the records are highly correlated to its underlying topology, we would like to investigate under what conditions such signal is smooth with regard to the network.

In addition to the above questions, there still exist certain unsolved theoretical issues in this work. When we discuss the coherence of the partial graph Fourier ensemble, we need to analyze the largest magnitude of the entries among each eigenvector. Thanks to the theory about the GFT, we have certain implications about why the coherence of the low frequency components is small. However, there is still no strict mathematical proof about what graphs are likely to have such property. In the area of graph theory, there have been plenty of studies analyzing the distribution of the eigenvalues while not much efforts have been laid on analyzing the distribution of eigenvectors. Hence, this question still remains an open problem. Another unsolved problem is related to GSCS. During experiments and simulations, we find out that the conventional CS decoding ℓ_1 programming still works well for the partial graph Fourier ensemble although such sensing matrix does not satisfy the conventional RIP. We conjecture that the reason of this phenomenon should be the same as that of the least square estimator but we currently lack solid theoretical analysis.

Appendix A

A.1 Proof of Theorem 4.3.2

We first need some tools to determine the property of the sensing matrix such that the $(\epsilon_\gamma, r)-$RAmP is satisfied. The following lemma is tailored from [18] to fit our need.

Lemma A.1.1. *Let Ψ be an $N \times N$ orthogonal matrix obeying $\Psi^T\Psi = NI$. Consider a fix set T and let Ω be a random set sampled using the Bernoulli model and $\mu_\Psi(T) = \max_{j\in T}|\Psi_{i,j}|$. Denote $Y = \frac{1}{M}\Psi^T_{\Omega T}\Psi_{\Omega T} - I$ where I is the identity matrix and $M = |\Omega|$. Then*

$$\mathbb{E}\|Y\| \leq C_R \cdot \mu_\Psi(T)\frac{\sqrt{|T|\log|T|}}{\sqrt{M}} \tag{A.1}$$

and

$$P(|\|Y\| - \mathbb{E}\|Y\|| > t) \leq 3\exp(-\frac{t}{\gamma B}log(1 + \frac{t}{1+\mathbb{E}\|Y\|})), \tag{A.2}$$

where $B \leq \mu^2_\Psi(T)|T|/M$ anb C_R is some small constant.

From this lemma, Candès and Romberg [18] further prove for $x \in \mathbb{R}^N$ be a sequence supported on a fixed set T, $\frac{M}{2}\|x\|_2^2 \leq \|\Psi_\Omega x\|_2^2 \leq \frac{3M}{2}\|x\|_2^2$. If we let $T = L_{j,\gamma}$, then it is exactly the $L_\gamma-$RIP. It is worth noting that there is one minor difference between the lemma here and the original work in [18]. Since in [18], the authors discuss the case where T is fixed but arbitrary, they define the coherence $\mu = \max|\Psi_{i,j}|$. However, in the scenario we are interested in, we merely concern about $L_{1,\gamma}-$RIP. Accordingly, the set T is not arbitrary but $T(L_{1,\gamma})$. Correspondingly, we can replace μ with $\mu_U(T)$ since only U_T is involved. This lemma can also be exploited as a useful tool for verifying the $(\epsilon_\gamma, r)-$RAmP.

Proof. In order to prove the conclusion, it is equivalent to upper the probability that $\Phi = \frac{\sqrt{N}}{\sqrt{M}} U_\Omega$ does not satisfy $\|\Phi u\|_2^2 \leq (1+\epsilon_\gamma) j^{2r} \|u\|_2^2$:

$$P(\|\Phi u\|_2^2 > (1+\epsilon_\gamma) j^{2r} \|u\|_2^2) \leq P(|\|\Phi u\|_2^2 - \|u\|_2^2| > [(1+\epsilon_\gamma) j^{2r} - 1] \|u\|_2^2) \tag{A.3}$$

$$= P(\|\frac{1}{M} \Phi_{\Omega T}^T \Phi_{\Omega T} - I\| > (1+\epsilon_\gamma) j^{2r} - 1) \tag{A.4}$$

Denote $Y = \frac{N}{M} \Phi_{\Omega T}^T \Phi_{\Omega T} - I$. Thus, the problem is now equivalent to bound $P(\|Y\| > (1+\epsilon_\gamma) j^{2r} - 1)$. From Lemma A.1.1, set $t = \frac{(1+\epsilon_\gamma) j^{2r} - 1}{2}$ and since C_R is small and $|T| \ll N$, we can find an M large enough to make $\mathbb{E}\|Y\| \leq t$ if $\frac{(1+\epsilon_\gamma) j^{2r} - 1}{2\mu_\Phi(T)}$ is not very small, then (A.4) is bounded by the righthand of (A.2). Accordingly, we can obtain

$$M \geq \frac{4 C_R^2 \mu^2(T) |T| \log |T|}{[(1+\epsilon_\gamma) j^{2r} - 1]^2} \tag{A.5}$$

which can be simplified as:

$$M \geq C_1 |T| \ln |T| (\frac{\mu(T)}{j^{2r}})^2 \tag{A.6}$$

where $C_1 = 4C_R^2/(1+\epsilon_\gamma)^2$. Since $B \leq \mu^2(T)|T|/M$, (A.2) gives

$$P(\|Y\| > 2t) \leq 3 \exp(-\frac{Mt \cdot \log(1 + \frac{t}{1+t})}{\mu^2(T)\gamma}) \tag{A.7}$$

and let it be bounded by δ which provides the following:

$$M \geq \ln(\frac{3}{\delta}) \frac{\gamma \mu^2(T)|T|}{t \cdot \log(\frac{1+2t}{1+t})} \tag{A.8}$$

where $t = \frac{(1+\epsilon_\gamma) j^{2r} - 1}{2}$. Since it is easy to pick proper ϵ_γ and r to bound $\log(\frac{1+2t}{1+t})$ away from zero, i.e. there exist a constant C such that $0 < C \leq \log(\frac{1+2t}{1+t}) < \log 2$, we can further simplify (A.8) as:

$$M \geq C_2 |T| \ln(\frac{3}{\delta}) (\frac{\mu(T)}{j^r})^2 \tag{A.9}$$

where $C_2 = \frac{\gamma}{(1+\epsilon_\gamma) \log(\frac{1+2t}{1+t})}$. Combining (A.6) with (A.9), we can see that:

$$M \geq \max\{C_1 |T| \ln |T| (\frac{\mu(T)}{j^{2r}})^2, C_2 |T| \ln(\frac{3}{\delta}) (\frac{\mu(T)}{j^r})^2\}, \tag{A.10}$$

which gives the conclusion □

A.2 Proof of Theorem 4.3.5

The proof of this theorem is exactly the same as in [10] since the only difference here is we assume the signal to be linear compressible while their work considers generally compressible signals. We include it here for the completeness of this thesis.

Proof. In this proof, we represent θ_γ as the γ term linear approximation of the original signal θ. To bound $\|\Phi(\theta - \theta_\gamma)\|_2$, we write θ as:

$$\theta = \theta_\gamma + \sum_{2}^{\lceil \frac{N}{\gamma} \rceil} \theta_{T_j},$$

where θ_{T_j}, according to Definition 16, is the difference between the $j\gamma$ term linear approximation and the $(j-1)\gamma$ term linear approximation. Since Φ has the $(\epsilon_\gamma, r)-$RAmP for the linear residual subspaces $L_{j,\gamma}$ and $r = s - 1$, we obtain:

$$\|\Phi(\theta - \theta_\gamma)\|_2 = \|\Phi(\sum_{2}^{\lceil \frac{N}{\gamma} \rceil} \theta_{T_j})\|_2 \leq \sum_{2}^{\lceil \frac{N}{\gamma} \rceil} \|\Phi\theta_{T_j}\|_2 \tag{A.11}$$

$$\leq \sum_{2}^{\lceil \frac{N}{\gamma} \rceil} \sqrt{1+\epsilon_\gamma} j^{s-1} \|\theta_{T_j}\|_2. \tag{A.12}$$

Since x is a linear compressible signal, the norm of each piece of its GFT θ can be bounded as

$$\|\theta_{T_j}\|_2 = \|\theta_{j\gamma} - \theta(j-1)\gamma\|_2 \leq \|\theta - \theta_{j\gamma}\|_2 + \|\theta - \theta_{(j-1)\gamma}\|_2 \tag{A.13}$$

$$\leq SK^{-s}((j-1)^{-s} + j^{-s}). \tag{A.14}$$

Combining this bound with A.12, we obtain

$$\|\Phi(\theta-\theta_\gamma)\|_2 \leq \sum_{2}^{\lceil\frac{N}{\gamma}\rceil} \sqrt{1+\epsilon_\gamma} j^{s-1} \|\theta_{T_j}\|_2 \tag{A.15}$$

$$\leq \frac{\sqrt{1+\epsilon_\gamma}}{K^s} S \sum_{2}^{\lceil\frac{N}{\gamma}\rceil} \frac{j^{s-1}}{(j-1)^s} + \frac{j^{s-1}}{j^s} \tag{A.16}$$

$$\tag{A.17}$$

$$\leq \frac{\sqrt{1+\epsilon_\gamma}}{K^s} S \sum_{2}^{\lceil\frac{N}{\gamma}\rceil} \frac{1}{j(1-1/j)^s} + \frac{1}{j} \tag{A.18}$$

$$\leq \frac{\sqrt{1+\epsilon_\gamma}}{K^s} S \sum_{2}^{\lceil\frac{N}{\gamma}\rceil} \frac{2^s}{j} + \frac{1}{j} \tag{A.19}$$

$$\leq (2^s+1)\frac{\sqrt{1+\epsilon_\gamma}}{K^s} S \sum_{2}^{\lceil\frac{N}{\gamma}\rceil} \frac{1}{j} \tag{A.20}$$

By using Euler-Maclaurin summations, that $\sum_2^{\lceil\frac{N}{\gamma}\rceil} \frac{1}{j} \leq ln\lceil\frac{N}{\gamma}\rceil$, we can easily obtain the conclusion. □

A.3 Proof of Theorem 4.3.6

Proof. Let $\theta_{(\gamma)}$ be the first γ entries of θ. Consequently, it is a $\gamma \times 1$ vector and the least square estimate $\hat{\theta}_{(\gamma)} = \Phi_\gamma^\dagger y$ while $\hat{\theta}_{\gamma^c} = 0$. Hence, $\hat{\theta} = [\hat{\theta}_{(\gamma)}^T, \hat{\theta}_{\gamma^c}^T]^T$. Let θ_γ denote the $\gamma-$term linear approximation of θ, then:

$$\|\theta-\hat{\theta}\|_2 = \|\theta-\theta_\gamma+\theta_\gamma-\hat{\theta}\|_2 \tag{A.21}$$

$$\geq \frac{1}{\sqrt{2}}(\|\theta-\theta_\gamma\|_2 + \|\theta_\gamma-\hat{\theta}\|_2) \tag{A.22}$$

which gives the lower bound immediately. The inequality is due to the fact that the support of $\theta - \theta_\gamma$ is disjoint to that of $\theta_\gamma - \hat{\theta}$. For the upper bound, it is straightforward to see that:

$$\|\theta - \hat{\theta}\|_2 \leq \|\theta - \theta_\gamma\|_2 + \|\theta_\gamma - \hat{\theta}\|_2 \tag{A.23}$$

$$= \|\theta - \theta_\gamma\|_2 + \|\theta_{(\gamma)} - \Phi_\gamma^\dagger y\|_2 \tag{A.24}$$

The second equality is due to the fact that θ_γ and $\hat{\theta}$ are 0 outside the support of T_1. Notice that $\theta_{(\gamma)} = \Phi_\gamma^\dagger \Phi_\gamma \theta_{(\gamma)} = \Phi_\gamma^\dagger \Phi \theta_\gamma$ and $y = \Phi\theta$, the recovery error can be further bounded by:

$$\|\theta - \hat{\theta}\|_2 \leq \|\theta - \theta_\gamma\|_2 + \|\Phi_\gamma^\dagger \Phi(\theta - \theta_\gamma)\|_2 \tag{A.25}$$

$$\leq \|\theta - \theta_\gamma\|_2 + \|\Phi_\gamma^\dagger\|_2 \|\Phi(\theta - \theta_\gamma)\|_2 \tag{A.26}$$

Since Φ has the $L_{1,\gamma}-$RIP, $\frac{\|\Phi_\gamma u\|_2}{\|u\|_2} \in [1 - \delta_\gamma, 1 + \delta_\gamma]$, i.e., the smallest singular value of Φ_γ is smaller than $\sqrt{1 - \delta_\gamma}$. Denote it by σ_{min} Hence, $\|\Phi_\gamma^\dagger\|_2 = \frac{1}{\sigma_{min}} \leq \frac{1}{\sqrt{1-\delta_\gamma}}$. Moreover, since $\theta \in \mathbb{L}_s$ and Φ has the $(\epsilon_\gamma, r)-$RAmP, we can bound $\|\Phi(\theta - \theta_\gamma)\|_2$ by $C_s \sqrt{1 + \epsilon_\gamma} S \gamma^{-s} ln\lceil \frac{N}{\gamma} \rceil$, which completes the proof. □

References

[1] Sparselab. *[Online]. Available: http://http://sparselab.stanford.edu/*, 2007.

[2] Cimis data. *[Online]. Available: http://www.cimis.water.ca.gov*, 2011.

[3] A. Agaskar and Y.M. Lu. An uncertainty principle for functions defined on graphs. In *Proceedings of SPIE*, volume 8138, page 81380T, 2011.

[4] A. Agaskar and Y.M. Lu. Uncertainty principles for signals defined on graphs: bounds and characterizations. In *The 37th International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, 2012.

[5] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer networks*, 38(4):393–422, 2002.

[6] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak. Compressive wireless sensing. In *Proceedings of the 5th international conference on Information processing in sensor networks*, pages 134–142. ACM, 2006.

[7] W. Bajwa, A. Sayeed, and R. Nowak. Matched source-channel communication for field estimation in wireless sensor networks. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, pages 44–es. IEEE Press, 2005.

[8] W.U. Bajwa, J.D. Haupt, G.M. Raz, S.J. Wright, and R.D. Nowak. Toeplitz-structured compressed sensing matrices. In *Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on*, pages 294–298. IEEE, 2007.

[9] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3):253–263, 2008.

[10] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982–2001, 2010.

[11] M. Belkin and P. Niyogi. Using manifold structure for partially labeled classification. *Advances in Neural Information Processing Systems*, 15:929–936, 2002.

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

[13] M. Belkin and P. Niyogi. Convergence of laplacian eigenmaps. In *In Advances in Neural Information Processing Systems*, volume 19, pages 129–137. The MIT Press, 2007.

[14] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[15] M. Ben-Chen and C. Gotsman. On the optimality of spectral compression of mesh data. *ACM Transactions on Graphics (TOG)*, 24(1):60–80, 2005.

[16] T. Blumensath and M.E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[17] T. Blumensath and M.E. Davies. Sampling theorems for signals from the union of finite-dimensional linear subspaces. *IEEE Transactions on Information Theory,*, 55(4):1872–1882, 2009.

[18] E. Candès and J. Romberg. Sparsity and incoherence in compressive sampling. *Inverse problems*, 23:969, 2007.

[19] E.J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, 2008.

[20] E.J. Candes and J. Romberg. Quantitative robust uncertainty principles and optimally sparse decompositions. *Foundations of Computational Mathematics*, 6(2):227–254, 2006.

[21] E.J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[22] E.J. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406–5425, 2006.

[23] V. Cevher, M. Duarte, and R.G. Baraniuk. Distributed target localization via spatial sparsity. In *European Signal Processing Conference (EUSIPCO)*, 2008.

[24] Y. Chen, D. Bindel, H.H. Song, and R.H. Katz. Algebra-based scalable overlay network monitoring: algorithms, evaluation, and applications. *IEEE/ACM Transactions on Networking*, 15(5):1084–1097, 2007.

[25] A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k-term approximation. *American Mathematical Society*, 22(1):211–231, 2009.

[26] R.R. Coifman and M. Maggioni. Diffusion wavelets. *Applied and Computational Harmonic Analysis*, 21(1):53–94, 2006.

[27] M. Crovella and E. Kolaczyk. Graph wavelets for spatial traffic analysis. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1848–1857. IEEE, 2003.

[28] W. Dai. Subspace pursuit for compressive sensing: Closing the gap between performance and complexity. Technical report, DTIC Document, 2008.

[29] D.L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[30] D.L. Donoho, I. Drori, Y. Tsaig, and J.L. Starck. *Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit*. Department of Statistics, Stanford University, 2006.

[31] M.F. Duarte and R.G. Baraniuk. Spectral compressive sensing. *preprint*, 2010.

[32] M.F. Duarte and Y.C. Eldar. Structured compressed sensing: from theory to applications. *Signal Processing, IEEE Transactions on*, 59(9):4053–4085, 2011.

[33] M.F. Duarte, S. Sarvotham, D. Baron, M.B. Wakin, and R.G. Baraniuk. Distributed compressed sensing of jointly sparse signals. In *Asilomar Conf. Signals, Sys., Comput*, pages 1537–1541, 2005.

[34] L. Grafakos. *Classical Fourier Analysis*. Springer Verlag, 2008.

[35] R.M. Gray. *Toeplitz and circulant matrices: A review*. Information Systems Laboratory, Stanford University, 1971.

[36] D.K. Hammond, P. Vandergheynst, and R. Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, 2011.

[37] J. Haupt and R. Nowak. Signal reconstruction from noisy random projections. *IEEE Transactions on Information Theory*, 52(9):4036–4048, 2006.

[38] M. Kanso and M. Rabbat. Compressed rf tomography for wireless sensor networks: Centralized and decentralized approaches. *Distributed Computing in Sensor Systems*, pages 173–186, 2009.

[39] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 279–286. ACM Press/Addison-Wesley Publishing Co., 2000.

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

[41] S.G. Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 2008.

[42] R. Masiero, G. Quer, D. Munaretto, M. Rossi, J. Widmer, and M. Zorzi. Data acquisition through joint compressive sensing and principal component analysis. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6. IEEE, 2009.

[43] R. Masiero, G. Quer, M. Rossi, and M. Zorzi. A Bayesian analysis of compressive sensing data recovery in wireless sensor networks. In *Ultra Modern Telecommunications Workshops, 2009. ICUMT'09. International Conference on*, pages 1–6. IEEE, 2009.

[44] M. Muger. The Discrete Fourier Transform and its properties. *[Online]. Available: http://www.math.ru.nl/ mueger/BV.pdf*, 2010.

[45] D. Needell and J.A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

[46] R. Nowak, U. Mitra, and R. Willett. Estimating inhomogeneous fields using wireless sensor networks. *Selected Areas in Communications, IEEE Journal on*, 22(6):999–1006, 2004.

[47] I. Pesenson. Sampling in Paley-Wiener spaces on combinatorial graphs. *Transactions of the American Mathematical Society*, 360(10):5603, 2008.

[48] I.Z. Pesenson and M.Z. Pesenson. Sampling, filtering and sparse approximations on combinatorial graphs. *Journal of Fourier Analysis and Applications*, 16(6):921–942, 2010.

[49] H.V. Poor. *An introduction to signal detection and estimation*. Springer, 1994.

[50] M. Rabbat, J. Haupt, A. Singh, and R. Nowak. Decentralized compression and predistribution via randomized gossiping. In *Proceedings of the 5th international conference on Information processing in sensor networks*, pages 51–59. ACM, 2006.

[51] H. Rauhut. Circulant and toeplitz matrices in compressed sensing. *Arxiv preprint arXiv:0902.4394*, 2009.

[52] H. Rauhut. Compressive sensing and structured random matrices. *Theoretical Foundations and Numerical Methods for Sparse Recovery*, 9:1–92, 2010.

[53] A. Ribeiro and G.B. Giannakis. Bandwidth-constrained distributed estimation for wireless sensor networks-part i: Gaussian case. *Signal Processing, IEEE Transactions on*, 54(3):1131–1143, 2006.

[54] M. Rudelson and R. Vershynin. Sparse reconstruction by convex relaxation: Fourier and Gaussian measurements. In *2006 40th Annual Conference on Information Sciences and Systems*, pages 207–212. IEEE, 2006.

[55] S.E. Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

[56] I.D. Schizas, G.B. Giannakis, and Z.Q. Luo. Distributed estimation using reduced-dimensionality sensor observations. *Signal Processing, IEEE Transactions on*, 55(8):4284–4299, 2007.

[57] D.I. Shuman, P. Vandergheynst, and P. Frossard. Chebyshev polynomial approximation for distributed signal processing. *Arxiv preprint arXiv:1105.1891*, 2011.

[58] D.A. Spielman. Spectral graph theory. *Lecture Notes, Yale University*, 2009.

[59] M. Stojnic, F. Parvaresh, and B. Hassibi. On the reconstruction of block-sparse signals with an optimal number of measurements. *IEEE Transactions on Signal Processing,*, 57(8):3075–3085, 2009.

[60] D. Ustebay, R. Castro, and M. Rabbat. Efficient decentralized approximation via selective gossip. *IEEE Journal of Selected Topics in Signal Processing*, 5(4):805–816, 2011.

[61] G.K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.

[62] W. Wang, M. Garofalakis, and K. Ramchandran. Distributed sparse random projections for refinable approximation. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 331–339. ACM, 2007.

[63] J.J. Xiao, A. Ribeiro, Z.Q. Luo, and G.B. Giannakis. Distributed compression-estimation using wireless sensor networks. *IEEE Signal Processing Magazine*, 23(4):27–41, 2006.

[64] G. Zhu, H. Yang, R. Yan, J. Ren, B. Li, and Y. Lai. Uncovering evolutionary ages of nodes in complex networks. *Arxiv preprint arXiv:1107.1938*, 2011.

[65] X. Zhu. Semi-supervised learning literature survey. 2005.

[66] X. Zhu, J. Kandola, J. Lafferty, and Z. Ghahramani. Graph kernels by spectral transforms. 2005.

[67] X. Zhu and M. Rabbat. Approximating signals supported on graphs. In *The 37th International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, 2012.

[68] X. Zhu and M. Rabbat. Graph spectral compressed sensing for sensor networks. In *The 37th International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, 2012.