Application-driven Compressed Sensing

by

Sungwon Lee

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
Doctor of Philosophy
(ELECTRICAL ENGINEERING)

August 2013

Dedication

To my greatest parents ··· Hunchul Lee
Soonok Kim

To my loving grandparents ··· Jindoo Lee
Jakyoung Huh

To my uncle ··· Bonjin Byun

To my best friend ··· Ga-Young Suh

Acknowledgements

I would like to express my sincere appreciation and gratitude to my advisor, Prof. Ortega, for his support and patience throughout my graduate studies at the University of Southern California. He has been the best advisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered. His philosophy of guidance has trained me as one of researchers who are able to think critically and keep asking myself why.

I owe my gratitude to Prof. Krishnamachari and Prof. Neumann for serving as members of my dissertation committee and Prof. Kuo and Prof. Dimakis for serving as committee members of my qualify exam. Their valuable comments and suggestions were really appreciated.

I would also like to say thank you to all my colleagues in USC and friends in our research group for valuable discussion and collaboration: Prof. Krishnamachari, Prof. Hashemi, Sundeep Pattem, Chenliang Du, Jaehoon Kim, In Suk Chong, Godwin Shen, Woo-Shik Kim, Hey-Yeon Cheong, Kun-Han Lee, Sean McPherson, Zihong Fan, Greg Lin, Sunil K. Narang, Amin Rezapour, Hilmi E. Egilmez, Akshay Gadde, Yung-Hsuan Chao, Jason Tokayer, Yongzhe Wang, Lingyan Sheng, Gloria Halfacre, Mary Francis, Allan Weber, Seth Scafani, and Talyia Veal.

Many friends have supported and encouraged me to keep focused on what I originally dreamed of. Their support and care helped me overcome setbacks and stay focused on my graduate study. I greatly appreciate their friendship; Hoshik

Kim, Juyong Brian Kim, Euihyun Kang, Soyeon Lee, Hyungil Chae, Christopher S. Chang, SangHyun Chang, Jeehwan Kim, Oh-Hoon Kwon, Jae Kyoung Suh, Jae-sun Seo, Seung-hyun Son, Jeansoo Khim, Jinhwan Lee, Youngsoo Sohn, Andrew Mc Son, Jungwoo Kim, Donnie Hojun Kim, Jaehoon Kim, Ahryon Cho, DGM families, and Gahee Kim.

I appreciate Ga-Young Suh her supporting and encouraging me in difficult times. I heartfully thank my uncle, Bonjin Byun, for his constant care and concern from my first day in USC. Most importantly, none of this would have been possible without the love and support of my parents and grandparents. They to whom this dissertation is dedicated to, has been a constant source of love, concern, support and strength all these years. I would like to express my heart-felt gratitude to my family.

Table of Contents

List of Tables

List of Figures

Abstract

Traditional compressed sensing (CS) approaches have been focused on the goal of reducing the number of measurements while achieving satisfactory reconstruction. Reducing the number of measurements can directly lead to reductions in costs in some applications, e.g., the scanning time in fast magnetic resonance imaging (MRI) or the sampling rate in analog-to-information conversion. However, in some other applications, minimizing the number of measurements by itself does not provide a better solution in terms of system complexity due to additional application-driven constraints.

In general, those constraints affect the choice of either measurement basis or sparsifying basis. For example, if the total cost of collecting measurements is a crucial factor as compared to the reconstruction accuracy, reducing the number of measurements does not guarantee better performance because the increase of measurement cost can exceed the gain achieved by the increase of the number of measurements. Thus, the design of an efficient measurement basis should consider the total cost for measurements as well as the reconstruction accuracy. Also, in coding applications where signals are first captured and then compressed, better performance can be achieved by adaptively selecting a transform or sparsifying basis and then signaling the chosen transform to the decoder. For instance, for piecewise smooth signals, where sharp edges exist between smooth regions, edge-adaptive transforms can provide sparser representation at the cost of some overhead.

Thus, the design of sparsifying basis should be optimized with respect to a given measurement basis, while the signaling overhead is minimized. These observations motivated us to investigate efficient design schemes for CS that can provide better reconstruction while minimizing the application-driven costs.

In this thesis, we study the optimization of compressed sensing in three different applications, each of which imposes a different set of constraints: i) efficient data-gathering in wireless sensor networks (WSN), ii) depth map compression using a graph-based transform, and iii) fast target localization using a single-chip ultra-wideband (UWB) radar. Under these application-driven constraints, we study how to minimize application specific costs while minimizing the mutual coherence in order to achieve satisfactory reconstruction using CS.

In sensor networks, we take explicitly into consideration the cost of each measurement (rather than minimizing the number of measurements), and optimize the measurement matrix that leads to energy efficient data-gathering in WSN. For depth map compression, the constraint to consider is the total bitrate, including both the number of bits for measurements and the bit overhead to code the edge map required for the construction of graph-based transform (GBT). To improve overall performance, we propose a greedy algorithm that selects for each block the GBT that minimizes a metric accounting for both the edge structure of the block and the characteristics of the measurement matrix. For fast localization of objects using a UWB ranging system, we design an efficient measurement system that is constructed using a low-density parity-check (LDPC) matrix, designed to satisfy hardware-driven constraints. To enhance performance, we propose a window-based reweighted L_1 minimization that outperforms other existing algorithms in high noise environments.

Chapter 1

Introduction

Traditional compressed sensing (CS) approaches have been focused on reducing the number of measurements while achieving satisfactory reconstruction. To achieve satisfactory performance, two main conditions should be satisfied: (i) sparseness of signal to be reconstructed and (ii) incoherence between measurement (sensing) basis and sparsifying (sparsity-inducing) basis [12, 23]. Under these conditions, a reduction in the number of measurements can directly lead to lower complexity in some applications, in terms of application specific metrics such as the scanning time in fast magnetic resonance image (MRI) or the sampling rate in analog-to-information conversion [53, 55]. However, in other applications additional constraints should be considered, because fewer measurements do not always provide better performances when the trade-off between complexity and reconstruction quality is considered.

In this thesis, we investigate the optimization of compressed sensing under application-driven constraints. There could be many possible applications but we consider three specific examples: i) efficient data-gathering on wireless sensor network (constrained sensing matrix), ii) depth map compression using graph-based

transforms (adaptive sparsifying basis), and iii) fast target localization using single-chip UWB radar (hardware-driven constrains). Each application has its own constraints to satisfy and costs to minimize. Each problem can be addressed with a design philosophy satisfying both application-driven and compressed sensing constraints, while providing satisfactory reconstruction accuracy. While the results in this thesis are presented for specific applications, we expect that this design philosophy can be applied to various scenarios.

Table 1.1 summarizes the choices of sensing and sparsifying basis for each application. The bases to be optimized with respect to the application specific constraints are highlighted in the table. Table 1.2 describes the application specific constraints or costs to minimize for each application.

Table 1.1: Choice of sensing / sparsifying basis for each application

Application	Sensing Basis ($\mathbf{\Phi}$)	Sparsifying Basis ($\mathbf{\Psi}$)
WSN	**Sparse block-diagonal random**	Wavelet or GBT
Depth map	Hadamard	**GBT**
UWB radar	**Non-negative integer**	Spatial (Identity)

In the WSN application, the transport cost to collect sensor information should be considered for efficient data-gathering of WSN data. Since the total cost of measurements in WSN is closely related to the construction of the sensing matrix, the challenge is to design a power-efficient sensing matrix, while preserving incoherence with respect to a given sparsifying basis.

For depth map compression, the additional cost to consider is the total number of bits to encode both measurements and edge map, where the edge map is closely related to the construction of the sparsifying basis. In contrast to the WSN application, the sensing matrix is fixed (the Hadamard matrix is chosen in our work) and

Table 1.2: Overview of application specific constraints / cost

Application	Constraints / Cost
WSN	Transport cost for measurement
Depth map	Bits for measurements and edge map
UWB radar	Scanning time under hardware-driven constraints

the goal is to optimize the sparsifying basis (graph-based transform) by minimizing the number of bits required for the construction of the transform while achieving satisfactory reconstruction.

For fast localization of objects using single-chip UWB ranging system, we need to design an efficient measurement system satisfying hardware-driven constraints: non-negative integer entries in the measurement (sensing) matrix, constant row-wise sum of non-zero entries in the matrix, and a unique structure characterized by Kronecker product. Also, a non-linear reconstruction technique for UWB signal needs to be designed for better reconstruction.

Before discussing more details about our design for each application, we first provide an overview of the basic mathematical results that provide guarantees for robust recovery of sparse signals in CS.

1.1 Compressed Sensing (CS)

Compressed Sensing (CS) refers to a general class of systems where the goal is to recover an N-dimensional sample signal ($\boldsymbol{x}$) having a sparse representation in one basis, the goal is to recover it from a small number of projections (smaller than N) onto a second basis that is incoherent with the first [11, 23].

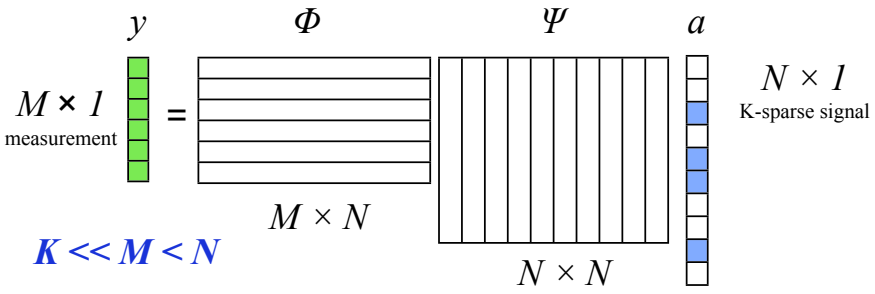


Figure 1.1: General matrix formulation of compressed sensing.

More formally, if a signal, $\boldsymbol{x} \in \Re^N$, is sparse in a given basis, $\boldsymbol{\Psi}$ (i.e., the sparsity inducing basis), $\boldsymbol{x} = \boldsymbol{\Psi a}, |\boldsymbol{a}|_0 = K$, where $K \ll N$, then theoretically we can reconstruct the original signal with $M = O(K \log N)$ measurements by finding the sparsest solution to an under-determined, or ill-conditioned, linear system of equations, $\boldsymbol{y} = \boldsymbol{\Phi x} = \boldsymbol{\Phi\Psi a} = \boldsymbol{Ua}$, where $\boldsymbol{U}$ is known as the holographic basis as shown in Fig. 1.1. Reconstruction is possible by solving the convex unconstrained optimization problem:

$$\min_{\boldsymbol{a}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{Ua}\|_2^2 + \gamma\|\boldsymbol{a}\|_1. \tag{1.1}$$

Multiple results have been derived to establish under what conditions $\boldsymbol{x}$ can be successfully recovered (refer to [1, 6] for more details). Here we summarize some of these results, which will be useful in deriving some of our approaches. As an example, a unique solution can be obtained when $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ satisfy the restricted isometry property (RIP) [10].

Definition 1.1.1. *$K - RIP$ [10]*

An $M \times N$ matrix $\boldsymbol{U} = \boldsymbol{\Phi\Psi}$ has the K-restricted isometry property (K-RIP) with constant $\delta_K \in [0, 1]$ if, given an integer K, we have that:

$$(1 - \delta_K)\|\boldsymbol{a}\|_2^2 \leq \|\boldsymbol{Ua}\|_2^2 \leq (1 + \delta_K)\|\boldsymbol{a}\|_2^2$$

holds for any K-sparse vector $\boldsymbol{a}$.

The definition of K-RIP indicates that all submatrices of $\boldsymbol{U}$ of size $M \times K$ are close to an isometry, and thus distance preserving with a small margin, δ_K. While the design of $\boldsymbol{U}$ satisfying the K-RIP is an NP-Complete problem, random sensing matrices, $\boldsymbol{\Phi}$, lead to $\boldsymbol{U}$ $(= \boldsymbol{\Phi\Psi})$ satisfying K-RIP with high probability. Random matrices whose entries are i.i.d. Gaussian, Bernoulli, or more generally subgaussian, *universally* satisfy K-RIP with any choice of orthonormal basis matrix, $\boldsymbol{\Psi}$, leading to perfect reconstruction with high probability when $M = O(K \log(N/K))$ [10, 13, 23].

Aside from the random matrix approach, the general problem of finding optimal matrices satisfying K-RIP is an NP-Complete problem. Thus, mutual coherence [8, 72] has been proposed as an alternative measure to characterize matrices that are likely to meet the RIP condition. Mutual coherence serves as a rough characterization of the degree of similarity between the sparsity and measurement systems and is defined as follows:

Definition 1.1.2. *Mutual coherence [8]*
For two matrices with normalized row vectors($\boldsymbol{\Phi}$*) and column vectors(*$\boldsymbol{\Psi}$*), mutual coherence is defined as* $\mu(\boldsymbol{U}) = \max_{i,j} |\boldsymbol{U}(i,j)| = \max_{i,j} |\boldsymbol{\phi(i)\psi(j)}|$*, where* $\boldsymbol{\phi(i)}$ *is a row vector of* $\boldsymbol{\Phi}$ *and* $\boldsymbol{\psi(j)}$ *is a column vector of* $\boldsymbol{\Psi}$*. Note that* $\mu(\boldsymbol{U}) \in [0, 1]$

For μ to be close to its minimum value, each of the measurement vectors must be spread out in the $\boldsymbol{\Psi}$ domain. A small value of $\mu(\boldsymbol{U})$ indicates that $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are incoherent with each other, i.e., no element of one basis $(\boldsymbol{\Psi})$ has a sparse representation in terms of the other basis $(\boldsymbol{\Phi})$.

In the case when both Φ and Ψ are orthogonal, the minimum number of measurements for perfect reconstruction can be computed as follows.

Theorem 1.1.3. *Minimum number of measurements [8]*
Let $\boldsymbol{U} = \boldsymbol{\Phi\Psi}$ *be an* $N \times N$ *orthogonal matrix (*$\boldsymbol{U^T U} = \boldsymbol{I}$*) with* $|U(i,j)| \leq \mu(\boldsymbol{U})$. *For a given signal* $\boldsymbol{x} = \boldsymbol{\Psi a}$, *suppose that* $\boldsymbol{a}$ *is supported on a fixed (arbitrary) set* T *with* K *non-zero entries, and assume that measurements,* $\boldsymbol{y} = \boldsymbol{\Phi x}$, *are retrieved by random downsampling. Then,* L_1 *optimization can recover* $\boldsymbol{x}$ *exactly with high probability if the number of measurements* M *satisfies*

$$M = O(K\mu^2(\boldsymbol{U})N \log N) \tag{1.2}$$

The bound on the number of measurements provided by Theorem 1.1.3 decreases as $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ become increasingly incoherent, i.e., the minimum number of measurements for perfect reconstruction is determined by μ for given K and N. In this thesis, we investigate how to reduce μ while satisfying application specific constraints. In the applications of data-gathering in WSN and depth map compression, the orthogonal bases, $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$, are studied to find a better solution with measurements obtained by a separate downsampling. But for the UWB application, we investigate how to find the actual (rectangular) sensing basis without the separate downsampling for a given $\boldsymbol{\Psi} = \boldsymbol{I}$.

1.2 Efficient data-gathering in WSN

In sensor networks, energy efficient data manipulation and transmission is very important for data gathering, due to significant power constraints on the sensors. This constraint has motivated the study of joint routing and compression for power-efficient data gathering of locally correlated sensor network data. Most of the early

works were theoretical in nature and, while providing important insights, ignored the practical details of how compression is to be achieved [19,62,80]. More recently, it has been shown how practical compression schemes such as distributed wavelets can be adapted to work efficiently with various routing strategies [63, 75, 76]. The existing transform-based techniques can reduce the number of bits to be transmitted to the sink, thus achieving overall power savings. These transform techniques are essentially critically sampled approaches, so that their cost of gathering scales up with the number of sensors, which could be undesirable when large deployments are considered. Although the number of measurements in CS, M, is also proportional to the dimension of signal, N, the rate of increase is lower than that for the critically sampled approaches because it is a logarithmic function of N, $M = O(K \log N)$.

CS has been considered as a potential alternative in this context, as the number of measurements required depends on the characteristics (sparseness) of the signal and also the dimension of the signal [8, 11, 23]. With recent interest in WSN, researchers have proposed various ways to apply CS to WSN in order to reduce the gathering costs [26, 52, 60, 69]. But those approaches focused on the total number of measurements rather than the total transport cost of measurements. In [83], it was shown that CS could also operate using *sparse* random projections (SRP) but this work does not consider transport cost to collect measurements in a *multi-hop* network. In [47, 68], the potential benefits of CS for sensor network applications have been recognized but significant obstacles remain for it to become competitive with more established (e.g., transform-based) data gathering and compression techniques. A primary reason is that CS theoretical developments have focused on *minimizing the number of measurements* (i.e., the number of samples captured), rather than on *minimizing the cost of each measurement*.

In the WSN data-gathering application, the key observation is that an efficient measurement system needs to be *both sparse* (few sensors contribute samples to each measurement) and *spatially-localized* (the sensors that contribute to each measurement are close to each other) in order to be competitive in terms of transport cost. In [48], we introduced the use of a cluster-based measurement system for CS; each measurement was obtained by a linear combination of samples captured *within a single cluster*, and clusters were selected in order to contain sensor nodes that are close to each other. Based on the clustering scheme, we extend this approach by analyzing how the choice of spatial clusters affects the reconstruction accuracy, for a given spatially-localized sparsity basis. Moreover, we propose novel clustering techniques that take into consideration both transport cost and reconstruction quality [45].

More specifically, we have two main contributions in this thesis. First, we introduce the concept of *maximum* energy overlap between clusters and basis functions, which we denote β. If basis functions and clusters have similar spatial localization, most of the energy of a given basis function is likely to be concentrated in a few clusters, which means that only measurements taken from those clusters are likely to contribute to reconstructing a signal that contains that specific basis function. Since the measurement system is not aware *a priori* of where signals will be localized, it needs to gather enough measurements to reconstruct signals with any spatial localization, and since each cluster overlaps only a few such basis functions, it will need to have a larger number of measurements. Conversely, for the same number of measurements, as the energy of the basis functions is more evenly distributed over clusters (smaller β), this could lead to better reconstruction. To verify this,

we provide a proof that the minimum number of measurements for perfect reconstruction is proportional to β. Therefore, for given basis functions, we can estimate performance of different clustering schemes by computing β.

Second, we propose a centralized iterative algorithm with a design parameter, λ, for joint optimization of the energy overlap and distance between nodes in each cluster. A joint optimization is required because there exists a *tradeoff* between β and the distance. To achieve smaller β (which leads to fewer required measurements for a certain level of reconstruction accuracy), each basis function should be overlapped with more clusters. This means that the nodes within a cluster tend to be farther from each other because basis functions are localized in space. Since total transport cost is a function of the number of measurements and transport cost per measurement, the trade-off allows reducing the number of measurements at the cost of increasing transport cost per measurement. By joint optimization using an appropriately chosen λ, we can achieve a good trade-off between transport cost and reconstruction accuracy. Compared with another CS technique (which showed the best performance in [48]), our simulation results show that our proposed approach with joint optimization achieves almost 50% reduction in transport costs at the same level of mean squared error in the reconstruction.

Also, we extend our work to a practical situation where the sensors are randomly deployed. In order to sparsify the sensor data measured at randomly positioned sensors, we propose to use graph-based transform (GBT) that can achieve a sparse representation of the sensor data [78]. We first represent the random topology as a graph then construct a sparsifying basis by placing eigenvectors of the Laplacian matrix of the graph. With the GBT, we propose a heuristic design of the data-gathering that the aggregations happen at the sensors with fewer neighbors in the graph than a empirically chosen threshold. In our simulations, compared to

other methods, our proposed approach shows better performance in the total power consumption at the same level of MSE in the reconstruction.

1.3 Depth map compression for 3D-TV

In the problem of depth map compression for 3D-TV, the performance gain with standard image or video coding techniques is limited because the depth map has different characteristics from other natural images or standard video sequences. We can characterize a depth map as i) a gray-scale video, ii) containing piecewise smooth signals, and iii) with sharp edges where depth changes across different objects or around objects-to-background (background-to-objects) transitions. It has been shown that edges play a crucial role in the rendered views; errors in edge information translate into geometry errors in the interpolated view, which leads to significant quality degradation [39, 40].

These characteristics have motivated edge-adaptive coding tools that apply transforms within smooth regions without filtering across edges. Examples include edge-aware DCTs [31, 66, 89, 90] or wavelets [7, 21, 65, 67, 79]. The edge-aware DCT approaches require that the transform block size be adaptive to edge location or the pixel values be rearranged in order to be aligned with the dominant direction of edges in a block. The wavelet-based approaches are not appropriate to block based coding architectures, which are dominant in image and video coding standards such as JPEG and H.264/AVC. CS based methods have also been proposed recently [25,73]. These methods sparsify depth map data using the DCT, which has been widely used for block-based image and video compression, but is inefficient for coding blocks containing arbitrary shaped edges. This motivates us to consider similar CS architectures but using better sparsifying basis.

To improve the efficiency of depth map coding, we propose a new CS framework based on a graph signal representation [46]. The graph based transform (GBT) has been shown to be better for sparse representation of depth map data, especially when arbitrary edges such as diagonal or mixture of horizontal and vertical edges exist in the block [38, 74]. The basic idea is that for each block an edge structure can be identified, and from it an edge adaptive graph transform can be derived. We first investigate the tradeoff between the cost (related to the overhead required to encode edge map) and the reconstruction quality (related to the incoherence between GBT and Hadamard sensing matrix). As more edges are used as a part of the transform, the coherence decreases, so that we can achieve better reconstruction with a fixed number of measurements, but the amount of overhead information required to represent the edge map also increases.

To find a better edge map, we propose a greedy algorithm to minimize a cost function that takes into account the mutual coherence. Since computing the mutual coherence at every iteration of the algorithm is computationally expensive, we approximate the average mutual coherence between the Hadamard sensing matrix and GBT with a technique that does not require the explicit construction of GBT. At every iteration of the algorithm the cost function can be computed using only the edge map and Hadamard sensing matrix. In our proposed algorithm, the complicated construction of the GBT is only required once the optimized edge map is found. Also, since our approach applies to blocks of arbitrary size, we apply our block-based approach to H.264/AVC reference software JM17.1 and compare the performance to the case where H.264/AVC is used for depth map compression. With the proposed method, a 3.7 dB PSNR gain or 38% bitrate saving is observed on average when applied to three depth map sequences.

1.4 Fast localization using UWB radar

For fast localization of objects using an ultra-wideband (UWB) ranging system, we propose a CS framework based on a recently developed hardware [15]. This hardware utilizes a *ranging* technique by sending multiple pulses, then averaging the received pulses in short time intervals (*windows*), each corresponding to a certain roundtrip time of the reflected pulse. Assuming the environment is relatively static, the receiver can localize an object at a specific distance by selecting a corresponding window and determining if the window contains reflected signal. The averaging within a chosen window provides robustness to noise. It also requires less power consumption, because power is only consumed during the measurement window, which can represent a small percentage of time. However, a limitation of this scheme comes from sequential sampling, i.e., candidate object locations have to be probed in sequence, so that the time required to locate an object will be proportional to the number of measurement windows. In this work, we propose techniques that can significantly reduce the scanning time, with no increase in overall power consumption, and can also operate in high noise environments. The key observation is that in many situations the number of objects that can be localized in the environment is small relative to the number of locations that are probed. In a static environment, this allows us to probe several locations simultaneously, so that each measurement combines reflections at several distances. Processing can then be used to extract the actual position information from the combined observations.

In the context of radar applications, many CS-based approaches have been proposed that exploit the sparse structure of UWB signals [12,23]. In [2], it was shown that the received signal can be digitized at a rate much lower than the Nyquist rate, but the high noise case scenarios and total power consumption constraints that we

mainly focus on in this work are not considered. Also, a precise CS-UWB positioning system was proposed that exploited the redundancy of UWB signal captured at multiple receivers in order to localize a transmitter [84, 85], but the ADC rate is still higher than what can be achieved with the UWB hardware platform we build upon [15]. As a CS approach tightly coupled to hardware, a Random-Modulation Pre-Integrator (RMPI) was proposed to achieve low-rate ADC by random modulation in analog domain [86–88] but the random modulation of signals contaminated by powerful noise in analog domain does not provide robust signal recovery.

To design a system that is robust to high noise and consumes less power while providing reliable localization, we propose a CS technique [44] tightly coupled to the capabilities of recently developed hardware [15]. First, we observe that the UWB signal is sparse if few object exists since the UWB signal is highly localized in time. Combined with the UWB ranging system, this leads to a special structure where sparse non-zero entries are clustered into a few groups (windows). The number of groups is equal to the number of objects in the region of interest. Second, we design an efficient measurement system subject to several constraints imposed by the hardware. The constraints include (i) non-negative integer entries in the sensing matrix (ii) constant row-wise sum of entries in the matrix (iii) non-zero entries of each row can exist only at the positions with a constant shift, which leads to unique structure characterized by a Kronecker product. Under these constraints, we construct a sensing matrix by using a low-density parity-check (LDPC) matrix that has recently been shown to be a good measurement system in [20,50]. Third, to enhance the localization performance, we propose a window-based reweighted L_1 minimization that shows good performance for aforementioned signal model and measurement system. In our simulations, we compare our proposed method with other existing reconstruction algorithms with respect to several metrics evaluating

localization performance. Our simulation results show that our proposed method can achieve reliable target-localization while using only 40% of the sampling time required by the corresponding sequential scanning scheme, even in a highly-noisy environment.

The rest of the dissertation is organized as follows. The efficient data-gathering for WSN using spatially-localized CS is described in Chapter 2. The depth map compression exploiting optimized GBT with CS-related constraint is proposed in Chapter 3. The fast localization of UWB radar by CS is presented in Chapter 4. Conclusions and future work are discussed in Chapter 5.

Chapter 2

Spatially-Localized Compressed Sensing for Efficient Data-gathering in Multi-Hop Sensor Networks

In sensor networks, energy efficient data manipulation / transmission is very important for data gathering, due to significant power constraints on the sensors. CS has been proposed as a potential solution because it requires capturing a smaller number of samples for successful reconstruction of sparse data. Traditional CS does not take explicitly into consideration the cost of each measurement (it simply tries to minimize the number of measurements), and this ignores the need to transport measurements over the sensor network. In this chapter, we study CS approaches for sensor networks that are spatially-localized, thus reducing the cost of data gathering. In particular, we study the reconstruction accuracy properties of a new distributed measurement system that constructs measurements within spatially-localized clusters. We first introduce the concept of maximum energy overlap between clusters and basis functions (β), and show that β can be used to estimate the minimum number of measurements required for accurate reconstruction. Based on this metric, we propose a centralized iterative algorithm for joint

optimization of the energy overlap and the distance between sensors in each cluster. Our simulation results show that we can achieve significant savings in transport cost with small reconstruction error. We also extend our work to the case of sensors placed at random positions. We show that smooth sensor data can be sparsified using a GBT based on the network topology and propose a heuristic design of the sensing matrix that takes into account the number of neighbors of the sensors in the graph. In our simulations, compared to other methods, our proposed approach shows better performance in the total power consumption at the same level of MSE in the reconstruction.

The rest of this chapter is organized as follows. We first introduce previous work related to WSN data gathering in Section 2.1 then formulate it in a CS framework in Section 2.2. Based on this problem formulation, we propose a spatially localized projection scheme associated with an energy overlap between projection basis and sparsifying basis in Section 2.3. Then we provide a mathematical proof that the energy overlap is a good metric to determine the number of measurements for perfect reconstruction in Section 2.4. Techniques to find a power-efficient clustering scheme are discussed in Section 2.5 and verified by our simulations in Section 2.6. We also extend our work to the case where sensors are placed at random positions and provide preliminary results in Section 2.7.

2.1 Related Work

Joint routing and compression has been studied for efficient data gathering of locally correlated sensor network data. Most of the early works were theoretical in nature and, while providing important insights, ignored the practical details of how compression is to be achieved [19, 62, 80]. More recently, it has been shown how

practical compression schemes such as distributed wavelets can be adapted to work efficiently with various routing strategies [16, 63, 75, 76].

Existing transform-based techniques, including wavelet based approaches [16,75, 76,81] and the distributed KLT [33], can reduce the number of bits to be transmitted to the sink thus achieving overall power savings. These transform techniques are essentially critically sampled approaches, so that their cost of gathering scales up with the number of sensors, which could be undesirable when large deployments are considered.

CS has been considered as a potential alternative in this context, as the number of samples required (i.e., number of sensors that need to transmit data), depends on the characteristics (sparseness) of the signal [8,11,23]. In addition, CS is also potentially attractive for wireless sensor networks because most computations take place at the decoder (sink), rather than at the encoder (sensors), and thus sensors with minimal computational power can efficiently encode data. Also, CS potentially increases the security of data transmission because partial measurements intercepted by malicious users are not sufficient for decoding (i.e., global information about all measurements is needed).

With recent interest in WSN, researchers have proposed various ways to apply CS to WSN. In [69], the authors studied a scenario where ultimately every sensor has an approximation of the network data by using gossip algorithms. But, this work assumes a single-hop network communication and does not consider the transmission cost of pre-distribution phase (gossiping phase). Also, an innovative CS approach was proposed that makes it possible to gather sensor data without any inter-sensor communication, based on a distributed compressed sensing approach [4, 26]. But, if a signal does not fit the corresponding signal model (joint

sparsity model), overall performance may not be good. More specifically, the assumption that there exists common information such as common data supports (i.e., the locations of the nonzero coefficients) among sensor measurements is unrealistic.

Some other work exploits the spatial correlations among data measured in each sensor with CS. In [83], it was shown that CS could also operate using *sparse* random projections (SRP). The implication for a single-hop sensor network is that good field reconstruction can be obtained even when only a small, randomly chosen fraction of sensors report their measurements. But this work does not consider the transport cost required to collect measurements in a *multi-hop* network. In [52], CS was proposed to detect abnormal sensor data or uncorrelated data in adjacent neighboring sensors, but the performance of this approach in terms of reconstructing all sensor data is likely to be limited. For the large scale networks, inter-flow network coding was combined to address high link failure in sensor networks [60]. But this approach results in much larger delay in the measurement transmission as compared to other CS approaches.

In [47,68], the potential benefits of CS for sensor network applications have been recognized but significant obstacles remain for it to become competitive with more established (e.g., transform-based) data gathering and compression techniques. A primary reason is that CS theoretical developments have focused on *minimizing the number of measurements* (i.e., the number of samples captured), rather than on *minimizing the cost of each measurement*. Thus, in this work, we propose to optimize CS to minimize the transport cost of measurements while providing satisfactory reconstruction accuracy [45].

2.2 Problem Formulation

In this work, we assume that $\boldsymbol{x} \in \Re^N$ is a vector containing measurements obtained by N sensors in a 2D region at a given time and $\boldsymbol{x}$ is K-sparse in a given sparsifying basis $\tilde{\boldsymbol{\Psi}}$. The N-sample signal ($\boldsymbol{x}$) can be recovered from M measurements or projections ($M < N$) onto a sensing (measurement) basis, $\boldsymbol{\Phi}$ if $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are incoherent [11, 23] as explained in Chapter 1. For efficient data-gathering from sensors spread over space to the sink located at the center of the network, we consider distributed measurement strategies that are both sparse *and* spatially localized.

2.2.1 Low-cost sparse projection based on clustering

In order to design distributed measurements strategies that are both sparse *and* spatially localized, we propose dividing the network into clusters of adjacent nodes and forcing projections to be obtained only from nodes within a cluster. As an example, we first consider two simple clustering approaches. Assume that all clusters contain the same number of nodes. When N_c clusters are used, each cluster will contain $\frac{N}{N_c}$ nodes. In "square clustering", the network is partitioned into a certain number of equal-size square regions. Alternatively, in "SPT-based clustering", we first construct the shortest path tree (SPT) then, based on that, we iteratively construct clusters from leaf nodes to the sink. These clustering schemes are first applied to equally spaced sensors on a 2D regular grid, then extended to the sensors positioned at irregular positions at the end of this chapter.

Any chosen clustering scheme can be represented in CS terms by generating the corresponding measurement matrix, $\boldsymbol{\Phi}$, and using it to reconstruct the original signal. As shown in Fig. 2.1, each row of $\boldsymbol{\Phi}$ represents the aggregation corresponding to one measurement: we place non-zero (e.g., random) coefficients in the positions

corresponding to sensors that provide their data for a specific measurement and the other positions are set to zero. Thus, the sparsity of a particular measurement in $\mathbf{\Phi}$ depends on the number of active nodes participating in this aggregation. We expect this approach to be more efficient than traditional CS because the sensors in each cluster are relatively close to each other (leading to a spatially-localized projection) and the number of sensors in each cluster is only a small fraction of the number of sensors in the network (leading to a sparse projection).

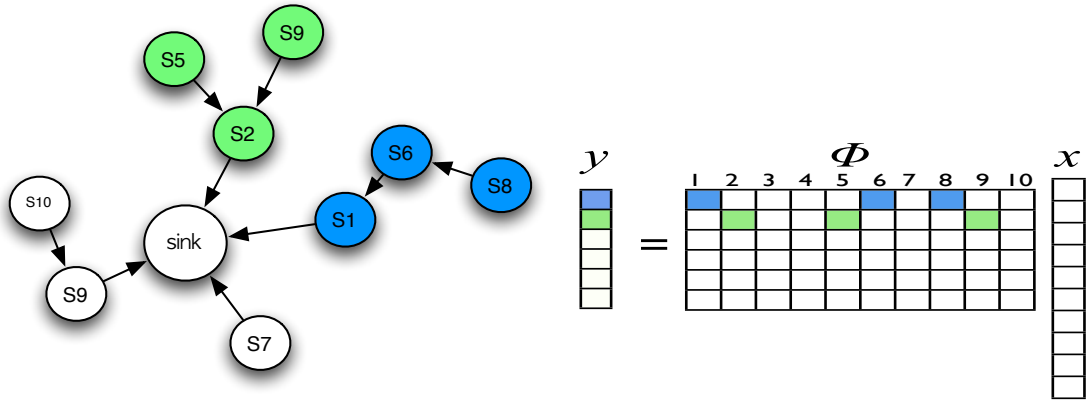


Figure 2.1: Link between CS measurements and data aggregation in WSN

In order to make the design simpler, we consider non-overlapped clusters, which leads to a block-diagonal structure for $\mathbf{\Phi}$. Note that recent work [22,32], seeking to achieve fast CS computation, has also proposed measurement matrices with a block-diagonal structure, with results comparable to those of dense random projections. Our work, however, is motivated by achieving spatially localized projections so that our choice of block-diagonal structure will be constrained by the relative positions of the sensors (each block corresponds to a cluster).

2.2.2 Sparsity-inducing basis and cluster selection

While it is clear that localized gathering leads to lower communication costs, it is not obvious how it may impact reconstruction quality. Thus, an important goal of this work is to study the interaction between localized gathering and reconstruction. A key observation is that in order to achieve both efficient routing and adequate reconstruction accuracy, the structure of the sparsity-inducing basis should be considered. As an example, consider the case where signals captured by the sensor network can be represented by a "global" basis, e.g., DCT, where each basis spans all the sensors in the network. Then the optimally incoherent measurement matrix will be the identity matrix, $\boldsymbol{I}$, and thus a good measurement strategy is simply to sample $K \log N$ randomly chosen sensors and then forward each measurement directly to the sink (no aggregation). Alternatively, for a completely localized basis, e.g., $\boldsymbol{\Psi} = \boldsymbol{I}$, a dense projection may be best for reconstruction accuracy. However, once the transport costs have been taken into account, the best solution is to just transmit the non-zero samples to the sink via the SPT. In other words, even if CS theory suggests a given type of measurements (e.g., dense projection for the $\boldsymbol{\Psi} = \boldsymbol{I}$ case), applying these directly may not lead to an efficient routing and therefore efficient distributed CS may not be achievable.

In this work, we first consider intermediate cases, in particular those where localized bases with different spatial resolutions are considered (e.g., wavelets). In [8] it was shown that a partial Fourier measurement matrix is incoherent with wavelet bases at fine scales. However, such a dense projection is not suitable for low-cost data gathering for the reasons discussed above. Next we explore appropriate spatially-localized gathering for data that can be represented in localized bases such as wavelets.

2.3 Efficient clustering for spatially-localized CS

2.3.1 Independent vs. Joint reconstruction

To help us understand how to select a clustering scheme that is appropriate for CS, we first compare two types of reconstruction: independent reconstruction and joint reconstruction. Suppose that we construct a series of clusters of nodes and collect a certain number of local measurements from each cluster. As an example, consider 2 clusters and the corresponding localized projections, $\boldsymbol{\phi_1}$ and $\boldsymbol{\phi_2}$, with a given sparsifying basis, $\boldsymbol{\Psi}$:

$$\boldsymbol{U} = \boldsymbol{\Phi\Psi} = \begin{bmatrix} \boldsymbol{\phi_1} & 0 \\ 0 & \boldsymbol{\phi_2} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi_1} & \boldsymbol{\psi_2} \\ \boldsymbol{\psi_3} & \boldsymbol{\psi_4} \end{bmatrix}, \; where \; \boldsymbol{U} = \begin{bmatrix} \boldsymbol{U_1} \\ \boldsymbol{U_2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\phi_1\psi_1} & \boldsymbol{\phi_1\psi_2} \\ \boldsymbol{\phi_2\psi_3} & \boldsymbol{\phi_2\psi_4} \end{bmatrix}. \quad (2.1)$$

Since the two clusters do not overlap with each other, the measurement matrix, $\boldsymbol{\Phi}$ in (2.1), has a block diagonal structure. For joint reconstruction, the original sparsifying basis, $\boldsymbol{\Psi}$, is employed. But, for independent reconstruction, data in the first cluster are reconstructed with partial basis functions, $\boldsymbol{\psi_1}$ and $\boldsymbol{\psi_2}$, and those in the second cluster are with $\boldsymbol{\psi_3}$ and $\boldsymbol{\psi_4}$ thus, when N_c clusters are involved, independent reconstruction should be performed N_c times, once for each cluster.

Joint reconstruction is expected to outperform independent reconstruction because measurements collected from a cluster can also convey information about data in other clusters, since basis functions that overlap with more than one clusters can be identified with measurements from those clusters. For example, the measurements from the first cluster can help to reconstruct the signal in the second cluster if there exists a basis function overlapped with both clusters.

Based on this observation the key intuition in our work is that clustering schemes that have overlap with more basis functions should be preferred. The degree of overlap between basis functions and clusters can be measured in many different ways. In this work, we propose to use as a metric the energy of the basis functions captured by each cluster. We will discuss the energy overlap in more detail in Section 2.3.3 and verify that joint reconstruction outperforms independent reconstruction through our simulations.

2.3.2 Spatially-Localized Projections in CS

An aggregation path in a sensor network can be represented by a row of the measurement matrix, $\mathbf{\Phi}$. We place non-zero (possibly random) coefficients in the positions corresponding to active sensors that provide their data for a specific measurement, while the other positions are set to zero, which means that the sparsity of a particular measurement in the matrix depends on the number of nodes participating in each aggregation.

To express M measurements algebraically, we consider a down-sampling matrix, $\boldsymbol{Q}$, that chooses M measurements out of N with equal probability. This can be expressed as:

$$\boldsymbol{y}_{M\times 1} = \boldsymbol{Q}_{M\times N}\mathbf{\Phi}_{N\times N}\boldsymbol{x}_{N\times 1} \tag{2.2}$$

With respect to standard CS approaches, the actual (rectangular) measurement matrix is $\boldsymbol{Q}\mathbf{\Phi}$ but, in this work, we explicitly separate the actual measurements as a combination of projections onto an orthogonal matrix, $\mathbf{\Phi}$, followed by downsampling, $\boldsymbol{Q}$. This will be needed for our proof in Section 2.4.3.

Similarly, the aggregations within a cluster can be expressed as a set of rows of $\mathbf{\Phi}$. Since N_c non-overlapped clusters are considered, we can express the measurement

system as a block diagonal matrix that contains N_c square sub-matrices, $\boldsymbol{\Phi_i}$ on its diagonal, so that $\boldsymbol{\Phi_i}$ represents the aggregation scheme used for the i^{th} cluster in the network. Therefore, the dimension of $\boldsymbol{\Phi_i}$ is determined by the number of sensors contained in the i^{th} cluster.

Let us define $\boldsymbol{x}$ as an original input signal and $\boldsymbol{x_p^i}$ as a vector of samples measured in the i^{th} cluster. To associate $\boldsymbol{\Phi_i}$ with $\boldsymbol{x_p^i}$, we introduce a permutation matrix, $\boldsymbol{P}$, whose output is $\boldsymbol{x_p} = [\boldsymbol{x_p^1} \dots \boldsymbol{x_p^i} \dots \boldsymbol{x_p^{N_c}}]^T = \boldsymbol{Px}$. Thus, by multiplying $\boldsymbol{\Phi}$ with the output of $\boldsymbol{Px}$, we have

$$\boldsymbol{y} = \boldsymbol{Q\Phi Px}, \quad \text{where} \quad \boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi_1} & & & \\ & \boldsymbol{\Phi_2} & & \\ & & \ddots & \\ & & & \boldsymbol{\Phi_{N_c}} \end{bmatrix}. \tag{2.3}$$

The square block-diagonal matrix, $\boldsymbol{\Phi}$, is the measurement matrix and the clusters are uniquely defined by, $\boldsymbol{P}$, the clustering matrix. A different permutation $\boldsymbol{P}$ can be defined for each clustering method, so that in any clustering method we can write the projection as in (2.3). Note that $\boldsymbol{P}$ *by itself* does not determine clustering, what produces clustering is the block diagonal matrix, $\boldsymbol{\Phi}$.

We now discuss how the clustering matrix is related to the sparsifying basis matrix. Since a K-sparse signal is represented by K non-zero coefficients in a given basis $\tilde{\boldsymbol{\Psi}}$, $\boldsymbol{x} = \tilde{\boldsymbol{\Psi}}\boldsymbol{a}$, the measurements are obtained by

$$\boldsymbol{y} = \boldsymbol{Q\Phi Px} = \boldsymbol{Q\Phi}\left(\boldsymbol{P}\tilde{\boldsymbol{\Psi}}\right)\boldsymbol{a} = \boldsymbol{Q\Phi\Psi a}, \tag{2.4}$$

where we define $\boldsymbol{\Psi} = \boldsymbol{P}\tilde{\boldsymbol{\Psi}}$ as a permutation of $\tilde{\boldsymbol{\Psi}}$ After permutation by the clustering matrix, the measurement matrices for each cluster, $\boldsymbol{\Phi_i}$, are correctly associated with data measured by sensors in the corresponding clusters, $\boldsymbol{x_p^i}$.

In summary, a cluster-based measurement system leads to a block diagonal measurement matrix with appropriate permutation, $\boldsymbol{P}$, related to the physical positions of sensors. Note that recent work [22, 32], seeking to achieve fast CS computation, has also proposed measurement matrices with a block-diagonal structure, with results comparable to those of dense random projections. Our work, however, is motivated by goal of achieving spatially localized projections, so that our choice of block-diagonal structure will be constrained by the *deterministic* positions of the sensors, instead of using a uniformly *random* permutation as proposed in [22].

2.3.3 Average Energy Overlap

To characterize the distribution of energy of the basis functions with respect to the clusters, we present a metric, E_{oa}, and an analysis of the worst-case scenario. Given $\boldsymbol{\Psi} = \boldsymbol{P}\tilde{\boldsymbol{\Psi}}$, for a given $\boldsymbol{P}$, suppose that N_c is the number of clusters and C_i is a set of nodes contained in the i^{th} cluster. The energy overlap between the i^{th} cluster and the j^{th} basis vector, $E_o(i,j)$, can be defined as:

$$E_o(i,j) = \sum_{k \in C_i} \psi(j,k)^2. \tag{2.5}$$

Then, the average energy overlap per overlapped basis can be a good indicator of distribution of energy of a basis function across multiple clusters. Since the basis

functions are normalized, the maximum overlap is 1. For each cluster, $E_{oa}(i)$ is computed as

$$E_{oa}(i) = \frac{1}{N} \sum_{j=1}^{N} E_o(i,j),\ \forall i \in \{1, 2, \cdots, N_c\}, \tag{2.6}$$

$E_{oa}(i)$ is the average energy localized in the i^{th} cluster. Intuitively, this metric shows how much the average energy of the basis functions is localized in each cluster. Thus, as energy of the basis functions is more evenly distributed over the clusters, E_{oa} decreases, which leads to better reconstruction performance with joint reconstruction. While this metric is a good indicator of the average distribution of basis functions, our results indicated that the worst case overlap may be better at predicting reconstruction performance.

It would be useful to have a metric to determine the number of projections required from each local cluster in order to achieve a certain level of reconstruction performance. We first define what the worst-case is, then propose a method to characterize the 'worst-case scenario' performance. If the global sparsity is K, the worst case scenario is when all K basis vectors supporting data are completely contained in a single cluster. In this case, $O(K)$ projections would be required from each cluster. There are two reasons for this. First, since the identity of the cluster where bases are concentrated is not known *a priori*, it is not possible to concentrate projections within that cluster without measuring information in the others. Second, projections from other clusters not overlapped with those basis vectors do not contribute to reconstruction performance as much as projections from the overlapped cluster. Thus, instead of the average energy overlap, we consider the maximum energy overlap as a metric of reconstruction quality in a spatially localized measurement system. The derivation of the number of measurements for

perfect reconstruction in terms of the maximum energy overlap will be discussed in Section 2.4.

2.3.4 Maximum Energy Overlap

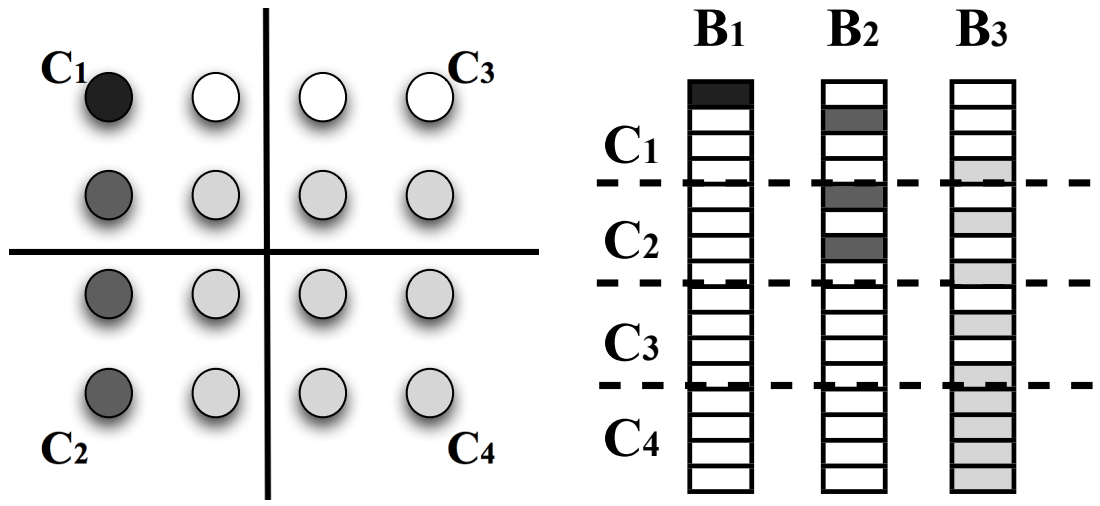
To quantify the distribution of energy overlap over clusters, we define the maximum energy overlap for the basis $\mathbf{\Psi}$, $\beta(\mathbf{\Psi})$, as follows:

Maximum energy overlap, $\beta(\mathbf{\Psi})$

$$\beta(\mathbf{\Psi}) = \beta(\boldsymbol{P}\tilde{\mathbf{\Psi}}) = \max_{i,j} \sum_{l} \mathbf{\Psi}_i^2(l,j) \ , \ \beta(\mathbf{\Psi}) \in [0,1]$$

β represents the maximum amount of energy of a basis functions captured by a single cluster. The matrix $\mathbf{\Psi}_i$ is the rectangular sub-matrix corresponding to the i^{th} cluster. For example, as depicted in Fig. 2.2 (b), we first compute the sum of squared entries (colored cells) for each pair of (B_i,C_j); For B_1, energy overlap is 1 with C_1 and zero with the other clusters. Then we take the maximum value of the computed sums. If β is 1 (maximum value), it indicates that there exists at least one basis function completely covered by a cluster in space such as the overlap between B_1 and C_1 in Fig. 2.2 (a). In contrast, small β means that most basis functions are overlapped with multiple clusters in space.

Intuitively, measurements taken from a cluster can also convey information about data in other clusters when basis functions overlap with more than one cluster, e.g., B_2 in Fig. 2.2 (a) can be identified with measurements from those clusters (C_1 and C_2). This is one of the reasons why, as will be shown experimentally in Section 2.6.1, joint reconstruction outperforms independent reconstruction.



(a) overlap in spatial domain

(b) overlap in sparsifying basis

Figure 2.2: (a) Illustration of energy overlap for a 4×4 grid network of 16 sensors. The network is divided into 4 square clusters and 3 basis functions are considered. The bases have spatial resolution of 1, 3 and 9 sensors, for B_1, B_2 and B_3, respectively. (b) Permuted sparsifying basis matrix, $\mathbf{\Psi} = \boldsymbol{P}\tilde{\mathbf{\Psi}}$. The entries of each basis function (column vector of $\mathbf{\Psi}$) is filled with colors if non-zero coefficients exist and white otherwise. Note that 13 more basis functions exist but are omitted here. The maximum energy overlap is 1 in this case, since B_1 is completely contained in C_1.

If a specific basis function is completely contained within a cluster, e.g. B_1, then only measurements from C_1 are likely to contribute to reconstructing a signal that contains B_1. Thus, as discussed in Section 2.3.3, the worst case scenario where all K basis functions supporting data are completely contained in a single cluster, e.g., B_1 in Fig. 2.2 (a) significantly increases the number of measurements required to achieve a good reconstruction. Thus, as basis functions are overlapped with more clusters, we will have a potentially higher chance to reconstruct the signal correctly. To further improve localized CS performance, a clustering scheme that minimizes overlap should be chosen. In the next section, we will show how β affects reconstruction accuracy and determines the minimum number of measurements for perfect reconstruction.

2.4 Theoretical Result

As shown in previous sections, the maximum energy overlap, β, is determined for given sparsifying basis, $\tilde{\boldsymbol{\Psi}}$, block-diagonal matrix, $\boldsymbol{\Phi}$, and a clustering scheme, $\boldsymbol{P}$. Here, we show how β affects reconstruction accuracy by deriving the minimum number of measurements for perfect reconstruction as a function of β.

2.4.1 Definitions and Assumptions

We assume that N_c non-overlapped clusters contain the same number of sensors. Thus, $\boldsymbol{\Phi}$ has N_c square sub-matrices with size of $N/N_c \times N/N_c$ along its diagonal. Therefore, if each sub-matrix, $\boldsymbol{\phi_i}$, is orthogonal, then $\boldsymbol{\Phi}$ is also orthogonal. Based on the problem formulation, our main result is based on three assumptions. First, the sparsifying basis, $\tilde{\boldsymbol{\Psi}}$, is orthogonal. Second, the maximum absolute value of entries in the sparsifying basis is bounded, $\max_{i,j} |\tilde{\boldsymbol{\Psi}}(i,j)| \leq 1/\sqrt{\log N}$, so that we do not consider degenerate cases, such as the canonical basis in spatial domain ($\tilde{\boldsymbol{\Psi}} = \boldsymbol{I}$). This assumption is satisfied by bases such as the DCT or the Daubechies wavelets with a sufficient number of levels of decomposition [22]. Lastly, the sub-measurement matrix, $\boldsymbol{\phi_i}$, is an orthogonalized i.i.d. Gaussian matrix: $\phi_i(j,k) \sim N(0, N_c/N)$. Thus $\boldsymbol{\Phi^T \Phi} = \boldsymbol{I}_N$ since the clusters are disjoint. In order to evaluate the coherence between the measurement matrix, $\boldsymbol{\Phi}$, and the permuted sparsifying basis matrix, $\boldsymbol{\Psi}(= \boldsymbol{P}\tilde{\boldsymbol{\Psi}})$, we define an $N \times N$ matrix $\boldsymbol{U}$:

$$
\begin{aligned}
\boldsymbol{U}_{N\times N} &= \begin{bmatrix} \boldsymbol{\Phi_1} & & & \\ & \boldsymbol{\Phi_2} & & \\ & & \ddots & \\ & & & \boldsymbol{\Phi_{N_c}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Psi_1} \\ \boldsymbol{\Psi_2} \\ \vdots \\ \boldsymbol{\Psi_{N_c}} \end{bmatrix}, \\
&= \begin{bmatrix} \boldsymbol{\Phi_1\Psi_1} \\ \vdots \\ \boldsymbol{\Phi_{N_c}\Psi_{N_c}} \end{bmatrix}, \qquad (2.7) \\
&= \begin{bmatrix} \boldsymbol{U_1} \\ \vdots \\ \boldsymbol{U_{N_c}} \end{bmatrix}. \qquad (2.8)
\end{aligned}
$$

By assumption, all square sub-matrices $\boldsymbol{\Phi_i}$ have the same size, and therefore all the $\boldsymbol{U_i}$ also have the same size. But our work can be extended to the case that the clusters have different sizes. Moreover $\boldsymbol{U}$ is an orthogonal matrix because $\boldsymbol{\Phi}$ and $\tilde{\boldsymbol{\Psi}}$ are orthogonal by assumption and the clustering matrix, $\boldsymbol{P}$, is a permutation matrix, so that:

$$
\boldsymbol{U^T U} = (\boldsymbol{\Phi P \tilde{\Psi}})^T(\boldsymbol{\Phi P \tilde{\Psi}}) = (\boldsymbol{\tilde{\Psi}^T P^T \Phi^T})(\boldsymbol{\Phi P \tilde{\Psi}}) = \boldsymbol{I_N} \qquad (2.9)
$$

The mutual coherence, μ, can be computed using Definition 1.1.2 then we can apply it to Theorem 1.1.3 because $\boldsymbol{U}$ is orthogonal. This helps us to derive the minimum number of measurements in terms of the number of clusters, N_c, sparsity, K, and the number of sensors, N.

2.4.2 Main Result

To obtain a bound on the number of measurements, we first derive an asymptotic upper bound on mutual coherence. With this bound, we can attain the minimum number of measurements for perfect reconstruction by using Theorem 1.1.3 if the aforementioned conditions are satisfied.

Proposition 2.4.1. *If all the sub-measurement matrices, $\boldsymbol{\phi_i}$, are orthogonalized i.i.d. Gaussian, $N(0, \frac{N_c}{N})$, and the orthogonal sparsifying basis, $\tilde{\boldsymbol{\Psi}}$, and the clustering matrix, $\boldsymbol{P}$, are known a priori, then $\mu(\boldsymbol{U})$ is asymptotically bounded by*

$$Pr\left[\mu(\boldsymbol{U}) \leq O\big(\sqrt{\beta \frac{N_c}{N} \log N}\big)\right] = 1 - O\big(\frac{1}{N}\big). \tag{2.10}$$

Proposition 2.4.1 quantifies the probability that coherence exceeds a certain bound. The probability that coherence is not bounded by $O\big(\sqrt{\beta \frac{N_c}{N} \log N}\big)$ is close to 0 as N increases. For the proof, the main technical tools are large deviation inequalities of sum of independent random variables. Specifically, the result is derived from Bernstein's deviation inequality [51] and a union bound for the supremum of a random process. Refer to Section 2.4.3 for details of proof.

With aforementioned assumptions and Proposition 2.4.1, we can characterize the impact of β on reconstruction accuracy in terms of the number of measurements.

Theorem 2.4.2. *For a given signal* $\boldsymbol{x} = \boldsymbol{\Psi a}$ *with* $|\boldsymbol{a}|_0 = K$ *and a clustering scheme with parameter* $\beta \in [0, 1]$*, the* L_1 *optimization can recover* $\boldsymbol{x}$ *exactly with high probability if the number of measurements* M *satisfies*

$$M = O(K\beta N_c \log^2 N). \tag{2.11}$$

Based on the bound on coherence of Proposition 2.4.1, we can derive Theorem 2.4.2, the minimum number of measurements for perfect reconstruction, by using Theorem 1.1.3 and the fact that $\boldsymbol{U}$ is orthogonal by (2.9).

Note that the bound in Theorem 2.4.2 is nearly identical to the bound obtained for the SRM case in [22], where clusters are defined based on random permutation matrices (i.e., $O(K\beta N_c \log^2 N)$ vs. $O(KN_c \log^2 N)$). Our result shows that random permutation matrices can be further optimized to have smaller β, which leads to smaller number of measurements for the same level of reconstruction accuracy.

In general, the number of measurements is proportional to the maximum energy overlap because basis functions with more uniformly distributed energy increase the probability of correct reconstruction. Also, the number of measurements is proportional to the number of clusters, N_c. This implies that a sparser measurement matrix (larger N_c) requires more measurements for the same level of reconstruction, as also shown in previous work [22, 83].

Note that there exists a *tradeoff* between β and the distance between nodes belonging to the same cluster. A decrease in β can be achieved when each basis function overlaps a larger number of clusters. Since basis functions are spatially localized, that means that nodes within a cluster will tend to be farther from each other, so that they can cover more different basis, which in turn leads to an increase

of transport cost per measurement. Since total transport cost is a function of both the number of measurements and the transport cost per measurement, this trade-off allows reducing the number of measurements at the cost of increasing transport cost per measurement. In order to construct clusters that take into consideration both reconstruction quality and transport costs, we propose a centralized iterative algorithm, which will be presented in Section 2.5.

2.4.3 Proof of Proposition 2.4.1

In this section, we present the details of the proof of Proposition 2.4.1. The goal is to bound the coherence in terms of β with high probability. The sketch of proof is similar to that of SRM in [22] but the details are different. First, in our problem, the randomness is due to the coefficients of the measurement matrix, $\Phi(i,j)$, rather than the uniform permutation as in SRM. Second, we consider an additional quantity, β, as well as the number of clusters, N_c.

Before going into the details of proof, we first approximate a bound related to $E[U_i^2(j,k)]$. Basically, $U_i(j,k)$ is the inner product between the j^{th} row of $\mathbf{\Phi_i}$ and k^{th} column of $\mathbf{\Psi}$ as shown in (2.8). Denote $X_l = \Phi_i(j,l)\Psi_i(l,k)$ so that $U_i(j,k) = \sum_l \Phi_i(j,l)\Psi_i(l,k) = \sum_l X_l$. Define $\gamma = \max_{i,j} |\mathbf{\Psi}(i,j)| \leq 1/\sqrt{\log N}$ by assumption and let $\Phi(j,k) \sim N(0, N_c/N)$. Therefore, $|X_l|$ can be bounded by the product of γ and three times of the standard deviation of $\Phi(j,k)$ with high probability, i.e.,

$$
\begin{aligned}
|X_i| &= |\Phi_i(j,l)\Psi_i(l,k)| \\
&\leq 3\sigma\gamma \\
&\leq 3\sqrt{\frac{N_c}{N\log N}}.
\end{aligned}
$$

The next approximation is concerned with $E[U_i^2(j,k)]$. Note that $Var(U_i(j,k))$ is equal to $E[U_i^2(j,k)]$ because $E(U_i(j,k)) = 0$. $Var(U_i(j,k))$ can be approximated as a function of β as follows:

$$
\begin{aligned}
Var(U_i(j,k)) &= E[U_i(j,k)^2] \\
&= E\left[(\sum_{l=1}^{N/N_c} \Phi_i(j,l)\Psi_i(l,k))^2\right] && (2.12) \\
&= E\left[\sum_{l=1}^{N/N_c} \Phi_i^2(j,l)\Psi_i^2(l,k)\right] \\
&= \sum_{l=1}^{N/N_c} E[\Phi_i^2(j,l)]\Psi_i^2(l,k) && (2.13) \\
&= \sum_{l=1}^{N/N_c} \frac{N_c}{N}\Psi_i^2(l,k) \\
&\leq \frac{N_c}{N}\beta && \text{(by definition of } \beta)
\end{aligned}
$$

In (2.12), the cross terms are zero since $E(\Phi_i(j,l)) = 0$ and the $\Phi_i(j,l)$'s are independent.

Now, we present the details of the proof of Proposition 2.4.1. The main technical tools are large deviation inequalities of sum of independent random variables. More

specifically, the proof is derived from Bernstein's deviation inequality [51] and a union bound for the supremum of a random process. We first approximate the probability that mutual coherence is bounded by α using a union bound for the supremum of a random process:

$$
\begin{aligned}
& Pr[\mu \leq \alpha] \\
& = Pr[\max_{j,k} |\boldsymbol{U}(j,k)| \leq \alpha] \\
& = 1 - Pr[\max_{j,k} |\boldsymbol{U}(j,k)| > \alpha] \\
& = 1 - Pr[\bigcup_{i} (\max_{j,k} |\boldsymbol{U_i}(j,k)| > \alpha)] \\
& \geq 1 - \sum_{i=1}^{N_c} \sum_{j=1}^{N/N_c} \sum_{k=1}^{N} Pr[|\boldsymbol{U_i}(j,k)| > \alpha] \qquad \text{(by Union Bound)}
\end{aligned}
$$

Then we use Bernstein's deviation inequality (Theorem 2.4.3) to approximate the bound of the tail probability of $U_i(j,k)$. Note that the $U_i(j,k)$'s are independent random variables because, by assumption, the $\Phi(i,j)$'s are i.i.d. Gaussian random variables with zero mean.

Theorem 2.4.3. *Bernstein's inequality [51]*
If $X_1, X_2, \ldots, X_n$ are independent (not necessarily identical) and zero-mean random variables, and $|X_i| \leq C, \forall i$, then

$$
Pr\left[\left|\sum_{i=1}^{n} X_i\right| \geq \alpha\right] \leq 2\exp\left(-\frac{\alpha^2/2}{\sum_{i=1}^{i=n} E[X_i^2] + C\alpha/3}\right) \tag{2.14}
$$

Denoting again $X_l = \Phi_i(j,l)\Psi_i(l,k)$, by the approximation, we have

$$C = 3\sqrt{\frac{N_c}{N \log N}} \tag{2.15}$$

and

$$\sum E[X_i^2] = Var(U_i(j,k)) \leq \frac{N_c}{N}\beta \tag{2.16}$$

By substituting (2.15) and (2.16) in Bernstein's inequality,

$$\begin{aligned}
&Pr[\mu \leq \alpha] \\
&\geq 1 - \sum_{i=1}^{N_c}\sum_{j=1}^{N/N_c}\sum_{k=1}^{N} 2\exp\left(-\frac{\alpha^2/2}{Var(\boldsymbol{U_i}(j,k)) + M\alpha/3}\right) \quad \text{(by Berstein's inequality)} \\
&\geq 1 - 2N^2 \exp\left(-\frac{\alpha^2/2}{(\beta N_c/N) + (3\sqrt{N_c/N \log N})\alpha/3}\right) \qquad \text{(by approximation)} \\
&= 1 - \exp\left(\log 2N^2 - \frac{\alpha^2/2}{(\beta N_c/N) + (\sqrt{N_c/N \log N})\alpha}\right)
\end{aligned}$$

We need to find the minimum α, denoted α^*, such that the probability, $Pr[\mu \leq \alpha^*]$, asymptotically goes to '1'. To achieve the asymptotic behavior, the 2^{nd} term (*exp* term) in the last inequality should be zero with large N, which means that the 2^{nd} order polynomial inside the *exp* term should be negative. The polynomial can be expressed as a function of α as:

$$f(\alpha) = -\alpha^2/2 + \sqrt{\frac{N_c}{N \log N}} \log 2N^2\alpha + \frac{\beta N_c}{N} \log 2N^2 \tag{2.17}$$

To find α^* for large N, we first check the characteristics of $f(\alpha)$. Since $f(0) > 0$, $f'(0) > 0$, and $f''(0) < 0$, the larger root of $f(\alpha)$ is the minimum α such that $Pr[\mu \leq \alpha^*] = 1$ for large N. Algebraically, from the computation of the larger root of $f(\alpha)$ in (2.17), we can conclude that

$$Pr[\mu \leq \alpha^*] = 1 - O(1/N),$$
$$where \;\; \alpha^* = O(\sqrt{\beta \frac{N_c}{N} \log N})$$

□

With the asymptotic bound of coherence above, we can derive Theorem 2.4.2, the minimum number of measurements for perfect reconstruction, by Theorem 1.1.3.

2.5 Centralized iterative clustering algorithm

To achieve a good tradeoff between transport cost and reconstruction accuracy, we need to jointly optimize β and the distance between nodes in a cluster. This motivated us to design a centralized iterative algorithm that can generate clusters based on a cost function taking into consideration both β and distance within cluster.

2.5.1 Cost function

For a given undirected graph $G = (V, E)$, we assume that the sparsifying basis, $\mathbf{\Psi}$, is known *a priori* and all the basis functions (columns of $\mathbf{\Psi}$) are normalized to 1 so that $\beta \in [0, 1]$. Also, N nodes are placed along a square grid in a field.

In order to construct N_c clusters that minimize transport costs, while guaranteeing perfect reconstruction for sparsity K. To achieve the goal, we design a greedy algorithm based on our propose cost function. The algorithm starts from (i) N_c initial nodes, one for each cluster, then (ii) at every iteration, we find edges connected to each of the clusters and compute the weights, (iii) to find the edge with the minimum weight, and (iv) we add the edge to the cluster. In short, at each iteration, we want to decide what node to be added to which cluster among all nodes not assigned to one of clusters in previous iterations.

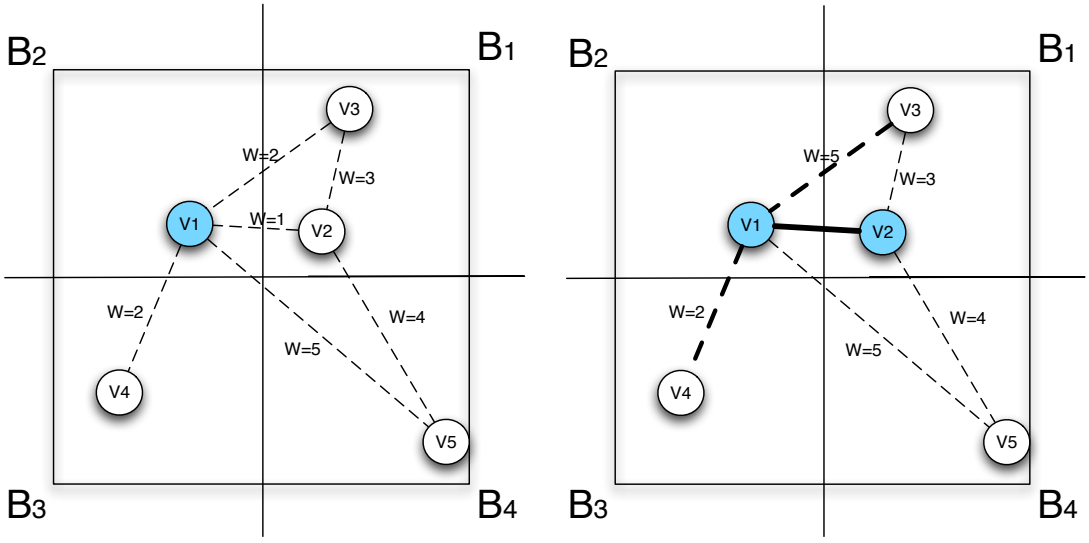
We first assume that the transport cost depends on the distance between nodes and define the distance *in hops* as $D(e)$ for an edge, $e \in E$, connecting two nodes, i.e., the smallest number of hops between two nodes in a multi-hop network. But β cannot be defined based on just nodes only since it depends on the clusters. Thus, we need to define $\beta(e, C_i)$ with respect to an edge, $e \in E$, and a given cluster, $C_i \in \{C_1, C_2, \cdots, C_{N_c}\}$. Assume that the edge e is not an edge connecting two nodes in the same cluster, C_i.

We first define a cost function with respect to an edge e and the i^{th} cluster C_i as:

$$W(e, C_i) = D(e) + \lambda\beta(e, C_i),\ \lambda > 0, \tag{2.18}$$

where $\beta(e)$ is the maximum energy overlap when an edge, e, is connected to the cluster C_i. Thus, once a node is added to a cluster, energy overlap of the edges connected to the cluster changes so that the edge weights should also change. As an example, from Fig. 2.3 (a) to (b), the weight of an edge, $e = (v_1, v_3)$, changes from 3 to 5 because v_3 and the cluster, $\{v_1, v_2\}$ is overlapped with the same basis function, B_1, which leads to the increase of the corresponding β. The cost function,

$W(e, C_i)$, will be used in our proposed algorithm to find an optimized clusters in Chapter 2.5.2



(a) the i^{th} iteration (b) the $(i+1)^{th}$ iteration

Figure 2.3: Illustration of update of edge weights from (a) to (b). There exist 4 square basis functions (B_i) and 5 nodes (v_i) connected by edges, (v_i, v_j), with different weights. Assume that the initial node for clustering is v_1. At the i^{th} step, a cluster is formed by $\{v_1, v_2\}$ because (v_1, v_2) has the minimum weight of 1. Since the cluster changes, we need to update the weights if necessary. For example, the weight of an edge, (v_1, v_3), changes from 3 to 5 because v_3 and the cluster, $\{v_1, v_2\}$ is overlapped with the same basis function, B_1, which increases the β with respect to the cluster.

2.5.2 Algorithm details

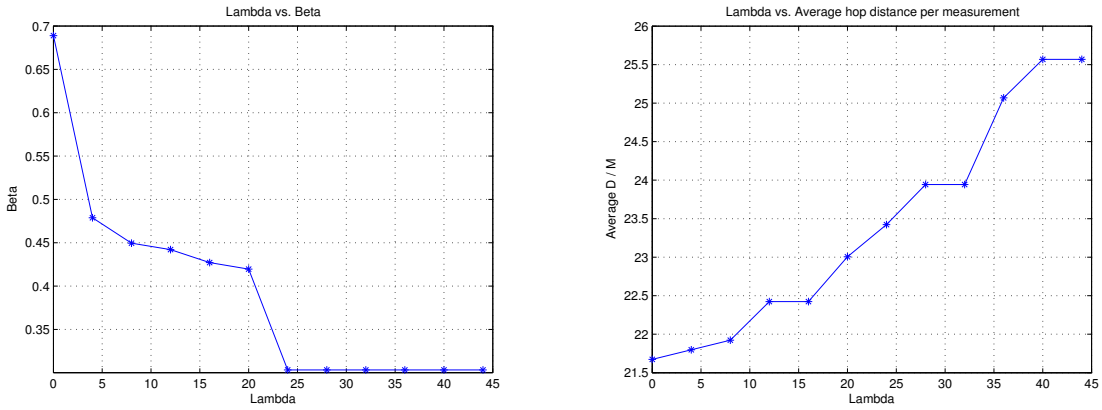
The goal of the algorithm is to construct N_c clusters that minimize transport costs, while guaranteeing perfect reconstruction for sparsity K. Transport costs depend on the distance between nodes and the number of measurements transmitted, which in turn depends on β. Thus we need joint optimization of β and the distance between nodes. For joint optimization, our proposed algorithm iteratively grows the clusters by finding an edges with minimum cost and adding to one of clusters.

To find a set of edges to form N_c clusters such that the total weight of the edges, $W(e, C_i)\ \forall i$, is minimized, we design an algorithm based on a greedy local heuristic. The algorithm starts from N_c initial nodes, one for each cluster; we deterministically chose N_c nodes located on the grid with equal distance to the adjacent starting nodes. At every iteration, we find edges connected to each of the clusters and compute the weights, $W(e, C_i)$. Then, the edge with the minimum weight is added to the cluster. This procedure continues until every node is assigned to one of the N_c clusters. Refer to Algorithm 1.

Algorithm 1 Joint Optimization of β and D

Given an undirected graph, $G(V, E)$, such that $|V| = N$.
Assign N_c nodes to clusters; one for each cluster, V_{C_i}.
$E_{C_i} = \varnothing,\ \forall i$.
for $k = 1$ to $N - N_c$ **do**
  Find $E_n = \{(v_1, v_2) \mid v_1 \in V, v_2 \in V_{C_i}, \forall i\}$
  Compute $W(e, C_i) = D(e) + \lambda\beta(e, C_i),\ \forall e \in E_n\ and\ \forall C_i$
  $(C_{i_{min}}, e_{min}) = \arg\min_{\forall C_i,\, e \in E_n} W(e, C_i)$
  $v_{min} = \left\{v_1 \mid e_{min} = (v_1, v_2), v_2 \in V_{C_{i_{min}}}\right\}$
  Add e_{min} to $E_{C_{i_{min}}}$ and v_{min} to $V_{C_{i_{min}}}$.
  Remove edges $\in \{e \mid e = (v_{min}, v), \forall v \in V_{C_i} \forall i\}$ from E.
  Remove v_{min} from V.
end for

The algorithm is similar to Prim's algorithm [43] for finding minimum spanning trees (MSTs). Given weights of edges, we choose an edge with minimum weight at every step as in Prim's algorithm. However, we have additional requirements as compared to Prim's algorithm. First, our algorithm finds N_c clusters with minimum total edge weights instead of an MST. Thus, an edge with the minimum weight is added to one of clusters to which the edge is connected rather than to a tree. Second, Prim's algorithm runs under the assumption that the weights of the edges do not change but, in our problem, the edge weights should be updated at every step.



(a) λ vs. β (b) λ vs. average D per measurement

Figure 2.4: Joint optimization of different λ. By running the algorithm with Daubechies 4 basis with 2^{nd} level of decomposition, 256 sensors are separated into 16 clusters. Different choices of λ generate different results; as λ increases, the edge weights are more sensitive to the change of β so that β decreases at the cost of increasing D.

Given a sparsifying basis and the positions of sensors, Algorithm 1 generates a set of N_c clusters by minimizing edge weights associated with a λ. Since edge weight $W(e, C_i) = D(e) + \lambda\beta(e, C_i)$, the design parameter λ controls the balance between two competing terms: $\beta(e, C_i)$ and $D(e)$. As λ increases, $\beta(e, C_i)$ is a more dominant factor than $D(e)$ so that the edges with smaller $\beta(e, C_i)$ have higher chance to be added to a cluster, which means that the spatial extent of clusters increases. As shown in Fig. 2.4, as λ increases, the final β (Definition 2.3.1) decreases but the average distance per measurement increases. However, it is not clear how to determine the best λ with respect to both reconstruction accuracy and transport cost.

The minimum number of measurements, M, decreases thanks to the decrease of β while the distance between nodes within the same cluster, D, increases. Since transport cost is determined by $C = MD$, different λ affects the overall transport cost and there could exist a (or a range of) good λ^* that achieves large energy

savings with the similar level of reconstruction accuracy. Thus, given a target reconstruction accuracy, we experimentally search for the minimum transport cost that provides that desired accuracy.

2.6 Simulation Result

The simulation consists of four parts. First, we compare performance between independent reconstruction and joint reconstruction. Second, we compare performance between two different routing schemes without the optimization of the maximum energy overlap, then compare the performance of SPT-based clustering to other CS techniques. Third, we verify Theorem 2.4.2 by examining the correlation between the estimated M_{est} by Theorem 2.4.2 and the minimum M_{sim} measured by simulation. Fourth, we evaluate the performance of the joint optimization in terms of transport cost and reconstruction quality. The details of different simulation environments will be described in following subsections.

2.6.1 Joint reconstruction vs. independent reconstruction

For the simulation, we used 500 realizations generated with sparsity $K = 55$, where non-zero coefficients were randomly selected. In the network, 1024 nodes are deployed on the square grid and error free communication is assumed. We do not assume any priority is given to specific clusters for measurements, i.e., we collect the same number of localized measurements for each cluster. With localized projection within each cluster, data is reconstructed jointly and independently with Gradient Pursuit for Sparse Reconstruction (GPSR) [29]. To evaluate performance, SNR is used to evaluate reconstruction accuracy. In this simulation, we focus on the reconstruction accuracy with increasing measurements.

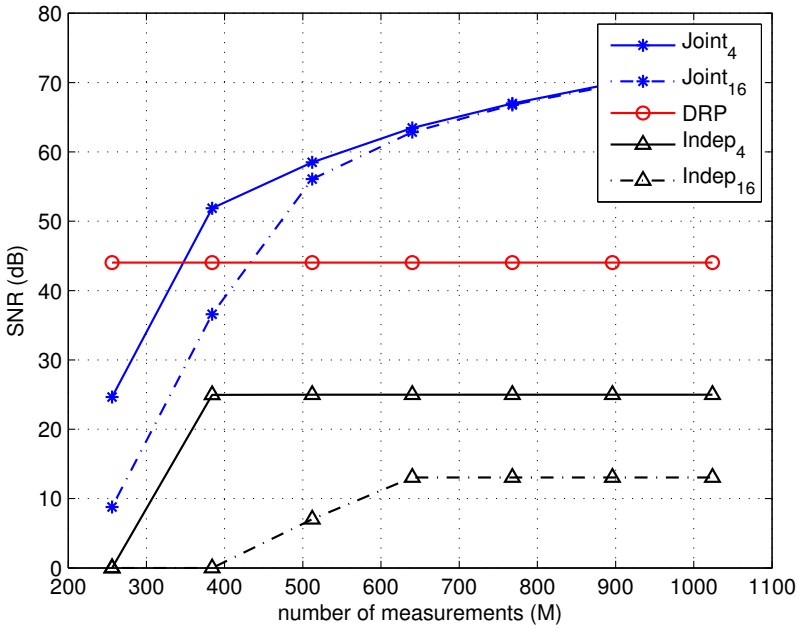


Figure 2.5: Independent reconstruction vs. joint reconstruction when the sparsifying basis is Haar basis with decomposition level of 5. For comparison, the square-clustering scheme with two different number of clusters (4 and 16 clusters) are used.

To compare independent reconstruction with joint reconstruction, we used a square-clustering scheme with two different number of clusters (4 and 16 clusters) and data sparse on the Haar basis with 5 levels of decomposition. In Fig. 2.5, dense random projection (DRP) corresponds to the case where 256 global measurements (i.e., each measurement is an aggregate of data from all the nodes in the network) are transmitted to the sink and then data is reconstructed using joint reconstruction. Other curves are generated from localized measurements in each cluster and the two types of reconstruction are applied.

Fig. 2.5 shows that joint reconstruction outperforms independent reconstruction as expected based on the discussion of Section 2.3.1. In all following simulations only joint reconstruction is used.

2.6.2 SPT-based clustering

For the simulation, we used the same setup as in Section 2.6.1. But, in this simulation, we consider two types of clustering with different number of clusters (16, 64, and 256 clusters): square-clustering and SPT-based clustering. For cost evaluation, transmission cost is estimated as $\sum(bit) \times (distance)^2$, as was done in [75, 76], although this could be extended to use more realistic cost metrics. We plot reconstruction SNR as a function of the rate of the transport cost of each scheme with respect to the transport cost in the raw data gathering without compression. While we use the cost ratio for evaluation, this simulation is mainly focused on the comparison between square and SPT-based comparison.

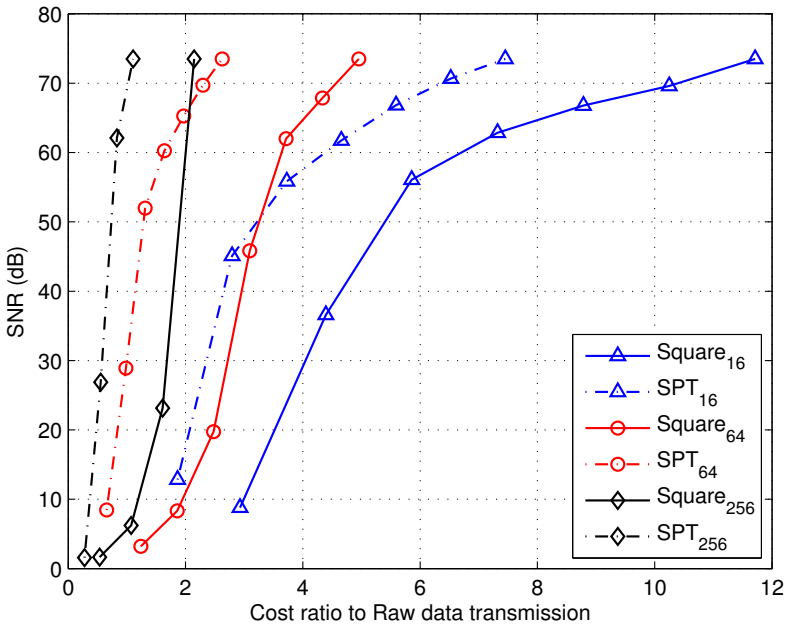


Figure 2.6: Cost ratio to raw data gathering vs. SNR with different number of clusters and clustering schemes when the sparsifying basis is Haar basis with decomposition level of 5.

With joint reconstruction and Haar basis, Fig. 2.6 shows that SPT-based clustering outperforms square clustering for different number of clusters (N_c). As N_c increases, reconstruction accuracy decreases because the measurement matrix becomes sparser as network is separated into more equal-size clusters. However, once the transport costs have been taken into account, more clusters show better performance because cost per measurement decreases and SPT-based clustering always outperform square clustering. Since we also observed this trend for different bases, we will focus on SPT-based clustering in following simulation.

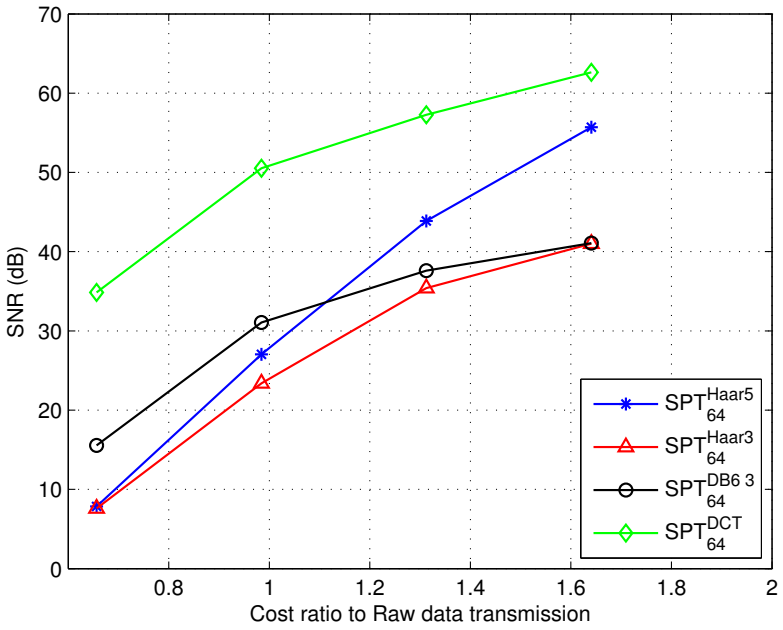


Figure 2.7: Performance comparison in terms of cost ratio with respect to raw data gathering vs. SNR for different basis functions and 64 SPT-based clusters.

In order to investigate effects of the spatial localzation of signals (determined by the spatial extent of basis functions), we generate data with the same sparsity in different bases and fix SPT-based clustering with 64 clusters. For comparison, we consider three different cases: (i) DCT basis, where each basis vectors have

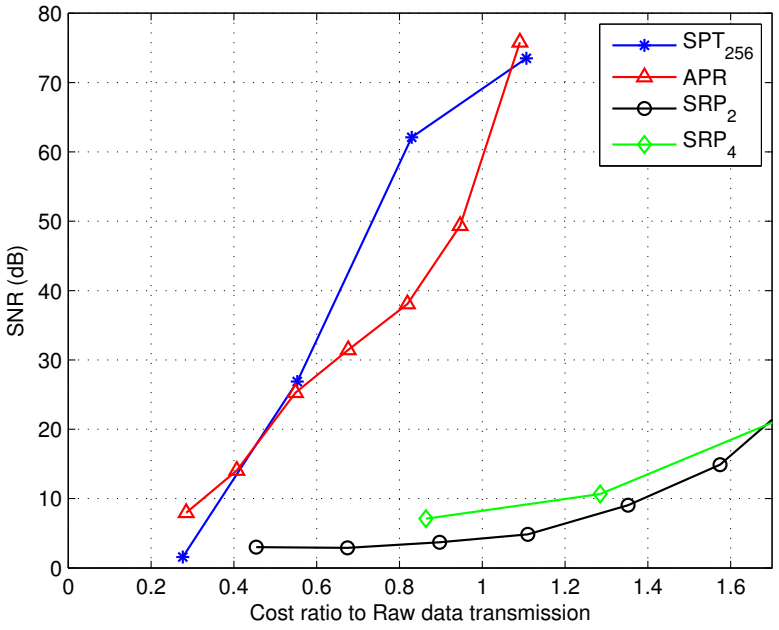


Figure 2.8: Performance comparison in terms of cost ratio with respect to raw data gathering vs. SNR for 256 SPT-based clusters with other CS approaches with Haar basis with level of decomposition of 5.

high overlaps in energy which distributed throughout the network, (ii) Haar basis, where the basis vectors have less overlap and the energy distribution of each basis varies from being very localized to being spread out over the whole field for different basis functions at a level, and (iii) Daubechies (DB6) basis, where the overlaps and distribution are intermediate between DCT and Haar. The result in Fig. 2.7 shows that, for the same clustering scheme and data with the same sparsity in a given basis, the gains from joint reconstruction depend on how"well-spread" the energy in the basis vectors is, that is, results are better when data is sparse in more "global" bases.

Fig. 2.8 shows that our approach outperforms other CS approaches [68,83]. APR corresponds to a scheme where aggregation occurs along the shortest path to the

sink and all the sensors on the paths provide their data for measurements [68]. SRP with different parameter, $s' = s \times n$, represents a scheme that randomly chooses s' nodes without considering routing, then transmits data to the sink via SPT with opportunistic aggregation [83].

In the comparison, SRP performs worse than the others because, as we expected, taking samples from random nodes for each measurement significantly increases total transmission cost. Our approach and APR are comparable in terms of transmission cost but our approach shows better reconstruction.

2.6.3 Reconstruction accuracy and β

In this simulation, we consider 3 different 2-D Daubechies basis with 2 levels of decomposition: DB4, DB6, and DB8. For each of the basis, we generate 1000 realizations with three different sparsity levels (K): 20, 38, and 55. In each realization, K basis functions are chosen at random and assigned random coefficients. In the network, 1024 nodes are deployed on a square grid in a region of interest and a sink is located at center of the field collecting measurements from sensors with error free communication. With M measurements, data is jointly reconstructed using the GPSR [29].

To verify Theorem 2.4.2, we first measure the minimum number of measurements, M_{sim}, required for perfect reconstruction in our simulation. To measure M_{sim}, we first evaluate reconstruction accuracy using the perfect reconstruction rate (Prr). We consider that for a given realization perfect reconstruction is achieved if $\max|x - \hat{x}| < 10^{-3}$. For 1000 synthesized sparse data, Prr is defined as a ratio of the number of data perfectly reconstructed to 1000. The minimum M_{sim} for perfect

Table 2.1: Correlation coefficient, $r \in [-1, 1]$, between M_{est} and M_{sim}

$N = 1024, N_c = 16$			
	DB_4	DB_6	DB_8
K=20	0.49	0.44	0.50
K=38	0.59	0.52	0.61
K=55	0.68	0.57	0.63

reconstruction is the smallest M that satisfies a perfect reconstruction rate larger than 0.99.

To collect spatially-localized projections, N sensors are separated into N_c non-overlapped clusters with the same size; every cluster contains N/N_c sensors. We consider 20 different clustering schemes. For each clustering scheme, we divide N sensors in the field into 16 localized clusters with radial shape going from the sink to the boundary of the network. To generate 20 different clustering schemes, we rotate the 16 clusters with a certain angle for each clustering scheme.

To estimate M_{est}, for given $\mathbf{\Psi}$ and K, we first compute β for each clustering scheme so that we have 20 different values of β, one for each clustering scheme. Then, M_{est} is computed following Theorem 2.4.2. To check if M_{est} and M_{sim} are correlated to each other, we use Pearson's linear correlation coefficient, $r \in [-1, 1]$, which measures the linear dependence between two variables [64].

As shown in Table 2.1, the correlation value is around 0.55 for different K and $\mathbf{\Psi}$, which shows that β affects reconstruction accuracy in terms of the minimum number of measurements for perfect reconstruction. However, our bound is not tight enough to estimate the exact number of measurements because there exists a gap between M_{est} and M_{sim}. However, β is a useful metric because, based on β, we can compare different clustering schemes and also design a clustering scheme by optimizing β as discussed in Section 2.5.

Since the maximum energy overlap is a worst case measure, β can be misleading. For example, suppose there exists a basis function and cluster for which the energy overlap is 1 (perfect overlap), but the others are relatively small; in this case by definition $\beta = 1$. However, successful reconstruction is possible with high quality if the basis function associated to the large energy overlap is not in the data support. Therefore, it would be meaningful to examine the impact of maximum energy overlap of each basis function and clusters on the level of error observed for that basis function.

For each basis function, we define β_i by measuring the maximum energy overlap between the i^{th} basis function and all clusters. Thus, the β_i characterizes the distribution of energy of individual basis functions. Table 2.2 shows the average correlation between β_i and the error related to the corresponding basis function. The simulation results show that basis functions with larger β_i generate more errors. This implies that basis functions with concentrated energy on fewer clusters have lower probability to be identified by spatially-localized measurements. While the results do not show very high correlation; sufficient to reliably estimate the number of measurements required for perfect reconstruction, is well estimated, β can be a useful metric to find a better clustering scheme, which will be discussed in Section 2.6.4.

Table 2.2: Correlation coefficient, $r \in [-1, 1]$, between maximum energy overlap and MSE for each basis

$N = 1024, N_c = 16, M = 410$			
	DB_4	DB_6	DB_8
K=20	0.31	0.58	0.52
K=38	0.48	0.62	0.55
K=55	0.47	0.63	0.64

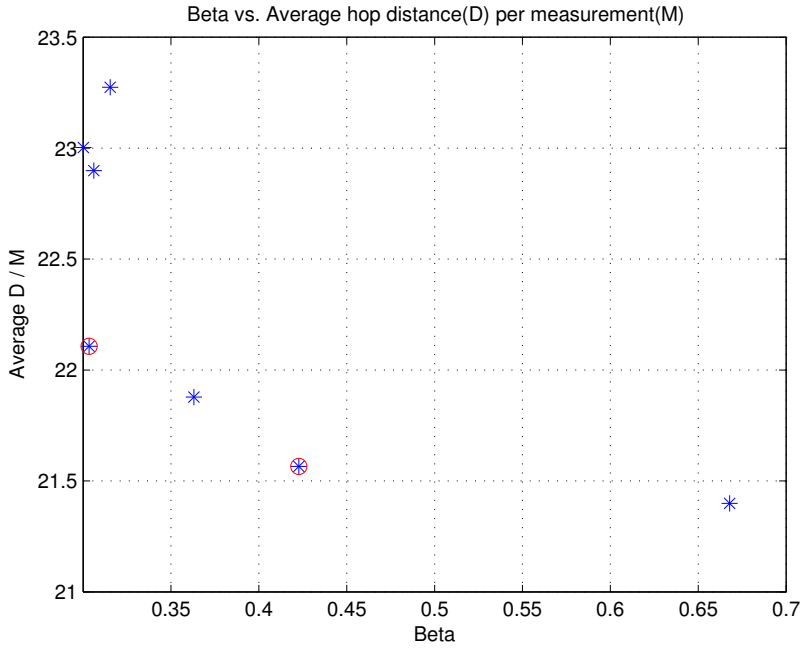


Figure 2.9: β vs. average number of hops per measurement. Each point corresponds to the result of Algorithm 1 with different λ. The points with red circle are chosen for the evaluation of transport cost and MSE in Fig. 2.10

2.6.4 Joint optimization

To evaluate the joint optimization, we use mean squared error (MSE) as a metric for reconstruction accuracy. This is because, in practice, we are more interested in the level of error associated to a specific transport cost. Since we allocated the same number of bits for each measurement, transmission cost, $\sum$ (bit) $\times$ (number of hops), depends on the product of the number of measurements with the distance in hops. In our simulation, we consider a signal with sparsity $K = 38$ in 2D Daubechies-4 basis with 2 levels of decomposition. Located at center of the field, a sink collects M measurements from 64 clusters. For energy efficiency, measurements from each cluster are routed to the sink along shortest path. For comparison with other CS

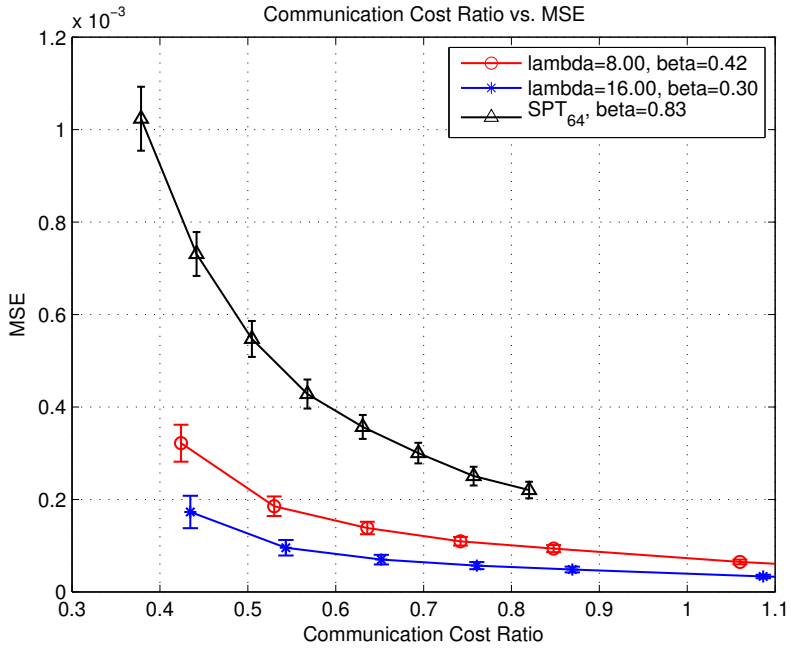


Figure 2.10: Transport cost ratio vs. MSE. The x-axis is the ratio of total transport cost of spatially-localized CS to the cost for raw data gathering without any manipulation. We compare performance of results by joint optimization with two different λ's with that of SPT_{64} in [48]

approaches, we consider a clustering scheme based on shortest path tree (SPT) that showed the best performance in Section 2.6.2

As discussed in Section 2.5, in general, smaller β can be achieved by increasing distance between nodes in the same cluster. The tradeoff can also be observed in Fig. 2.9; as λ increases, we can achieve smaller β but this increases hop distance per measurement as discussed in Section 2.5. In addition, Fig. 2.9 shows that, as λ increases, β decreases quickly but becomes saturated at some point. After that, transport cost increases without improvement of β. Thus, we can expect that one of λ's around the saturation point will correspond to a good operating point showing better performance in terms of total transport cost and reconstruction accuracy.

Fig. 2.10 shows the overall performance with different λ. Each curve shows the average MSE over 100 realizations and the variation in reconstruction accuracy is expressed as three times of standard deviation. As expected, a λ with the value of 16 located at the sharp transition in Fig. 2.9 shows the best performance. With the best λ and 1024 sensors divided into 64 clusters, we can achieve 40% cost saving with respect to raw data-gathering with small mean squared error ($\leq 1 \times 10^{-4}$). Compared with SPT-based clustering scheme in [48], our clustering scheme with joint optimization achieves almost 50% reduction in transport costs at the same level of mean squared error in the reconstruction. Also, Fig. 2.10 confirms that for a given transport cost β is a good predictor of reconstruction quality; lower β leads to better reconstruction quality. The clusters in the SPT-based clustering scheme consume less energy to construct a measurement. However, the savings in transport cost is compensated by a larger number of measurements required for the same level of reconstruction quality which can be explained by large value of $\beta(= 0.83)$.

2.7 Extension to irregularly positioned sensors

Up to this point, we have proposed an energy-efficient data gathering scheme in WSN. This approach has two major limitations in practice: (i) regular positions of sensors on the 2D grid and (ii) K-sparse synthesized data in a given sparsifying basis $\mathbf{\Psi}$. The regular topology is useful for monitoring buildings, bridges, or power plants but is not appropriate for many other applications such as monitoring of habitat, wild fire, or battle field. This motivate us to study how the CS-based approach can be extended to data-gathering with irregularly positioned sensors. Also, k-sparse data in a given $\mathbf{\Psi}$ is unlikely to happen in reality. Most signals

are compressible in irregular bases, which encourages us to investigate the use of graph-based transforms (GBT) that can provide a sparse representation of realistic sensor data.

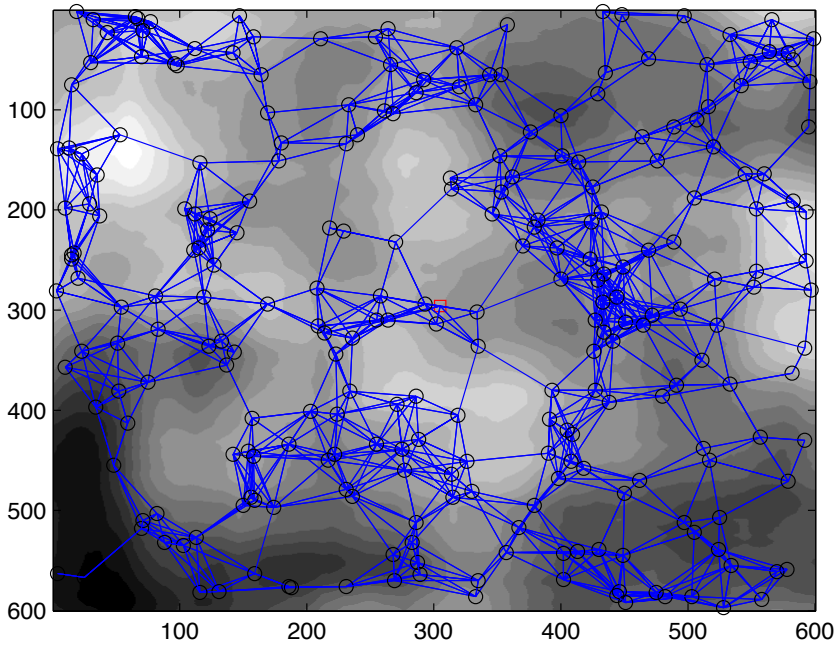


Figure 2.11: 256 sensors in irregular positions and the corresponding graph. The communication range is set as the minimum distance that results in a connected graph.

In this work, we consider a realistic data model with irregularly positioned sensors. We first generate data that is independent of the sparsifying basis, using a second order AR model as shown in Fig. 2.11. In the noise-free data, the correlation of data between two nodes increases as the distance between them decreases. On the smooth field of data, 256 sensors are randomly deployed, and we measure then transmit the data to the sink positioned at the center of the field along the SPT.

Under these realistic assumptions, we use graph-based transforms (GBT) as a sparsifying basis because the transform can be applied to various deployments of sensors if the topology is represented by a graph. For the construction of GBT, we first represent the WSN as a graph, $G(V, E)$ with nodes (sensors) and links (connections) between sensors as illustrated in Fig. 2.11. Note that the links can exist only if the two sensors are within a specific range. In this work, we set the range as the minimum such that the resulting graph is connected (i.e., there are no disconnected subgraphs) as shown in Fig. 2.11. Since the sensor data is likely to be highly correlated between adjacent sensors, the links between distant sensors that are farther apart can be disconnected for a sparser representation. Note that the GBT is only required at the sink to reconstruct the received signal, so that sensors in the field do not need to know the complete topology of the network in order to transmit data. In addition, It is possible for nodes to decide locally on how to transmit data, so that centralized coordination is not required.

From the graph, the adjacency matrix $\boldsymbol{A}$ is formed, where $\boldsymbol{A}(i,j) = \boldsymbol{A}(j,i) = 1$ if the distance between sensor i and j is smaller than the minimum communication range. Otherwise $\boldsymbol{A}(i,j) = \boldsymbol{A}(j,i) = 0$. Then we define the degree matrix $\boldsymbol{D}$, where $\boldsymbol{D}(i,i)$ is the number of links connected to the i^{th} sensor and $\boldsymbol{D}(i,j) = 0, \ \ \forall i \neq j$. Finally, the Laplacian matrix can be defined as:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A} = \begin{cases} -1, & if \ \ (i,j) \in E \\ d_i, & if \ \ i = j \\ 0, & otherwise \end{cases} \tag{2.19}$$

After the eigenvalue decomposition, we use the eigenvector matrix as a sparsifying basis, $\boldsymbol{\Psi}$, whose columns are the eigenvectors of the Laplacian matrix, $\boldsymbol{L}$. Note that $\boldsymbol{\Psi}$ is orthogonal because $\boldsymbol{L}$ is symmetric, leading to real eigenvalues and

a set of orthogonal eigenvectors (refer to [34, 78] for more details). Fig. 2.12 shows the performance of the GBT as a sparsifying basis. Although the sensor data is not perfectly sparse, the GBT shows a good compressibility, i.e., more than 99% of energy is compacted in a few GBT coefficients.

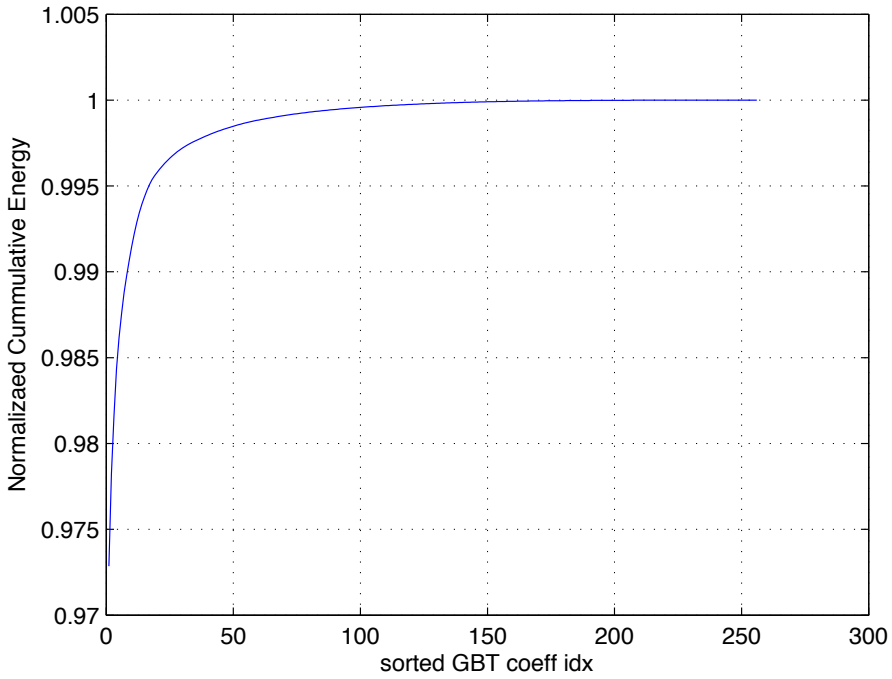


Figure 2.12: Compressible WSN data in the GBT. The x-axis shows the indices of GBT coefficients and the y-axis shows the cumulated sum of normalized energy of GBT basis functions. As shown in Fig. 2.11, the data is generated by a 2^{nd} AR model with the AR filter $H(z) = \frac{1}{(1-\rho e^{jw_0}z^{-1})(1-\rho e^{-jw_0}z^{-1})}$, where $\rho = 0.99$ and $w_0 = 359$

For CS-based data-gathering, we consider two approaches: SPT aggregation and GBT-aware aggregation. For the SPT aggregation (same as APR in Section 2.6.2), we randomly choose a certain number of sensors, and aggregate data of all the sensors on the SPT. Then, the linear combinations of data with Gaussian random coefficients are transmitted along the SPT [68]. Alternatively, the GBT-aware

aggregation selectively chooses the sensors by considering the number of the links connected to the sensors in the graph that is used for GBT construction as in Fig. 2.11. From a certain number of randomly chosen sensors, the aggregation happens along the SPT as in the first approach. But, an aggregation takes place at a sensor if the number of neighbors connected to the sensor is less than a threshold. Otherwise, the sensor relays the received data to its parent sensor along SPT. Since the GBT can be interpreted as a spectral decomposition over the links, the aggregation over the sensors with fewer neighbors in the graph increases the amount overlap, which leads to better reconstruction as discussed in Section 2.4. The threshold is empirically chosen in this work. Once the threshold is determined, we do not need to update it if the topology remains the same and the sensor data is sparsely represented by the constructed GBT, so that our propose approach does not require a lot of coordination if the topology is known locally. Also, since the aggregation decision depends on a characteristic of local network, this approach can be done in a decentralized way.

In this simulation, a second order AR model is used to generate 50 realizations with high spatial data correlation as shown in Fig. 2.13. More specifically, the AR filter $H(z) = \frac{1}{(1-\rho e^{jw_0}z^{-1})(1-\rho e^{-jw_0}z^{-1})}$, where $\rho = 0.99$ and $w_0 = 359$. For the simulation, 256 sensors are randomly positioned in the 600×600 grid and the data measured at each sensor is represented using 12 bits. Also, the measurements (or down-sampled data) are transmitted along the SPT as shown in Fig. 2.13. Note that the locations of the sensors do not change throughout our simulations. Also, for measuring energy consumption, we adopt a realistic cost model proposed in [35,82]. Energy in the sensors is dissipated when both transmitting, $E_T(k, D)$, and receiving data, $E_R(k)$. The energy consumption in k bit transmission over a distance D is

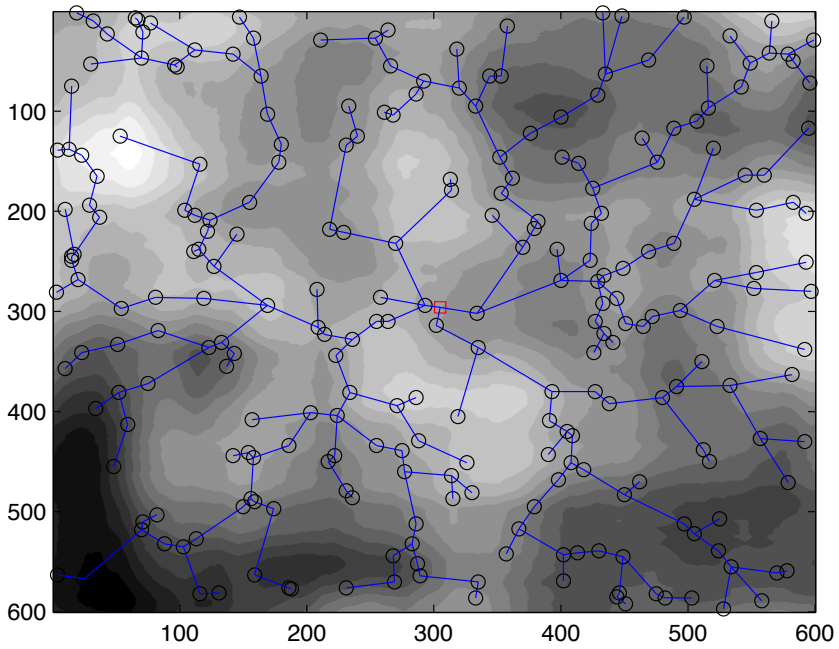


Figure 2.13: 256 sensors in irregular positions and the corresponding SPT.

$E_T(k, D) = E_{elec}k + \varepsilon_{amp}kD^2$ Joules and the consumption in k bit reception is $E_R(k) = E_{elec}k$.

In our simulation, we compare four different approaches: (i) SPT aggregation (CS_{SPT}) [68], (ii) GBT interpolation($itpl_{GBT}$) [58], (iii) raw transmission without any compression, and (iv) our proposed method (CS_{GBT}). For $itpl_{GBT}$, we randomly choose a certain number of sensors and transmit the sampled data to the sink along the SPT. Then, data is reconstructed by the graph interpolation technique proposed in [58]. The curve for the raw data transmission is generated with different levels of quantization, but the other curves use a fixed quantization step size. For CS_{GBT}, we empirically choose the threshold as 5, thus the aggregation happens at the nodes on SPT if the nodes have fewer than 5 neighbors in the graph in Fig. 2.11. The result in Fig. 2.14 shows that our proposed approach outperforms

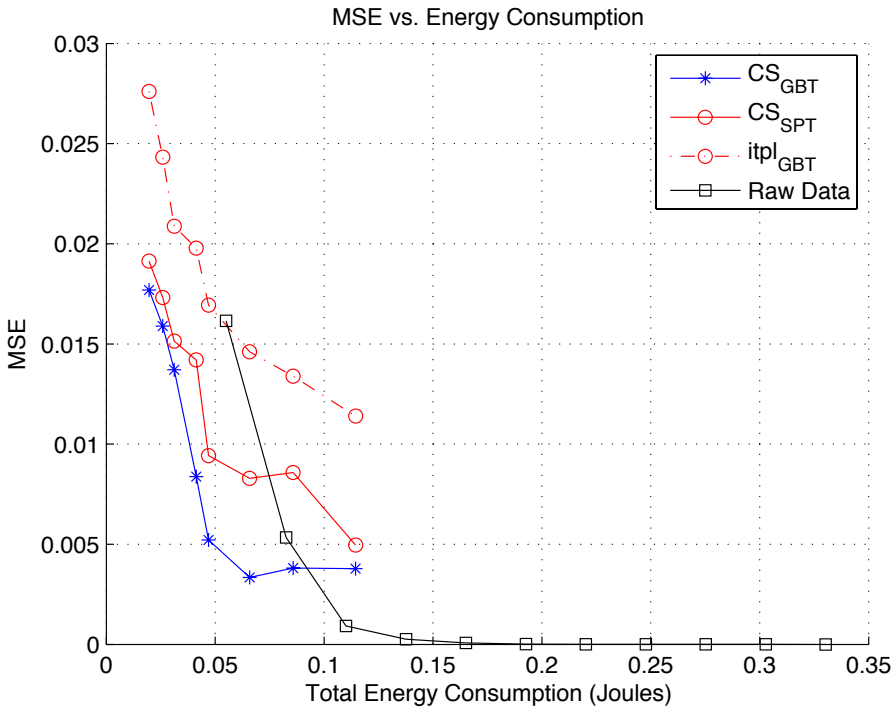


Figure 2.14: Total energy consumption vs. MSE. The x-axis is the total energy consumption in Joules and the y-axis is MSE. The curves are generated by taking averages over 50 realizations of the sensor data.

the other methods in terms of the energy consumption and the reconstruction accuracy. The GBT interpolation technique shows worse performance because it assumes a bandlimited graph signal supported only at frequencies $[0, w]$, but the cutoff frequency,w, in the graph is not small enough with respect to the downsampling rate (i.e., the ratio of the number of sensors providing samples with respect to the total number of sensors), so that the reconstruction quality is degraded.

2.8 Conclusion

To achieve energy efficient data gathering in WSN, we exploit a sparse and spatially-localized CS measurement system that is aware of transport cost per measurement

by constructing measurements within spatially-localized clusters. However, while the spatially-localized measurement system leads to lower transport cost, it is not obvious how it affects reconstruction accuracy. Thus, we first introduce a metric to measure the maximum energy overlap between clusters and basis functions, β. Then we show that the metric has an impact on reconstruction accuracy with respect to the number of measurements for perfect reconstruction. By exploiting the tradeoff between β and distance between sensors in clusters, we propose a centralized iterative algorithm with a design parameter, λ, to construct clusters that are jointly aware of energy efficiency and reconstruction quality. Our simulation results show that, with an an appropriately chosen λ, we can achieve significant savings in transport cost with small reconstruction error. Also, we extend our work to a WSN that consist of sensors in irregular positions. This work shows a preliminary result that the sensor data can be sparsely represented by GBT and our proposed approach shows a promising performance compared to other existing methods.

Chapter 3

Adaptive Compressed Sensing for Depth map Compression Using Graph-based Transform

3.1 Introduction

Standard compressed sensing (CS) theory prescribes that robust signal recovery is possible when a signal is sparse in a given sparsifying basis. Based on the signal characteristics, the sparsifying basis is often assumed to be known *a priori* at the decoder. However, for coding applications where signals are first captured and then compressed, better performance can be achieved by adaptively selecting a transform or sparsifying basis and then signaling the chosen transform to the decoder. For instance, for piecewise smooth signals, where sharp edges exist between smooth regions, edge-adaptive transforms can provide sparser representation at the cost of some overhead.

In this work, we consider block-based depth map compression as an example application. Previous work has shown that edge adaptive transforms can be more efficient than standard transforms (e.g., DCT) due to the piecewise smooth nature of these signals [74]. Moreover, correct representation of edges is important because errors in edge information lead to significant degradation of the quality of

interpolated views in 3-D TV applications [39, 40]. For depth map compression, CS-based methods have been recently proposed. CS is applied by either projecting the depth map on a random sensing matrix (Cartesian grid sampling technique) [73] or down-sampling 2D-DCT coefficient [25]. However, performance gains achieved by these techniques are limited because the standard DCT is chosen as the sparsifying basis, which is inefficient for coding blocks containing arbitrarily shaped edges (i.e., neither vertical, nor horizontal) separating smooth regions. To preserve the edge information, many researchers have investigated efficient transforms for depth map, that avoid filtering across edges [54, 56]. However, these methods have limitations. For example, Platelets [56] have a fixed approximation error because depth maps are not exactly piece-wise planar. Shape-Adaptive Wavelets [54] are not amenable to a block based coding architecture, which has been widely adopted in international standards for image and video coding such as H.264/AVC.

To overcome those limitations of existing transforms, a graph based transform (GBT) has been proposed [74] that can achieve a sparse representation, even when arbitrary edges exist in a block. It also has the advantage that it can be easily applied within a block-based coding architecture. GBT is based on representing each block as a graph, where each vertex corresponds to a pixel, and vertices are linked only when no strong edges are present between the corresponding pixels. For any given block with arbitrary size, different graphs can be chosen, leading to different transforms, which depend on the edge structure and therefore require that overhead bits be sent to the decoder. In [74] it was shown that these adaptive GBTs improved performance as compared to DCT-only methods, even when the overhead was taken into account. This work was further extended in [38], which proposed a simple cost function and a search technique to optimize the GBT selection for each

block, balancing the increased sparseness achievable if more edges are considered with the added overhead required for transmitting this information to the decoder.

In this thesis, we propose a novel CS framework where the adaptive GBT is used as a block-adaptive sparsifying basis. We consider the problem of, given a specific sensing matrix, optimizing the sparsifying basis. In this work, we first fix the sensing matrix as a Hadamard matrix because (i) the retrieval of measurements is computationally simple at the encoder and (ii) the computation of our approximated mutual coherence in Section 3.3 can be greatly simplified. Then, we optimize the choice of GBT by taking into account the reconstruction quality and the overhead required to specify the GBT. Note that the approach in [38] aims at selecting a GBT that provides maximum sparsity for a block, without requiring excessive overhead. A key result in our work is to show that maximum sparsity *does not* guarantee optimal performance when using CS. As studied in [8, 45], CS reconstruction depends not only on the sparsity of signal representation but also on the mutual coherence between sensing matrix and sparsifying basis. Therefore, a GBT providing the sparsest representation of depth data is not necessarily maximally incoherent with a given Hadamard sensing matrix. Thus, joint optimization is required to select the best GBT for a given depth map, taking into account rate overhead (to specify the transform), sparsity of the representation and mutual coherence. We propose a greedy iterative algorithm that evaluates a metric for different edge configurations before selecting one. This algorithm uses a low-complexity estimate of the mutual coherence, so that explicit construction of the GBT at the encoder is only required once the edge map has been selected (i.e., it is not required in the iterative process leading to this selection). The proposed block-adaptive CS approach is integrated within an H.264 codec. When evaluating its intra coding performance on three

depth map sequences, we observe 3.8 dB PSNR gain in the quality of interpolated views obtained from the decoded depth map, or an average of 39% bitrate savings.

The rest of this chapter is organized as follows. We first formulate the problem in Section 3.2. Then we provide a theoretical result showing that a good GBT can be found by an approximation of mutual coherence in Section 3.3 and the performance is verified by our simulation results in Section 3.4.

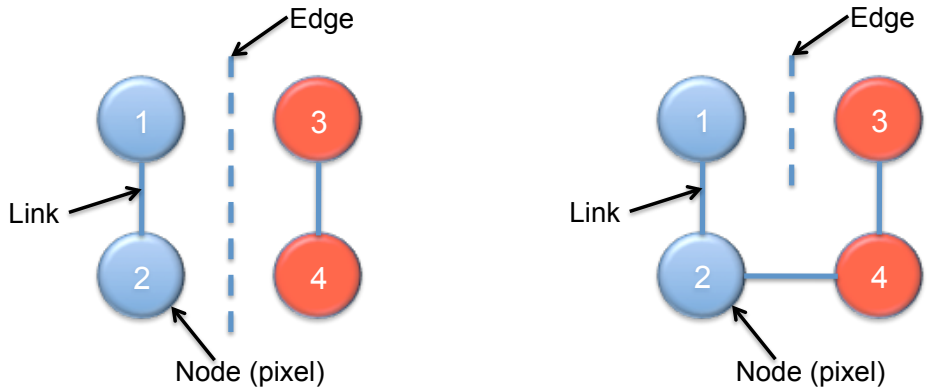
3.2 Problem Formulation

For the construction of GBT, we first represent each depth block as a graph, $G(V, E)$ with nodes (pixels) and links (connections) between nodes as illustrated in Fig. 3.1. Note that we use the term "edge" to refer to image edges in order to avoid confusion with the links in the graph. A link is present in the graph only when no edge was selected between the two corresponding pixels. Later in this chapter, we will discuss how edges between pixels will be identified, which is closely related to the optimization of GBT.

In this work, we assume 4-neighbor connectivity for the pixels, so that each node, V, can have at most 4 links. From the graph, the adjacency matrix $\boldsymbol{A}$ is formed, where $\boldsymbol{A}(i,j) = \boldsymbol{A}(j,i) = 1$ if pixel positions i and j are immediate neighbors not separated by an edge. Otherwise $\boldsymbol{A}(i,j) = \boldsymbol{A}(j,i) = 0$. Then we define the degree matrix $\boldsymbol{D}$, where $\boldsymbol{D}(i,i)$ is the number of links connected to the i^{th} pixel and $\boldsymbol{D}(i,j) = 0, \ \forall i \neq j$. Finally, the Laplacian matrix can be defined as:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A} = \begin{cases} -1 & if \ \ (i,j) \in E \\ d_i & if \ \ i = j \\ 0 & otherwise \end{cases} \tag{3.1}$$

Note that $\boldsymbol{L}$ is symmetric, leading to real eigenvalues and a set of orthogonal eigenvectors. Thus we define the GBT for a given graph as the eigenvector matrix, $\boldsymbol{\Psi}$, whose columns are the eigenvectors of the Laplacian $\boldsymbol{L}$ of the graph. Since $\boldsymbol{\Psi}$ is orthogonal, its inverse is $\boldsymbol{\Psi}^T$.



(a) Example 1 (b) Example 2

Figure 3.1: 2-by-2 block examples: we have four pixels (nodes) in the block and assume that there exists one edge between pixels. (a) the edge separates those pixels into two partitions. Since the links between nodes in the graph are assumed not to go across the edge, we have two sub-graphs where 2 nodes are connected to each other by a link. (b) while an edge exists, the edge does not separate pixels into two partitions, generating only one graph.

As an example, consider the four pixels separated by an edge as shown in Fig. 3.1. By considering the pixels as nodes and assuming the links between pixels cannot go across an edge, the block can be represented as a graph, $G(V, E)$. To construct the GBT from the graph in Fig. 3.1 (a), we first construct the adjacency matrix, $\boldsymbol{A_1}$, and the degree matrix, $\boldsymbol{D_1}$, which has only 1's on its diagonal because each pixel is connected to only one other pixel, due to the existence of an edge:

$$\boldsymbol{A_1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \qquad \boldsymbol{D_1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.2}$$

From $\boldsymbol{D_1}$ and $\boldsymbol{A_1}$, the Laplacian matrix, $\boldsymbol{L_1}$, is defined as in (3.1). To construct the GBT, $\boldsymbol{\Psi_1}$, from $\boldsymbol{L_1}$, we first apply the eigenvalue decomposition to $\boldsymbol{L_1}$ then we form $\boldsymbol{\Psi_1}$ with column vectors corresponding to the eigenvectors of $\boldsymbol{L_1}$:

$$\boldsymbol{L_1} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad \boldsymbol{\Psi_1} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix}. \tag{3.3}$$

Similarly, the GBT for the example of Fig. 3.1 (b), $\boldsymbol{\Psi_2}$, can be generated as follows.

$$\boldsymbol{A_2} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \qquad \boldsymbol{D_2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}, \tag{3.4}$$

$$\boldsymbol{L_2} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & 0 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}, \quad \boldsymbol{\Psi_2} = \begin{bmatrix} 0.5 & 0.65 & -0.5 & 0.27 \\ 0.5 & 0.27 & 0.5 & -0.65 \\ 0.5 & -0.65 & -0.5 & -0.27 \\ 0.5 & -0.27 & 0.5 & 0.65 \end{bmatrix}. \tag{3.5}$$

For better visualization, the j^{th} basis function of $\boldsymbol{\Psi_i}$, $\boldsymbol{\Psi_i^j}$, can be shown as 2×2 matrices corresponding to the pixel locations in Fig. 3.1, so that we have:

$$\boldsymbol{\Psi_1^1} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix}, \quad \boldsymbol{\Psi_1^2} = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix}, \tag{3.6}$$

$$\boldsymbol{\Psi_1^3} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix}, \quad \boldsymbol{\Psi_1^4} = \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix}, \tag{3.7}$$

$$\boldsymbol{\Psi_2^1} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}, \quad \boldsymbol{\Psi_2^2} = \begin{bmatrix} 0.65 & -0.65 \\ 0.27 & -0.27 \end{bmatrix}, \tag{3.8}$$

$$\boldsymbol{\Psi_2^3} = \begin{bmatrix} -0.5 & -0.5 \\ 0.5 & 0.5 \end{bmatrix}, \quad \boldsymbol{\Psi_2^4} = \begin{bmatrix} 0.27 & -0.27 \\ -0.65 & 0.65 \end{bmatrix}. \tag{3.9}$$

One observation from the two examples in Fig. 3.1 is that the bases (column vectors in $\boldsymbol{\Psi_1}$ and $\boldsymbol{\Psi_2}$) can take arbitrary values across an image edge. That is, there is no need for one of the basis functions to provide a smooth approximation to an image discontinuity. Since the edge in the 2^{nd} example does not separate 4 nodes into more than one partitions, all the basis functions in $\boldsymbol{\Psi_2}$ are global bases that are completely different from those in $\boldsymbol{\Psi_1}$. These show that the different edges (edge map) lead to different GBTs that provide different sparsity in the representation. A spectral decomposition, defined as the projection of a signal onto the eigenvectors of $\boldsymbol{L}$ (equivalently, projection onto $\boldsymbol{\Psi^t}$), can be interpreted as providing the "frequency" contents of the graph signal [34, 78]. Note that in the example of Fig. 3.1(b) the corresponding bases in (3.8) and (3.9) behave as 1-D bases of increasing frequency (more zero crossings) as we follow the links from node 1 to node 3, avoiding the edge between them.

In our depth map compression application, block-adaptive GBTs are applied to residual blocks obtained after intra/inter prediction, where the graph from which the GBT is derived is chosen based on the edges present in each residual block. For each block, these edges could be detected by applying a simple threshold to the difference between neighboring residual pixel values [74]. However, using the same threshold for all blocks does not take into account the overhead required to transmit the chosen edge map to the decoder, which tends to increase with the number of edges. Thus, two blocks may achieve similar levels of sparsity for a given threshold, but the block where more edges are identified may require a higher overall rate. As an alternative, the work in [38] seeks to find the optimized edge map for each block by considering this overhead. In this work, we also investigate the selection of edge maps that are optimized in order to consume fewer bits, while providing satisfactory reconstruction by compressed sensing.

A key observation in our work is that the optimized GBT (which [38] attempts to obtain) may not provide optimal performance if CS is used. This is because performance depends both on the level of sparsity in the representation and on the incoherence between the sparsity basis and the measurement matrix, which is very important for reconstruction, as studied in [8, 45]. For any GBT chosen as the sparsifying basis, $\mathbf{\Psi}$, we can compute the mutual coherence with a fixed Hadamard sensing matrix, $\mathbf{\Phi}$, $\mu(\mathbf{\Phi\Psi})$. Based on the mutual coherence, the minimum number of measurements for perfect reconstruction can be computed as $M = O(K\mu^2(\boldsymbol{U})N \log N)$ [8]. The lower bound on the number of measurements decreases as $\mathbf{\Phi}$ and $\mathbf{\Psi}$ become increasingly incoherent. Thus, if we can estimate how μ varies as a function of the chosen GBT, then we can also compute a bound on the number of measurements needed, which will predict achievable performance.

With this, and with an estimate of the cost required to encode the corresponding edge map, it becomes possible to design a good GBT.

3.3 Optimizing GBT for CS

3.3.1 Bound on the mutual coherence

We first derive a bound on the mutual coherence for a given GBT matrix, $\mathbf{\Psi}$, and the Hadamard matrix, $\mathbf{\Phi}$. Since both matrices are deterministic, the mutual coherence is also deterministic, and could be computed for each candidate GBT. However, GBT construction is a complex operation as it requires finding all the eigenvectors of the Laplacian matrix. The complexity grows as the size of graph (equivalently, the block size in depth map compression) increases. Even if there exist only a few GBTs that are truly useful and for those we could precompute the mutual coherence, the number of useful GBT candidates also increases with the graph size, which leads to larger memory requirements. Thus it would be desirable to avoid having to construct GBTs at every stage of the search for the optimized GBT. In what follows, we derive upper and lower bounds on the mutual coherence then use their average to estimate the mutual coherence of the block.

We first derive the upper bound of the mutual coherence.

Theorem 3.3.1. *For a given graph $G(V, E)$, the mutual coherence, μ, between the Hadamard sensing matrix, $\mathbf{\Phi}$, and a graph-based transform matrix, $\mathbf{\Psi}$, satisfies*

$$\mu(E) \leq \sqrt{\frac{\max_{\forall i} N_{G_i}}{N}},$$

where N_{G_i} denotes the size of the i^{th} sub-graph (equivalently, the number of pixels in the sub-graph).

If a graph is connected, then the mutual coherence is bounded by 1 because the DC component of the Hadamard basis is identical to the eigenvector corresponding to the zero eigenvalue of the graph Laplacian. In contrast, a fully disconnected graph where all the pixels are separated by edges can achieve the minimum bound for the mutual coherence. However, this increases the overhead to encode the edge map so that the coding gain is limited. The proof is trivial because all the entries of the Hadamard matrix are $\pm 1/\sqrt{N}$ and all the basis functions of GBT (columns of $\mathbf{\Psi}$) are normalized to 1, so that the maximum absolute value of the inner-products is bounded by the maximum size of a group normalized by N. Next, a lower bound on mutual coherence is derived.

Theorem 3.3.2. *For a given graph $G(V, E)$, mutual coherence, μ, between an arbitrary sensing matrix, $\mathbf{\Phi}$ and a graph-based transform, $\mathbf{\Psi}$, satisfies*

$$\mu(E) \geq \max_{\forall k} \sqrt{\frac{\sum_{(l,m)\in E} \left(\mathbf{\Phi}(k,l) - \mathbf{\Phi}(k,m)\right)^2}{2|E|}},$$

where $|E|$ is the total the number of links between nodes.

Since we consider 4×4 blocks with 4-neighbor connectivity in this work, the maximum $|E|$ is equal to 24 if $G(V, E)$ is a fully-connected graph. The numerator of the bound is a sum of squared differences of $\mathbf{\Phi}(i, j)$ corresponding to connected pixels. The bound indicates that the lower bound of the mutual coherence increases as more pixels corresponding to high variation of $\mathbf{\Phi}$ are connected. Since $\mathbf{\Phi}$ is a

Hadamard matrix in this work, this corresponds to counting the number of zero crossings in the graph for all Hadamard bases.

The proof is based on the fact that $\boldsymbol{x^T L x} = \sum_{(i,j)\in E}(x(i) - x(j))^2$, for any $\boldsymbol{x} \in \Re^V$. Let $\boldsymbol{x}^T = \boldsymbol{\Phi}(i,:)$. Since $L = \boldsymbol{\Psi\Lambda\Psi}^T$,

$$\boldsymbol{x^T L x} = \boldsymbol{\Phi}(i,:)(\boldsymbol{\Psi}\Lambda\boldsymbol{\Psi}^T)\boldsymbol{\Phi}(i,:)^T \tag{3.10}$$

$$= \sum_{(l,m)\in E} \left(\boldsymbol{\Phi}(i,l) - \boldsymbol{\Phi}(i,m)\right)^2, \;\; i \in \{1,2,\ldots,N\} \tag{3.11}$$

(3.10) can be rewritten as follows:

$$\boldsymbol{x^T L x} = \boldsymbol{\Phi}(i,:)(\boldsymbol{\Psi}\Lambda\boldsymbol{\Psi}^T)\boldsymbol{\Phi}(i,:)^T \tag{3.12}$$

$$= \boldsymbol{U}(i,:)\boldsymbol{\Lambda}\boldsymbol{U}(i,:)^T \tag{3.13}$$

$$= \sum_j \lambda_j \boldsymbol{U}(i,j)^2, \tag{3.14}$$

where $\boldsymbol{U} = \boldsymbol{\Phi\Psi}$ and λ_j is the j^{th} eigenvalue of $\boldsymbol{L}$. From (3.11) and (3.14):

$$\sum_{(l,m)\in E} \left(\boldsymbol{\Phi}(i,l) - \boldsymbol{\Phi}(i,m)\right)^2 = \sum_j \lambda_j \boldsymbol{U}(i,j)^2 \tag{3.15}$$

$$\leq \left(\sum_j \lambda_j\right) \max_{\forall j} \left(\boldsymbol{U}(i,j)^2\right), \tag{3.16}$$

where $\sum_i \lambda_i = Trace(\boldsymbol{L}) = 2|E|$ because the total sum of diagonal entries in $\boldsymbol{L}$ is the twice of the total number of links between pixels. From (3.16), we have

$$\max_{\forall j} |\boldsymbol{U}(i,j)| \geq \sqrt{\frac{\sum_{(l,m)\in E} \left(\boldsymbol{\Phi}(i,l) - \boldsymbol{\Phi}(i,m)\right)^2}{2|E|}}. \tag{3.17}$$

Thus, the lower bound of the mutual coherence is derived:

$$\begin{aligned} \mu(E) &= \max_{\forall(i,j)} |\boldsymbol{U}(i,j)| = \max_{\forall i} \left(\max_{\forall j} |\boldsymbol{U}(i,j)| \right) \\ &\geq \max_{\forall i} \sqrt{\frac{\sum_{(l,m)\in E} \left(\boldsymbol{\Phi}(i,l) - \boldsymbol{\Phi}(i,m)\right)^2}{2|E|}}. \end{aligned} \tag{3.18}$$

Note that both lower bound, μ_{lower}, and upper bound, μ_{upper}, can be computed without constructing the GBT. The upper bound is determined by the maximum size of a disconnected sub-graph in the graph and the lower bound by the edge map and the given Hadamard sensing matrix. To approximate the mutual coherence between the two bases for a given graph, $G(V,E)$, we take the average $\mu_{avg}(E) = \frac{\mu_{lower}(E)+\mu_{upper}(E)}{2}$. Since the mutual coherence is the maximum correlation between two bases, the mutual coherence can be misleading, especially when only a few correlations are large but the others are small. Thus, instead of looking at the maximum correlation, the average correlation provides a better estimate for the CS performance as studied in [27]. Thus, μ_{avg} can be used as an alternative metric instead of the original mutual coherence. The averaged mutual coherence will be used to approximate the rate for CS measurements to find optimized adjacency matrix, which will be covered in the following section.

3.3.2 Iterative GBT construction for CS

To find the best sparsifying basis, $\mathbf{\Psi}$, we iteratively evaluate a series of adjacency matrices using their average mutual coherence, μ_{avg}. We assume 4-neighbor connectivity in 4×4 block, so that there exist 12 horizontal edges and 12 vertical edges. Instead of searching the whole space of 2^{24} possible adjacency matrices, we propose a greedy algorithm to find an optimized adjacency matrix (Algorithm 2). By defining a cost function, the cost for removing each edge can be calculated.

Algorithm 2 Optimization of adjacency matrix, $\boldsymbol{A}$

Given an undirected graph, $G(V, E)$, such that $|V| = 16$ and $|E| = 0$.

Construct $G_0(V, E_0)$ such that $e_{i,j} \in E_0, \ \forall e_{i,j} = \{(v_i, v_j) | v_i == v_j\}$.

$E_{max} = 24$, $E_s = E - E_0$

$C_{min} = \log_2(\frac{\left(\sum_{\forall(i,j)\in E_0}(v_i - v_j)^2\right)\mu^2_{avg}(E_0)}{2Q^2}) + \lambda \cdot m(E_0)$

for $k = 1$ to $E_{max} - |E_0|$ **do**

$e_{min} = \arg\min_{\forall e=(v_i,v_j)\in E_s} |v_i - v_j|$

Remove e_{min} from E_s

$E_k = E_{k-1}$

Add e_{min} to E_k.

$C_k = \log_2(\frac{\left(\sum_{\forall(i,j)\in E_k}(v_i - v_j)^2\right)\mu^2_{avg}(E_k)}{2Q^2}) + \lambda \cdot m(E_k)$.

if $C_k \leq C_{min}$ **then**

$C_{min} = C_k$

else

Remove e_{min} from E_k

end if

end for

return $E_{out} = E_k$

The basic idea of Algorithm 2 is as follows. In the initial graph, there are only links between pixels that have the same value. This would be equivalent to having a very small threshold to determine whether two pixels have an edge between them. At each iteration, the algorithm finds the link with the minimum pixel difference among those that have not been examined in previous iterations. This link is added if the updated cost is smaller than the one in the previous iteration. The algorithm repeats the same procedure until all the links have been searched excluding the links in the initial state. Thus, for 16 nodes on the 4×4 grid with 4-neighbor connectivity, the maximum number of iterations is 24 if no link exists at the initial graph. Algorithm 2 optimizes the graph in order to find a good set of links providing better performance in terms of the number of bits for encoding and the reconstruction quality estimated by the average mutual coherence in Section 3.3. The set of links to be found by the algorithm, E_{out}, can be uniquely represented by adjacency matrix or edge map.

The cost function of a given graph, $G(V, E)$, is defined as:

$$\begin{aligned} \mathrm{C(E)} &= \mathrm{Cost}_{\mathrm{measurement\_rate}} + \lambda \cdot \mathrm{Cost}_{\mathrm{edge\_rate}} \\ &= \log_2 \left(\frac{\left(\sum_{\forall (i,j) \in E} (v_i - v_j)^2 \right) \mu_{avg}^2(E)}{2Q^2} \right) + \lambda \cdot m(E), \end{aligned} \tag{3.19}$$

The edge rate, $m(E)$, is the number of bits required to code E (equivalently, the corresponding adjacency matrix or edge map), which can be represented using 24 bits with 4-neighbor connectivity, and then compressed further using entropy coding. The scaling factor λ can be applied to control the trade-off between the coefficient rate and edge rate, which is empirically determined in our simulation. $\mathbf{v}$ is a vector representing the input depth map block and thus v_i is the value of

pixel i. a_{ij} is the element in the adjacency matrix that correspond to the link between pixels i and j. Thus, $\sum_{\forall(i,j)\in E}(v_i - v_j)^2$ is the sum of squared differences between connected pixels, which provides an estimate the cost transmitting the GBT coefficients and approximates their sparseness. Then, we divide this with the square of quantization step size, Q. Note that the cost function is identical to the one proposed in [38] except for $\mu^2_{avg}(E)$.

For an intermediate graph, $G(V, E_k)$ in Algorithm 2, the cost function, $C(E_k)$ is defined as:

$$\mathrm{C(E_k)} \;=\; \log_2\left(\frac{\left(\sum_{\forall(i,j)\in E_k}(v_i - v_j)^2\right)\mu^2_{avg}(E_k)}{2Q^2}\right) + \lambda \cdot m(E_k) \qquad (3.20)$$

The average mutual coherence, $\mu_{avg}(E)$, is needed to estimate the rate required to transmit the measurements, since the number of measurements is proportional to $K\mu^2 \log N$ as studied in [8]. Note that we can ignore the $\log N$ term because total number of pixels in each block does not change during the algorithm.

3.4 Simulation Results

The simulation is based on H.264/AVC reference software JM17.1. For simplicity, only a 4×4 blocksize is used in our simulation for the transform, although this can be easily extended to other block sizes. As test sequences in our simulation, we use only intra-frames of depth map sequences Ballet, Newspaper, and Mobile. With RD optimization with respect to H.264/AVC, GBT and CS-GBT, the encoder chooses the best mode and transmits extra bits to signal the transform mode for each block. For CS-GBT, the encoder encodes 4 Hadamard measurements corresponding to the

Methods	RD Optimization	Overhead
H.264/AVC	DCT	None
GBT	DCT + GBT	Optimized $\boldsymbol{A}$ [38]
CS-GBT-1	DCT + GBT + CS	Optimized $\boldsymbol{A}$ [38]
CS-GBT-2	DCT + GBT + CS	Optimized $\boldsymbol{A}$ for CS

Table 3.1: For comparison, four different methods are considered. First, with H.264/AVC, only 4×4 DCT is enabled, thus no overhead exists. Second, DCT and GBT are enabled in RD optimization and, for GBT, the adjacency matrix, $\boldsymbol{A}$, is optimized as in [38] with overhead transmitted to the decoder for each block that uses GBT. Third, with the same $\boldsymbol{A}$, an additional transform mode, CS, is considered. Lastly, DCT, GBT, and CS are considered in RD optimization as before but the adjacency matrix, $\boldsymbol{A}$, is optimized as discussed in Section 3.3.2.

4 lowest frequency bases then scalar quantization is applied to the measurements associated with a QP value. In this work, we do not consider an adaptive choice of measurements because the encoder is required to signal the indices of the chosen measurements to the decoder, which increases the overhead bits. It may be possible to optimize the choice of Hadamard projections while taking into account the overhead, but this is left for future work. To reconstruct the depth map from the Hadamard measurements, MOSEK C-library [5] is employed to solve the L_1 minimization, which is then integrated into H.264/AVC reference software JM17.1.

For comparison, we construct the GBT matrix using two different greedy algorithms with different cost metric as shown in Table 3.1: i) GBT construction without mutual coherence [38] (GBT and CS-GBT-1) ii) GBT construction with mutual coherence discussed in Section 3.3.2 (CS-GBT-2). The scaling factor λ in (3.19) is empirically chosen as 0.03 which equals to the one in the cost function of [38] because, in our simulation, the change of the scaling factor does not affect much the overall performance. For both cases, the resulting adjacency matrices

are entropy coded and sent to the decoder. The decoder can construct the equivalent GBT matrix from the losslessly-encoded adjacency matrix (equivalently, edge map).

For CS-GBT-1 and CS-GBT-2, one can choose between DCT, GBT, and CS to achieve the best performance. For example, for each block, the RD cost can be calculated for DCT, GBT, and CS, and the best approach with the smallest RD cost can be selected. The overhead indicating the chosen transform is encoded into the bitstream for each block, and the optimized adjacency matrix is provided only for blocks coded using GBT or CS. Similarly, for GBT, DCT and GBT are considered in the RD optimization and the optimized adjacency matrix is transmitted to the decoder only for blocks coded using GBT. We consider QP values of 24, 28, 32, and 36 to encode depth maps. As a reference, we also compare those approaches to H.264/AVC for the depth map compression. The reconstruction quality is evaluated using PSNR calculated by comparing the ground truth video and the synthesized video using the decoded depth maps.

From the RD curves in Fig. 3.2, it is shown that the CS-GBT-2 outperforms H.264/AVC. Also, our proposed approach shows better performance than GBT and CS-GBT-1. This indicates that taking into account explicitly the mutual coherence, as we propose, leads to improvements over simply optimizing the GBT for sparsity (with the metric from [38]). Noticeable PSNR improvements over other methods

Sequence	BD-PSNR (dB)	BD-bitrate (%)
Ballet	0.9 dB	-49.4 %
Newspaper	1.5 dB	-26.8 %
Mobile	9.2 dB	-42.8 %
Average	3.9 dB	-39.7 %

Table 3.2: BD-PSNR/bitrate results of CS-GBT-2 compared to H.264/AVC.

Sequence	BD-PSNR (dB)	BD-bitrate (%)
Ballet	0.3 dB	-7.8 %
Newspaper	0.9 dB	-16.1 %
Mobile	2.4 dB	-9.7 %
Average	1.2 dB	-11.2 %

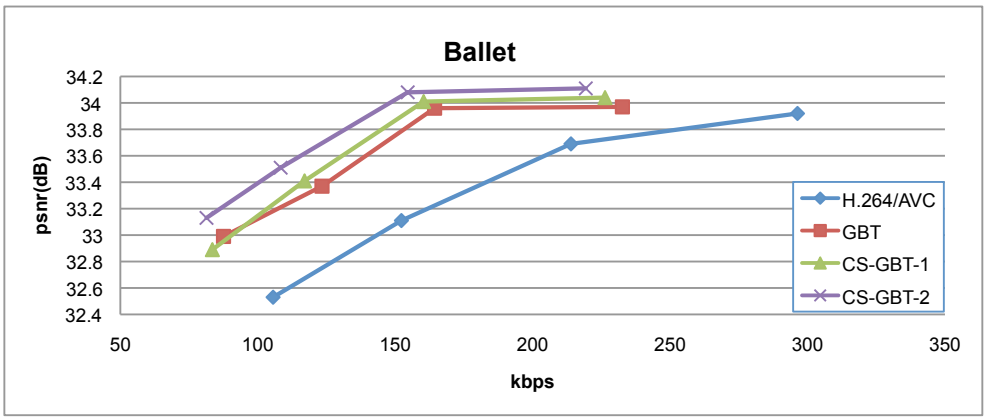
Table 3.3: BD-PSNR/bitrate results of CS-GBT-2 compared to GBT.

are observed because, with our optimized adjacency matrix for CS, more blocks are chosen to be coded using Hadamard measurements. The performance also depends on the number of strong edges in a frame and the level of noise around the edges. Among three sequences in our simulation, the Mobile sequence contains stronger edges along the object boundary with relatively less noise. Thus depth edges are better preserved than those in the other sequences, which leads to the best performance in Fig. 3.4. Also, the perceptual improvement in Ballet sequence is shown in Fig. 3.3. As marked by blue circles, we can notice clear edges reconstructed by our proposed approach. The results for three different sequences are shown in Tables 3.2 and 3.3 in terms of BD-PSNR and BD-bitrate.

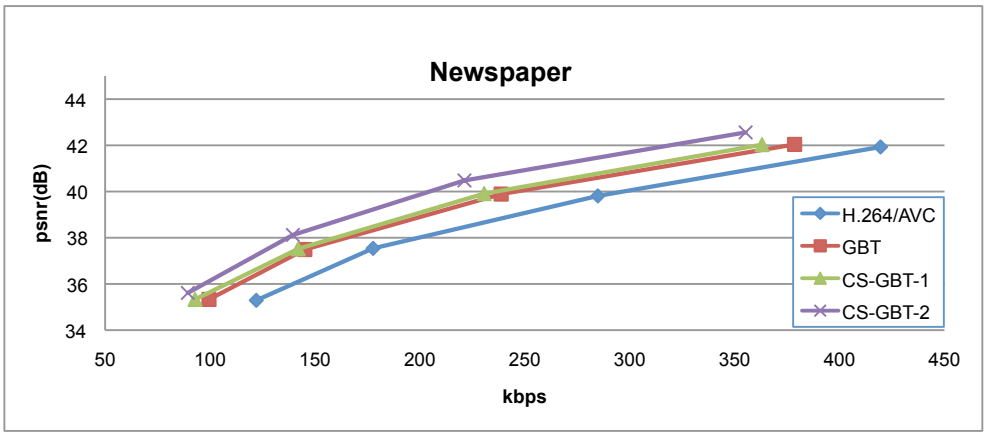
3.5 Conclusion

For depth map compression, we propose a novel CS approach where the adaptive GBT is used as a block-adaptive sparsifying basis. Based on the observation that maximum sparsity does not guarantee optimal performance when using CS, we propose a greedy algorithm that selects for each block a GBT that minimizes a metric that takes into consideration both the edge structure of the block and the

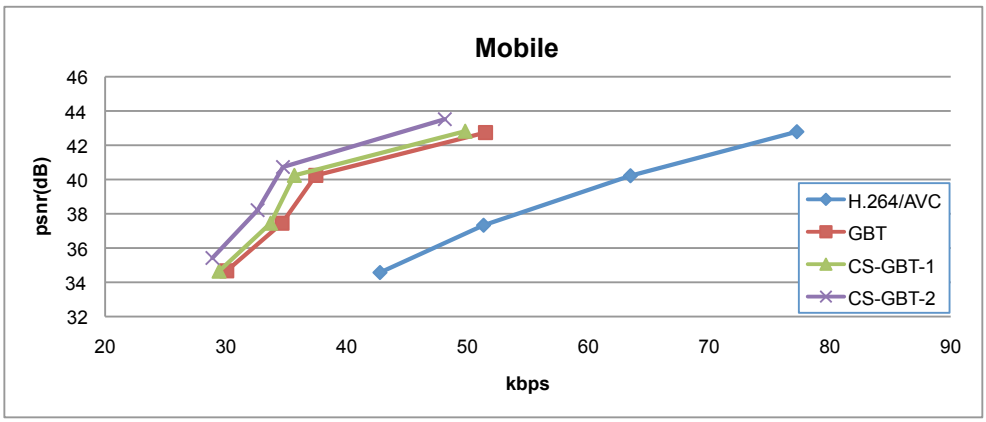
characteristics of the CS measurement matrix, using an estimate of average mutual coherence. As compared to coding using H.264/AVC, the proposed approach applied to intra-frames shows a significant gain for interpolated views.



(a) Ballet

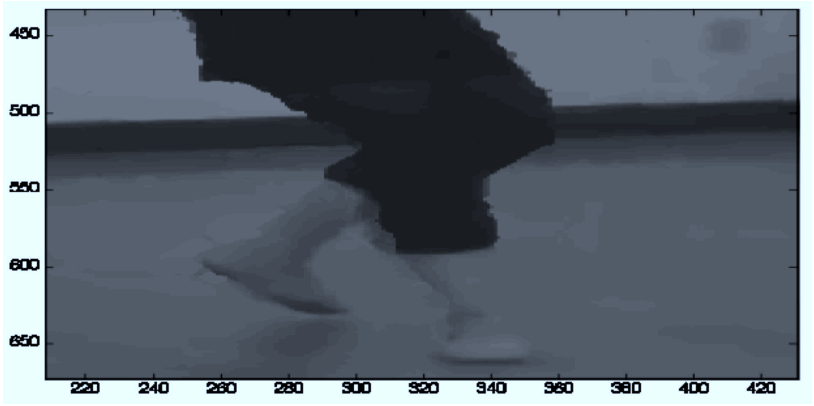


(b) Newspaper

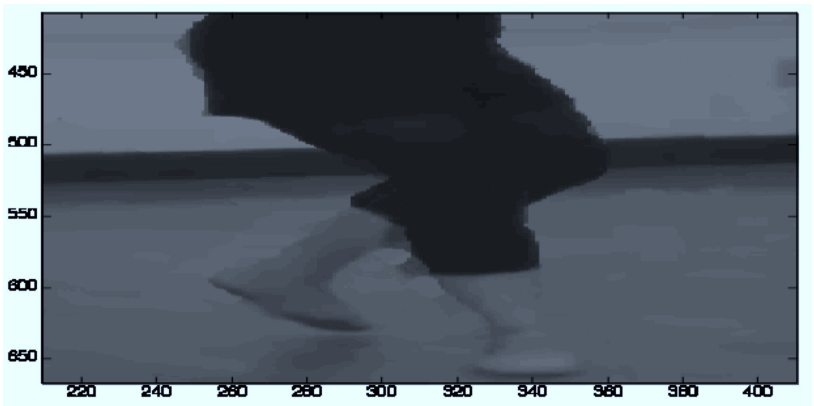


(c) Mobile

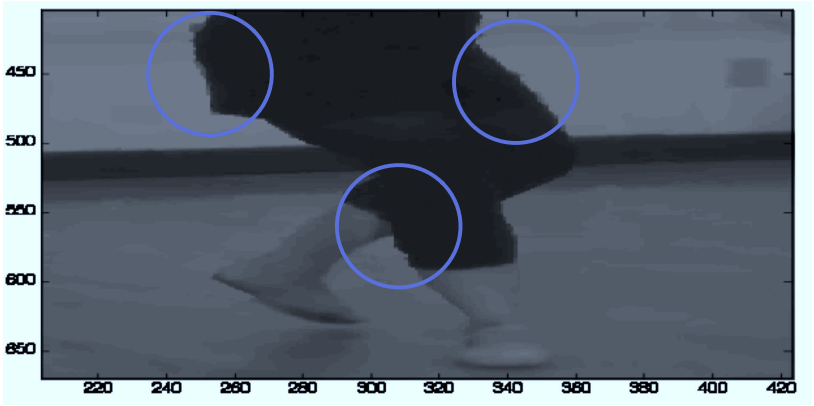
Figure 3.2: RD performance comparison between i) H.264/AVC, ii) GBT [38], iii) CS-GBT-1, and iv) CS-GBT-2 for different sequences: (a) Ballet (b) Newspaper (c) Mobile.



(a) H.264/AVC

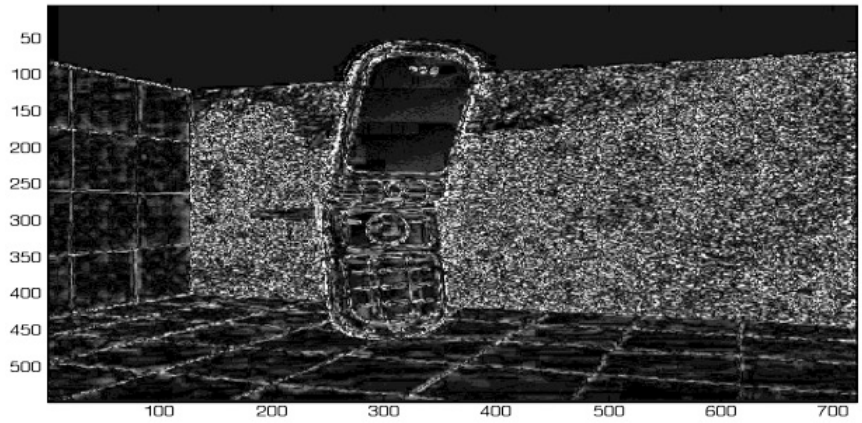


(b) GBT

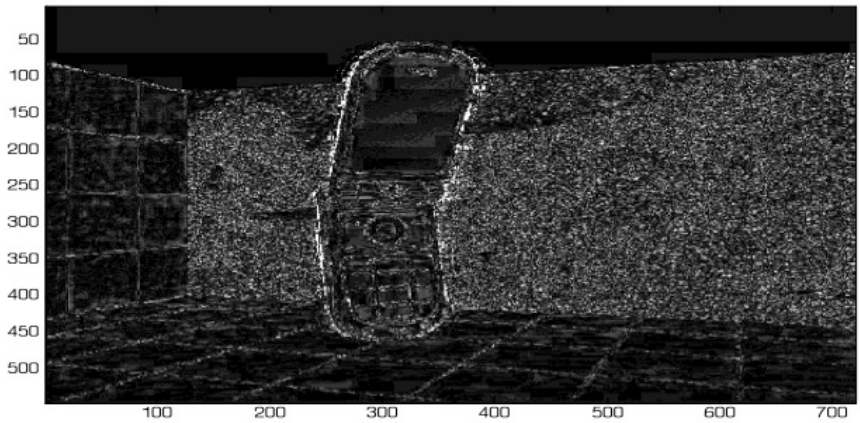


(c) CS-GBT-2

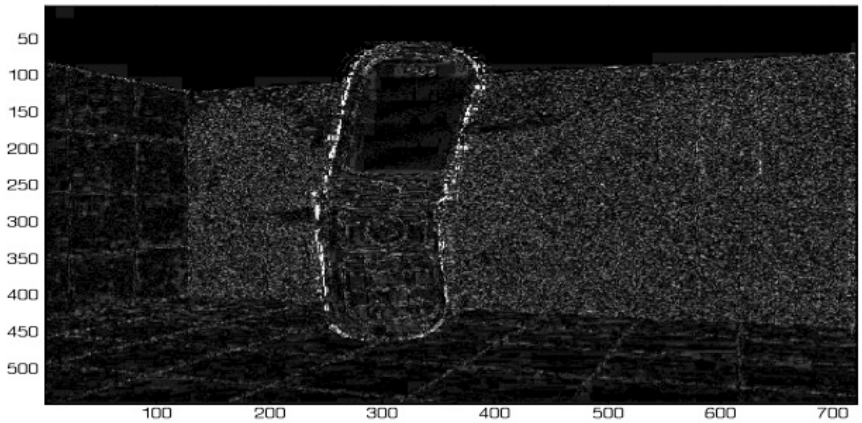
Figure 3.3: Perceptual improvement in Ballet sequence (QP 24): comparison of i) H.264/AVC, ii) GBT, and iii) CS-GBT-2



(a) H.264/AVC



(b) GBT



(c) CS-GBT-2

Figure 3.4: The absolute difference between the synthesized view with and without depth map compression in Mobile sequence (QP 24): comparison of i) H.264/AVC, ii) GBT, and iii) CS-GBT-2

Chapter 4

Hardware-driven Compressive Sampling for Fast Target Localization using Single-chip UWB Radar Sensor

4.1 Introduction

Ultra-wideband (UWB) systems have been utilized for important applications such as radar tracking of objects and monitoring breathing or heartbeats of humans [14, 18]. For example, breathing monitoring can be achieved by localizing the subject's chest movement, which is critical for people who are under severe injury or sedation after surgery. This requires a precise and fast localization of objects with high resolution. Compared to available solutions using video camera techniques, UWB provides benefits of higher spatial depth resolution [49].

In general, UWB radar sensors employ two types of detection schemes: (i) energy detection [42] or (ii) direct sampling [15, 24]. Energy detection achieves low power consumption and has a simple architecture due to the nature of correlator-based detection circuitry. At the expense of higher power consumption, direct sampling enables reconstruction of the reflected waveform in the whole detection

range and therefore provides an opportunity for advanced signal processing to extract additional information [14]. A practical challenge for UWB radar design is to overcome the low SNR from each received pulse due to the UWB emission spectrum mask posed by Federal Communications Commission (FCC) [17].

A recently developed hardware [15, 24] combines the direct sampling approach with a *ranging* technique (we will call the combined technique *sequential sampling* in the rest of this chapter). The ranging technique works by sending multiple pulses then averaging the received pulses in short time intervals (*windows*), each corresponding to a certain roundtrip time of the reflected pulse. Assuming the environment is relatively static, the receiver can localize an object at a specific distance by selecting the window corresponding to that distance and determining if the window contains reflected signal. The averaging within a chosen window provides robustness to noise. It also requires less power consumption, because power is only consumed during the measurement window, which can represent a small percentage of the overall operating time. However, a limitation of this scheme comes from sequential sampling, i.e., candidate object locations have to be probed in sequence, so that the time required to locate an object will be proportional to the number of measurement windows.

In this work we propose a novel technique that can significantly reduce the scanning time, with no increase in overall power consumption. The key observation is that in many situations the number of objects that can be observed is small relative to the number of locations that are probed. This allows us to probe several locations simultaneously, so that each measurement combines reflections at several distances. Processing can then be used extract the actual position information from the combined observations. Our approach is based on applying compressed

sensing (CS) principles with a design that is tightly coupled to the UWB hardware platform.

In the context of radar applications, many researchers have proposed CS-based approaches that exploit the sparse structure of UWB signals [12, 23]. In [2], authors showed that the received signal can be digitized at a rate much lower than the Nyquist rate, without a need for matched filters. But important issues, such as performance in the high noise case and total power consumption are not considered. Similarly, CS was applied to UWB detection applications, but with a mostly theoretical focus [36, 57, 61] or with experiments in a relatively simple environment [77]. Also, a precise CS-UWB positioning system was proposed by exploiting the redundancy of UWB signal captured at multiple receivers to localize a transmitter [84, 85]. While this work achieved low ADC sampling rate, the rate is still higher than what can be achieved with the UWB hardware platform we build upon, and its performance is not as robust in high noise environments. As a CS approach tightly coupled to hardware, the Random-Modulation Pre-Integrator (RMPI) was proposed to achieve low-rate ADC by random modulation in analog domain [86–88] but the random modulation of signals contaminated by powerful noise in analog domain does not provide robust signal recovery. Also, the analog random modulation does not provide the flexibility to accommodate different sampling algorithms.

In this work, we propose a CS technique tightly coupled to the capabilities of recently developed hardware [15, 24] with the goal of achieving robustness to noise and low power consumption while providing reliable localization. To the best of our knowledge, this is the first work that exploits UWB sparsity in the context of a ranging technique for an object localization application. More specifically, there are three main contributions in this work.

First, we formulate the sparse structure of the signal of interest. The UWB signal is sparse if few objects are present because the UWB signal is highly localized in time. Combined with the UWB ranging system, this leads to a special structure where sparse non-zero entries are clustered into a few groups (windows). The number of windows where non-zero entries occur is equal to the number of objects in the region of interest. More details about our representation of UWB signals will be discussed in Section 4.2.2.

Second, we design an efficient measurement system subject to several constraints imposed by the hardware. The constraints include (i) non-negative integer entries in the sensing matrix (ii) constant row-wise sum of entries in the matrix (iii) non-zero entries of each row can exist only at positions with a constant shift, which leads to a unique structure characterized by a Kronecker product. Under these constraints, we construct a sensing matrix by using a low-density parity-check (LDPC) matrix that was recently shown to lead to a good measurement system in [20, 50].

Third, in order to enhance the localization performance, we propose a window-based reweighted L_1 minimization and show that it provides good performance for the abovementioned signal model and measurement system. In our simulations, we compare our proposed method with other existing reconstruction algorithms in terms of several metrics for evaluating localization performance. Our simulation results show that our proposed method can achieve reliable target-localization while using only 40% of the sampling time required by the corresponding sequential scanning scheme, even in a highly-noisy environment.

The rest of this chapter is organized as follows. In Section 4.2, we formulate the UWB ranging system in CS framework and provide how to approximate the total scanning time. Then we describe our proposed approach in Section 4.3 and the result is verified in our simulation in Section 4.4.

4.2 Problem Formulation

4.2.1 UWB Ranging System and Assumptions

We start by describing the hardware design in [15, 24], which serves as the basis for our design. This system can probe the presence of objects in a certain distance from the receiver as illustrated in Fig. 4.1. In this approach, short-time pulses are transmitted periodically. Assume that we would like to determine whether an object is present at a given distance range from the receiver. Given such a range of distances, we know that the reflected pulse will have a certain roundtrip time that will fall into a short time interval (window). Denoting *cycle* the interval between successive transmitted pulses, as shown in Fig. 4.1, we divide a cycle into *windows*, each corresponding to a small distance range.

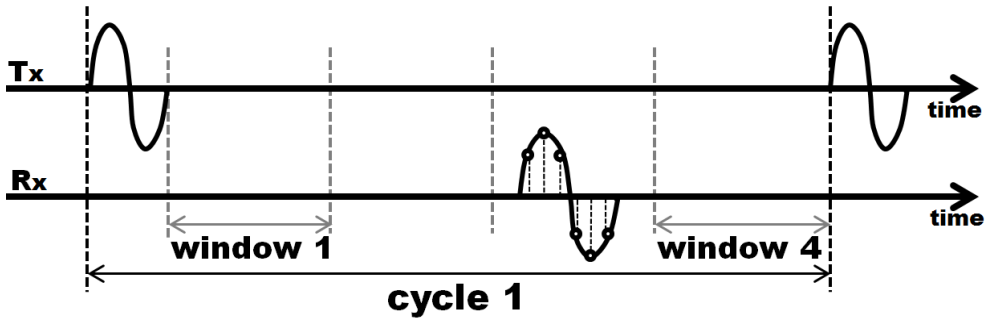


Figure 4.1: Basic motivation for UWB ranging system. After a pulse is transmitted, the observation time for the receiver can be divided into non-overlapped windows. Since the pulses and their reflections are narrow, we assume that the reflection localized in one of the windows. If a range is chosen, the receiver is able to measure a reflected signal during a specific window corresponding to the roundtrip time for that range. Within each cycle, the receiver only consumes power during a window, leading to low overall power consumption.

This hardware design for object localization application is based on two assumptions; (i) the environment and the objects of interest remain stationary within the time that multiple short-time pulses are transmitted, and (ii) the reflected pulses

reside within one of the windows. The first assumption is valid in practice because the hardware operates at picosecond scale for measuring the reflection in each cycle thus scanning more than 100 windows over multiple cycles can finish in the order of nanoseconds [24]. This fine-scale operation in time enables us to apply our proposed approach to practical applications such as breathing monitoring. The second assumption is from the observation that the transmitted pulses and their reflections are narrow compared to the time period of each window.

Under the aforementioned assumptions, the hardware design has several advantages: low power consumption, robustness to noise, and flexibility to accommodate different sampling algorithms. By selecting a specific range, the system cannot observe objects at other distances, because measurements are performed only within the chosen window. But, as a consequence, power consumption is significantly reduced, since no power is consumed during other window intervals, while averaging over multiple cycles increases robustness to noise, e.g., thermal noise from circuits, reflection from objects that are not of interest, etc. Note that noise can be significant in these scenarios. For example, the lower bound of SNR with the hardware design is about -21dB based on the discussion in [41]. Note that the system does not consume extra energy when the sampling switches from one window to another. This is an important fact which allows us to combine measurements over multiple windows as we propose next.

4.2.2 Window-based Sparse UWB Signal Model

Without noise, the sampled signal, $\boldsymbol{x}$, will be sparse because of two main reasons. First, UWB pulses are very narrow in time, so that the received signals are themselves sparse in the time domain, i.e., reflected pulses corresponding to an object

of interest are present in a short time interval. Second, the number of objects of interest is small compared to the number of windows. Thus, $\boldsymbol{x}$ has a special structure such that sparse non-zero entries are clustered within a few windows, with the number of windows where reflections are present being equal to the number of objects in the region of interest. In this work, we mainly focus on the second source of sparsity, leading to window-based sparse signal model.

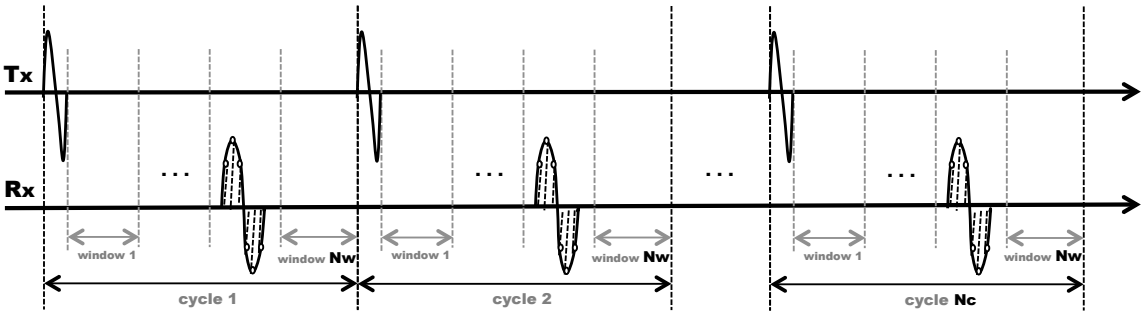


Figure 4.2: Illustration of the UWB sampling system with the parameters:N_C and N_W. Throughout N_C cycles, one pulse is periodically transmitted at the beginning of each cycle which consists of N_W non-overlapped windows. After taking summation of the reflections during N_C cycles in analog domain, N_S samples are collected in each window.

Let N_S be the number of samples in each window, and assume that the UWB-ranging system has N_W non-overlapped windows in each cycle as shown in Fig. 4.2. Assume that each cycle is long enough that it can capture all reflections of interest. For example, this would mean that, in an indoor environment, the cycle would be long enough to receive a reflection from the furthest point in the environment.

To introduce a signal to be reconstructed for the object localization, we first define the signal in noise-free environment then we will extend it to the actual measured signal with noise in Section 4.2.3. Suppose that we observe a noise-free signal $\boldsymbol{x_{i,j}}$ in the i^{th} window during the j^{th} cycle, where $\boldsymbol{x_{i,j}}$ is a vector with N_S samples. Under the assumption that the environment and the objects of interest

are stationary, we can approximate that the signals (i.e., $\boldsymbol{x_{i,j}},\ \forall j \in \{1, 2, \cdots, N_C\}$) observed in the i^{th} window over N_C cycles are identical to $\boldsymbol{x}[i]$:

$$\boldsymbol{x}[i] = \boldsymbol{x_{i,1}} = \boldsymbol{x_{i,2}} = \cdots = \boldsymbol{x_{i,N_W}}\ ,\ \forall i \in \{1, 2, \cdots, N_W\} \tag{4.1}$$

Thus, define a signal, $\boldsymbol{x}$, by concatenating N_W sub-signals, $\boldsymbol{x}[i], i \in 1, ..., N_W$:

$$\boldsymbol{x}^T = [\underbrace{x_1, \ldots, x_{N_S}}_{\boldsymbol{x}[1]}, \ldots, \underbrace{x_{N-N_S+1}, \ldots, x_N}_{\boldsymbol{x}[N_W]}]^T \tag{4.2}$$

Since every $\boldsymbol{x}[i]$ has length N_S, the dimension of $\boldsymbol{x}$ is $N = N_S N_W$. Our window-based signal model is similar to models such as block-sparsity, cluster-sparsity, or multiple measurement vector (MMV) model [3, 28, 70]. Compared to those signal models, our sparse signal, $\boldsymbol{x}$, has a common characteristic such that non-zero entries are grouped into a few (sparse) blocks, but the blocks are not overlapped with each other and their sizes are identical. Note that, to the best of our knowledge, this signal model has not been applied to a realistic UWB hardware with ranging capability.

4.2.3 UWB Measurement System and Matrix Formulation

In this work, we propose hardware-driven compressive sampling (HDCS) as an alternative to the sequential sampling scheme presented in [15]. The sequential sampling scheme scans *the same* window in each of the N_C cycles. This is repeated for every window until all the windows are scanned. In contrast, the HDCS scheme collects information about *multiple* windows over the same N_C cycles by scanning one window per cycle but switching the window to be scanned over the course of N_C cycles.

Before deriving a general formulation, we first consider the sequential sampling scheme. Let N_C be the number of cycles over which the receiver integrates before the ADC is activated. After the integrated analog waveform is sampled by the ADC, the measurements obtained from the i^{th} window, $\boldsymbol{y_i}$, can be represented as a linear combination of $\boldsymbol{x_{i,j}}$ with i.i.d. Gaussian noise, $\boldsymbol{n_j}$:

$$\boldsymbol{y_i} = \sum_{j=1}^{N_C} (\boldsymbol{x_{i,j}} + \boldsymbol{n_{i,j}}) \ , \ \forall i \in \{1, 2, \cdots, N_W\} \tag{4.3}$$

Since we assume a stationary environment where the signal to be reconstructed does not change over time and the sequential sampling scans each window over N_C cycles, we can simplify (4.3) by using (4.1):

$$\boldsymbol{y_i} = N_C \cdot \boldsymbol{x}[i] + \boldsymbol{n_i}, \tag{4.4}$$

where $\boldsymbol{n}_i$, is a summation of random variables following i.i.d. Gaussian distribution: $\boldsymbol{n}_i = \sum_{j=1}^{N_C} n_{i,j}, \ n_{i,j} \sim \mathrm{N}(0, \ \sigma_N^2)$. This process can be interpreted as a *diagonal* sensing matrix with diagonal term N_C by concatenating $\boldsymbol{y_i}$ as follows:

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y_1} \\ \boldsymbol{y_2} \\ \vdots \\ \boldsymbol{y_{N_W}} \end{bmatrix} = \begin{bmatrix} N_C\boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & N_C\boldsymbol{I} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & N_C\boldsymbol{I} \end{bmatrix}_{N \times N} \begin{bmatrix} \boldsymbol{x}[1] \\ \boldsymbol{x}[2] \\ \vdots \\ \boldsymbol{x}[N_W] \end{bmatrix} + \begin{bmatrix} \boldsymbol{n_1} \\ \boldsymbol{n_2} \\ \vdots \\ \boldsymbol{n_{N_W}} \end{bmatrix} \tag{4.5}$$

The sequential sampling of (4.5) involves a diagonal matrix (because each window is separately sampled) with equal diagonal terms because all windows are observed the same number of times (N_C). However, with our proposed HDCS scheme, the matrix does not necessarily have zero off-diagonal terms, since one

measurement, $\boldsymbol{y_i}$, can include information from multiple windows. Thus, a general HDCS scheme can be represented as follows:

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y_1} \\ \boldsymbol{y_2} \\ \vdots \\ \boldsymbol{y_{M_W}} \end{bmatrix} = \begin{bmatrix} N_C^{(1,1)}\boldsymbol{I} & N_C^{(1,2)}\boldsymbol{I} & \cdots & N_C^{(1,N_W)}\boldsymbol{I} \\ N_C^{(2,1)}\boldsymbol{I} & N_C^{(2,2)}\boldsymbol{I} & \cdots & N_C^{(2,N_W)}\boldsymbol{I} \\ \vdots & \vdots & \ddots & \vdots \\ N_C^{(M_W,1)}\boldsymbol{I} & N_C^{(M_W,2)}\boldsymbol{I} & \cdots & N_C^{(M_W,N_W)}\boldsymbol{I} \end{bmatrix}_{M\times N} \begin{bmatrix} \boldsymbol{x}[1] \\ \boldsymbol{x}[2] \\ \vdots \\ \boldsymbol{x}[M_W] \end{bmatrix} + \begin{bmatrix} \boldsymbol{n_1} \\ \boldsymbol{n_2} \\ \vdots \\ \boldsymbol{n_{M_W}} \end{bmatrix}, \tag{4.6}$$

where $\sum_{j=1}^{N_W} N_C^{(i,j)} = N_C,\ \forall i \in \{1, 2, \cdots, M_W\}$ and $M_W < N_W$ (equivalently, $M < N$). $N_C^{(i,j)}$ represents the number of times information from window j is included in measurement i. In general, $M_W < N_W$ because we can reduce the number of measurements by aggregating information from multiple windows. Both sampling schemes can be represented by a sensing matrix, $\boldsymbol{\Phi}$ and a noise vector, $\boldsymbol{n}$:

$$\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{n} \tag{4.7}$$

Since the UWB ranging hardware of interest obtains measurements under the assumption that the whole reflection is always captured in a window, we can formulate $\boldsymbol{\Phi}$ as a matrix containing blocks with dimension N_S-by-N_S, each corresponding to a specific "ranging" window in the sequential sampling or a specific "measurement" window in the HDCS sampling. Note that there exist two kinds of windows. In HDCS, a "measurement" window aggregates information from multiple "ranging" windows. Also, $\boldsymbol{\Phi}$ contains non-negative integer entries indicating the number of cycles to be integrated in order to obtain measurements. More specifically, $\Phi(i, j)$ indicates the number of cycles over which $x(j)$ is integrated in order to obtain $y(i)$.

Thus, we can easily compute the total scanning time by taking summation of all the entries in $\mathbf{\Phi}$: $\sum_{\forall i,j} \mathbf{\Phi}(i,j)$.

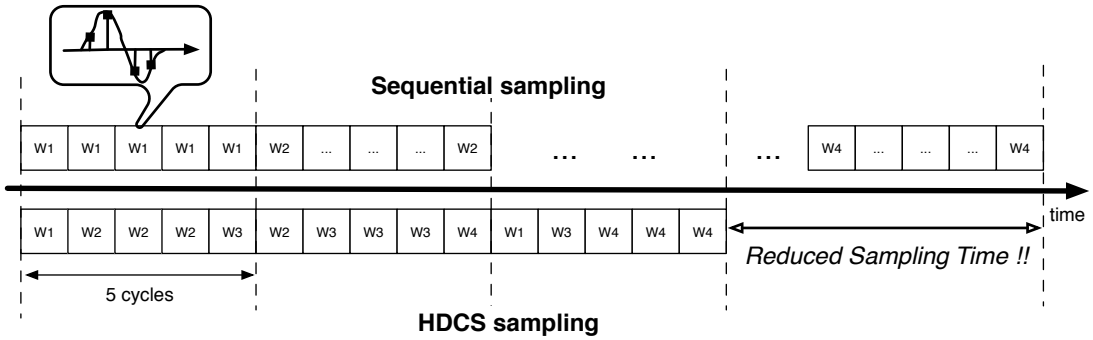


Figure 4.3: Advantage of HDCS scheme. This shows comparison of two different samping schemes formulated as two matrices in (4.8). While Sequential sampling scans a single window during 5 cycles, HDCS collects information from multiple windows during 5 cycles, which could lead to power savings.

The sequential sampling scheme scans a window over N_C cycles until all windows have been scanned. Thus, this process can be interpreted as a *diagonal* sensing matrix with the N_C on its diagonal as in (4.5) whose total scanning time is $N_C N (= N_C N_S N_W)$. On the contrary, the HDCS scheme collects information about *multiple* windows from a measurement by scanning a certain combinations of windows during N_C cycles. Note that the intuition of HDSC is similar to that in Chapter 2, where energy-efficient data gathering can be achieved by collecting measurements from multiple clusters in a wireless sensor network. Thus, the HDCS can expedite the scanning procedure by reducing the number of measurements as illustrated in Fig. 4.3.

For better understanding, we provide an example illustrated in Fig. 4.3, assuming that we obtain a measurement from 4 windows consisting of 4 samples

throughout 5 cycles: $N_S = 4$, $N_W = 4$, and $N_C = 5$. This leads to two sensing matrices, $\mathbf{\Phi_1}$ and $\mathbf{\Phi_2}$ with sequential sampling and HDCS sampling, respectively:

$$\mathbf{\Phi_1} = \begin{bmatrix} 5\boldsymbol{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 5\boldsymbol{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 5\boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & 5\boldsymbol{I} \end{bmatrix}_{16\times 16}, \ \mathbf{\Phi_2} = \begin{bmatrix} 1\boldsymbol{I} & 3\boldsymbol{I} & 1\boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & 1\boldsymbol{I} & 3\boldsymbol{I} & 1\boldsymbol{I} \\ 1\boldsymbol{I} & \mathbf{0} & 1\boldsymbol{I} & 3\boldsymbol{I} \end{bmatrix}_{12\times 16} \tag{4.8}$$

Both $\mathbf{\Phi_1}$ and $\mathbf{\Phi_2}$ contain 4-by-4 identity matrices because we consider 4 samples in each window. As discussed earlier, the sequential sampling scheme generates a diagonal sensing matrix, $\mathbf{\Phi_1}$, with 5 ($=N_C$) on its diagonal. However, with HDCS scheme, $\mathbf{\Phi_2}$ has non-zero off-diagonal terms. If $\mathbf{\Phi_2}$ can give us the same level of reconstruction as $\mathbf{\Phi_1}$, we can achieve 0.75 scanning time reduction because, with HDCS, more windows are measured during 5 cycles for each measurement as shown in Fig. 4.3. The decrease of scanning time leads to the same amount of total power reduction.

Now, the challenge is how to design a *good* sensing matrix satisfying the constraints imposed by the hardware. In other words, arbitrary combinations of windows in the HDCS are not guaranteed to provide a reliable localization performance because CS reconstruction performance depends on the coherence of the sensing matrix. For example, there can be a better combination of the windows than the $[1, 3, 1]$ combination in Φ_2 in (4.8). Thus, we propose to optimize a sensing matrix for the HDCS in order to provide both faster and more reliable localization of the objects. The optimization for better performance will be discussed in more detail next.

4.3 Proposed Approach

4.3.1 LDPC Measurement System

In order to reduce the scanning time, the challenge is how to design a measurement mechanism that can achieve successful reconstruction with fewer measurements. With traditional CS, the random matrices, such as Gaussian random matrix and (uniform randomly) down-sampled Fourier matrix, have been exploited as sensing matrices because they satisfy the restricted isometry property (RIP) with high probability [23].

However, combined with UWB ranging system [15], these popular sensing matrices are no longer appropriate due to additional constraints: i) all the entries of the matrix should be non-negative integers because the entries indicate the number of cycles. This condition rules out popular sensing matrices such as random or Fourier matrices that have real entries. ii) the sum of entries in each row is fixed as a constant number of cycles, N_C. Thus, the scanning time is directly proportional to the number of rows in the sensing matrix. iii) non-zero entries of each row can exist only at the positions with constant shift of N_S. Thus, sensing matrix, $\boldsymbol{\Phi}$, can be formulated as a Kronecker product of the identity matrix with a matrix containing coefficients at the corresponding positions, $\boldsymbol{A}$ (generating sensing matrix):

$$\boldsymbol{\Phi}_{M\times N} = \begin{bmatrix} a_{(1,1)}\boldsymbol{I} & \dots & a_{(1,N_W)}\boldsymbol{I} \\ \vdots & \ddots & \vdots \\ a_{(M_W,1)}\boldsymbol{I} & \dots & a_{(M_W,N_W)}\boldsymbol{I} \end{bmatrix} = \boldsymbol{A}_{M_W\times N_W} \otimes \boldsymbol{I}_{N_S} \tag{4.9}$$

For example, the matrices in (4.8) can be represented following (4.9) as:

$$\boldsymbol{\Phi_1} = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}_{4\times4} \otimes \boldsymbol{I}_{4\times4} \tag{4.10}$$

$$\boldsymbol{\Phi_2} = \begin{bmatrix} 1 & 3 & 1 & 0 \\ 0 & 1 & 3 & 1 \\ 1 & 0 & 1 & 3 \end{bmatrix}_{3\times4} \otimes \boldsymbol{I}_{4\times4} \tag{4.11}$$

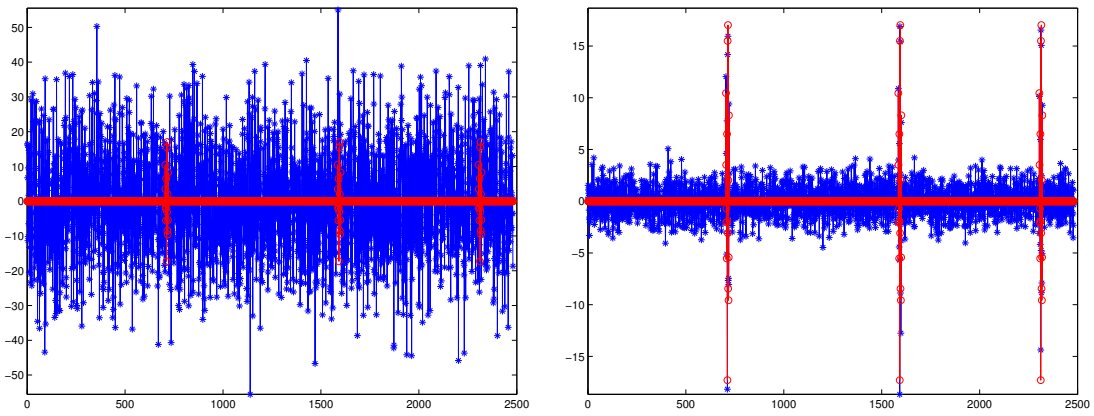
To satisfy these constraints, we propose to adopt low-density parity-check (LDPC) measurement system recently studied in [20,50]. In [20], the authors provide strong theoretical results showing that parity-check matrices corresponding to good channel codes can be used as provably good measurement matrices using basis pursuit reconstruction. In [50], the authors show that LDPC matrices significantly outperform other current CS matrices. Thus, by using an LDPC matrix as the generating sensing matrix, $\boldsymbol{A}$, we can construct a good sensing matrix, $\boldsymbol{\Phi}$, using (4.9), because the coherence of sensing matrix, $\boldsymbol{\Phi}$, is the same as that of the generating sensing matrix, $\boldsymbol{A}$. This can be easily shown as

$$\boldsymbol{U_\Phi} = \boldsymbol{\Phi^T\Phi} = (\boldsymbol{A}\otimes\boldsymbol{I})^T(\boldsymbol{A}\otimes\boldsymbol{I}) = \boldsymbol{A^TA}\otimes\boldsymbol{I} = \boldsymbol{U_A}\otimes\boldsymbol{I}$$

Thus, if $\boldsymbol{A}$ is a good CS measurement matrix, then $\boldsymbol{\Phi}$ in (4.9) is also a good sensing matrix. Also, since the LDPC matrices, $\boldsymbol{A}$, have the same number of 1's in each row, the resulting measurement matrices, $\boldsymbol{\Phi} = \boldsymbol{A}\otimes\boldsymbol{I}$, also satisfy the second condition.

4.3.2 Window-based Reweighted L_1 Minimization

As discussed earlier, our goal is to localize the objects in space. Equivalently, this means that we want to identify the data support (DS) which contains the non-zero entries, since this data support directly corresponds to the locations of objects in space. To identify data support with the sequential sampling scheme in [15], we first reconstruct a signal, $\hat{\boldsymbol{x}}$, by dividing integrated measurements by the number of cycles, N_C. Then, thresholding is applied with an empirically chosen threshold in order to determine data support. As shown in Fig. 4.4, we can achieve higher SNR by increasing the number of cycles, because the noise can be approximated as i.i.d. Gaussian noise as in [30, 37]. However, a larger number of cycles results in longer acquisition time (or higher power consumption) and lower temporal resolution if objects are moving.



(a) Without averaging (b) After 500 iterations

Figure 4.4: Effect of averaging in the sequential sampling scheme [15]: (a) The red plot indicates the original signal (ground truth) and the blue indicates measured signal including the ground truth plus strong noise, SNR=−21.5dB (b) Result of sequential sampling after averaging over 500 cycles.

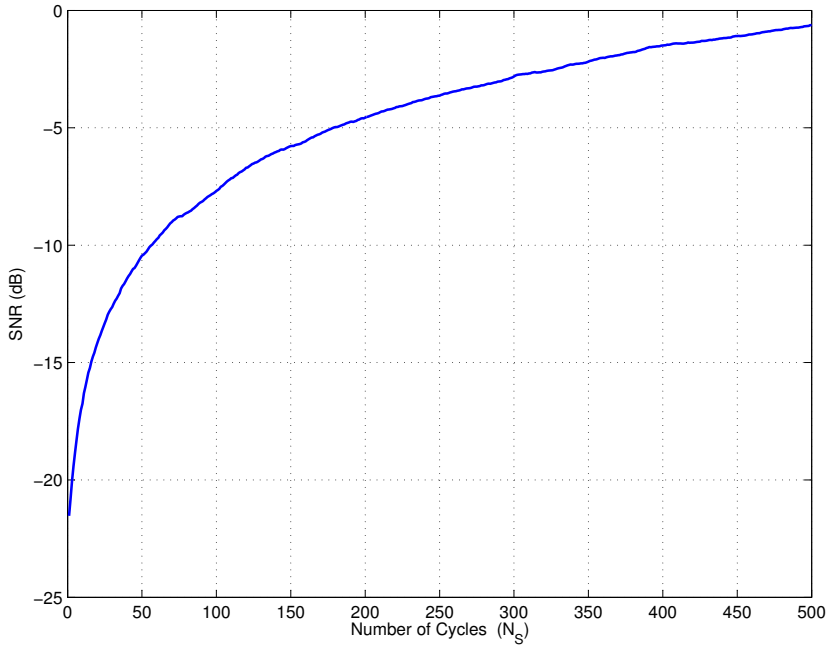


Figure 4.5: Effect of averaging in a sequential sampling scheme [15]. SNR increases as more samples are averaged (i.e.,the number of cycles increases) with high noise level ($\sigma_N = 30$) and system parameters ($N_S = 16$, $N_W = 155$). Note that even after 500 averaging operations, SNR is -0.63dB in this example.

With the HDCS scheme discussed in Section 4.3.1, we propose a two-phase localization process comprising: (i) *non-linear* signal reconstruction and (ii) thresholding. First, we reconstruct signal, $\hat{\boldsymbol{x}}$, from M measurements then we identify the data support by a simple thresholding. For signal reconstruction, we solve a non-linear optimization problem to find a solution to the under-determined system. For thresholding, a window is chosen as one of the possible data supports if the energy of the window is greater than a small value, $\|\hat{\boldsymbol{x}}[k]\|_2 > 0.001$. Note that the threshold is fixed throughout this work, and is not changed according to different noise level or different M.

For successful reconstruction of the signal, several previous works show promising results (refer to [70] for details). In [28] L_2/L_1 minimization was based on a

block-sparsity model. This approach seeks to minimize the sum of the L_2 norm of the signal over several windows subject to the data-fitting constraint:

$$\min \sum_{i=1}^{N_W} \|\boldsymbol{x}[i]\|_2 \;\; s.t. \;\; \|\boldsymbol{\Phi x} - \boldsymbol{y}\|_2 \leq \delta.$$

Theoretical results in [28] show that signals with block-sparsity can be successfully reconstructed with a sensing matrix satisfying block RIP. The signal model in [28] can also be applied to our problem because the reflection signal has a few non-zero entries which are clustered within a few windows. However, noisy measurements or integer sensing matrices are not considered in [28]. Another approach is iterative reweighted L_1 minimization (RL_1) [9, 71]. For the i^{th} iteration, this approach minimizes the L_1 norm of weighted sum of intermediate x_i, subject to data-fitting constraint:

$$\min \| \sum_{j=1}^{N} W_i(j) x_i(j) \|_1 \;\; s.t. \;\; \|\boldsymbol{\Phi x} - \boldsymbol{y}\|_2 \leq \delta.$$

The weight at $(i+1)^{th}$ iteration, $\boldsymbol{W}_{i+1}$, is computed as the inverse of the absolute value of x_i at previous iteration:

$$W_{i+1}(j) = \frac{1}{|x_i(j)| + \epsilon} \;\; , \; j \in \{1, \ldots, N\}$$

Here, ϵ is a small regularization term that prevents an infinite weight term from occurring when x_i is zero. The weight increases as the intermediate result becomes smaller, thus this leads to a solution closer to that of L_0 minimization because large values of x_i contribute to the metric as much as smaller values. However, since the weight update is an entry-wise operation, the windows of interest are not successfully identified when noise levels are high.

This intuition encouraged us to use an iterative window-based reweighted L_1 minimization (WRL_1). The algorithm uses the reweighting technique that has been used in iterative reweighted L_1 minimization with the only difference being that the weight is computed by window-wise operation; the weights for the entries belonging to the k^{th} window are computed as the L_1 norm of the partial intermediate signal, $\boldsymbol{x}[k]$, within that window. The weight vector of the k^{th} window at the i^{th} iteration, $\boldsymbol{W}_i[k]$, is

$$\boldsymbol{W}_i[k] = \mathbf{1}\frac{1}{\|\boldsymbol{x}_{i-1}[k]\|_1 + \epsilon} \ , \ k \in \{1, \ldots, N_W\} \tag{4.12}$$

Here $\mathbf{1}$ is a vector of dimension N_S with all entries equal to 1. The weights for the entries in the same window are updated with the same value. This window-based updating scheme was proposed as the adaptive group Lasso algorithm in [91] or reweighted M-Basis Pursuit in [70], but, to the best of our knowledge, we are the first to apply this reconstruction technique to UWB signal reconstruction.

4.4 Simulation Results

In our simulations, we consider realistic parameters such that 155 windows containing 16 samples in each window ($N_S = 16$, $N_W = 155$), which are similar to those used in the design of UWB radar hardware that we take as a starting point [15, 24] (16 samples in each of 128 windows). In [15, 24] it was shown that the entire detectable range is 15m and the range can be divided to a specific number of windows depending on the requirements of applications. With 155 windows in UWB radar hardware, the maximum spatial depth resolution is 9.6cm, which is desirable for object localization applications in an indoor environment.

Also we assume that each measurement is obtained from 500 cycles: $N_C = 500$. In the simulation, the goal is to localize three objects in the region of interest, which

is the same as finding three windows in signal, $\boldsymbol{x}$. We generated a data set of 80 realizations and each data contains three windows with non-zero entries indicating three objects in space; those windows are chosen randomly with a uniform distribution and the values of non-zero entries are generated by Gaussian distribution.

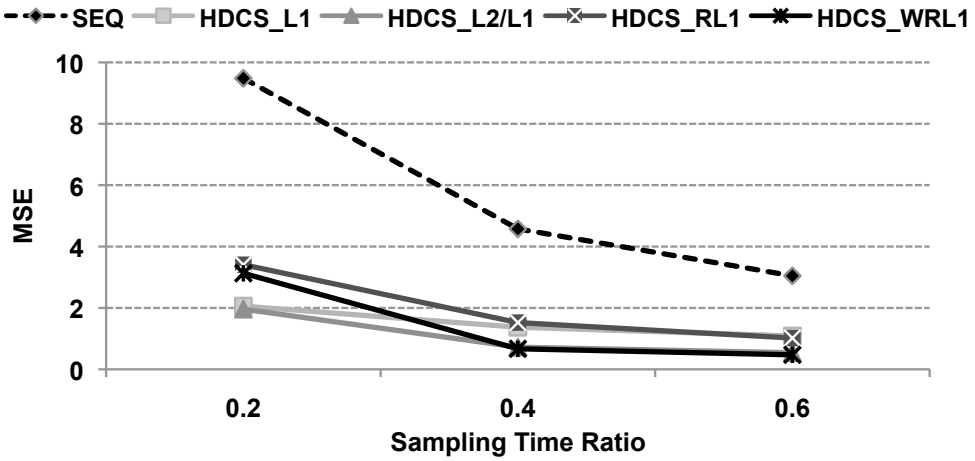


Figure 4.6: Cost ratio vs. MSE: For CS sampling schemes, cost is the total sampling time to collect M measurements. Since we fix the number of cycles, N_C, for every measurement as 500 in the simulation, the cost ratio is a ratio of the number of measurements to the dimension of signal, M/N. But, for sequential sampling scheme (noted 'SEQ' in the figure), we take N measurements with reduced N_C.

For the measurement system, we adopt an LDPC matrix as discussed in Section 4.3.1. For simulation, we first construct the generating sensing matrix, $\boldsymbol{A}$, using LDPC matrices with different number of rows, M_w, by changing the number of 1's in each column from 1 to 3 with that in each row fixed as 5. Then, the measurement matrix, $\boldsymbol{\Phi}$, is constructed by $\boldsymbol{\Phi} = 100\boldsymbol{A} \otimes \boldsymbol{I}_{N_S}$ as in (4.9), where the constant multiplier is $N_C/5 = 100$.

Also, we consider noisy measurements with three different noise levels, $\sigma_N \in \{10, 20, 30\}$, which generates very low SNR (approximately -16.5dB, -22.6dB and -26.1dB respectively on average over our data set). For object localization, we

first reconstruct $\hat{\boldsymbol{x}}$ using L_1 minimization techniques then identify the data support by examining L_2 norm of signals within each window, $\hat{\boldsymbol{x}}[k]$, as discussed in Section 4.3.2. For reconstruction, we compare window-based reweighted L_1 minimization ($HDCS_{WRL_1}$), with three other algorithms: traditional L_1 minimization ($HDCS_{L_1}$), L_2/L_1 minimization ($HDCS_{L_2/L_1}$), and reweighted L_1 minimization ($HDCS_{RL_1}$).

To evaluate performance, we need to measure localization quality as well as scanning time. The scanning time can be easily computed by counting the number of rows of sensing matrix, $\boldsymbol{\Phi}$, because the number of cycles is the same for all approaches ($N_C = 500$). Thus, the cost ratio is a ratio of the number of measurements to the dimension of signal, M/N. To evaluate localization quality, Mean Squared Error (MSE) can be used by measuring the entry-wise difference of values between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$. Fig. 4.6 shows the comparison of performance between our proposed reconstruction technique, $HDCS_{WRL_1}$, and other reconstruction techniques, in terms of MSE and scanning time. Note that for the sequential sampling scheme (noted 'SEQ' in Fig. 4.6), we collect N measurements with the reduced number of cycles, N'_C in order to compare to HDCS sampling at different total scanning time. Since the sequential sampling scheme requires to scan all the windows, we reduce the number of cycles, N_C, to $N'_C = round(N_C \frac{M}{N})$ then construct a measurement matrix as:

$$\boldsymbol{\Phi_{seq}} = N'_C \begin{bmatrix} \boldsymbol{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I} & \mathbf{0} & \mathbf{0} \\ \vdots & \mathbf{0} & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{I} \end{bmatrix}_{N \times N} \tag{4.13}$$

Since we consider three different sampling time ratios, $M/N \in \{0.2, 0.4, 0.6\}$, in Fig 4.6, the resulting $N'_C \in \{100, 200, 300\}$. Fig. 4.6 shows that HDCS schemes with

different reconstruction techniques achieve about five times better reconstruction quality with similar scanning time.

Although MSE is one of the most generic metric for reconstruction evaluation, it can be misleading because smaller MSE does not always guarantee better window identification. For example, perfect identification of data support (DS) can result in large MSE if a large difference exists between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ within the DS.

	HDCS_{L_1}	HDCS_{L_2/L_1}	HDCS_{RL_1}	HDCS_{WRL_1}
avg. mismatch (a)	537.539	132.629	160.562	4.258
avg. no. of DS (b)	26.461	8.124	10.562	3.067
mismatch / DS (a/b)	20.378	16.090	15.065	1.197
max. mismatch	18.371	5.202	5.910	0.989
F-measure	0.206	0.546	0.442	0.943

Table 4.1: Performance evaluation of identification of data support (DS): The acquisition time is reduced to 0.4 compared to sequential sampling scheme, $M/N = 0.4$, with noisy measurements, $\sigma_N = 30$.

Thus, we consider additional metrics to evaluate mismatch of data support (DS). In the comparisons, we rule out the averaging method in sequential sampling scheme because it requires a good threshold which should be adaptive to parameters such as the level of noise and number of cycles, N_C. First, we compute the number of candidates for DS in terms of the number of windows. As discussed earlier, the windows containing non-zero entries are formed as candidates for the estimated DS. Second, for each candidate, we compute minimum distance to any of the correct DS (ground truth). The distance is computed in terms of the number of windows. Then, the average of minimum distances can be interpreted as a metric to evaluate the error in object location in space. Third, we also consider the maximum mismatch of data support by taking the maximum of the minimum distances

	HDCS$_{L_1}$	HDCS$_{L_2/L_1}$	HDCS$_{RL_1}$	HDCS$_{WRL_1}$
avg. mismatch (a)	345.94	42.28	94.48	5.15
avg. no. of DS (b)	25.30	6.30	10.91	3.21
mismatch / DS (a/b)	13.56	6.74	8.45	1.59
max. mismatch	116	93	95	78
F-measure	0.16	0.23	0.27	0.32

Table 4.2: Performance evaluation of identification of data support (DS): The acquisition time is reduced to 0.2 compared to sequential sampling scheme, $M/N = 0.3$, with noisy measurements, $\sigma_N = 20$.

between the candidates and the ground truth to evaluate the performance in the worst case. Lastly, we compute F-measure discussed in [70] as

$$2\frac{|supp(\boldsymbol{x}) \cap supp(\hat{\boldsymbol{x}})|}{|supp(\boldsymbol{x})| + |supp(\hat{\boldsymbol{x}})|}, \tag{4.14}$$

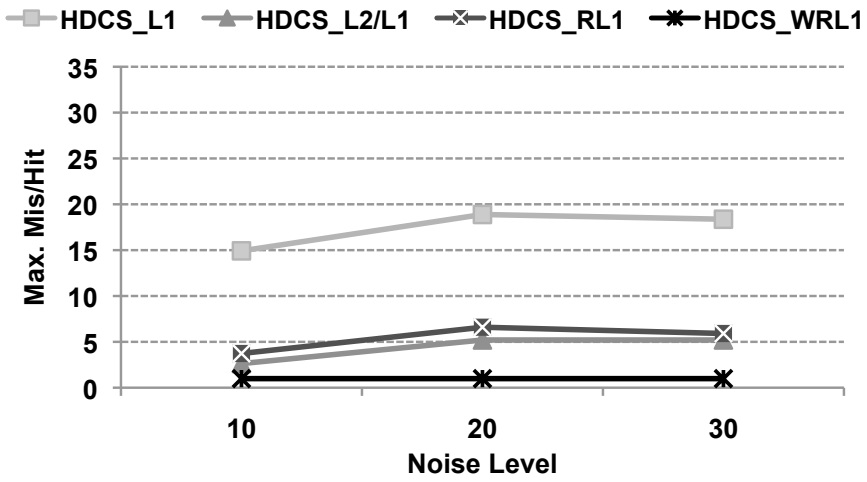
where $supp(\boldsymbol{x}) = \{i \in [1, \ldots, N_W] : \|\boldsymbol{x}[i]\|_2 > 0.001\}$. Note that the F-measure is equal to 1 when the data support of the reconstructed signal coincides exactly with the ground truth.

Table 4.1 shows the performance with respect to the metrics evaluating the ability to identify data support (DS). In the result, $HDCS_{L_1}$ and $HDCS_{L_2/L_1}$ approaches are non-iterative methods while the others are iterative algorithms, where the results after three iterations are presented. As shown in Tables 4.1 and 4.2, our proposed reconstruction technique outperforms all the other methods with respect to all the metrics we consider. Especially, the maximum mismatch of data support shows the performance in the worst case. In Table 4.1, $HDCS_{WRL_1}$ achieves a maximum mismatch of 0.98 windows on average, with only 40% of acquisition time needed for sequential sampling. This means that the average error in location is of the distance represented by a window in the worst case scenario.

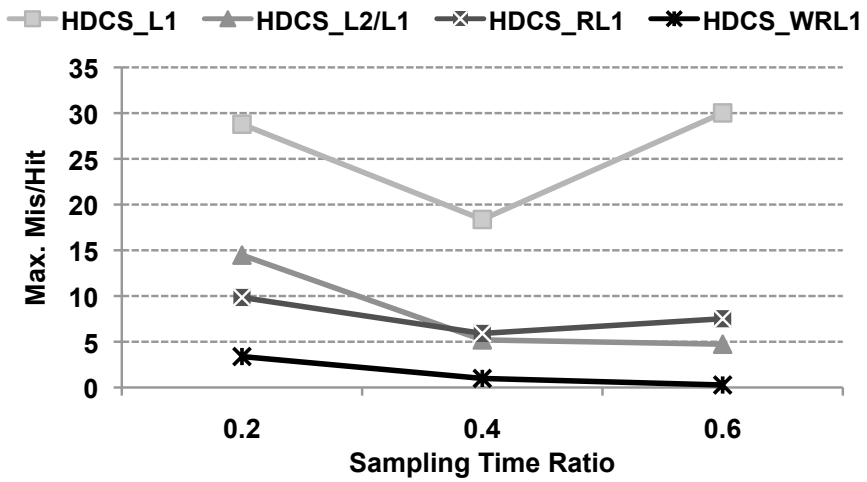
Figs. 4.7 and 4.8 compare performance with respect to abovementioned two metrics. In Fig. 4.7(a), $HDCS_{WRL_1}$ shows very small maximum mismatch at every noise level which is almost equal to 1. This indicates that our identified windows are mismatched at most by one window on average over 80 data. Also, $HDCS_{WRL_1}$ shows very stable performance at different noise levels. Fig. 4.7(b) shows that, in the highest level of noise we tested, $HDCS_{WRL_1}$ shows the best performance and it reaches to almost perfect reconstruction at 0.6 sampling time ratio. Similarly, in Fig. 4.8(a), $HDCS_{WRL_1}$ shows the highest F-measure at every noise level which is very close to 1. Also, it does not drop as the noise level increases as shown in Fig. 4.8(b).

4.5 Conclusion

To design an energy-efficient UWB ranging system, we propose a CS approach combined with a novel hardware architecture. we first formulate UWB signal representation with a special structure such that sparse non-zero entries are clustered into a few groups. Also, we design an efficient measurement system that is constructed by an LDPC matrix, which satisfies several constraints imposed by the hardware. To enhance performance, we propose a window-based reweighted L_1 minimization which outperforms other existing algorithms in our simulation. The result shows that our proposed method can achieve reliable target-localization while requiring only 40% of sampling time of the sequential sampling scheme in highly-noisy environment.

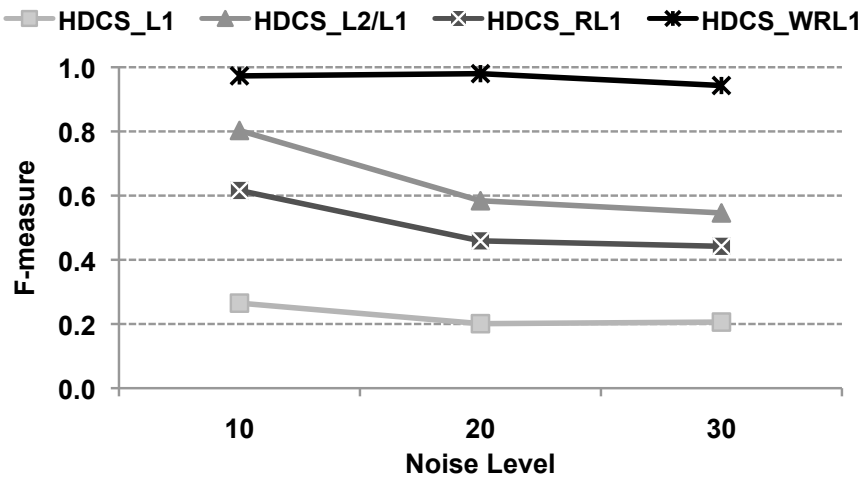


(a) σ_N vs. max. mismatch

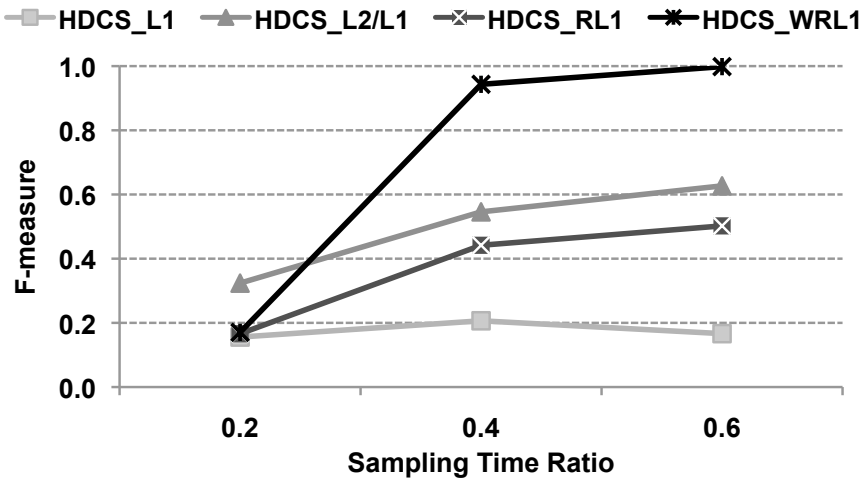


(b) sampling time ratio vs. max. mismatch

Figure 4.7: Performance comparison with respect to maximum mismatch of data support: (a) Fix sampling time ratio as 0.4 and compare performance at different noise levels. (b) Fix noise level as 30 and compare performance at different sampling time ratios.



(a) σ_N vs. F-measure



(b) sampling time ratio vs. F-measure

Figure 4.8: Performance comparison with respect to F-measure: (a) Fix sampling time ratio as 0.4 and compare performance at different noise levels. (b) Fix noise level as 30 and compare performance at different sampling time ratios.

Chapter 5

Conclusions and Future Work

5.1 Conclusion

We propose to optimize Compressed Sensing (CS) design choices under the constraints driven by various applications. Unlike in traditional CS approaches, we consider additional constraints driven by different application requirements such as transport cost in wireless sensor network, total bitrate in depth map compression, or scanning time in UWB ranging system. In these applications, reducing the number of measurements does not always provide a better solution if application specific constraints are not considered.

As we discussed earlier, in the application of WSN, the transport cost to collect the information from the sensors should be considered for the overall performance of data gathering system and this is closely related to the construction of measurement matrix. Thus, we propose to optimize the measurement matrix with respect to the given sparsifying matrix for the joint optimization of the transport cost and the reconstruction accuracy of sensor data for the data gathering. Our proposed approach achieves better performance, as compared to other existing CS techniques related to data gathering of sensor data in wireless sensor network. We also propose

a heuristic approach applied to a practical situation where sensors are randomly deployed over a field of interest.

For depth map compression, the additional cost we need to consider is the bit overhead to code the edge map required for the construction of graph-based transform (GBT). In contrast to the WSN application, we fix the sensing matrix as a Hadamard matrix and we optimize the sparsifying matrix to reduce the number of bits to represent depth map signal while achieving satisfactory reconstruction. We propose a greedy algorithm to optimize the joint optimization of the bit overhead and the reconstruction quality of depth map data and achieve a significant improvement over H.264/AVC.

For fast localization of objects using a UWB ranging system, we propose the design of an efficient measurement system that is constructed using low-density parity-check (LDPC) matrix, designed to satisfy several hardware-related constraints: non-negative integer entries in measurement (sensing) matrix, constant row-wise sum of non-zero entries in the matrix, and a unique structure characterized by Kronecker product. To enhance performance, we propose a window-based reweighted L_1 minimization that outperforms other existing algorithms in our simulation. The result shows that our proposed method can achieve reliable target-localization, while using only 40% of the scanning (sampling) time required by the sequential scanning scheme, even in high noise environments.

5.2 Future Work

The approaches for WSN and depth map compression are related to optimization of compressed sensing with respect to different constraints. Although we extended our work to a more general situation where the sensors are deployed at arbitrary

positions, the heuristic solution needs to be further investigated in order to generalize the approach by optimizing spatially-localized projection with respect to a given GBT. Since GBT is a tool to sparsify any generic signal defined on arbitrary graph, the generalized approach can also be applied to simplified cases such as both WSN and depth map compression.

Furthermore, the generalized framework can be extended to local construction of GBT as shown in [59]. Instead of global construction of GBT, GBT can be constructed with partial information about neighboring nodes in the graph. Thus, investigating how to optimize spatially-localized projection combined with localized GBT will be meaningful, which will lead to localized optimization of the general framework with less complexity.

While UWB radar results are promising, many questions remain. In practice, reconstruction techniques with less complexity will be required, so that they can run in real-time. Also, in order to verify the performance, integration with the hardware will be needed in practice. In addition, a theoretical analysis of the system to evaluate the potential localization performance by this reduced number of measurements or different LDPC matrices will be meaningful.

Reference List

[1] R. Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4):118–121, Jul. 2007.

[2] R. Baraniuk and P. Steeghs. Compressive radar imaging. In *IEEE Radar Conf*, pages 128–133, 2007.

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

[4] D. Baron, M. B. Wakin, M. F. Duarte, S. Sarvotham, and R. G. Baraniuk. Distributed compressed sensing. *arXiv:0901.3403*, Jan. 2009.

[5] MOSEK C-library. www.mosek.com.

[6] E. Candès. Compressive sampling. *Int. Congress of Mathematics*, 3:1433–1452, 2006.

[7] E. Candès and D. Donoho. *Curvelets - a surprisingly effective nonadaptive representation for objects with edges*. Vanderbilt University Press, 1999.

[8] E. Candès and J. Romberg. Sparsity and incoherence in compressive sampling. In *Inverse Problems*, June 2007.

[9] E. J. Candès, M. B. Wakin, and S. Boyd. Enhancing sparsity by reweighted l1 minimization. *Journal of Fourier Analysis and Applications,*, 14(5):877–905, Dec. 2008.

[10] E.J. Candès. Compressive sampling. *International Congress of Mathematicians*, 3:14331452, 2006.

[11] E.J. Candès, J. Romberg, and T. Tao. Robust uncertainity principles : exact signal reconstruction from highly incomplete frequency information. In *IEEE Transactions on Information Theory*, February 2006.

[12] E.J. Candès, J.K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207, 2006.

[13] E.J. Candès and T. Tao. Near-optimal signal recovery from random projections and universal encoding strategies. *IEEE Trans. Inform. Theory*, 52(12):5406 – 5425, 2006.

[14] S. Chang, M. Wolf, and J. W. Burdick. Human detection and tracking via ultra-wideband (UWB) radar. *IEEE International Conference on Robotics and Automation*, May 2010.

[15] T. Chu, J. Roderick, S. Chang, T. Mercer, C. Du, and H. Hashemi. A short-range UWB IR CMOS sensor for human feature detection. *IEEE International Solid-State Circuits Conference Digest of Technical Papers*, pages 294–295, Feb. 2011.

[16] A. Ciancio, S. Pattem, A. Ortega, and B. Krishnamachari. Energy-efficient data representation and routing for wireless sensor networks based on a distributed wavelet compression algorithm. In *IPSN*, April 2006.

[17] Federal Communications Commission. First report and order, revision of part 15 of the commission's rules regarding ultra wideband transmission systems. *ET Docket*, pages 98–153, Feb. 2002.

[18] E. Conti, A. Filippi, and S. Tomasin. On the modulation of ultra wide band pulse radar signal by target vital signs. *International Symposium on Bioelectronics and Bioinformatics*, Dec. 2009.

[19] R. Cristescu, B. Beferull-Lozano, and M. Vetterli. On network correlated data gathering. In *INFOCOM*, March 2004.

[20] A. G. Dimakis, R. Smarandache, and P. O. Vontobel. LDPC codes for compressed sensing. *IEEE Transactions on Information Theory*, 58(5):3093–3114, May 2012.

[21] M. N. Do and M. Vetterli. The contourlet transform: An efficient directional multiresolution image representation. *IEEE Trans. Image Proc.*, 14(12):2091–2106, Dec. 2005.

[22] T.T. Do, T.D. Tran, and Lu Gan. Fast compressive sampling with structurally random matrices. In *ICASSP*, April 2008.

[23] D. L. Donoho. Compressed sensing. In *IEEE Transactions on Information Theory*, April 2006.

[24] C. Du and H. Hashemi. An UWB CMOS impulse radar. *To appear in IEEE Radio Frequency Integrated Circuits Symposium*, Jun, 2013.

[25] J. Duan, L. Zhang, R. Pan, and Y. Sun. An improved video coding scheme for depth map sequences based on compressed sensing. In *International Conference on Multimedia Technology (ICMT)*, Hangzhou, Aug. 2011.

[26] M. F. Duarte, M. B. Wakin, D. Baron, and R. G. Baraniuk. Universal distributed sensing via random projections. In *IPSN*, April 2006.

[27] M. Elad. Optimized projections for compressed sensing. *IEEE Trans. on Signal Processing*, 55(12):5695–5702, 2007.

[28] Y. Eldar and M. Mishali. Robust recovery of signals from a structured union of subspaces. *IEEE Transactions on Information Theory*, 55(11):5302–5316, Nov 2009.

[29] M.A.T. Figueiredo, R.D. Nowak, and S.J. Wright. Gradient projection for sparse reconstruction:application to compressed sensing and other inverse problems. In *IEEE Journal of Selected Topics in Signal Processing*, 2007.

[30] J. Foerster. Channel modeling sub-committee report final. *IEEE 802.15.SG3a Study Group*, Dec. 2002.

[31] J. Fu and B. Zeng. Directional discrete cosine transforms: A theoretical analysis. In *Proc. of IEEE Int. Conf. Acoustics, Speech and Signal Proc., ICASSP 2007*, volume I, pages 1105–1108, Honolulu, HI, USA, Apr. 2007.

[32] L. Gan, T. T. Do, and T. D. Tran. Fast compressive imaging using scrambled block hadamard ensemble. In *EUSIPCO*, sep 2008.

[33] M. Gastpar, P. Dragotti, and M. Vetterli. The distributed Karhunen-Loeve transform. In *MMSP*, December 2002.

[34] D. Hammond, P. Vandergheynst, and R. Gribonval. Wavelets on graphs via spectral graph theory. In *Elsevier: Appplied and Computational Harmonic Analysis*, volume 30, pages 129–150, Apr. 2010.

[35] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient routing protocols for wireless microsensor networks. *In Proc. of Hawaii Intl. Conf. on Sys. Sciences,*, Jan. 2000.

[36] M. Herman and T. Strohmer. High-resolution radar via compressed sensing. *IEEE Transactions on Signal Processing*, 57(6):2275–2284, June 2009.

[37] E. A. Homier and R. A. Scholtz. Hybrid fixed-dwell-time search techniques for rapid acquisition of ultra-wideband signals. *International Workshop on Ultra Wideband System*, 2003.

[38] W.-S. Kim. 3-D video coding system with enhanced rendered view quality. In *Ph.D. dissertation, University of Southern California*, 2011.

[39] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila. Depth map distortion analysis for view rendering and depth coding. In *Proc. of IEEE Int. Conf. Image Proc., ICIP 2009*, Cairo, Egypt, Nov. 2009.

[40] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila. Depth map coding with distortion estimation of rendered view. In *Proc. of IS&T/SPIE Electronic Imaging, VIPC 2010*, San Jose, CA, USA, Jan. 2010.

[41] J. D. Kraus. Antennas. In *McGraw-Hill*, pages 774–790, 1988.

[42] H. Krishnaswamy and H. Hashemi. A 4-channel 4-beam 24-to-26GHz spatio-temporal RAKE radar transceiver in 90nm CMOS for vehicular radar applications. *IEEE International Solid-State Circuits Conference Digest of Technical Papers*, pages 214–215, 2010.

[43] J.B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. In *Proceedings of the American Mathematical Society*, volume 7, pages 48–50, February 1956.

[44] S. Lee, C. Du, H. Hashemi, and A. Ortega. Hardware-driven compressive sampling for fast target localization using single-chip UWB radar sensor. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[45] S. Lee and A. Ortega. Joint optimization of transport cost and reconstruction for spatially- localized compressed sensing in multi-hop sensor networks. *APSIPA*, Dec. 2010.

[46] S. Lee and A. Ortega. Adaptive compressed sensing for depthmap compression using graph-based transform. In *Proceedings of International Conference on Image Processing (ICIP)*, 2012.

[47] S. Lee, S. Pattem, M. Sathiamoorthy, B. Krishnamachari, and A. Ortega. Compressed sensing and routing in sensor networks. In *USC CENG Technical Report*, April 2009.

[48] S. Lee, S. Pattem, M. Sathiamoorthy, B. Krishnamachari, and A. Ortega. Spatially-localized compressed sensing and routing in multi-hop sensor networks. In *GSN*, June 2009.

[49] C. Li, J. Ling, J. Li, and J. Lin. Accurate doppler radar noncontact vital sign detection using the relax algorithm. *IEEE Transactions on Instrumentation and Measurement*, 59(3):687–695, 2010.

[50] W. Lu, K. Kpalma, and J. Ronsin. Sparse binary matrices of LDPC codes for compressed sensing. *Data Compression Conference (DCC)*, 2012.

[51] G. Lugosi. Concentration-of-measure inequalities. In *Lecture notes*, February 2006.

[52] C. Luo, F. Wu, J. Sun, and C. W. Chen. Compressive data gathering for large-scale wireless sensor networks. *in Proc. of ACM Mobicom09*, page 145156, Sep. 2009.

[53] M. Lustig, D. Donoho, and J. M. Pauly. Sparse MRI: The application of compressed sensing for rapid mr imaging. In *MRM*, December 2007.

[54] M. Maitre and M. N. Do. Depth and depth-color coding using shape-adaptive wavelets. *Visual Communication and Image Representation*, Special Issue on Multicamera Imaging:513–522, Jul. 2010.

[55] T. Michaeli and Y.C. Eldar. Xampling at the rate of innovation. *IEEE Transactions on Signal Processing*, 60(3), Mar. 2012.

[56] Y. Morvan, D. Farin, and P. H. N. de With. Platelet-based coding of depth maps for the transmission of multiview images. In *Proc. of SPIE: Stereoscopic Displays and Applications (2006)*, volume 6055, San Jose, CA, USA, Jan. 2006.

[57] F.M. Naini, R. Gribonval, L. Jacques, and P. Vandergheynst. Compressive sampling of pulse trains: Spread the spectrum! *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009.

[58] S. K. Narang, A. Gadde, and A. Ortega. Signal processing techniques for interpolation in graph structured data. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[59] S. K. Narang and A. Ortega. Local two-channel critically sampled filter-banks on graphs. In *Proc. of IEEE Int. Conf. Image Proc., ICIP 2010*, Hong Kong, Sep. 2010.

[60] N. Nguyen, D. L. Jones, and S. Krishnamurthy. Netcompress: Coupling network coding and compressed sensing for efficient data communication in wsns. *in IEEE Workshop on SiPS*, page 356361, Oct. 2010.

[61] JL Paredes, GR Arce, and Z. Wang. Ultra-wideband compressed sensing: Channel estimation. *IEEE Journal of Selected Topics in Signal Processing*, 1(3):383–395, 2007.

[62] S. Pattem, B. Krishnamachari, and R. Govindan. The impact of spatial correlation on routing with compression in wireless sensor networks. In *IPSN*, April 2004.

[63] S. Pattem, G. Shen, Y. Chen, B. Krishnamachari, and A. Ortega. Senzip: An architecture for distributed en-route compression in wireless sensor networks. In *ESSA*, April 2009.

[64] K. Pearson. Mathematical contributions to the theory of evolution. iii. regression, heredity and panmixia. In *Philos. Trans. Royal Soc. London Ser.*, volume 187, pages 253–318, 1896.

[65] E. Le Pennec and S. Mallat. Sparse geometric image representations with bandelets. *IEEE Trans. Image Proc.*, 14(4):423–438, Apr. 2005.

[66] W. Philips. Comparison of techniques for intra-frame coding of arbitrarily shaped video object boundary blocks. *IEEE Trans. Circuits Syst. Video Technol.*, 9(7):1009–1012, Oct. 1999.

[67] D. D.-Y. Po and M. N. Do. The contourlet transform: An efficient directional multiresolution image representation. *IEEE Trans. Image Proc.*, 15(6):1610–1620, Jun. 2006.

[68] G. Quer, R. Masierto, D. Munaretto, M. Rossi, J. Widmer, and M. Zorzi. On the interplay between routing and signal representation for compressive sensing in wireless sensor network. In *ITA*, February 2009.

[69] M. Rabbat, J. Haupt, A. Singh, and R. Nowak. Decentralized compression and predistribution via randomized gossiping. In *IPSN*, April 2006.

[70] A. Rakotomamonjy. Surveying and comparing simultaneous sparse approximation (or group-lasso) algorithms. *Signal Processing*, 91:1505–1526, 2011.

[71] R.Saab, R.Chartrand, and O. Yilmaz. Stable sparse approximations via nonconvex optimization. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.

[72] M. Rudelson and R. Vershynin. Sparse reconstruction by convex relaxation: Fourier and gaussian measurements. *Annual Conference on Information Sciences and Systems*, pages 207 – 212, Mar. 2006.

[73] M. Sarkis and K. Diepold. Depth map compression via compressed sensing. In *Proc. of IEEE Int. Conf. Image Proc., ICIP 2009*, Cairo, Egypt, Nov. 2009.

[74] G. Shen, W.-S. Kim, S. K. Narang, A. Ortega, J. Lee, and H. Wey. Edge-adaptive transforms for efficient depth-map coding. In *Proc. of 28th Picture Coding Symposium, PCS 2010*, Nagoya, Japan, Dec. 2010.

[75] G. Shen and A. Ortega. Joint routing and 2D transform optimization for irregular sensor network grids using wavelet lifting. In *IPSN*, April 2008.

[76] G. Shen and A. Ortega. Transform-based distributed data gathering. *IEEE Transactions on Signal Processing*, 58(7), July 2010.

[77] G. Shi, J. Lin, X. Chen, F. Qi, D. Liu, and L. Zhang. UWB echo signal detection with ultra-low rate sampling based on compressed sensing. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 55(4):379–383, 2008.

[78] D. I Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. Signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular data domains. *SPM*, 2013.

[79] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, and P.L. Dragotti. Directionlets: Anisotropic multidirectional representation with separable filtering. *IEEE Trans. Image Proc.*, 15(7):1916–1933, Jul. 2006.

[80] P. von Rickenbach and R. Wattenhofer. Gathering correlated data in sensor networks. In *DIALM-POMC*. ACM, October 2004.

[81] R. Wagner, H. Choi, R. Baraniuk, and V. Delouille. Distributed wavelet transform for irregular sensor network grids. In *SSP*, July 2005.

[82] A. Wang and A. Chandraksan. Energy-efficient DSPs for wireless sensor networks. *IEEE Signal Processing Magazine*, 19(4):68–78, Jul. 2002.

[83] W. Wang, M. Garofalakis, and K. Ramchandran. Distributed sparse random projections for refinable approximation. In *IPSN*, April 2007.

[84] D. Yang, H. Li, G. D. Peterson, and A. Fathy. UWB signal acquisition in positioning systems: Bayesian compressed sensing with redundancy. *Conference on Information Sciences and Systems (CISS)*, 2009.

[85] D. Yang, H. Li, and G.D. Peterson. Compressed sensing TDOA for UWB positioning systems. *IEEE Radio and Wireless Symposium (RWS)*, 2011.

[86] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak. A 100mhz-2ghz 12.5x sub-nyquist rate receiver in 90nm cmos. *IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, 2012.

[87] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak. Design and implementation of a fully integrated compressed-sensing signal acquisition system. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[88] J. Yoo, C. Turnes, E. Nakamura, C. Le, S. Becker, E. Sovero, M. Wakin, M. Grant, J. Romberg, A. Emami-Neyestanak, and E. Candès. A compressed sensing parameter extraction platform for radar pulse signal acquisition. *IEEE*

Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS), PP(99):1–13, 2012.

[89] B. Zeng and J. Fu. Directional discrete cosine transforms for image coding. In *Proc. of IEEE Int. Conf. Multimedia and Expo, ICME 2006*, pages 721–724, Toronto, Canada, Jul. 2006.

[90] B. Zeng and J. Fu. Directional discrete cosine transforms - a new framework for image coding. *IEEE Trans. Circuits Syst. Video Technol.*, 18(3):305–313, Mar. 2008.

[91] H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.