GRAPH SPECTRAL COMPRESSED SENSING FOR SENSOR NETWORKS

Xiaofan Zhu and Michael Rabbat

Department of Electrical and Computer Engineering, McGill University
3480 University St, Montreal QC, Canada H3A 2A7

ABSTRACT

Consider a wireless sensor network with N sensor nodes measuring data which are correlated temporally or spatially. We consider the problem of reconstructing the original data by only transmitting $M \ll N$ sensor readings while guaranteeing that the reconstruction error is small. Assuming the original signal is "smooth" with respect to the network topology, our approach is to gather measurements from a random subset of nodes and then interpolate with respect to the graph Laplacian eigenbasis, leveraging ideas from compressed sensing. 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.

***Index Terms*—** Distributed estimation, graph Fourier transform, compressed sensing, wireless sensor networks.

1. INTRODUCTION

For many *wireless sensor network* (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. 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 whereby we directly gather a reduced number of informative measurements rather than gathering a large number of redundant measurements.

When describing a N-dimensional signal in terms of a given basis transformation (e.g., the Fourier transform), if we are given a budget of $\gamma \ll N$ values to represent the signal, then the best choice is to keep the γ transform coefficients with largest magnitude. Directly computing a basis transformation and locating the γ largest transform coefficients in a distributed manner is non-trivial and consumes more energy and bandwidth resources than simply sending the raw data from each sensor to a fusion center.

One promising solution to the above issue leverages developments in the area of *compressed sensing* (CS) [1, 2]. CS theory shows that, when the signal is sparse or compressible in the transform domain, we can utilize $M \ll N$ 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 [3, 4] 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, which results in non-trivial power consumption. Wang et al. [5] solve this problem by proposing sparse random sensing matrices, which significantly reduces the communication overhead.

In contrast to previous work, we focus on the particular case of estimating signals which are smooth with respect to a graph. In this paper, we propose a technique called *Graph Spectral Compressed Sensing* (GSCS). We show 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 *Graph Fourier Transform* (GFT) basis [6], 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 measurements. Consequently, both power consumption, bandwidth usage, and latency are reduced.

Our main contribution is two fold. First, to the best of our knowledge, most of the previous CS literature [4] considering data compression or field estimation assumes that the signals sampled are compressible in certain orthogonal domains (e.g., 2-d wavelets). 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 [3, 4] consider dense noisy 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 can successfully reduce both the energy consumption and query latency under the scheme of CS.

The rest of the paper is organized as follows: In Section 2, the basic idea of GSCS is introduced. In Section 3, detailed data gathering algorithms for WSNs with spatially and temporally correlated signals are proposed. In Section 4, both synthesized and real world data are utilized to verify the performance of our approach, and we conclude in Section 5.

2. GRAPH SPECTRAL COMPRESSED SENSING

The first step of GSCS is to generate a deterministic orthogonal transform basis where the signal is compressible. Here we utilize the Graph Fourier Transform.

2.1. The Graph Fourier Transform

Graph theory plays an important role in analyzing networks since networks can be well modeled by graphs. Two crucial tools for studying graphs are the adjacency and Laplacian matrices, which encode the topology of a graph. For an undirected, unweighted graph

$G = (V, E)$, in which E and V denotes the sets of edges and nodes respectively. The adjacency matrix A is an $N \times N$ matrix, where $N = |V|$ is the number of nodes, with entries $A_{i,j} \in \{0, 1\}$, where $A_{i,j} = 1$ if there is an edge between node i and node j, and $A_{i,j} = 0$ otherwise. The degree of node i, denoted by d_i, is the number of nodes connected to i. The degree matrix D is a diagonal matrix with entries $D_{i,i} = d_i$. The graph Laplacian is then $L = D - A$.

Let $\lambda_0 \leq \lambda_1 \leq \cdots, \leq \lambda_{N-1}$ denote the eigenvalues of L, with corresponding eigenvectors $u_i, i = 0, 1, \cdots, N-1$. We denote the Laplacian eigenbasis of the graph G by $U = [u_0, u_1, \cdots, u_{N-1}]$. From the discussion of [6], we know that the Laplacian eigenbasis can be regarded as a sort-of "Fourier transform" for signals supported on the nodes of G, and so we refer to U as the Graph Fourier Transform matrix. A signal $f \in \mathbb{R}^V$ supported on $G = (V, E)$ is said to be smooth if there exists a positive constant $C \ll \lambda_{N-1}$ such that $\|f\|_G^2 \leq C\|f\|^2$, where $\|f\|_G^2 = f^T L f$. Moreover, a smooth signal supported on G has GFT coefficients $\widehat{f}(\lambda_i) = \langle f, u_i \rangle$ with linearly decaying behavior; i.e., $|\widehat{f}(\lambda_i)| \leq S i^{-(s+1/2)}$ for constants $s, S > 0$. As discussed, e.g., in [7], compressible signals can often be defined by the decaying behavior of the non-linear approximation error. However, since we are concerned with linearly compressible signals here, we focus on the following class of signals.

Definition 1. For given $s > 0$, the set of s-linearly-compressible signals is defined as: $\mathbb{L}_s = \{f \in \mathbb{R}^N : \epsilon_l(\gamma, f) \leq S\gamma^{-s}, 1 \leq \gamma \leq N, S < \infty\}$, where $\epsilon_l(\gamma, f) = \sum_{i=\gamma}^{N-1} |\widehat{f}(\lambda_i)|^2$ is the γ-term linear approximation error.

In this paper, we are interested in smooth signals supported on graphs. In some cases, the graph of interest may be known (e.g., the network topology). In other cases (e.g., temporal correlation), we consider the problem of constructing a graph which is appropriate for compressing the given signal. This problem is studied in [6]: Given an arbitrary signal $x \in \mathbb{R}^N$, we can think of each coordinate of x as value associated with a particular node in a graph. We can construct the K-Nearest-Neighbor (KNN) graph by connecting nodes with similar values, and then use its Laplacian eigenbasis U. It has been shown [6] that with proper choice of K, the number of neighbors of each node, we can construct an underlying KNN graph where the signal x is smooth. Equivalently, we are able to find a graph such that the signal x is linearly compressible in the corresponding GFT basis. In real applications, we may not know the prior information about which nodes share similar values. However, we can generate the graph using, e.g., location information if the signal is spatially correlated or by previous estimates if the signal is temporally correlated. More detailed information is included in Section 3.

2.2. Compressed Sensing via Graph Fourier Transform Basis

Compressed Sensing [1, 2] is a very useful tool to handle sparse or compressible signals. Suppose instead of collecting all the coefficients of a signal $x \in \mathbb{R}^N$, we merely record M inner products (measurements) of x with $M \ll N$ pre-selected vectors. This can be represented as: $y = \Psi x = \Psi U \theta = \Phi \theta$, where $\Phi = \Psi U$ is the sensing matrix with dimension $M \times N$, U is the orthogonal basis where x is sparse or compressible and θ is the basis expansion coefficients. If the sensing matrix Φ satisfies certain conditions [1, 2], then we can reconstruct the original signal by solving the linear program (ℓ_1 decoding): $\min_\theta \|\theta\|_1 \ s.t. \ y = \Phi\theta$.

Candès [1] and Rudelson [8] discuss conditions that the structure of random matrices should satisfy to be valid CS sensing matrices: (i)The matrix should be orthogonal. (ii)The entries of the normalized $N \times N$ matrix should be uniformly 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 $M = O(\gamma \log^4 N)$ rows of such matrices, we can generate valid sensing matrices for CS [8]. The traditional Discrete Fourier Transform (DFT) basis is clearly a candidate fit for such criteria. As the GFT is considered the "Fourier" basis for signals supported on graphs, can the GFT basis be similarly treated as the DFT basis? It is straightforward to check that the GFT basis is orthogonal, but the second condition (bounded coherence) cannot be guaranteed. 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 2. 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.

It has been show in [9] that $\mu_U(T)$ 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 $O(\frac{1}{\sqrt{N}})$. Moreover, if we construct a connected symmetric KNN graph by choosing a small parameter K, where K is the number of neighbors a node should be connected to, $\lambda_0, \cdots, \lambda_i$ are likely to be small for $i \ll N$ and thus we can have bounded $\mu_U(T)$ where $T = \{1, 2, \cdots, i\}$.

Fortunately, the uniformly bounded condition can be relaxed if we utilize this prior information and the linear compressibility of the signals supported on graphs. For example, consider a sparse signal here. If the nonzero entries of the original signal have a fixed support T and U is the sensing matrix, then the behavior of submatrix U_{T^c} will not affect the recovery process, where T^c is the complementary set of T; i.e., we merely require $\mu_U(T) = O(\frac{1}{\sqrt{N}})$. The same conclusion can be generalized to linear compressible signals. Moreover, linear compressible signals can be well approximated by the linear approximation, i.e., the first γ GFT coefficients will contain most of the energy of the signal. Thus, we can rely on such prior knowledge and utilize a simple estimator called oracle estimator. Let $\Phi = U_\Omega$ denote the sensing matrix, where Ω is a subset of $1, 2, \cdots, N$ and U_Ω is a submatrix generated by selecting the corresponding rows from Ω. Also let Φ_γ denote the sub-matrix of Φ containing the first γ columns. The oracle estimator can be represented as $U_\gamma^\dagger y$ where $y = \Phi\theta$ is the measurement and $U_\gamma^\dagger$ is the Moore-Penrose pseudo inverse of U_γ. After we obtain $\hat{\theta}$, we can get the estimate $\hat{x}$ via the equation $\hat{x} = U\hat{\theta}$.

Theorem 2.1. *Let $x \in \mathbb{L}_s$ be an s-linear compressible signal and $T_j = \{(j-1)\gamma + 1, (j-1)\gamma + 2, \cdots, j\gamma - 1\}$. Let the sensing matrix be U_Ω. 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 $M = |\Omega|$ obeys $M \geq Const \cdot \gamma \cdot \ln(\frac{\gamma}{\delta})$ for some $\delta > 0$, then with probability $1 - \delta$, the estimate $\hat{x}$ obtained from the oracle estimator satisfies: $\frac{1}{\sqrt{2}}\|x - x_\gamma\|_2 \leq \|x - \hat{x}\|_2 \leq \|x - x_\gamma\|_2 + C \cdot S\gamma^{-s} ln\lceil \frac{N}{\gamma} \rceil$, where $C = \frac{C_s\sqrt{1+\epsilon_\gamma}}{\sqrt{1-\delta_\gamma}}$ and x_γ is the $\gamma-$term linear approximation.*

A detailed proof of this theorem is provided in the technical report [9]. The proof makes use of techniques developed in [7, 10, 11]. The theorem claims that if the entries of the original signal decay quickly, we can 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. Since the oracle estimator recovers the $\gamma-$term linear approximation of the original signal, the recovery error has a lower bound.

3. APPLICATION TO WIRELESS SENSOR NETWORKS

3.1. Spatially Correlated Signals

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 [6], 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 measurements y and the sensing matrix U_Ω. After the fusion center obtains the measurement y, i.e., the data readings from M sensor nodes, we can estimate the original signal x by utilizing the oracle estimator or conventional ℓ_1 decoding.

3.2. Temporally Correlated Signals

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 the work [6] and obtain its Laplacian matrix U by taking the eigenvalue decomposition of the Laplacian matrix L of G.

(3) At time t, the WSN randomly collects data from a 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 $y_t = U_{\Omega_t} x_t$.

(4) When the fusion center obtains the current measurement vector y_t, it recovers the current estimates $\hat{x}_t = U\hat{\theta}$ by utilizing the oracle estimator or conventional ℓ_1 decoding.

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

3.3. Power, Latency and Distortion

For a linear compressible signal, the upper bound shows that $\|x - x_\gamma\|_2 \leq Const \cdot S\gamma^{-s}$. 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 the 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 [4] and assuming the recovery errors are close for GSCS and conventional CS with the same number of measurements, for a WSN with N nodes, each sensor has to transmit M times in order to generate the measurement vector y, i.e., the total number of transmissions in the WSN is MN. However, by exploiting GSCS, we merely require M_2 nodes to transmit their readings, 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 $M_2 \ll NM_1$.

4. EXPERIMENTS

4.1. Synthesized Data

Fig.1 shows the performance of GSCS with oracle estimator and Basis Pursuit (BP), conventional CS via i.i.d. Gaussian random matrix and sparse random projection [5]. The signal is generated by the following model: 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 $s = 2$. For oracle estimator, we set the parameter $\gamma = round(\frac{M}{7})$ in all the experiments here for simplicity. We use the BPsolver routine of SparseLab2.1[1] to solve the ℓ_1 recovery problem. The algorithm is run for 500 trials to get the best, worst and average performance. It is worth noting that all the Monte Carlo experiments in this section use a fixed data set with random measurements. From Figure 2, we can see that for a linear 2-compressible signal, GSCS with the oracle estimator outperforms all the other methods when $M \ll N$. Its performance is only worse than that of others when $M \rightarrow N$. This is easy to understand since the recovery error for oracle 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.

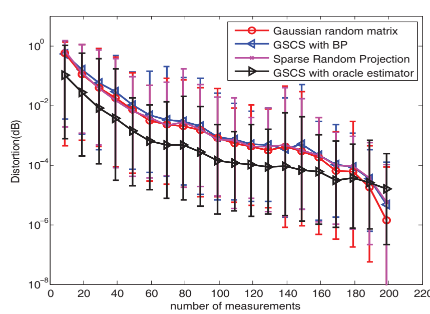


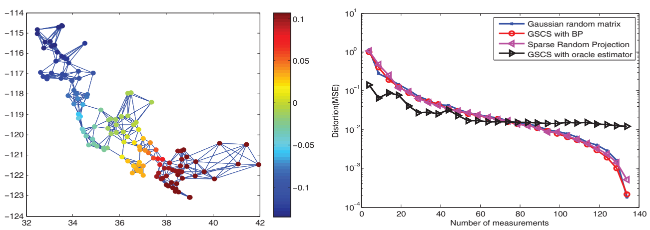
Fig. 1. Performance of GSCS with oracle estimator and Basis Pursuit (BP), conventional CS via i.i.d. Gaussian random matrix and sparse random projection. The averaged distortion over 200 trials is plotted while the best and worst performance is denoted by the error bar. The number of measurements M is from 9 to 199.

[1] http://sparselab.stanford.edu/

4.2. Real world data

In this section, we investigate the performance of GSCS on data from the California Irrigation Management Information System (CIMIS)[2]. This dataset is generated 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 run GSCS on solar radiation data across multiple sensors and multiple time points.

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. The resulting network is shown in Fig. 2(a), and Fig. 2(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 oracle estimator works clearly better than all the other methods when $M \ll N$. The distortion is computed for 100 different times and the average distortion is presented.

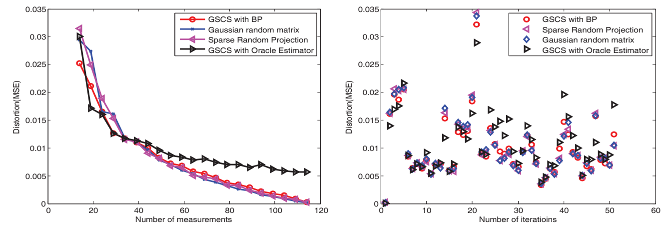


(a) (b)

Fig. 2. (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 oracle estimator and BP, conventional CS via i.i.d. Gaussian random matrix and sparse random projection. The figure plots distortion (mean squared error) as a function of the number of measurements, M.

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 of three months. First we set $r = 40$ and let the sensor data of the first 40 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 3.2 to estimate the original signals. Figure 3(a) shows how the number of measurements affects the performance of GSCS. The averaged MSE becomes fairly small when the number of measurements exceeds 20. The performance of GSCS with the oracle estimator is comparable to other methods when $M \ll N$. Daily readings might change quickly from the past and such signals don't exhibit strict linear compressibility. This is one disadvantage of the oracle estimator: it requires that signals strictly conform to the linear compressible model. Figure 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, we find that the large spikes of the error usually correspond to signals that deviate from the the day before.

[2]http://www.cimis.water.ca.gov



(a) (b)

Fig. 3. (a) Performance comparison of GSCS with oracle estimator and BP, conventional CS via i.i.d. Gaussian random matrix and sparse random projection on temporally correlated data as a function of number of measurements per day. The distortion is calculated by averaging over the total 52 daily readings. The parameter K is also set to 7. (b) Mean square error of each iteration. The number of measurements is set to 40.

5. CONCLUSION

In this paper, we propose a new technique called Graph Spectral Compressed Sensing. GSCS utilizes the partial Graph Fourier ensemble as the sensing matrix for smooth signals supported on graphs. 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,

6. REFERENCES

[1] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Transactions on Information Theory,*, vol. 52, no. 12, pp. 5406–5425, 2006.

[2] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory,*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] J. Haupt, W. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *IEEE Signal Processing Magazine,*, vol. 25, no. 2, pp. 92–101, 2008.

[4] 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*. ACM, 2006, pp. 134–142.

[5] 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*. ACM, 2007, pp. 331–339.

[6] X. Zhu and R. Rabbat, "Approximating signals supported on graphs," McGill University, Tech. Rep., 2011.

[7] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Transactions on Information Theory,*, vol. 56, no. 4, pp. 1982–2001, 2010.

[8] M. Rudelson and R. Vershynin, "Sparse reconstruction by convex relaxation: Fourier and gaussian measurements," in *40th Annual Conference on Information Sciences and Systems,*. IEEE, 2006, pp. 207–212.

[9] X. Zhu and R. Rabbat, "Graph spectral compressed sensing," McGill University, Tech. Rep., 2011.

[10] E. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[11] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse problems*, vol. 23, p. 969, 2007.