Coded Compressed Sensing with Applications to Wireless Communication

Lorne Applebaum

A Dissertation
Presented to the Faculty of Princeton University in Candidacy for the Degree of Doctor of Philosophy

Recommended for Acceptance by the Department of Electrical Engineering Adviser: Robert Calderbank

September 2012
Abstract

Compressed sensing is a new paradigm that exploits the sparsity of signals to reduce the number of measurements required to recover a representation. This is accomplished using the general notion of inner-products as measurements, encapsulated in “measurement matrices.” In this work, we focus on designing both measurement matrices as well as compressed sensing recovery algorithms. We consider several measurement code designs and recovery schemes with applications to particular systems. First, we consider chirp-coded compressed sensing measurements which, with a jointly designed recovery algorithm, is designed for computationally efficient recovery. For $M$ measurements, $O(M \log M)$ recovery is possible, a significant speed improvement over conventional random signals and recovery methods. Subsequently, we consider OFDM channel estimation in the context of compressed sensing and note that measurement matrices are restricted to the form of sub-Fourier matrices. We provide a method to manifest suitable matrices for recovering sparse channels by deterministically selecting a few pilot tones. Next, we consider how the compressed sensing paradigm can be used to build novel wireless systems. We design a multiuser detection scheme for random access on asynchronous channels. For this system, we develop new compressed sensing recovery theory and design a codebook suitable for the recovery of sparse sets of active users. Finally, we design a virtual full-duplex adhoc wireless network system using half-duplex hardware. In the system, nodes use codes containing “listening symbols” during which the devices sense the wireless channel. When active nodes in the network transmit simultaneously, each node inherits a unique compressed sensing problem to recover the data from its neighbors.

The work in this thesis shows how, with careful consideration of the application, compressed sensing with coded matrices can provide great performance improvements and novel system designs.
Acknowledgements

The work in this dissertation would not have been possible without the support of many wonderful people.

I would like to thank my family for supporting my decision to pursue graduate studies, their words of encouragement and their forgiveness for when I was sometimes too busy to phone home.

Stephen Howard and Rob Nowak deserve many thanks for welcoming me into their research spheres, showing me new ways of thinking about problems and inspiring me as a researcher. They were amazing hosts and I had a great time during my summers away from Princeton.

My colleagues and friends Waheed Bajwa, Marco Duarte and Jarvis Haupt were instrumental in many of the results found in this dissertation. They kept my research on track and were great fun to be around, both in front of the white-boards and outside of the office. I owe them much gratitude.

My fellow graduate students and researchers, both at Princeton and the places I have visited, kept new ideas flowing through my mind and maintained my work-life balance throughout my studies. I would especially like to thank Vaneet Aggarwal, Sushobhan Avasthi, Abhishek Bhattacharjee, Eugene Brevdo, Zachary Harmony, Anthony Hoffman, Aman Jain, Kevin Loutherback, Dustin Mixon, Andrea Nedic, Yury Polyanskiy, Joseph Salmon, Daniel Schwartz-Narbonne, and Eric Wang. They helped make this process an enjoyable one.

Ultimately, the greatest acknowledgments are reserved for my adviser Robert Calderbank. Professor Calderbank led me down the right paths and steered me away from the wrong ones while still giving me freedom to find my own way. He introduced me to, as well as facilitated my working with, many of the people I have mentioned above. None of the work presented here would have been possible without him.
# Contents

Abstract ........................................................................................................ iii
Acknowledgements ...................................................................................... iv

1 Introduction .................................................................................................. 1
  1.1 Notation .................................................................................................. 6

2 Chirp Codes: Compressed Sensing Measurements for Rapid Recovery ....... 7
  2.1 Introduction ............................................................................................ 8
  2.2 Compressed Sensing Background and Notation ....................................... 9
  2.3 Chirped Sensing Codes ........................................................................... 10
    2.3.1 Relation to Other Chirps and Alltop Gabor Frames ......................... 13
  2.4 RIP Analysis of Θ .................................................................................. 14
  2.5 Algorithm Guarantees ............................................................................ 17
    2.5.1 Single Witness Chirp Rate Detection .............................................. 19
    2.5.2 Frequency Parameter Selection ..................................................... 21
    2.5.3 Multiple Shift Chirp Rate Selection ............................................... 23
  2.6 $\Phi$ for Sparse Fourier Signals ............................................................. 24
  2.7 Computational Complexity ..................................................................... 25
  2.8 Performance ........................................................................................... 26
    2.8.1 Sparsity Requirements .................................................................... 27
2.8.2 Detection in Noise ............................................ 27
2.9 Conclusions and Extensions ................................. 28
2.A Proof of Lemma 2.2 ............................................. 30
2.B Proof of Theorem 2.4 ................................. 32

3 Compressed OFDM Channel Sensing ......................... 41
  3.1 Introduction ................................................. 41
  3.2 System Model .............................................. 44
  3.3 Useful Compressed Sensing Theory ......................... 46
  3.4 Deterministic Pilot Sequence and Training Data Selection .............................................. 48
  3.5 Main Results ................................................ 50
  3.6 Numerical Experiments and Discussion ..................... 52
  3.7 Conclusions .................................................. 55
  3.8 Appendix ..................................................... 56
    3.8.1 Proof of Lemma 3.2 ..................................... 56

4 Multi-user Detection for Asynchronous Random Access ........ 59
  4.1 Introduction ................................................. 60
  4.2 System Model .............................................. 64
  4.3 Multiuser Detection Using The Lasso ......................... 66
    4.3.1 Main Results ............................................. 67
    4.3.2 Computational Complexity ................................ 71
  4.4 Codewords for Multiuser Detection ......................... 73
    4.4.1 Random Rademacher Codewords: Guarantees ................................. 73
    4.4.2 Deterministic Codewords: Construction and Guarantees ......................... 77
  4.5 Numerical Results and Discussion ......................... 83
    4.5.1 Monte Carlo Experiments .................................. 83
    4.5.2 Discussion ................................................ 87
5 Codes for Random Access with Virtual Full Duplex

5.1 Introduction ............................................ 113
5.2 System Model ............................................ 115
5.3 A Small Choir Code ..................................... 117
  5.3.1 Worst-Case Coherence .............................. 118
  5.3.2 Average Coherence ................................. 120
  5.3.3 Spectral Norm ....................................... 123
  5.3.4 Receiver Fairness .................................. 127
5.4 Performance with Recovery Methods ..................... 128
  5.4.1 Restricted Isometry Property ..................... 129
  5.4.2 One-Step Thresholding ............................ 130
  5.4.3 The Lasso ........................................... 130
6 Summary and Future Directions 135
Chapter 1

Introduction

Gathering information from the external world into digital systems is an important problem that permeates many technologies. From digital cameras to cellular networks, systems are dependent on components that process physical signals and convert them to a form useful for digital manipulation. In some cases, faithful digital reproductions of the physical signal are necessary while in others detection or classification is desired. This interface with the real world has, in many applications, become a bottleneck. Moore’s law has marched forward delivering progressively more powerful computational abilities on digital information. However, Analog-to-Digital converters (A/Ds) and sensors have struggled to develop at a commensurate rate. This limits the bandwidth with which we can process signals or the detail of signals we can resolve. In other applications, such as wireless networks, the need to measure channel conditions and detect waiting users competes for resources that could otherwise be used to send and receive data. The effective use of sensing resources is a large determinant of the performance of modern technologies.

Over the past decade a new paradigm has emerged in signal processing with the potential to address the challenge of efficiently measuring signals. Techniques coined *compressed sensing* and *sparse signal recovery* have developed and with them has
come a renewed focus on sparse models of signals. We say a signal \( x \in \mathbb{R}^N \) is \( K \)-sparse in a basis \( \Psi \) if we can write

\[
x = \Psi s
\]

for some \( s \in \mathbb{R}^N \) with no more than \( K \ll N \) non-zero components. This notion of sparsity (as well as the generalization to compressibility\(^1\)) is commonly applied as a means to compress signals. Rather than store every element of \( x \) we can store the few non-zero elements in \( s \) and note their position. An example is transform coding for the compression of natural images. While the simplest representation of an image is a description of each pixel, we save on storage by transforming the image into its frequency or wavelet representation and discarding insignificant components. This use of signal sparsity requires all the elements of \( x \) to be measured before compression but it inspires the question of whether we can use the sparsity of \( x \) to not only compress it, but also measure it more efficiently. Why measure a signal only to discard much of what is measured? Compressed sensing has shown that we can indeed garner a complete characterization of a signal \( x \) using far fewer than \( N \) measurements by taking advantage of the sparsity.

Compressed sensing accomplishes this with a general notion of measurement. Whereas a sample is regarded as giving the value of a single element of \( x \), with compressed sensing we regard a measurement more generally as an inner product on \( x \). Thus, we can write a vector of measurements \( y \in \mathbb{R}^M \) as

\[
y = Ax
\]

\(^1\)Compressible signals are signals which are almost sparse in the sense that most of the signal is contained in a few strong components.
where the rows of $A$ define the measurements. Since we assume $M < N$, the matrix $A$ has a non-trivial kernel and the task of recovering a general non-sparse $x$ from $y$ is impossible. The striking result of compressed sensing is that, with care in our choice of the $M \times N$ matrix $A$ and using the fact that $x$ is sparse, we can in fact recover the signal.

The compressed sensing literature has grown to include various families of and conditions on matrices $A$ and as well as algorithms to recover $x$. In this thesis, we too provide developments in this vein. Particular to the work here, however, is the recognition that the methods of compressed sensing are most useful when the matrix $A$ is strongly connected to an application. While mathematically, it is convenient to assume $A$ can be selected arbitrarily, in reality measurements are influenced by physical models and the systems for which the measurements are made. For example, as we will see, channel estimation in orthogonal frequency-division multiplex (OFDM) systems naturally inherit measurements as rows in a Fourier matrix and asynchronous wireless channels grant $A$ a Toeplitz structure. Additionally, the structure and design of $A$ can greatly influence the recovery method used to retrieve $x$. If we deeply connect $A$ with the method used to recover $x$ there are great opportunities for performance improvements. In this thesis we consider the structure and design of matrices $A$ with a particular interest in wireless networks.

In Chapter 2, we begin generally with a design of compressed sensing measurements for real-time systems or other situations in which the speed of recovery from $y$ to $x$ is of significant concern. Chapter 2 can, in many ways, be contrasted with the large body of compressed sensing work in which random measurements are used to build the measurement matrices $A$. Random measurements conveniently divorce the selection of $A$ from the algorithm used to recover $x$. Random measurements have been shown to satisfy, with high probability, conditions that allow the recovery of $x$ with quite general methods. These general methods are most often optimization
algorithms which, for real-time systems, can be prohibitively expensive computationally. In contrast, the formulation of the measurements in Chapter 2 is algebraic in nature and co-designed with a recovery algorithm. We design the matrix such that the sparse signal can be recovered rapidly. In addition, since the matrix $\mathbf{A}$ is deterministically defined, it avoids many of the problems with storage, inherent with random measurements. The deterministic design of measurement matrices is a thread that permeates the work in this thesis as an important property for system designs. Chapter 2 serves as an initial look at the potential of deterministic coded compressed sensing.

We move into the sphere of wireless communications in Chapter 3 and consider the application of compressed sensing to OFDM channel estimation. OFDM is a common communication technique that uses the fast Fourier transform (FFT) to divide data across discrete frequency bins of a wireless channel. In order to communicate successfully, the channel coefficients of each bin must be estimated. This is typically done by receiving pilot symbols. However, each pilot symbol used occupies a bin that could otherwise be used for data transmission. The method developed in Chapter 3 provides an estimation technique for channels with few dominant signal paths (sparse in the time domain). Using the sparse properties of the channel, far fewer FFT bins need be occupied while still providing an accurate channel estimate. As OFDM pilot symbols, the measurements used to estimate the channel form rows of a Fourier matrix. The matrix $\mathbf{A}$ in the OFDM channel estimation problem cannot be arbitrary and instead takes a particular form. A deterministic procedure is provided to form the matrix $\mathbf{A}$ adhering to this requirement.

While Chapter 3 provides a way to use techniques of compressed sensing in existing systems, in Chapters 4 and 5 we consider how ideas from compressed sensing can build novel wireless systems. We target random access and adhoc network scenarios as they provide a natural notion of sparsity. A prominent task in these scenarios
is to detect the set of active users on the channel, which is small compared to the set of all users. Further, we can allow the additive nature of the wireless channel to form measurements as inner products. In Chapter 4 we focus on random access in asynchronous channels. In asynchronous channels a Toeplitz structure is imposed on the measurement matrix. Chapter 4 provides new compressed sensing theory and codeword/measurement designs adhering to the requirements of the asynchronous random access problem.

In Chapter 5 we develop theory and a code design for virtual full duplex in adhoc networks. We consider devices which share common circuitry for transmitting and receiving wireless signals so that only unidirectional communication can occur at any instant. Rather than scheduling communication on the network, we allow active users to communicate simultaneously with “listening symbols” defining periods of reception. By careful codeword design, each user inherits a unique compressed sensing problem which can be used to recover the data of the transmitters in its neighborhood. As a result, full duplex communication is achieved on the codeword time-scale.

The work in this thesis shows that the structure of $A$ is an important consideration for the application of ideas in compressed sensing. We see that wireless communication applications impose particular structures on $A$ and that by deterministically designing the matrix we can both exploit the structure and reap performance benefits.
1.1 Notation

Unless otherwise remarked, we use the following notational conventions.

\( \mathbf{x} \) Bold face lower-case letters denote vectors.

\( \mathbf{X}, \Psi \) Bold face upper-case letters and upper-case Greek letters denote matrices.

\( \mathbf{I} \) Identity matrix with size given by context.

\( \mathbf{0} \) All zero vector with size given by context.

\( (\cdot)^T \) Matrix/vector transpose.

\( (\cdot)^H \) Matrix/vector conjugate transpose.

\( \langle \mathbf{x}, \mathbf{y} \rangle \) Inner-product between vectors \( \mathbf{x} \) and \( \mathbf{y} \).

\( \mathbb{R} \) The set of real numbers.

\( \mathbb{C} \) The set of complex numbers.

\( \mathbb{N} \) The set of natural numbers.

\( \mathbb{Z} \) The set of integers.

\( \mathbb{Z}_M \) The set of integers modulo \( M \).

\( \mathbb{F}_M \) The Galois field of size \( M \).

\( \mathcal{F}\{\cdot\} \) The discrete Fourier transform.

\( \mathcal{N}(m, \sigma^2) \) The normal distribution with mean \( m \) and variance \( \sigma^2 \).

\( \text{binary}(\pm 1/\sqrt{N}, \mathbf{I}_N) \) \( N \)-length Rademacher distribution in which each entry independently takes value \( +1/\sqrt{N} \) or \( -1/\sqrt{N} \) each with probability \( 1/2 \).

\( \log(\cdot) \) The natural logarithm.

\( \mathbb{P}(\cdot) \) The probability of an event.

\( \mathbb{P}(\cdot|C) \) The probability of an event conditioned on \( C \).

\( \mathbb{E}[\cdot] \) The expectation of a random variable.

\( \lfloor \cdot \rfloor \) The floor operator.

\( a \mid b \) The integer \( a \) divides the integer \( b \).

\( a \nmid b \) The integer \( a \) does not divide the integer \( b \).
Chapter 2

Chirp Codes: Compressed Sensing Measurements for Rapid Recovery

Compressed sensing is a novel technique to acquire sparse signals with few measurements. Normally, compressed sensing uses random projections as measurements. Here we design deterministic measurements and an algorithm to accomplish signal recovery with computational efficiency. A measurement matrix is designed with chirp sequences forming the columns. Chirps are used since an efficient method using FFTs can recover the parameters of a small superposition. We show that this type of matrix is valid as compressed sensing measurements. This is done by bounding the eigenvalues of sub-matrices, as well as by an empirical comparison with random projections. Further, by implementing our algorithm, simulations show successful recovery of signals with sparsity levels similar to those possible by Matching Pursuit with random measurements. For $K$ sparse signals, our algorithm recovers the signal with computational complexity $O(KM \log M)$ for $M$ measurements. This is a significant improvement over conventional algorithms.
2.1 Introduction

In compressed sensing, the use of randomly generated projections to make measurements has the useful consequence of sidestepping the computationally difficult task of checking whether the measurements allow for signal recovery. By considering recovery stochastically, it has been shown that measurements generated from Gaussian or Bernoulli random variables allow for signal recovery with high probability. In some ways, the use of random measurements may be viewed as an analogy to random codes used by Shannon to prove theorems in channel coding. Though useful in proofs, purely random channel codes are never used in practice because encoding and decoding would be far too computationally intensive. Instead, practical channel codes are developed with an efficient coding and decoding scheme in mind. We have a similar situation in compressed sensing. Though $\ell_1$ minimization has been shown to recover the signal from random projections [1], it is computationally expensive. The question arises as to whether we can design projections to facilitate the rapid recovery of the signal. This is an issue of practical consequence. If compressed sensing is to be used in real-time systems, we must have a method which, in addition to reducing the number of measurements, is able to recover the signal quickly. Here we present a proof of concept scheme that accomplishes this.

A number of decoding schemes have been proposed that improve upon the $\ell_1$ minimization signal recovery technique (also known as Basis Pursuit). However, most schemes presuppose random measurements. Examples include Orthogonal Matching Pursuit [2] and its refinements [3]. In contrast, the scheme presented here exploits structure in deterministically designed measurements to make recovery much faster. There exists a small number of other schemes with less structurally random measurements [4–6]. The scheme presented here has lower recovery complexity.

The remainder of the chapter is organized as follows. In Section 2.2 we provide necessary background and notation and in Section 2.3 we introduce our encoding
scheme and the corresponding decoding algorithm. Section 2.4 provides analysis of our encoding matrix in terms of restricted isometry properties commonly employed in compressed sensing. In Section 2.5 we provide analytical guarantees for recovery using our decoding algorithm. In Section 2.6 we consider our scheme in the special case of Fourier signals. In Sections 2.7 and 2.8 we examine our algorithm in terms of computational complexity, signal recovery and robustness to noise.

2.2 Compressed Sensing Background and Notation

We consider discrete signals of finite length. Let $x$ be a length $N$ signal which we would like to sense and recover. We assume that $x$ is sparse in some orthonormal basis. Thus, we can write $x$ as

$$x = \Psi s \quad \quad (2.1)$$

where $s$ is a length $N$ vector with $K$ or fewer non-zero elements. We measure $x$ with $M < N$ projections which result in the vector $y$. The vectors projected upon are set as the rows of the $M \times N$ matrix $\Phi$ which gives

$$y = \Phi \Psi s = \Theta s \quad \quad (2.2)$$

where the second equality is by definition of $\Theta$. We are free to design $\Phi$ and thus $\Theta$. Though, if we design $\Theta$ we should remain aware that actual sensing of the signal is done with $\Phi$.

Since $\Theta$ is a wide matrix, solving for $s$ given $y$ is ill posed. However, using non-linear methods, we can leverage the fact that $s$ has at most $K$ non-zero elements. It has been shown in [7] that if $\Theta$ satisfies certain restricted isometry properties (RIP),
s can be recovered perfectly using an $\ell_1$ minimization. An important example is randomly generated $\Theta$. Several results exist showing that, when $K$ satisfies

$$K < cM/ \log(N/M),$$

(2.3)

with a known constant $c$, randomly generated matrices of various types satisfy RIP with high probability [8]. Thus, if a signal’s sparsity is bounded by (2.3), then it can be recovered from $M$ random measurements with high probability.

We will consider our designed $\Theta$ more precisely in terms of RIP in Section 2.4. There, we will also give an empirical comparison of the eigenvalue statistics of our designed $\Theta$ with those of randomly generated measurements showing that (2.3) applies to recoverability from chirp sensing codes.

### 2.3 Chirped Sensing Codes

We approach the recovery problem by noting that finding $s$ is equivalent to discovering which small linear combinations of the columns of $\Theta$ form $y$. We will design $\Theta$ to facilitate this. In particular, we will look at a $\Theta$ designed with chirp signals forming the columns.

We use the type of chirps analysed in [9]. A prime length $M$ chirp signal has the form

$$v_{m,r}(l) = \alpha \cdot e^{j2\pi ml/M + j\pi rl(l-M)/M} \quad m, r \in \mathbb{Z}_M$$

(2.4)

where $m$ is the base frequency and $r$ is the chirp rate. For a length $M$ signal, there are $M^2$ possible pairs $(m, r)$. We will form a $M \times M^2$ sized $\Theta$ that has columns filled with all $M^2$ uni-modular chirp signals (setting $\alpha = 1$ for notational convenience, though in Section 2.4 $\alpha = \frac{1}{\sqrt{M}}$ is used).
Consider a vector $y$, with entries indexed by $l$, formed from the linear combination of some chirp signals. Taking $\mathcal{I}$ as a subset of $\mathbb{Z}_M \times \mathbb{Z}_M$ defining the set of active pairs of base frequencies and chirps rates, we can write

$$y(l) = \sum_{(m,r) \in \mathcal{I}} \beta_{m,r} v_{m,r}(l)$$

(2.5)

which have base frequencies $m$ and chirp rates $r$. The chirp rates can be recovered from $y$ by looking at $\bar{y}(l)y(l+s)$, where the index $l+s$ is taken mod $M$. This gives

$$f_s(l) = \bar{y}(l)y(l+s) = \sum_{(m,r) \in \mathcal{I}} |\beta_{m,r}|^2 v_{m,r}(s)v_{sr,0}(l)$$

$$+ \sum_{\substack{(m,r) \in \mathcal{I} \\ (m',r') \in \mathcal{I} \\ (m,r) \neq (m',r')}} \beta_{m,r}\beta_{m',r'} v_{m,r}(s)v_{(m-m'+sr),(r-r')}(l)$$

(2.6)

where the cross terms in the second sum remain as chirps. We see that $f_s(l)$ is a signal that has sinusoids at the discrete frequencies $sr$ mod $M$. If $M$ is prime, this is a bijection from chirp rates to FFT bins. Furthermore, the remainder of the signal consists of the cross terms. Since the cross terms are chirps, their energy is spread across all FFT bins.

As long as $y$ consists of sufficiently few chirps ($s$ is sparse), taking a FFT of $f_s(l)$ results in a spectrum with significant peaks at locations corresponding to $sr$ mod $M$ from which we can glean chirp rates.

Upon discovering the chirp rate $r$ we can “dechirp” the signal $y(l)$ by multiplying by $e^{-j\pi rl(l-M)/M}$. This converts only the chirps with rate $r$ to sinusoids. Performing an FFT on the resulting signal can be used to retrieve the corresponding value(s) for $m$ and $\beta_{m,r}$.
Setting the elements of $\Theta$ as

$$[\Theta]_{l,k} = e^{j\pi r(l-M)/M} e^{j2\pi ml/M} \quad \text{with} \quad k = Mr + m \in \mathbb{Z}_M^2$$  (2.7)

we see that $y = \Theta s$ will have the form (2.5). Given $y$ formed using $\Theta$, we summarize the scheme described in Algorithm 2.1. The recovery of $\beta_i$ gives the value of an element in $s$ while the pair $(m_i, r_i)$ gives its location in $s$ as the index $Mr_i + m_i$.

Algorithm 2.1 Chirp recovery algorithm

1. Choose a $s \in \mathbb{Z}_M$, $s \neq 0$, and a stopping energy $\epsilon$.
2. Form $f(l) = \bar{y}(l)y(l + s)$ and take length $M$ FFT.
3. Find location of the peak in the FFT as $r_i \mod M$ and record the unique $r_i$ corresponding to the location.
4. Multiply $y(l)$ by $e^{-j\pi r_i l(l-M)/M}$ and take length $M$ FFT.
5. Find the location of the peak and record as $m_i$. Use the value of the peak to recover $\beta_i$.
6. Replace $y$ with $y - \beta_i e^{j2\pi ml/M + j\pi r_i l(l-M)/M}$
7. Repeat steps 2-6 until $\|y\|^2_2 < \epsilon$ or have iterated $K$ times.

At the expense of more computation, we can improve the performance of the algorithm by exploiting the availability of $f_s(l)$ for different delays $s$. By adding the FFT bins of each chirp rate $r$ with those of the FFTs formed from the other delays, we can mitigate the effect of noise and any significant values from cross terms in (2.6). A bin with a chirp at $r$ is correlated across different $s$ while with a white noise approximation of the cross term interference (or simply white noise), other bins are not correlated and have zero mean. In the extreme case, we take an FFT for all $M - 1$ possible shifts $s$.

The modified procedure is summarized in Algorithm 2.2.
Algorithm 2.2 Multiple shift chirp recovery algorithm

1. Choose a stopping energy $\epsilon$ and a set of shifts $W \subset \mathbb{Z}_M/\{0\}$.
2. Form $f_s(l) = \bar{y}(l)y(l + s)$ for every $s \in W$ and take length $M$ FFT of each.
3. Using $r_is \mod M$, reorganize the output of each FFT such that the bins are in order of increasing $r_i$.
4. Sum the magnitude-squared values of the reorganized FFTs and record the peak $r_i$.
5. Multiply $y(l)$ by $e^{-j\pi r_i(l-M)/M}$ and take length $M$ FFT.
6. Find the location of the peak and record as $m_i$. Use the value of the peak to recover $\beta_i$.
7. Replace $y$ with $y - \beta_i e^{j2\pi ml/M + j\pi rl/M}$.
8. Repeat steps 2-7 until $\|y\|_2^2 < \epsilon$ or have iterated $K$ times.

We derive recovery guarantees for these algorithms in Section 2.5. Further, we compare the performance of original algorithm with this modified algorithm in Sections 2.8.1 and 2.8.2.

2.3.1 Relation to Other Chirps and Alltop Gabor Frames

The chirps defined in (2.4) are a slight modification of the chirps originally used in [10]. To see the relation more clearly, consider $\omega = e^{2\pi i/M}$ and write an alternative representation as

$$\tilde{v}_{m,r} = \alpha \omega^{ml} \omega^{2^{-1}rl^2}$$

(2.8)

where we have written $2^{-1}$ to emphasize the multiplicative inverse of 2 in $\mathbb{Z}_M$ rather than taking a square root of $\omega$. To see that this equivalent to (2.4) consider that, for all $m,r,l$, both chirps take values in the group $< \omega >$. This is trivially clear in (2.8) and can be seen in (2.4) after noting one of $l$ or $(l-M)$ must be even. Next consider that $\tilde{v}_{m,r}(l)^2 = v_{m,r}(l)^2$. Since $(\cdot)^2$ is bijective on the prime-ordered group.
\( <\omega> (-x \notin <\omega> \forall x \in <\omega>) \), it must be that \( \tilde{v}_{m,r}(l) = v_{m,r}(l) \). A multiplication of chirp rates by \( 2^{-1} \) converts the chirps in [10] to the chirps used here.

The chirps of in (2.4) are also related to vectors taken from an Alltop Gabor frame. In [11], it is shown that the set of chirps in (2.4) are “wiggling equivalent” to a Gabor frame generated by the Alltop function. That is, with an additional phase multiplying each chirp, the two sets are the same. Gabor frames are popular in compressed sensing literature and many of the results and techniques presented here can be applied.

### 2.4 RIP Analysis of \( \Theta \)

Having described the structure of \( \Theta \) in (2.7), here we consider the matrix in terms of restricted isometry properties giving signal recovery guarantees. Like much of the compressed sensing literature, the properties of interest here are defined in terms of restricted isometry constants [7]. For \( \Theta \) scaled such that its columns have unit norm, they are defined as the smallest \( \delta_K \) such that

\[
(1 - \delta_K) \|x\|^2 \leq \|\Theta_S x\|^2 \leq (1 + \delta_K) \|x\|^2 \quad \forall x, |S| = K
\]

(2.9)

where \( \Theta_S \) is the sub-matrix of \( \Theta \) using the \( K \) columns specified in the set \( S \). The constants \( \delta_K \) are equivalently bounds on the eigenvalues of \( \Theta_S^H \Theta_S \) close to unity. Bounds on the restricted isometry constants can give guarantees on the recoverability. For example, from [7] we have the following lemma.

**Lemma 2.1.** Suppose that \( K \geq 1 \) is such that \( \delta_{2K} < 1 \), and let \( s \) be a \( K \)-sparse signal with measurements \( y \). Then \( s \) is the unique minimizer to

\[
\min \|d\|_{\ell_0} \quad s.t. \quad \Theta d = y
\]

(2.10)
where the $\ell_0$-norm is the count of non-zero elements.

Using arguments similar to the uniqueness arguments in [12], here we calculate a bound on the sparsity $K$ for $\delta_{2K} < 1$ to hold for $\Theta$.

**Theorem 2.1.** Suppose that $K < (\sqrt{M} + 1)/2$ and let $s$ be a $K$-sparse signal with chirp code measurements $y$. Then $s$ is the unique minimizer of (2.10).

**Proof.** Let $|S| = 2K$ and $\lambda$ be an arbitrary eigenvalue of $G = \Theta_S^H \Theta_S$. By the Gershgorin circle theorem,

$$|\lambda - 1| \leq \sum_{i \neq j} |G_{ij}| \quad \forall j$$

(2.11)

The values $G_{i,j}$, $i \neq j$ are the inner products between non-identical normalized chirps. From the work in [9], for prime $M$ these inner products satisfy $|G_{i,j}| \leq 1/\sqrt{M}$. Thus,

$$|\lambda - 1| \leq \frac{1}{\sqrt{M}} (2K - 1)$$

(2.12)

Since this is true for all eigenvalues of $\Theta_S^H \Theta_S$, the upper-bound above also applies to $\delta_{2K}$. Applying Lemma 2.1 upon the bound gives the result. \qed

More restrictive conditions than Lemma 2.1 exist for recovery by $\ell_1$ minimization. These can be approached in a similar fashion to Theorem 2.1. However, rather than examine this directly, we will now show that chirp measurements can perform as well as randomly generated measurements. To do this, we consider the restricted isometry properties in a stochastic manner.

The standard compressed sensing formulation described in the literature considers the signal $s$ as fixed and examines recovery using of random measurements. The possibility of recovery is inherited from the randomness in the measurements. However, here we have developed deterministic measurements $\Theta$. This makes it difficult to find
the values of $\delta_K$ for non-random $\Theta$ and, in general, requires the computationally difficult problem of checking all $\binom{N}{K}$ possible $S$. However, we conjecture that, considering the signal rather than the measurements stochastically, a randomly generated sparse $s$ can be recovered from $\Theta$ with high probability. Empirical results presented here support this claim.

Recall that $\Theta$ satisfies RIP when $\Theta_S^H \Theta_S$ has eigenvalues sufficiently close to 1 for appropriately sized $S$. The sets $S$ relate to the possible supports of an $K$-sparse $s$. Considering these sets randomly, we consider the probability that $\Theta_S^H \Theta_S$ has eigenvalues appropriately close to 1.

After scaling $\Theta$ so that its columns have unit norm, we compare the statistics of eigenvalues of its Gram matrices to those of a matrix with Gaussian entries of zero mean and variance $\frac{1}{M}$. From well-known compressed sensing results, this Gaussian matrix is known to satisfy RIP with high probability when (2.3) is satisfied [7].

Figure 2.1 shows the sample means and standard deviations of the maximum and minimum eigenvalues of $\Theta_S^H \Theta_S$ for varying $K$. For every value $K$, sets $S$ are generated uniformly random over all sets and the statistics are accumulated from
10,000 samples. A value of $M = 67$ was used for the simulation. For comparison, the sample means of the maximum and minimum eigenvalues of the Gram matrices of the Gaussian measurements are also shown.

As a specific example, if we consider Figure 2.1 in terms of Lemma 2.1 we see that, with $|S| \leq 16$, the eigenvalues of $\Theta_S^H \Theta_S$ within 1 of unity with high probability. This implies $\ell_0$ minimization can recover a random 8-sparse signal with high probability.

More generally we note that from Figure 2.1 we see that the eigenvalues of $\Theta_S^H \Theta_S$ are, on average, closer to 1 by more than a standard deviation compared to the corresponding eigenvalues of Gaussian measurements. Thus, if Gaussian measurements satisfy any condition based on the RIP constants, then our $\Theta$ will also be able to recover a random $K$-sparse signal with high probability. Results are similar for other values of $M$.

It is important to note that here we have analyzed the measurement matrix $\Theta$ in isolation of the algorithm presented in Section 2.3. In this section, we have shown that $\Theta$ is suitable as compressed sensing measurements in general. In Section 2.5 and Section 2.8, we examine how the measurements perform jointly with our corresponding algorithm.

## 2.5 Algorithm Guarantees

In this section we provide sufficient conditions for the correct recovery of $s$ from $y$ using the algorithm defined in Section 2.3.

In order to simplify the analysis, we consider the case when no chirp rate $r$ has multiple base frequencies associated. We assume every element of $I$ has a distinct chirp rate. We justify this assumption using a probabilistic argument. That is, if $I$ is selected uniformly at random then the assumption holds with high probability when $|I|$ is small. More precisely, we have the following lemma.
Lemma 2.2. Assume $\mathcal{I}$ is selected uniformly at random from all $K$ sized subsets of $\mathbb{Z}_M \times \mathbb{Z}_M$. Then, if $K \leq M^{1/2-\epsilon}$ for some $\epsilon > 0$, the members of $\mathcal{I}$ have distinct second coordinates with probability exceeding $1 - \frac{2}{M^2}$.

The lemma is from [13]. Its argument is replicated in Appendix 2.A for convenience.

In order to analyze the algorithm we will use the following useful properties of the chirp signals (2.4).

- Unitary Fourier Transform of a chirp is a chirp [9, Theorem 1].

$$\mathcal{F}\{v_{x,y}\}(t') = \alpha_{x,y}v_{xy^{-1},(-y^{-1})}(t')$$  \hspace{1cm} (2.13)

where $\alpha_{x,y} = \mathcal{F}\{v_{x,y}\}(0)$ which is calculable and $|\alpha_{x,y}| = 1$ [9, Proposition 5].

Further, by considering $\mathcal{F}\{v_{x,y}\} = \mathcal{F}\{v_{x,0}v_{0,y}\}$ we can say $\alpha_{x,y} = \alpha_{0,y}v_{0,(-y^{-1})}(x)$.

- Time scaling

$$v_{x,y}(st) = v_{xs,ys^2}(t)$$  \hspace{1cm} (2.14)

- Change of time variable

$$v_{ax+bs^2,cs^2}(t) = v_{at,2bt+ct^2}(s)$$  \hspace{1cm} (2.15)

We begin by applying (2.13) to apply the Fourier transform on (2.6) to find

$$\mathcal{F}\{f_s\}(t') = \sqrt{M} \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|^2v_{x,y}(s)\delta(t' - ys)$$

$$+ \sum_{(x,y) \in \mathcal{I}} \beta_{x,y}\beta_{x',y'}^*v_{x,y}(s)\alpha(x-x'+ys, (y-y')^{-1})(x-x'+s)(y-y')^{-1}(t')$$  \hspace{1cm} (2.16)
Using the fact that \( \alpha_{x,y} = \alpha_{0,0}v_{0,-1}(x) \) we can write

\[
\alpha_{(x-x'+sy),(y-y')} = \alpha_{y-y'}v_{0,(-(y-y')^{-1})}(x - x' + sy)
\]

\[
= \alpha_{y-y'}v_{0,(-(y-y')^{-1})}(x - x')v_{-(x-x')y(y-y')^{-1},-y^2(y-y')^{-1}}(s)
\]

which gives

\[
\mathcal{F}\{f_s\}(t') = \sqrt{M} \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|^2 v_{x,y}(s) \delta(t' - ys)
\]

\[
+ \sum_{(x,y) \in \mathcal{I}} \beta_{x,y}\beta^*_{x',y'} \alpha_{y-y'}v_{0,(-(y-y')^{-1})}(x - x')
\]

\[
\times v_{x-(x-x')y(y-y')^{-1},-y^2(y-y')^{-1}}(s)
\]

\[
\times v_{(x-x'+sy)(y-y')^{-1},-(y-y')^{-1}}(t')
\]  \hspace{1cm} (2.17)

Finally, permuting the FFT bins moving \( ys \mapsto y \) yields

\[
\mathcal{F}\{f_s\}(t') = \sqrt{M} \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|^2 v_{x,y}(s) \delta(t' - y)
\]

\[
+ \sum_{(x,y) \in \mathcal{I}} \beta_{x,y}\beta^*_{x',y'} \alpha_{y-y'}v_{0,(-(y-y')^{-1})}(x - x')
\]

\[
\times v_{x-(x-x')y(y-y')^{-1},-y^2(y-y')^{-1}}(s)
\]

\[
\times v_{s(x-x'+sy)(y-y')^{-1},-(s^2(y-y')^{-1})}(t')
\]  \hspace{1cm} (2.18)

2.5.1 Single Witness Chirp Rate Detection

By using Lemma 2.2, we assume that summands in the first sum of (2.19) are orthogonal. That is, there is no such \( p \neq q \) such that both \( (p,y) \in \mathcal{I} \) and \( (q,y) \in \mathcal{I} \) (we
assume only one base frequency per chirp rate is active). Under this assumption we need not worry about phases due to \(v_{x,y}(s)\) canceling in the sum.

Let \((p, q)\) be an arbitrary pair in \(\mathcal{I}\). Using the triangle inequality, the magnitude at the FFT bin for \(q\) is lower bounded by

\[
|\mathcal{F}\{f_s\}(q's)| \geq \sqrt{M}|\beta_{p,q}|^2 - \sum_{\substack{(x,y) \in \mathcal{I} \\ (x',y') \in \mathcal{I} \\ (x,y) \neq (x',y')}} |\beta_{x,y}||\beta_{x',y'}| \tag{2.20}
\]

The magnitude of any FFT bin \(t'\) not corresponding to some chirp rate in \(\mathcal{I}\) is upper bounded by

\[
|\mathcal{F}\{f_s\}(t's)| \leq \sum_{\substack{(x,y) \in \mathcal{I} \\ (x',y') \in \mathcal{I} \\ (x,y) \neq (x',y')}} |\beta_{x,y}||\beta_{x',y'}| \tag{2.21}
\]

We can detect the chirp rate \(q\) as long as (2.20) is greater than (2.21). This is guaranteed when

\[
\sqrt{M}|\beta_{p,q}|^2 \geq 2 \sum_{\substack{(x,y) \in \mathcal{I} \\ (x',y') \in \mathcal{I} \\ (x,y) \neq (x',y')}} |\beta_{x,y}||\beta_{x',y'}| 
\]

\[
= 2 \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}| \left[ \sum_{(x',y') \in \mathcal{I}} |\beta_{x',y'}| - |\beta_{x,y}| \right] \tag{2.22}
\]

\[
= 2 \left( \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}| \right)^2 - 2 \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|^2 
\]

\[
= 2 \left( K(K - 1) \text{AVG}^2[|\beta|] - K \text{VAR}[|\beta|] \right)
\]
where for the last equality we have set $K = |\mathcal{I}|$ while AVG and VAR denote the empirical average and variance defined as

$$\text{AVG}[||\beta||] = \frac{1}{K} \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|$$

$$\text{VAR}[||\beta||] = \frac{1}{K} \sum_{(x,y) \in \mathcal{I}} |\beta_{x,y}|^2 - \text{AVG}^2[||\beta||]$$

(2.23)

Defining LAR as the peak to average ratio

$$\text{LAR}[||\beta||] = \max_{(p,q) \in \mathcal{I}} \frac{|\beta_{p,q}|}{\text{AVG}[||\beta||]}$$

(2.24)

we see that (2.22) is satisfied and a correct chirp rate is detected when

$$\text{LAR}^2[||\beta||] > \frac{2K(K-1)}{\sqrt{M}}$$

(2.25)

which informs us that $K$ should scale as $\sqrt{M}$ for single witness detection. These results are summarized in the following theorem.

**Theorem 2.2.** If the set $\mathcal{I}$ of active chirps have distinct chirp rates and if $K = |\mathcal{I}|$ satisfies $K(K-1) < \frac{\sqrt{M}}{2}$, each iteration of Algorithm 2.1 recovers the chirp rate of the chirp component with the largest coefficient.

Lemma 2.2 implies that the condition on $\mathcal{I}$ in Theorem 2.2 occurs with high probability when $\mathcal{I}$ is selected at random.

### 2.5.2 Frequency Parameter Selection

The ultimate step in the algorithm is to extract the frequency parameter. We assume we have found $q \in \mathcal{I}$ for the largest value $\beta_{p,q}$ in the set of $\{\beta_{x,y}\}$. To find $p$, we return to the signal $y(t)$, de-chirp it with $v_{0,-q}$ and consider its Fourier transform.
The de-chirped signal has the form

\[ v_{0,-q}(t)y(t) = \beta_{p,q}v_{p,0} + \sum_{(x,y) \in I/(p,q)} \beta_{x,y}v_{x,y-q}(t) \]  

resulting in a Fourier transform of the form

\[ \mathcal{F}\{v_{0,-q}y\}(t') = \sqrt{M}\beta_{p,q}\delta(t' - p) + \sum_{(x,y) \in I/(p,q)} \beta_{x,y}v_{x(y-q)^{-1},-(y-q)^{-1}}(t') \]  

At this point, we can ask under what condition is the frequency bin \( t' = p \) the largest. The magnitude at the bin \( p \) is given by

\[ \left| \sqrt{M}\beta_{p,q} - \sum_{(x,y) \in I/(p,q)} \beta_{x,y}v_{x(y-q)^{-1},-(y-q)^{-1}}(p) \right| \geq \sqrt{M}|\beta_{p,q}| - \sum_{(x,y) \in I/(p,q)} |\beta_{x,y}| \]  

The magnitude of the other bins is given by

\[ \left| \sum_{(x,y) \in I/(p,q)} \beta_{x,y}v_{x(y-q)^{-1},-(y-q)^{-1}}(t') \right| \leq \sum_{(x,y) \in I/(p,q)} |\beta_{x,y}| \]  

Thus, the peak at bin \( p \) is the largest when

\[ \sqrt{M}|\beta_{p,q}| > 2 \sum_{(x,y) \in I/(p,q)} |\beta_{x,y}| \]

\[ = 2K \text{AVG}[|\beta|] - 2|\beta_{p,q}| \]  

Thus, we correctly detect the frequency parameter \( p \) when

\[ \text{LAR}[|\beta|] > \frac{2K}{\sqrt{M}} \]  

which is summarized in the following theorem.
Theorem 2.3. Assume the set $I$ of active chirps have distinct chirps rates and the chirp rate detection step has correctly recovered the chirp rate with largest coefficient. Then, if $K = |I|$ satisfies $K < \sqrt{\frac{M+2}{2}}$, each iteration of Algorithm 2.1 or Algorithm 2.2 recovers the base frequency of the chirp component with the largest coefficient.

As before, Lemma 2.2 provides that the condition on $I$ in Theorem 2.3 occurs with high probability when $I$ is selected at random.

2.5.3 Multiple Shift Chirp Rate Selection

To consider the performance of the multiple shift chirp recovery in Algorithm 2.2, we consider the sum

$$M(t') = \sum_{s \in \mathcal{W}} |\mathcal{F}\{f_s\}(t's)|^2 \quad (2.32)$$

where the terms are the squared magnitudes of the reorganized FFT bins in (2.19) for each $s \in \mathcal{W}$. The theorem below tells us when we can guarantee that the peak of $M(t')$ coincides with the chirp of largest coefficient in $\beta$. However, it requires the following technical condition on $I$.

Condition 2.1 (Sufficiently Generic $I$). Let $(x,y), (x',y'), (p,q), (p',q')$ be pairs in $I$ and consider the two equations

$$(x - p) - (x - x')y(y - y')^{-1} + (p - p')q(q - q')^{-1} + t [(x - x')(y - y')^{-1} - (p - p')(q - q')^{-1}] = 0 \quad (2.33)$$

and

$$(y - q) - y^2(y - y')^{-1} + q^2(q - q')^{-1} + 2t [y(y - y')^{-1} - q(q - q')^{-1}] + t^2 [(q - q')^{-1} - (y - y')^{-1}] = 0. \quad (2.34)$$
We say $\mathcal{I}$ is sufficiently generic with constant $c_1$ if there are no more than $K^2$ solutions simultaneously satisfying (2.33) and (2.33) in $\mathcal{I} \times \mathcal{I} \times \mathcal{I} \times \mathcal{I}$ for each $t \in \mathbb{Z}_M$.

**Theorem 2.4.** Assume $\mathcal{I}$ has distinct chirp rates, satisfies Condition 2.1 and has $|\mathcal{I}| = K$. Then if $K \leq cM^{3/8}$, each iteration of Algorithm 2.2 with $\mathcal{W} = \mathbb{Z}_M \setminus \{0\}$ recovers the chirp rate of the chirp component of largest coefficient, where $c$ is a constant dependent only on $\text{LAR}[|\beta|]$.

Theorem 2.4 is proved in Appendix 2.B.

Taking $\mathcal{I}$ to be selected uniformly at random, Lemma 2.2 provides the distinct chirp rate condition with high probability. We also conjecture that random $\mathcal{I}$ satisfy Condition 2.1 with a constant $c_1$ independent of $M$ and $K$ with high probability. Simulations support this conjecture.

Theorem 2.4 illustrates the advantage of using multiple shifts. Whereas with a single shift, we required $K$ to scale as $M^{1/4}$, using all possible shifts allows $K$ to scale as $M^{3/8}$. Further, the simulations in Section 2.8 show that we need not use all shifts to achieve this full benefit.

### 2.6 $\Phi$ for Sparse Fourier Signals

Here we consider the use of Chirp codes in the special case of sparse Fourier signals. Though we are interested in being able to determine the combination of columns of $\Theta$, measurements are taken upon $x$ and thus are made with the matrix $\Phi = \Theta \Psi^{-1}$. We are therefore concerned with $\Phi$ for implementation. When $\Psi$ is the Fourier matrix (if $x$ is a sparse superposition of sinusoids), we can find the structure of $\Phi$ directly.

As described above, we set $\Theta$ to have the $l,k$ entry of the form

$$[\Theta]_{l,k} = e^{j\pi l(l-M)M} e^{j2\pi mlM} \quad \text{with} \quad k = Mr + m \in \mathbb{Z}_{M^2}$$
which we re-write using (2.8) as

\[ [\Theta]_{l,k} = e^{\frac{j2\pi r}{M} (2-1)^2} e^{\frac{j2\pi m}{M}} \quad \text{with} \quad k = Mr + m \in \mathbb{Z}_{M^2} \]

This construction of \( \Theta \) groups the columns of chirps in blocks of chirp rates.

Since \( \Psi \) is the Fourier matrix, \( \Phi = \Theta \Psi^{-1} \) is a matrix with rows formed by the \( M^2 \) length Fourier transform of the rows of \( \Theta \). The Fourier transform of the \( l \)th row of \( \Theta \) is given by

\[
\frac{1}{\sqrt{M}} \sum_{k=0}^{M^2-1} [\Theta]_{l,k} e^{-\frac{j2\pi k\omega}{M^2}} = \sum_{r=0}^{M-1} \sum_{m=0}^{M-1} e^{\frac{j2\pi r}{M} (2-1)^2} e^{\frac{j2\pi m}{M} \frac{l}{M}} e^{\frac{-j2\pi r}{M^2} \frac{l}{M}} e^{\frac{-j2\pi m}{M^2} \frac{l}{M}} e^{\frac{j2\pi \omega}{M^2} \frac{l}{M}} e^{\frac{-j2\pi \omega}{M^2} \frac{l}{M}}
\]

\[
= \left[ \sum_{r=0}^{M-1} e^{\frac{j2\pi r}{M} (2-1)^2} \right] \left[ \sum_{m=0}^{M-1} e^{\frac{j2\pi m}{M} (Ml-\omega)} \right]
\]

\[
= M \delta_{(2-1)^2-\omega} e^{j\pi (Ml-\omega)/(1/M-1/M^2)}
\]

\[
\times \frac{\sin (\pi (Ml-\omega)/M)}{\sin (\pi (Ml-\omega)/M^2)}
\]

where \( \delta_i \) is the Kronecker delta with \( i \) taken mod \( M \). Thus, the rows of \( \Phi \) are periodic trains of delta functions modulated by a sinc function. This means that the measurements \( y \) can be formed simply as a weighted sum of a sparse number of samples of \( x \). Thus, the encoding of \( y \) has a relatively low computational cost.

This formulation of \( \Phi \), along with the scenario of sparse signals in the Fourier domain, is used in the simulations illustrated later.

### 2.7 Computational Complexity

Reconstruction in compressed sensing is normally done by solving a linear program minimizing \( \|s\|_1 \). As remarked earlier, this method is computationally intensive and has complexity \( O(N^3) \). An alternative scheme is the greedy Matching Pursuit algo-
rithm with complexity $O(MKN)$, which we use in simulations for comparison. Here, we consider the complexity of the Chirp Code algorithm.

By using the chirp decoding algorithm we leverage the efficiency of the FFT. Similar to Matching Pursuit, the algorithm iteratively pulls out the strongest signals. In this algorithm, each “peel” requires the computation of two FFTs of length $M$: a first, to extract the coded chirp rate, and a second to extract the coded frequency. Since, for a $K$-sparse signal approximately $K$ peels are required, the complexity of the computation is

$$O(KM \log M)$$  \hspace{1cm} (2.36)

As noted in Section 2.3 we can trade additional computation for improved performance by using multiple delays $s$ to form $f(l)$. In the most computationally intensive case, we perform $M$ length $M$ FFTs for each peel which gives an overall complexity of

$$O(KM^2 \log M)$$  \hspace{1cm} (2.37)

In terms of computation, the algorithm is a significant improvement upon $\ell_1$ minimization and Matching Pursuit. For comparison, a table of various algorithms and their complexities can be found in [5].

### 2.8 Performance

Here we present some simulation results characterizing the performance of the algorithm. The simulations were produced using a measurement matrix formed as described in Section 2.3 acting upon a sparse signal with active entries chosen uniformly at random. We compare its performance against the matching pursuit algorithm using a Gaussian random $\Theta$. 

26
2.8.1 Sparsity Requirements

An important examination is whether the algorithm’s improved computation complexity degrades the sparsity level at which signals can be recovered. We look at the sparsity requirements for various signal lengths while using $M = \sqrt{N}$ measurements.

Figures 2.2 and 2.3 compare the reconstruction error of the chirp sensing code algorithm with that of matching pursuit for signal lengths $N = 127^2$ and $N = 257^2$. A signal comprised of a small number of sinusoids was measured and reconstructed. We include the two chirp sensing code algorithms: using a single shift as well as using multiple shifts. We see that when all shifts are utilized, the chirp sensing algorithm is able to outperform matching pursuit, successfully reconstructing signals containing more sinusoids. In fact, this performance is achieved with fewer than the complete set of shifts.

2.8.2 Detection in Noise

Strictly speaking, $s$ is sparse if it has very few non-zero elements. However, this is not a good model of practical signals. Practical signals will have small values in all
elements of $s$ either due to noise or components that can be discarded. A practical recovery algorithm must be able to work under these circumstances. Further, it is important to know at which noise levels the algorithm can operate.

Figure 2.4 compares the performance of the chirped sensing code algorithm with matching pursuit. In the figure, we examine the detection of a single sinusoid in noise. We consider a correct detection if the first “peel” of the signal corresponds to the sinusoid. The figure was generated by simulation using $M = 41$ measurements and length $N = 41^2$ signals. Probabilities were estimated using 1000 samples. Both the single shift $O(KM \log M)$ algorithm and the $O(KM^2 \log M)$ algorithm using all shifts are included in the comparison.

We see that the algorithm that uses all the shifts achieves the performance of matching pursuit. These results can be compared to those in [14].

2.9 Conclusions and Extensions

The chirped sensing codes we introduced here are an illustration of how, by particularly selecting measurements in $\Phi$, we can utilize a computationally efficient recon-
struction scheme. The choice of the measurements in $\Phi$ were made such that from $\Theta$, its form in the sparse basis, we can recover small linear combinations of columns. In particular, we designed a $\Theta$ filled with columns of chirps since we have an efficient method to recover chirp rates and frequencies from a small superposition. Further, with this design of $\Theta$, $\Phi$ has a convenient form in the case of sparse Fourier signals.

Unlike most compressed sensing literature, we used deterministic measurements. By finding bound on restricted isometry constants $\delta_K$, we can provide guarantees on recoverability from our designed $\Theta$. In addition, we considered a modified version of the RIP which regards the signal, rather than the measurements, stochastically. Empirical evidence showed that the majority of Gram matrices of $\Theta$ have eigenvalues closer to 1 than correspondingly sized Gram matrices of random Gaussian measurements. This, in turn showed that a signal recoverable from the random measurements is very likely recoverable from measurements made with $\Theta$.

Signal recovery from our measurements was also shown by the implementation of our decoding algorithm. The recovery exploited the efficiency of the FFT in each of two steps: the first to recover the chirp rates and second to recover the chirp frequency. By identifying the chirp rates and frequencies, the superimposed columns
of $\Theta$ are determined. In simulation the algorithm was shown to equal the performance of matching pursuit in noise resilience and exceed matching pursuit’s performance in signal sparsity requirements.

A limitation of the chirp code algorithm is the restriction $M \geq \sqrt{N}$. This derives from the size of the family of length $M$ chirps which necessitates $\Theta$ be $M \times M^2$ or narrower. As a result, this limits the algorithm’s abilities in situations where $M$ must be small. Stemming from this work, a similar algorithm based on second-order Reed-Muller codes found in [15] addresses this. Second-order Reed-Muller codes can be viewed analogously to chirps and decoding can be done using the Fast Hadamard Transform in place of the FFTs. The class of length $M$ second-order Reed-Muller codes is very large, essentially removing this lower bound on the number of measurements.

2.A Proof of Lemma 2.2

This proof of the lemma is from [13].

The total number of choices for $I$ is

$$M^2(M^2 - 1) \cdots (M^2 - (K - 1)) = \frac{(M^2)!}{(M^2 - K)!},$$

while the number of choices of $I$ with distinct chirp-rate $r$-coordinates is

$$M^2(M^2 - M) \cdots (M^2 - M(K - 1)) = M^K \frac{M!}{(M - K)!}.$$

Combining these, our task is to show that the following grows to 1 polynomially:

$$\Pr\left(\{r_k\}_{k=1}^K \text{ distinct}\right) = M^K \frac{M!}{(M - K)!} \frac{(M^2 - K)!}{(M^2)!}.$$
Applying Stirling’s approximation \( n! = \sqrt{2\pi n} \left( \frac{n}{e} \right)^n (1 + O(\frac{1}{n})) \) gives

\[
\Pr\left( \{r_k\}_{k=1}^K \text{ distinct} \right) \geq M^K \frac{\sqrt{2\pi M} \left( \frac{M}{e} \right)^M (1 - C\sqrt{M})}{\sqrt{2\pi} (M-K)(M-K)(1+\frac{C}{M-K})} \cdot \frac{\sqrt{2\pi (M^2-K)} \left( \frac{M^2-K}{e} \right)^{M^2-K} (1 - \frac{C}{M^2-K})}{\sqrt{2\pi M^2} \left( \frac{M^2}{e} \right)^{M^2} (1+\frac{C}{M^2})}.
\]

(2.38)

The product of the first two factors of (2.38) is

\[
\frac{\sqrt{M} - C}{\sqrt{M}} \cdot \frac{\sqrt{M^2 - K} - C}{\sqrt{M^2 - K}} = \frac{\sqrt{M^3 - MK} + \text{l.o.t.}}{\sqrt{M^3 - M^2 K} + \text{l.o.t.}},
\]

and so for sufficiently large \( M \), this product is \( \geq 1 \). We can also bound the last factor of (2.38):

\[
\frac{(M^2-K)^{M^2-K}}{(M^2)^M} = \frac{1}{(M^2)^K} \left( \frac{M^2-K}{M^2} \right)^{M^2-K} = \frac{1}{M^{2K}} (1 - K/M^2)^{M^2-K} \geq \frac{1}{M^{2K}} (1 - K/M^2)^2 \quad (2.39)
\]

To further bound (2.39), we claim that \( (1 + \frac{x}{n})^n \geq e^x (1 - \frac{x^2}{n}) \); we will check this in the case where \( y := \frac{x}{n} \) is sufficiently small, since this is all we need. Indeed, defining \( g(y) := (1 + y)^n - e^n (1 - ny^2) \), it is easy to verify that \( g(0) = g'(0) = 0 \), while \( g''(0) = n \), meaning \( g(y) \geq 0 \) for sufficiently small \( y \). Applying this to (2.39), we have the following bound for sufficiently large \( M \):

\[
\frac{(M^2-K)^{M^2-K}}{(M^2)^M} \geq \frac{1}{M^{2K}} (1 - K/M^2)^{M^2-K} \geq \frac{1}{M^{2K}} e^{-K} (1 - K^2/M^4).
\]

Continuing (2.38), we therefore have

\[
\Pr\left( \{r_k\}_{k=1}^K \text{ distinct} \right) \geq M^K \frac{M^M}{(M-K)^M-K} \cdot \frac{1}{M^{2K}} e^{-K} (1 - K^2/M^4).
\]
for sufficiently large $M$. Finally, since

$$M^K \frac{M^M}{(M-K)^{M-K}} \cdot \frac{1}{M^{2K}} = \frac{1}{(1 - \frac{K}{M})^{M-K}} \geq \frac{1}{e^{-\frac{K}{M}(M-K)}} = e^{K - \frac{K^2}{M}} \geq e^K (1 - \frac{K^2}{M}),$$

we have our desired high-probability bound:

$$\Pr\left(\{r_k\}_{k=1}^K \text{ distinct}\right) \geq \left(1 - \frac{K^2}{M}\right) \left(1 - \frac{K^2}{M^2}\right) \geq 1 - 2\frac{K^2}{M} \geq 1 - \frac{2}{M^2}.$$ 

### 2.B Proof of Theorem 2.4

To prove Theorem 2.4, we begin with equation (2.32) given by

$$M(t') = \sum_{s \in W} |\mathcal{F}\{f_s\}(t')^2|$$

$$= \sum_{s \in W} \mathcal{F}\{f_s\}(t') (\mathcal{F}\{f_s\}(t'))^*$$

(2.40)

We would like to calculate the $M(t')$. To aid in calculation, we will fix an arbitrary $t'$ and rewrite (2.19) as

$$\mathcal{F}\{f_s\}(t') = \sum_{(x,y) \in I} F_s((x,y), (x', y'))$$

(2.41)

where for $(x, y) = (x', y')$

$$F_s((x, y), (x, y)) = \sqrt{M} |\beta_{x,y}|^2 v_{x,y}(s) \delta(t' - y)$$

(2.42)
and for \( (x, y) \neq (x', y') \)

\[
F_s((x, y), (x', y')) = \beta_{x,y} \beta^*_{x',y'} \alpha_{y-y'} v_{0, -(y-y')^{-1}}(x - x') \\
\times v_{x-(x-x')y(y-y')^{-1}, y-y^2(y-y')^{-1}}(s) \\
\times v_{s(x-x'+sy)(y-y')^{-1}, -(s^2(y-y')^{-1})}(t')
\]  (2.43)

Using this substitution we can write

\[
|\mathcal{F}\{f_s\}(t's)|^2 = \sum_{(x,y)\in I} \sum_{(p,q)\in I} F_s((x, y), (x', y')) F_s^*((p, q), (p', q'))
\]  (2.44)

We calculate the summand for several cases, depending on relations between \( x, y, x', y', p, q, p', q' \), which cover all the terms of the sum. We use the fact that \( I \) consists of distinct chirp rates (justified by Lemma 2.2) to restrict the set of cases required.

For each of the cases, we consider the contribution to (2.32) at two FFT bins. We consider the contribution to \( M(r) \) for some \((l, r)\in I\) and \( M(t') \) for some \( t' \) not associated with any active element in \( I \).

**Case 1:** \( (x, y) = (x', y') = (p, q) = (p', q') \)

There are \( K = |I| \) summands in (2.44) that fall in this category. In this case, the sum over the four pairs collapse to a single pair.

\[
F_s((x, y), (x, y)) F_s^*((x, y), (x, y)) = M|\beta_{x,y}|^4 \delta(t' - y)
\]  (2.45)

The contribution to (2.32) is

\[
M^{(1)}(r) = |W|M|\beta_{l,r}|^4
\]  (2.46)
for some \((l, r) \in \mathcal{I}\) and \(M^{(1)}(t') = 0\) for any frequency \(t'\) not associated with an active chirp.

**Case 2:** \((x, y) = (x', y'), (p, q) = (p', q')\) with \((x, y) \neq (p, q)\)

There are \(K(K - 1)\) terms in (2.44) which fall in this category. Under the assumption of distinct chirp rates in \(\mathcal{I}\) this assumption \(\delta(t' - y)\delta(t' - q) = 0\). Thus, for these summands we have that \(F_s((x, y), (x, y))F^*_s((p, q), (p, q)) = 0\).

**Case 3:** \((x, y) = (x', y'), (p, q) \neq (p', q')\)

There are \(K^2(K - 1)\) terms in (2.44) which are in this case. We, once again, make a substitution to clarify calculation. For a fixed of values \((p, q), (p', q')\) we define \(\Gamma, \epsilon, \zeta, \eta, \theta, \kappa\) such that

\[
F_s((p, q), (p', q')) = \Gamma v_{\epsilon, \zeta}(s)v_{\eta, \theta, \kappa, s^2}(t) \quad (2.47)
\]

where

\[
\Gamma = \beta_{p, q}^p\beta_{p', q'}^q\alpha_{q - q'}v_{0, (q - q')^{-1}}(p - p') \quad (2.48)
\]

\[
\epsilon = p - (p - p')q(q - q')^{-1} \quad (2.49)
\]

\[
\zeta = q - q^2(q - q')^{-1} \quad (2.50)
\]

\[
\eta = (p - p')(q - q')^{-1} \quad (2.51)
\]

\[
\theta = q(q - q')^{-1} \quad (2.52)
\]

\[
\kappa = -(q - q')^{-1} \quad (2.53)
\]

\[
(2.54)
\]
With these definitions we have

\[
F_s((x, y), (x, y)) F_s^*((p, q), (p', q')) = \sqrt{M} |\beta_{x, y}|^2 v_{x, y}(s) \delta(t' - y) \\
\times \Gamma^* v_{-\epsilon, -\zeta}(s) v_{-\eta y - \theta s^2, -\kappa s^2}(t) \\
= \sqrt{M} |\beta_{x, y}|^2 \Gamma^* \delta(t' - y) \\
\times v_{x - \epsilon, y - \zeta}(s) v_{-\eta y - \theta s^2, -\kappa s^2}(y) \\
= \sqrt{M} |\beta_{x, y}|^2 \Gamma^* \delta(t' - y) \\
\times v_{x - \epsilon, y - \zeta}(s) v_{-\eta y - \theta y^2, -\kappa y^2}(s)
\]

where in the second equality we apply the change of variable identity (2.15). A calculation of the subscripts gives

\[
x - \epsilon - \eta y = (x - p) - (y - q)(p - p')(q - q')^{-1} \quad (2.56)
\]
\[
y - \zeta - 2\theta y - \kappa y^2 = (y - q) [1 - (q - q')^{-1}(q - y)] \quad (2.57)
\]

To address trivial chirps, we note that (2.56) and (2.57) are simultaneously zero only when \((p, q) = (x, y)\) or \((p', q') = (x, y)\). We deal with these special cases separately below.

Returning to (2.32) we find that, due to the \(\delta(\cdot)\) functions, the contribution in FFT bin \(t'\) not corresponding to any active chirp is \(M^{(3)}(t') = 0\). For \((l, r) \in \mathcal{I}\), we have

\[
|M^{(3)}(r)| \leq 2\sqrt{M} |\beta_{l, r}|^3 \sum_{(p', q') \in \mathcal{I}} \sum_{(p, q) \in \mathcal{I}} |\beta_{p', q'}| \sum_{s \in \mathcal{W}} 1 \\
+ \sqrt{M} |\beta_{l, r}|^2 \sum_{(p, q) \in \mathcal{I}} \sum_{(p', q') \in \mathcal{I}} |\beta_{p, q}| |\beta_{p', q'}| \sum_{s \in \mathcal{W}} v(s)
\]  

(2.58)
with an application of the triangle inequality. The first summation corresponds to the two special cases of \((p, q) = (l, r)\) and \((p', q') = (l, r)\) which result in a trivial chirp. In the second sum, we have removed the subscripts of \(v(s)\) for clarity. We do this since we only need the fact that both subscripts are not simultaneously zero. Later, we will bound this type of sum. The pedantic reader can read ahead or trust that our bound is not a function of \(p, q, p', q'\).

Completing the summations to write the sums in terms of empirical average \(\text{AVG}[|\beta|]\) gives

\[
|M^{(3)}(r)| \leq 2\sqrt{M} |\beta_{l,r}|^3 |W| K \text{AVG}[|\beta|] + \sqrt{M} |\beta_{l,r}|^2 \left| \sum_{s \in W} v(s) \right| K^2 \text{AVG}^2[|\beta|] \tag{2.59}
\]

**Remark 2.1.** By taking into account the missing terms when completing the sums, we can write a tighter bound in terms of the moments \(\mu_i(|\beta|)\) of \(|\beta|\) which give

\[
|M^{(3)}(r)| \leq 2\sqrt{M} |\beta_{l,r}|^3 |W| K \mu_1(|\beta|) - 2\sqrt{M} |W| |\beta_{l,r}|^4
+ \sqrt{M} |\beta_{l,r}|^2 \left| \sum_{s \in W} v(s) \right| \left( K^2 \mu_1^2(|\beta|) + 2 |\beta_{l,r}|^2 - 2K |\beta_{l,r}| \mu_1(|\beta|) - K \mu_2(|\beta|) \right).
\]

However, we will be using (2.59) to prove the result.

**Case 4:** \((x, y) \neq (x', y'), (p, q) = (p', q')\)

There are \(K^2(K - 1)\) terms in (2.44) which are in this case. With a simple change of variables, the sum over this case is the complex conjugate of the above case. As a result \(M^{(4)}(t') = 0\) for any \(t'\) not corresponding to an active chirp and, for \((l, r) \in \mathcal{I}\), \(M^{(4)}(r)\) obeys the same bound as (2.59).
**Case 5:** $(x, y) \neq (x', y'), (p, q) \neq (p', q')$ with $(x, y, x', y') \neq (p, q, p', q')$

There are $[K(K - 1)] [K(K - 1) - 1]$ summands in (2.44) which falls in this category. Similarly to the earlier case, we make the following definitions to clarify calculations.

We write

$$F_s((x, y), (x', y')) = Gv_{a,b}(s)v_{c+ds^2,es^2}(t)$$

(2.60)

where

$$G = \beta_{x,y}\beta_{x',y'}^\alpha y - y' v_{0,(y-y')^{-1}}(x - x')$$

(2.61)

$$a = x - (x - x')y(y - y')^{-1}$$

(2.62)

$$b = y - y^2(y - y')^{-1}$$

(2.63)

$$c = (x - x')(y - y')^{-1}$$

(2.64)

$$d = y(y - y')^{-1}$$

(2.65)

$$e = -(y - y')^{-1}$$

(2.66)

and we also define $\Gamma, \epsilon, \zeta, \eta, \theta$ identically to the earlier case. Using these definitions we have

$$F_s((x, y), (x', y'))F_s^*((p, q), (p', q')) = G\Gamma^{*}v_{a-\epsilon,b-\zeta}(s)v_{(c-\eta)s+(d-\theta)s^2,(e-\kappa)s^2}(t)$$

(2.67)

$$= G\Gamma^{*}v_{a-\epsilon+c-(c-\eta)t,b-\zeta+2(d-\theta)t+(e-\kappa)t^2}(s)$$

where we have applied the change of variables identity (2.15) in the second equation.

Calculating the subscripts gives

$$a - \epsilon + (c - \eta)t$$

$$= (x - p) - (x - x')y(y - y')^{-1} + (p - p')q(q - q')^{-1}$$

$$+ t \left[ (x - x')(y - y')^{-1} - (p - p')(q - q')^{-1} \right]$$

(2.68)
\[ b - \zeta + 2(d - \theta)t + (e - \kappa)t^2 = (y - q) - y^2(y - y')^{-1} + q^2(q - q')^{-1} + 2t \left[ y(y - y')^{-1} - q(q - q')^{-1} \right] + t^2 \left[ (q - q')^{-1} - (y - y')^{-1} \right] \tag{2.69} \]

Equations (2.68) and (2.69) form Condition 2.1 on \( I \) in Theorem 2.4. Let \( J_1 \subset I \times I \times I \times I \) be the tuples that define this case. Further, let \( J_2 \subset J_1 \) be the set of solutions simultaneously satisfying (2.68) and (2.69). By assumption, \( |J_2| \leq c_1K^2 \).

Returning to (2.32), the contribution from this case is bounded by

\[
|M^{(5)}(t')| \leq |W| \left| \sum_{J_2} |\beta_{x,y}| |\beta_{x',y'}| |\beta_{p,q}| |\beta_{p',q'}| \right| + \left| \sum_{s \in W} v(s) \sum_{J_1 \setminus J_2} |\beta_{x,y}| |\beta_{x',y'}| |\beta_{p,q}| |\beta_{p',q'}| \right|
\leq c_1K^2|W| \max_{(x,y) \in I} |\beta_{x,y}|^4 + \left| \sum_{s \in W} v(s) \right| K^4 \text{AVG}^4(\beta) \tag{2.70} \]

where the first sum separates the trivial chirps. The second inequality bounds the product in the first sum using the max while the second sum is bounded by completing the sum from \( J_1 \setminus J_2 \) to \( I \times I \times I \times I \). For an active chirp \((l, r) \in I\), \( |M^{(5)}(r)| \) also obeys this bound.

**Case 6:** \((x, y) \neq (x', y'), (p, q) \neq (p', q')\) with \((x, y, x', y') = (p, q, p', q')\)

In this simple case,

\[ |F_s((x, y), (x', y'))|^2 = |\beta_{x,y}|^2 |\beta_{x',y'}|^2 \tag{2.71} \]

There are \( K(K - 1) \) such summands in (2.44) and \( M^{(6)}(t') = M^{(6)}(r) \).
Combining the contributions from each of the cases we have that for an active chirp \((l, r) \in \mathcal{I}\),
\[
M(r) \geq M^{(1)}(r) - |M^{(3)}(r)| - |M^{(4)}(r)| - |M^{(6)}(r)| + M^{(6)}(r)
\]
while for a \(t'\) not associated with an active chirp,
\[
M(t') \leq |M^{(5)}(t')| + M^{(6)}(t').
\]
Thus, \(M(r) \geq M(t')\) is guaranteed when
\[
M^{(1)}(r) \geq |M^{(3)}(r)| + |M^{(4)}(r)| + |M^{(5)}(r)| + |M^{(5)}(t')| \tag{2.72}
\]
where we have used that \(M^{(6)}(r) = M^{(6)}(t')\).

**Lemma 2.3.** For \(v(s)\) a non-trivial chirp and for \(\mathcal{W} = \mathbb{Z}_M \setminus \{0\}\),
\[
\left| \sum_{s \in \mathcal{W}} v(s) \right| \leq \sqrt{M} - 1.
\]

**Proof.** Consider the \(\mathcal{F}\{v\}(0)\). If \(v(s)\) is of the form \(v_{a,0}, \ a \neq 0\), it is a pure non-trivial tone and thus \(\mathcal{F}\{v\}(0) = 0\). Otherwise, \(v(s)\) has a non-trivial chirp rate and \(|\mathcal{F}\{v\}(0)| = 1 [9, Proposition 5]\).

Thus, using the definition of the unitary Fourier transform we have
\[
\sum_{s \in \mathcal{W}} v(s) + 1 \leq \sqrt{M}.
\]

\(\square\)
Taking $\mathcal{W} = Z_M \setminus \{0\}$ and applying (2.46), (2.59), (2.70), Lemma 2.3 and assuming our $\beta_{l,r}$ of interest is the largest coefficient in $\beta$, we can say (2.72) is satisfied when

$$M^2|\beta_{l,r}|^4 - M|\beta_{l,r}|^4 \geq 4M^{3/2}K|\beta_{l,r}|^3 \text{AVG}||\beta||$$

$$+ 2MK^2 \left(||\beta_{l,r}||^2 \text{AVG}^2||\beta|| + ||\beta_{l,r}||^4\right) + 2\sqrt{M}K^4 \text{AVG}^4||\beta||$$

which, upon dividing by $|\beta_{l,r}|^4$ yields

$$M^2 - M \geq 4M^{3/2}K\text{LAR}^{-1}||\beta||$$

$$+ 2MK^2 \left(\text{LAR}^{-2}||\beta|| + 1\right) + 2\sqrt{M}K^4\text{LAR}^{-4}||\beta|| \quad (2.73)$$

Since the last term is dominant, we can say there exists a constant $c$, dependent only on $\text{LAR}||\beta||$, such that $cM^2 \geq \sqrt{M}K^4$ implies (2.73) is satisfied. This completes the proof.
Chapter 3

Compressed OFDM Channel Sensing

This chapter examines the problem of multipath channel estimation in single-antenna orthogonal frequency division multiplexing (OFDM) systems. In particular, we study the problem of pilot assisted channel estimation in wideband OFDM systems, where the time-domain (discrete) channel is approximately sparse. Existing works on this topic established that techniques from the compressed sensing literature can yield accurate channel estimates using a relatively small number of pilot tones, provided the pilots are selected \textit{randomly}. Here, we describe a general purpose procedure for \textit{deterministic} selection of pilot tones to be used for channel estimation, and establish guarantees for channel estimation accuracy using these sequences along with recovery techniques from the compressed sensing literature. Simulation results are presented to demonstrate the effectiveness of the proposed procedure in practice.

3.1 Introduction

Two key metrics used to evaluate the performance of wireless systems include: (i) the bit-error rate (BER) and (ii) the spectral efficiency (i.e., bits transmitted per second
per Hz). It is generally recognized that the BER in wideband wireless systems can be significantly reduced if the receiver has knowledge of the underlying multipath channel response; the so-called coherent communications [16]. In practice, however, the channel response is seldom (if ever) known to the receiver. Instead, it needs to be periodically estimated at the receiver in order to reap the benefits of coherent communications.

Our focus in this chapter is on multipath channel estimation for single-antenna orthogonal frequency-division multiplexing (OFDM) wireless systems [17]. There are two classes of methods that are commonly employed for channel estimation in OFDM systems, namely, training-based methods and blind methods. Blind methods attempt to estimate the channel by making use of the statistics of the unknown data only. Therefore, blind channel estimation has the potential to yield a lower BER without affecting the systems’s spectral efficiency. Blind methods, however, tend to be effective only if the underlying multipath channel remains constant over a large number of OFDM symbols. This is clearly a disadvantage in the case of a mobile system where the underlying channel can change from one symbol to the next symbol.

Training-based methods, on the other hand, try to estimate the channel by transmitting the unknown data multiplexed with some training data already known to the receiver. Such methods are preferred for channel estimation in mobile OFDM systems since they yield reliable estimates even if the channel changes from one symbol to the next symbol. The most prevalent form of training-based channel estimation in OFDM systems involves dedicating a few of the OFDM subcarriers (tones) solely for transmitting the training data (pilots) [18]. The key questions that arise in such pilot-assisted channel estimation (PACE) methods include: (i) which, and how many, OFDM tones should be used as pilot tones? and (ii) how does the choice of pilot tones affect the channel estimation error? Note that the former question directly impacts
the spectral efficiency, while the latter question directly impacts the BER in wireless systems.

Rinne and Renfors [19] and Negi and Cioffi [20] made some of the first attempts to concretely answer the above questions for PACE methods in OFDM systems. In particular, it has been argued in [20] that: (i) the best set of pilot tones corresponds to a set of cardinality equal to the length, $L$, of the underlying (discrete) multipath channel, with the pilot tones equally spaced within the OFDM subcarriers, and (ii) this set of equally spaced pilot tones results in a channel estimation error of $\frac{L\sigma^2}{E_{tr}}$, where $\sigma^2$ denotes the receiver noise variance and $E_{tr}$ denotes the training energy. The channel estimation results of [20] are based on the maximum likelihood (ML) criterion and are in fact optimal for narrowband OFDM systems. Many wideband OFDM systems of recent interest, such as underwater acoustic systems [21], digital television systems [22] and residential ultrawideband systems [23], however, correspond to underlying (discrete) multipath channels in which a large number of time-domain channel coefficients tend to have very small magnitudes; see [24,25] for a mathematical justification of this phenomenon from the channel modeling perspective.

Conventional linear PACE methods based on the ML criterion fail to capitalize on this anticipated structure (i.e., approximate sparsity or compressibility) of the underlying multipath channels in wideband OFDM systems. In contrast, the main contribution of this chapter is a nonlinear PACE scheme that makes use of a deterministically chosen set of pilot tones having cardinality much smaller than $L$ and a convex optimization-based reconstruction method, known as the Dantzig selector [26], to result in a channel estimation error that is significantly smaller than that achievable using the traditional PACE techniques. Stated differently, the PACE scheme proposed here is significantly superior to the ones proposed in [19,20] in terms of both the spectral efficiency (number of pilot tones is smaller than $L$) and the BER (channel estimation error is smaller than $\frac{L\sigma^2}{E_{tr}}$) in wideband OFDM systems.
In terms of relationship to previous work, note that the results reported in this chapter leverage some of the recent advances in the field of compressed sensing (CS) [27]. This makes our approach to PACE in OFDM systems somewhat similar to the ones proposed independently in [28–30]. However, note that [28–30] require using randomly chosen sets of pilot tones in order to provide concrete guarantees for finite training energy. In contrast, to the best of our knowledge this is the first chapter in the literature that provides rigorous guarantees for CS-based PACE methods for OFDM systems using deterministically chosen sets of pilot tones (deterministic probe designs for multiuser multi-antenna OFDM systems were recently examined in [31], for systems employing linear least-squares channel estimation).

Finally, note that recently Schniter [32] has also proposed a novel channel estimation scheme for wideband OFDM systems. The scheme proposed in [32] can be termed as semi-blind since it makes use of both the training data and the statistics of the unknown data to carry out joint channel estimation and data decoding using belief-propagation ideas. The primary difference between [32] and our work is that [32] assumes the underlying multipath channel to have no more than $S \ll L$ nonzero time-domain channel coefficients (i.e., $S$-sparse channels), whereas we assume the multipath channel to be approximately sparse (i.e., compressible) but not necessarily exactly sparse.

### 3.2 System Model

In this section, we describe the problem setup and accompanying assumptions for PACE in wideband OFDM systems. For the sake of this exposition, we restrict ourselves to the canonical discrete channel and system model; we refer the reader to [25] for the relationship between the discrete-time mathematical model and the continuous-time physical setup (see also the simulation setup and (3.9) in Section 3.6).
To begin, we assume that the transmitter communicates with the receiver over a discrete multipath channel of length $L$, $h = [h_0 \ h_1 \ ... \ h_{L-1}]^T \in \mathbb{C}^L$, that remains fixed for a period of $N+L$ with $N \gg L$. The main assumption that we make here concerns the structure of $h$ in wideband wireless systems. Specifically, we assume that the $j$-th largest (in magnitude) entry, $h_{(j)}$, of $h$ obeys

$$|h_{(j)}| \leq B \cdot j^{-\alpha - 1/2} \quad (3.1)$$

for some $B > 0$ and $\alpha > 0$. The parameter $\alpha$ here controls the rate of decay of the magnitudes of the ordered entries of $h$ and we term any $h$ that satisfies (3.1) as $\alpha$-compressible.

Next, we assume that the (unknown and training) data is transmitted using an OFDM symbol that consists of a total of $N$ subcarriers and has an $L$-length cyclic prefix. Using $d = [d_0 \ d_1 \ ... \ d_{N-1}]^T \in \mathbb{C}^N$ in this case to denote the data transmitted over each of the $N$ OFDM tones, it can then be easily shown that the received data vector $y \in \mathbb{C}^N$ at the receiver is related to the transmitted data vector $d$ as follows [16]:

$$y = Hd + w. \quad (3.2)$$

Here, $w \in \mathbb{C}^N$ represents a zero-mean additive white Gaussian noise (AWGN) vector of variance $\sigma^2$, while $H \in \mathbb{C}^{N \times N}$ is a diagonal matrix comprising of the $N$ OFDM channel coefficients that are related to $h$ as $\{H_{kk} = \sum_{\ell=0}^{L-1} F_{k\ell}h_{\ell}\}$ with $F_{k\ell} = e^{-j \frac{2\pi}{N} k\ell}$ denoting the $(k, \ell)$th element of the $N$-point discrete Fourier transform (DFT) matrix.

In order to carry out PACE under this setup, we can now proceed as follows. First, we select a set of indices $\mathcal{P} \subset \{0, 1, \ldots, N-1\}$, corresponding to the set of pilot tones, of cardinality $N_{tr} = |\mathcal{P}|$. Next, we construct a training data vector $d_{tr} \in \mathbb{C}^{N_{tr}}$ having energy $\|d_{tr}\|^2_2 = \mathcal{E}_{tr}$ and transmit this vector using the pilot tones specified by $\mathcal{P}$; in
other words, we have that \( d_{|P} = d_{tr} \), where \( d_{|P} \) denotes the restriction of \( d \) to the indices in \( P \). Then defining \( y_{tr} = y_{|P} \), it is easy to see because of the diagonal nature of the OFDM channel matrix \( H \) [cf. (3.2)] that

\[
y_{tr} = H_{|P \times P} d_{tr} + w_{|P}
\]

(3.3)

where \( H_{|P \times P} \) denotes the restriction of \( H \) to the ordered pairs in \( P \times P \). Further, (3.3) can be easily expressed in terms of the (time-domain) multipath channel \( h \) by noting that

\[
H_{|P \times P} d_{tr} = D_{tr} A h
\]

where \( D_{tr} = \text{diag}(d_{tr}) \) and \( A \) is an \( N_{tr} \times L \) matrix that comprises \( \{ [F_{p0} F_{p1} \ldots F_{p(L-1)}] : p \in P \} \) as its rows. Therefore, defining \( X = D_{tr} A \), the goal of any PACE scheme in OFDM systems is to (i) specify the pair \( (P, d_{tr}) \) such that \( N_{tr} = |P| \) and \( \|d_{tr}\|_2 = \mathcal{E}_{tr} \), and (ii) provide a reliable estimate of the underlying multipath channel \( h \) from the corresponding received training data vector

\[
y_{tr} = X h + w_{|P}.
\]

### 3.3 Useful Compressed Sensing Theory

The PACE scheme proposed in this chapter leverages existing concepts from the CS literature. In particular, we will employ a reconstruction method whose success relies upon a normalized version of the measurement matrix \( X = D_{tr} A \) described above satisfying the Restricted Isometry Property (RIP) [7]. Introduced briefly in Section 2.4, we give a more concrete definition of RIP here.

**Definition 3.1** (Restricted Isometry Property). An \( N_{tr} \times L \) matrix \( Z \) with unit norm columns satisfies the restricted isometry property of order \( S \in \mathbb{N} \) with parameter
$\delta_S \in [0,1]$—or, in shorthand, $Z$ satisfies RIP($S, \delta_S$)—if

$$(1 - \delta_S)\|v\|_2^2 \leq \|Zv\|_2^2 \leq (1 + \delta_S)\|v\|_2^2$$

holds for all $v \in \mathbb{C}^L$ having no more than $S$ nonzero entries.

In words, the RIP of order $S$ says that the matrix $Z$ acts like an near isometry on all $S$-sparse vectors $v$. For example, if $Z$ is the identity matrix (say, of dimension $L$), then it satisfies the RIP trivially for any $S = 1, 2, \ldots, L$ with $\delta_S = 0$.

The cases of particular interest in the CS literature, however, correspond to when $Z$ has fewer rows than columns. Indeed, the RIP has been widely adopted in the CS literature, and many results have been established for procedures that guarantee reliable and efficient reconstruction of sparse and compressible signals from a relatively small number of linear measurements obtained via a measurement matrix satisfying the RIP. In the following, we will leverage the results for one such reconstruction method, known as the Dantzig selector, which was proposed in [26] and which is particularly well-suited for measurements corrupted by stochastic noise. The following result is a complex-valued variant of the result originally reported in [26]; see [33, Th. 2.13] for further details.

**Lemma 3.1** (The Dantzig Selector). Let $Z$ be a measurement matrix satisfying RIP($2S, \delta_{2S}$) with $\delta_{2S} < 1/3$ for some $S \in \mathbb{N}$. Let $\gamma = Z\beta + \eta$ be a vector of noisy measurements of $\beta \in \mathbb{C}^L$, where $\eta$ is an AWGN vector with variance $\sigma^2$. Choose $\lambda_L = \sqrt{2(1 + a)\log L}$ for any $a \geq 0$. Then the estimator

$$\hat{\beta} = \arg \min_{v \in \mathbb{C}^L} \|v\|_1 \text{ subject to } \|Z^H(\gamma - Zv)\|_\infty \leq \sigma \lambda_L,$$

holds for all $v \in \mathbb{C}^L$. 

47
where the notation $Z^H$ denotes the Hermitian, or conjugate transpose, of $Z$, satisfies

$$
\|\hat{\beta} - \beta\|_2^2 \leq c_0 \min_{1 \leq k \leq S} \left( \sigma \lambda L \sqrt{k} + \frac{\|\beta_k - \beta\|_1}{\sqrt{k}} \right)^2,
$$

with probability at least \( 1 - 2 \left( \sqrt{\pi(1 + a) \log L} \cdot L^a \right)^{-1} \), where $\beta_k$ is the best $k$-term approximation of $\beta$, formed by setting all but the $k$ largest entries (in magnitude) of $\beta$ to zero, and the constant $c_0 = 16 / (1 - 3\delta_2 S)^2$.

### 3.4 Deterministic Pilot Sequence and Training Data Selection

In this section, we describe our proposed procedure for deterministic selection of the pair $(P, d_{tr})$ corresponding to the set of pilot tones and training data. Our procedure, which is based on the method outlined in [34], is quite general and in fact gives rise to a family of pilot tone/training data selection procedures, each of which is fully-described by a small number of integer-valued parameters. We will see in the following section that each selection $(P, d_{tr})$ results in a set of pilot tones and corresponding training data whose performance for estimating compressible multipath channels using the Dantzig selector can be rigorously quantified.

In order to describe our selection procedure, we first assume that the number of subcarriers, $N$, is prime. Under this condition, our procedure can be described as follows. Begin by first selecting an integer $R \geq 2$. Next, choose a set of $R$ integers denoted $\{a_i\}_{i=1}^R$ such that $a_R$ is relatively prime to $N$ (i.e., $a_R \in \{1, 2, \ldots, N-1\}$), while $a_i \in \{0, 1, 2, \ldots, N-1\}$ for the remaining $i \neq R$. The integers $\{a_i\}_{i=1}^R$ become the coefficients of a degree-$R$ polynomial $Q(m) = a_1 m + \cdots + a_R m^R$. Finally, choose an integer $M \geq 1$ and construct a multiset $\mathcal{T}$ by evaluating $Q(m) \mod N$ for integers
Procedure 3.1: Deterministic Pilot/Training Data Selection

1. Select an integer \( R \geq 2 \).

2. Choose integers \( a_R \in \{1,2,\ldots,N-1\} \) and \( a_i \in \{0,1,2,\ldots,N-1\} \) for \( i = 1,\ldots,R-1 \).

3. Construct the polynomial \( Q(m) = a_1 m + \cdots + a_R m^R \).

4. Choose an integer \( M \geq 1 \) and form the multiset of integers \( \mathcal{T} = \{Q(m) \mod N : m = 1,2,\ldots,M\} \).

5. Select the set of pilots to be the unique elements of \( \mathcal{T} \).

6. Select the training data vector entries according to \( \mathbf{d}_{tr}(p) = \sqrt{C_p \mathcal{E}_{tr}}/M \) for \( p \in \mathcal{P} \), where \( C_p \) denotes the multiplicity of each \( p \in \mathcal{P} \) in the multiset \( \mathcal{T} \).

---

\( m = 1,2,\ldots,M \). Formally, we have that \( \mathcal{T} = \{Q(m) \mod N : m = 1,2,\ldots,M\} \) with multiplicities.

Now, the set of pilot tones \( \mathcal{P} \) is the (sub)set of unique elements in \( \mathcal{T} \). Note that if each element of \( \mathcal{T} \) appears with multiplicity 1 then \( \mathcal{P} = \mathcal{T} \), otherwise \( \mathcal{P} \subset \mathcal{T} \). The entries of the training data vector \( \mathbf{d}_{tr} \) corresponding to the pilots \( p \in \mathcal{P} \), \( \{\mathbf{d}_{tr}(p)\}_{p \in \mathcal{P}} \), are functions of the multiplicity of the elements \( p \in \mathcal{P} \) in the multiset \( \mathcal{T} \). Specifically, let \( C_p \) denote the number of times the element \( p \in \mathcal{P} \) appears in \( \mathcal{T} \) and note that the cardinality of \( \mathcal{T} \) is equal to \( M \). Then, we select the training data associated with the pilot \( p \in \mathcal{P} \) as

\[
\mathbf{d}_{tr}(p) = \sqrt{C_p \mathcal{E}_{tr}}/M \tag{3.5}
\]

where \( \mathcal{E}_{tr} \) is the training data energy as described earlier. Notice that, by construction, we have \( \sum_{p \in \mathcal{P}} C_p = M \), and so the selection in (3.5) ensures \( \|\mathbf{d}_{tr}\|^2 = \sum_{p \in \mathcal{P}} d_{tr}^2(p) = \mathcal{E}_{tr} \), as required. This entire pilot sequence and training data selection procedure is summarized as Procedure 3.1.
Notice that this procedure provides considerable flexibility in selecting a pair \((\mathcal{P}, d_{tr})\), and the selection of each pair is fully-parameterized by the polynomial degree \(R (\geq 2)\), the number of polynomial evaluation points \(M\), the coefficients \(\{a_i\}_{i=1}^{R}\), and the number of subcarriers \(N\) (which is assumed to be prime). The next section establishes conditions under which these deterministic selections of \((\mathcal{P}, d_{tr})\) enable provably accurate estimation of compressible multipath channels using the Dantzig selector.

### 3.5 Main Results

In the previous section, we specified a deterministic procedure for selecting the pair \((\mathcal{P}, d_{tr})\) for PACE in OFDM systems. We now claim that Procedure 3.1 can result in a measurement matrix \(X = D_{tr}A\) that facilitates reconstruction of compressible multipath channels using the Dantzig selector. To that end, we follow the approach proposed in [34] to obtain the following lemma, which outlines conditions under which our deterministic selection of \((\mathcal{P}, d_{tr})\) corresponds to a normalized matrix \(\Psi = (\mathcal{E}_{tr})^{-1/2}X\) satisfying the RIP. The proof of this lemma is provided in the appendix.

**Lemma 3.2.** Suppose \(N > 2\) is prime, and let \((\mathcal{P}, d_{tr})\) be selected according to Procedure 3.1, with parameters \(R\) (the polynomial degree), \(M\) (the number of polynomial evaluation points), and \(\{a_i\}_{i=0}^{R}\) (the polynomial coefficients). Let

\[
\Psi = (\mathcal{E}_{tr})^{-1/2}D_{tr}A
\]

where (as above) \(D_{tr} = \text{diag}(d_{tr})\) and \(A\) is an \(N_{tr} \times L\) matrix that comprises \(
\begin{bmatrix}
F_{p0} & F_{p1} & \cdots & F_{p(L-1)}
\end{bmatrix}
\) : \(p \in \mathcal{P}\) as its rows. Choose \(R \geq 2\), and any \(\epsilon_1 \in (0, 1)\) and \(\epsilon_2 \in (0, \epsilon_1)\). There exists a constant \(c = c(N, \epsilon_2)\) such that whenever the number of evaluation points \(M\) satisfies, \(N^{1/(R-\epsilon_1)} \leq M \leq N\), the matrix \(\Psi\) satisfies
RIP($S, \delta_S$) for any $\delta_S \in (0, 1)$ and

$$S \leq c(N, \epsilon_2)\delta_S M^{(\epsilon_1-\epsilon_2)/2^{R-1}}. \tag{3.7}$$

Now, taken together with the results of Lemma 3.1, the results of Lemma 3.2 allow us to obtain the following theorem.

**Theorem 3.1.** Suppose that the time-domain multipath channel $h \in \mathbb{C}^L$ obeys (3.1). Define $y'_{tr} = (e_{tr})^{-1/2} y_{tr}$; in other words, $y'_{tr} = \Psi h + w'$, where $w'$ is an AWGN vector with variance $\text{SNR}^{-1} = \sigma^2_{e_{tr}}$.

Select $(P, d_{tr})$ according to Procedure 3.1, such that for any choice of $\epsilon_1 \in (0, 1)$ and $\epsilon_2 \in (0, \epsilon_1)$ the number of polynomial evaluation points satisfies

$$M > \max \left\{ N^{1/(R-\epsilon_1)}, \left( \frac{6}{c(N, \epsilon_2)} \right)^{2^{R-1}/1-\epsilon_2} B^{2^{R-1}(1+\epsilon_2)/(\epsilon_1-\epsilon_2)} \cdot \text{SNR}^{2^{R-1}(2\alpha+1)/(4(\epsilon_1-\epsilon_2))} \right\} \tag{3.8}$$

where $c(N, \epsilon_2)$ is the same constant as in Lemma 3.2. Finally, choose $\lambda_L = \sqrt{2(1+a)\log L}$ for any $a \geq 0$. Then the reconstruction error of the Dantzig selector channel estimate

$$\hat{h} = \arg \min_{v \in \mathbb{C}^L} \|v\|_1 \text{ subject to } \|\Psi^H (y'_{tr} - \Psi v)\|_\infty \leq \frac{\lambda_L}{\sqrt{\text{SNR}}},$$

satisfies

$$\|\hat{h} - h\|_2^2 \leq c'_0 B^{2\alpha+1} \text{SNR}^{-2\alpha/(2\alpha+1)} \log L,$$

with probability at least $1 - 2 \left( \sqrt{\pi(1+a)\log L} \cdot L^{\alpha} \right)^{-1}$, where $c'_0$ is an absolute constant.

The proof of Theorem 3.1 follows from the fact that the condition (3.8) implies that the matrix $\Psi$ described in Lemma 3.2 satisfies RIP($2S, \delta_{2S}$) with $S \geq B^{\alpha+1/2} \cdot \text{SNR}^{2\alpha+1}$.  

51
and $\delta_{2S} < 1/3$. Thus, we may apply Lemma 3.1 to obtain the stated estimation error bounds.

A few comments are in order regarding the result of the theorem. First, as our pilot tone selection procedure is entirely deterministic, the probabilistic statement is with respect to the randomness in the additive noise vector $w'$. Second, note that ignoring constants, our estimation error bounds scale like $\|\hat{h} - h\|_2^2 \lesssim \text{SNR}^{-\frac{2\alpha}{\alpha+1}} \log L$; in other words, our error guarantees exhibit only a logarithmic dependence on the channel length $L$. Compare this with the estimation error bounds using ML estimation, which are on the order of $L/\text{SNR}$. When $\alpha$ is large the error bound obtained here is $\sim \log(L)/\text{SNR}$, which may represent a significant scaling improvement over the ML-based estimation methods.

Finally, note that for our results to hold $M$ must exceed some minimum value as specified in (3.8). Without specifying a relationship between $N$ and $L$, and given that the modular arithmetic nature of Procedure 3.1 makes it difficult to precisely quantify $|\mathcal{P}|$ (though we have $|\mathcal{P}| \leq M$ trivially), the spectral efficiency of the proposed method is difficult to quantify analytically. In Section 3.6 we examine the estimation error as a function of the number of pilot tones $|\mathcal{P}|$ via simulation.

### 3.6 Numerical Experiments and Discussion

In this section, we present results from numerical experiments which illustrate and verify our results. Though our procedure deterministically selects the set of pilot tones, we use Monte Carlo experiments with random channel realizations and noise.

An important and novel feature of our results is their applicability to compressible channels (compared to the strictly sparse channels addressed in [32]). This is reflected in the model used for $h$. In the numerical experiments, we generate $h$ as the convolution of the response from $S$ point scatterers with that of a low pass filter.
This is given by

\[ h_j = \sum_{i=1}^{S} \beta_i \text{sinc}(j - W\tau_i) \quad (3.9) \]

where \( \beta_i \) and \( \tau_i \) are the coefficients and continuous-time delays of the multipath scatterers, while \( W \) is the two-sided bandwidth of the system. Since the \( \tau_i \)'s are not taken to be multiples of \( 1/W \), \( h \) is not strictly sparse. We again refer the reader to [25] for further discussion of the relationship between the discrete-time mathematical model and the continuous-time physical setup.

More particularly, for our simulations we take \( W = 25.12 \text{MHz} \) with \( S = 6 \) scatterers and \( \tau_i \) distributed uniformly on \([0, 12.7 \mu\text{sec}]\). These parameters are based on the “Brazil B” channel reported in [35]. Sampled at \( 1/W \), this results in a discrete channel length of \( L = 320 \). Further, we generate \( \beta_i \) as zero-mean Gaussian and normalize \( h \) such that \( \|h\|_2 = 1 \). A typical channel response is shown in Figure 3.1. Outside of the generation of \( h \), we use \( \sigma = 0.02\sqrt{2} \) and take the training data to have energy \( \mathcal{E}_{tr} = 1 \).

Results from our numerical experiments are shown in Figure 3.2 where we plot the mean squared-error (MSE) \( \|\hat{h} - h\|_2^2 \) of the channel estimate, averaged over 100
Monte Carlo trials, as a function of the number of pilot tones $|\mathcal{P}|$. We include results using various primes $N$ and polynomials $Q(m)$. For comparison with previous work, we also include recovery using the Dantzig selector from both equally spaced and randomly selected pilot tones.

To place the results in context, note that the classic results of [20] require $|\mathcal{P}| = L$ pilot tones to achieve an MSE of $\frac{L}{\text{SNR}} = 0.256$. Figure 3.2 shows that this MSE is achieved by our procedure using approximately 30 rather than 320 pilots while using additional pilots reduces the MSE further. Moreover, as a function of $|\mathcal{P}|$ the MSE of our procedure matches that of randomly selected tones. Random subselections of Fourier matrices, studied in the context of channel estimation in [28], are known to have near-optimal RIP guarantees. In contrast, our selection performs as well while having the advantage of being deterministically constructed.

Though the bounds of (3.7) suggest that recovery can be a function of the prime $N$ and polynomial degree $R \geq 2$, a strong dependence was not found empirically. This is apparent since the plots for various $N$ and polynomials roughly coincide. This, however, may be due to our theory providing worst-case guarantees over possible channel responses and polynomials which requires exhaustive search to validate.
empirically. The polynomials chosen for the experiments displayed in Figure 3.2 have the coefficients $a_i = 1$ for $i = 1, \ldots, R$.

An examination of the numerical experiments with evenly spaced pilots shows the utility of selecting pilots using the non-linear polynomials $Q(m)$. When $|\mathcal{P}| = L$, [20] finds the optimal set of pilots to be evenly spaced within the $N$ subcarriers. Though the estimation technique of [20] becomes ill-posed, one might expect that evenly spaced tones may also be effective when $|\mathcal{P}| < L$ and the Dantzig selector is used to estimate $h$. We test this idea numerically and find this is not the case. Figure 3.2 shows that pilot tones selected as our procedure prescribes outperform evenly spaced tones, allowing superior estimation using fewer pilots.

Finally, the MSE of our procedure can be further reduced by “debiasing” the estimate. As noted in [26], the estimate of the Dantzig selector is improved when, after the initial estimate, a least-squares step is performed to fit the data on the estimate’s support. For example, with 180 pilot tones, Figure 3.2 shows our procedure results in a MSE of 0.16. But with the additional debiasing step, the MSE is reduced to 0.10. In the interest of space, we have chosen not to display the debiased results in this exposition.

### 3.7 Conclusions

In this chapter we proposed a general purpose procedure for deterministic selection of pilot tones for estimation of approximately sparse (or compressible) multipath channels in single-antenna OFDM systems. Our approach utilized estimation techniques from the compressed sensing literature, and was based on establishing the RIP for certain deterministically subsampled discrete Fourier transform matrices, as in [34].

It is interesting to note that our pilot tone selection procedure provides considerable flexibility when selecting parameters, such as the order $R$ of the polynomial.
Q(m) and its integer coefficients \( \{a_i\}_{i=1}^R \). That said, our guarantees apply to all selections of these parameters which satisfy the conditions outlined in Procedure 3.1. It would be illustrative to perform a more comprehensive evaluation of the performance of our pilot tone selection and channel estimation procedure to determine whether there is a strong dependence on the selection of these parameters over a wide range of possible choices. We defer this to a future effort.

### 3.8 Appendix

#### 3.8.1 Proof of Lemma 3.2

Assume that a matrix \( Z \) has unit-norm columns, then the worst-case coherence \( \mu(Z) \) of \( Z \) is defined to be the largest (in magnitude) inner product between unique columns of \( Z \). Formally, if \( Z_i \) denotes the \( i \)th column of \( Z \), then

\[
\mu(Z) = \max_{i,j, i \neq j} |Z_i^H Z_j|.
\]

A general procedure for parlaying the coherence of a matrix into a statement of RIP was described, for example, in [36,37]. In particular, Geršgorin’s theorem [38] can be applied to bound the extremal eigenvalues of the Gram matrix \( G = Z^H Z \). It follows that for a specified \( \delta_S \) a matrix \( Z \) with coherence \( \mu(Z) \) (and unit-norm columns) satisfies RIP\((S, \delta_S)\) for \( S \leq \delta_S/\mu(Z) \).

Our goal, then, is to obtain an upper bound on the coherence of the matrix

\[
\Psi = (\mathcal{E}_{tr})^{-1/2} D_{tr} A,
\]

where \( D_{tr} = \text{diag}(d_{tr}) \) and \( A \) is an \( N_{tr} \times L \) matrix that comprises \( \left\{ F_{p0} \ F_{p1} \ \ldots \ F_{p(L-1)} \right\} : p \in \mathcal{P} \) as its rows. Let \( \Psi_i, i = 1, 2, \ldots, L \) denote the columns of \( \Psi \). Then, the entries of the Gram matrix \( G \) of \( \Psi \) are given by the expres-
\[ G(k, \ell) = \sum_{p \in P} \frac{C_p}{M} F_{pk} F_{p\ell} \]
\[ = M^{-1} \sum_{p \in P} C_p \exp \left( -j \frac{2\pi}{N} p(\ell - k) \right). \]

Now, notice that because of how we defined the multiplicity terms \( \{C_p\}_{p \in P} \), we can write the expression for \( G(k, \ell) \) equivalently as a sum over the indices \( m = 1, 2, \ldots, M \), in terms of the elements of the multiset \( T \). That is,

\[ G(k, \ell) = M^{-1} \sum_{m=1}^{M} \exp \left( -j \frac{2\pi}{N} (\ell - k)Q(m) \right). \]

Note that \( G(k, k) = 1 \), implying that the columns of \( \Psi \) are unit norm.

It remains to bound the coherence of \( \Psi \), which is just the largest value of \(|G(k, \ell)|\) when \( k \neq \ell \). For that, we follow the approach described in [34]. We make use of the following lemma which is credited to H. Weyl [39], and appears in its present form in [40].

**Lemma 3.3** (Weyl). Let \( R \geq 2 \), and let \( P(m) = b_1 m + \cdots + b_R m^R \), where \( b_R = \alpha/N + \theta/N^2 \), \( |\theta| \leq 1 \), and \( \gcd(\alpha, N) = 1 \). If, for \( 0 < \epsilon_2 < \epsilon_1 < 1 \), the condition \( M^{\epsilon_1} \leq N \leq M^{R-\epsilon_1} \) holds for some integer \( M \), then

\[ M^{-1} \left| \sum_{m=1}^{M} \exp \left( 2\pi j P(m) \right) \right| \leq \gamma(R, \epsilon_2) \cdot M^{(\epsilon_2-\epsilon_1)/2^{R-1}}. \]

For completeness, we note the constant \( \gamma(R, \epsilon_2) \) is given by

\[ \gamma(R, \epsilon_2) = 2 \left[ \left( \frac{64 R}{\epsilon_2} \right) \left( \frac{R^2}{\epsilon_2 \log 2} \right)^{\exp(R^2/\epsilon_2)} R! \right]^{1/2^{R-1}}. \]
Let $P(m) = (\ell - k)Q(m)/N$, then $b_R = (\ell - k)a_R$. For $\ell, k \in \{0, 1, 2, \ldots, N - 1\}$ and $\ell \neq k$, it follows that $\gcd(b_R, N) = 1$ since $a_R$ was selected to be relatively prime to $N$. If $M \geq N^{1/(R - \epsilon_1)}$ then Lemma 3.3 implies that the worst-case coherence of $\Psi$ is no more than $\gamma(R, \epsilon_2) \cdot M^{(\epsilon_2 - \epsilon_1)/2^{R-1}}$. The stated results follow from Geršgorin’s theorem.
Chapter 4

Multi-user Detection for
Asynchronous Random Access

Many applications in cellular systems and sensor networks involve a random subset of a large number of users asynchronously reporting activity to a base station. This chapter examines the problem of multiuser detection (MUD) in random access channels for such applications. Traditional orthogonal signaling ignores the random nature of user activity in this problem and limits the total number of users to be on the order of the number of signal space dimensions. Contention-based schemes, on the other hand, suffer from delays caused by colliding transmissions and the hidden node problem. In contrast, this chapter presents a novel pairing of an asynchronous non-orthogonal code-division random access scheme with a convex optimization-based MUD algorithm that overcomes the issues associated with orthogonal signaling and contention-based methods. Two key distinguishing features of the proposed MUD algorithm are that it does not require knowledge of the delay or channel state information of every user and it has polynomial-time computational complexity. The main analytical contribution of this chapter is the relationship between the performance of the proposed MUD algorithm in the presence of arbitrary or random delays and
two simple metrics of the set of user codewords. The study of these metrics is then focused on two specific sets of codewords, random binary codewords and specially constructed algebraic codewords, for asynchronous random access. The ensuing analysis confirms that the proposed scheme together with either of these two codeword sets significantly outperforms the orthogonal signaling-based random access in terms of the total number of users in the system.

4.1 Introduction

Many applications of wireless networks require servicing a large number of users that share limited communication resources. In particular, the term random access is commonly used to describe a setup where a random subset of users in the network communicate with a base station (BS) in an uncoordinated fashion [41]. In this chapter, we study random access in large networks for the case when active users transmit single bits to the BS. This so-called “on–off” random access channel (RAC) [42] represents an abstraction that arises frequently in many wireless networks. In third-generation cellular systems, for example, control channels used for scheduling requests can be modeled as on–off RACs; in this case, users requesting permissions to send data to the BS can be thought of as transmitting 1’s and inactive users can be thought of as transmitting 0’s. Similarly, uplinks in wireless sensor networks deployed for target detection can also be modeled as on–off RACs; in this case, sensors that detect a target can be made to transmit 1’s and sensors that have nothing to report can be thought of as transmitting 0’s.¹

The primary objective of the BS in on–off RACs is to reliably and efficiently carry out multiuser detection (MUD), which translates into recovery of the set of active

¹The focus of this chapter is on servicing a large number of users that share limited communication resources in the uplink. Limiting ourselves to on–off RACs in this case helps up isolate the key issues associated with designing arbitrary RACs involving (multiple-bit) packet transmissions in large networks.
users in our case. The two biggest impediments to this goal are that (i) random access tends to be asynchronous in nature, and (ii) it is quite difficult, if not impossible, for the BS to know the channel state information (CSI) of every user. Given a fixed number of temporal signal space dimensions $M$ in the uplink, the system-design goal therefore is to simultaneously maximize the total number of users $N$ in the network and the average number of active users $k$ that the BS can reliably handle without requiring knowledge of the delays or CSIs of the individual users at the BS.

Traditional approaches to random access fall significantly short of this design objective. In random access methods based on orthogonal signaling, the $M$ signal space dimensions are orthogonally spread among the $N$ users in either time, frequency, or code [41]. While this establishes a dedicated, interference-free channel between each user and the BS, this approach ignores the random nature of user activity in RACs. Therefore, by its very structure, random access based on orthogonal signaling dictates the relationship $k \leq N \leq M$. On the other hand, contention-based random access schemes such as ALOHA and carrier sense multiple access (CSMA) do take advantage of the random user activity [43]. However, significant problems arise in these schemes when the average number of active users $k$ and/or the total number of users $N$ gets large [43]. In the case of ALOHA, collisions and retransmissions accumulate to significant delays as $k$ becomes large. In the case of CSMA, the number of potential “hidden nodes” grows as $N$ increases, resulting in unintended and unrecognized collisions in large networks.

Cellular systems, partly because of the aforementioned reasons, typically resort to the use of matched filter receivers on uplink control channels. Such receivers correspond to single-user detection (SUD) since they detect each user independently, treating the interference from other active users as noise. However, despite the effectiveness of these receivers in today’s cellular systems, SUD schemes also have significant pitfalls. In particular, such schemes tend to have suboptimal performance since
they do not carry out joint detection and they tend to be prone to the “near–far” effect [42].

In order to overcome the issues associated with orthogonal signaling, contention-based methods and SUD schemes, we present in this chapter a novel code-division random access (CDRA) scheme that spreads the uplink communication resources in a non-orthogonal manner among the $N$ users and leverages the random user activity to service significantly more total users than $M$. A key distinguishing feature of the proposed scheme is that it makes use of a convex optimization-based MUD algorithm that does not require knowledge of the delays or CSIs of the users at the BS. In addition, we present an efficient implementation of the proposed algorithm based on the fast Fourier transform (FFT) that ensures that its computational complexity at worst differs by a logarithmic factor from an oracle-based MUD algorithm that has perfect knowledge of the user delays. Our main analytical contribution is the relationship between the probability of error $P_{err}$ of the proposed MUD algorithm in the presence of arbitrary or random delays and two metrics of the set of codewords assigned to the users. We then make use of these metrics to analyze two specific sets of codewords, random binary codewords and specially constructed (deterministic) algebraic codewords, for the proposed random access scheme. Specifically, we show that both these codewords allow our scheme to successfully manage an average number of active users that is almost linear in $M$: $k \lesssim M/(\tau \log (N\tau))$ for arbitrary delays and $k \lesssim M/\log (N\tau)$ for uniformly random delays, where $\tau$ denotes maximum delay in the network. More importantly, we show that the set of random codewords enable our scheme to service a number of total users that (ignoring $\tau$) is super-polynomial in $M$, $N \lesssim \exp(O(M^{1/3}))$, while the set of deterministic codes, which facilitate efficient codeword construction and storage, enable it to service a number of total users that is polynomial in $M$, $N \lesssim M^t$ for any reasonably sized $t \geq 2$.\footnote{Recall the “Big–O” notation: $f(n) = O(g(n))$ (alternatively, $f(n) \lesssim g(n)$) if $\exists c_o > 0, n_o : \forall n \geq n_o, f(n) \leq c_o g(n)$.
}
It is useful at this point to also consider non-orthogonal code-division multiple access (CDMA), which—like our scheme—also spreads the uplink communication resources in a non-orthogonal manner among $N > M$ users [41]. However, despite similarities at the codeword-assignment level, there are significant differences between non-orthogonal CDMA and the work presented here. First, non-orthogonal CDMA is used for applications in which a fixed set of users continually communicate with the BS, whereas our scheme corresponds to a random subset of users in a large network communicating single bits to the BS. Second, MUD schemes for non-orthogonal CDMA require that the BS has knowledge of the individual user delays, whereas we assume—partly because of the random user activity—that user delays are unknown at the BS.

In terms of related prior work, Fletcher et al. [42] have also recently studied the problem of MUD in on–off RACs. However, the results in [42]—while similar in spirit to the ones in here—are limited by the facts that [42]: (i) assumes perfect synchronization among the $N$ users, which is hard to guarantee in practical settings for large $N$; (ii) assumes that CSIs of the individual users are available to the BS in certain cases, which is difficult—if not impossible—to justify for the case of fading RACs; and (iii) only guarantees that the probability of error $P_{err}$ at the BS goes to zero asymptotically in $N$, which does not shed light on the scaling of $P_{err}$. More recently, we have become aware of the independent and simultaneous work in [44] and [45] that also considers on–off RACs in the context of configuration in ad-hoc wireless networks. However, [44] and [45] also make a synchronization assumption similar to [42]. Finally, the work presented here also has implications in the area of sparse signal recovery, and it relates to some recent work in model selection and compressed sensing [46, 47]. We defer a detailed discussion of these implications and relationships to later parts of the chapter.
The remainder of the chapter is organized as follows. In Section 4.2, we introduce our system model and accompanying assumptions. In Section 4.3, we describe our approach to MUD for asynchronous (non-orthogonal) CDRA and specify its performance for both arbitrary and random delays in terms of two metrics of the set of user codewords. In Section 4.4, we specialize the results of Section 4.3 to random binary codewords and specially constructed algebraic codewords. We finally conclude in Section 4.5 by reporting results of some numerical experiments and discussing connections of our work in the area of sparse signal recovery.

4.2 System Model

In this section, we formalize the problem of MUD in asynchronous on–off RACs by introducing our system model and accompanying assumptions. To begin, we assume that there are a total of \( N \) users in the network that communicate with the BS using waveforms of duration \( T \) and (two-sided) bandwidth \( W \); in other words, the total number of temporal signal space dimensions (degrees of freedom) in the uplink are \( M = TW \). In this chapter, we propose that users communicate using spread spectrum waveforms:

\[
x_i(t) = \sqrt{E_i} \sum_{n=0}^{M-1} x^i_n g(t - nT_c), \quad t \in [0, T),
\]

(4.1)

where \( g(t) \) is a unit-energy prototype pulse \((\int |g(t)|^2 dt = 1)\), \( T_c \approx \frac{1}{W} \) is the chip duration, \( E_i \) denotes the transmit power of the \( i \)-th user, and

\[
x_i = \begin{bmatrix} x^i_0 & x^i_1 & \ldots & x^i_{M-1} \end{bmatrix}^T, \quad i = 1, \ldots, N
\]

(4.2)

is the \( M \)-length real-valued codeword of unit energy \((\|x_i\|_2 = 1)\) assigned to the \( i \)-th user.
In the context of on–off RACs, we assume that on average a total of \(k\) of the \(N\) users transmit 1’s at time \(t = 0\) (without loss of generality), resulting in the following received signal at the BS

\[
y(t) = \sum_{i=1}^{N} h_i \delta_i x_i(t - \tau_i) + w(t).
\]

(4.3)

Here, \(h_i \in \mathbb{R}\) and \(\tau_i \in \mathbb{R}_+\) are the channel fading coefficient\(^3\) and the delay\(^4\) associated with the \(i\)-th user, respectively, \(w(t)\) is additive white Gaussian noise (AWGN) introduced by the receiver circuitry, and \(\{\delta_i\}\) are independent 0–1 Bernoulli random variables that model the random activation of the \(N\) users in the sense that \(\mathbb{P}(\delta_i = 1) = k/N\). Finally, we assume that user transmissions undergo independent fading and each \(h_i\) has a symmetric distribution on \(\mathbb{R}\) (e.g., Rayleigh fading with \(h_i\) distributed as \(\mathcal{N}(0, \rho_i^2)\)).

Next, we define the individual discrete delays \(\tau_i' \in \mathbb{Z}_+\) as \(\tau_i' \overset{\text{def}}{=} \lfloor \frac{\tau_i}{T_c} \rfloor\) and define the maximum discrete delay \(\tau \in \mathbb{Z}_+\) in the system as an upper bound on the delays satisfying \(\tau \geq \max_i \tau_i'\). It is easy to see that the received signal \(y(t)\) at the BS can be sampled at the chip rate to obtain an equivalent discrete approximation

\[
y \approx \sum_{i=1}^{N} h_i \delta_i \sqrt{\mathcal{E}_i} \bar{x}_i + \mathbf{w},
\]

(4.4)

which tends to be quite accurate as long as point sampling is employed and \(g(t)\) is close to being a square pulse. Here, the AWGN vector \(\mathbf{w}\) is distributed as \(\mathcal{N}(\mathbf{0}_{M+\tau}, \mathbf{I}_{M+\tau})\), i.e., the instantaneous received signal to noise ratio (SNR) of the active users is \(\mathcal{E}_i |h_i|^2\),

\(^3\)We take fading coefficients in \(\mathbb{R}\) since we are assuming real-valued codewords. Modifications for the complex-valued case are tedious but straightforward.

\(^4\)One of the major differences between [42,44] and the setup in here is that it is assumed in [42,44] that \(\max_{i,j} (\tau_i - \tau_j) < T_c\) whereas we do not make this assumption since it is nearly impossible to satisfy this condition for large-enough values of \(N\).
and the vectors $\tilde{x}_i \in \mathbb{R}^{M+\tau}$ are defined as

$$\tilde{x}_i = \begin{bmatrix} 0_{\tau_i}^T & x_i^T & 0_{\tau-\tau_i}^T \end{bmatrix}^T, \quad i = 1, \ldots, N.$$  \hspace{1cm} (4.5)

The assumptions we make here are that (i) the maximum delay $\tau$ is known at the BS and (ii) each user has knowledge of the SNR at which its transmitted signal arrives at the BS (in other words, the $i$-th user knows $|h_i|$). Both these assumptions are quite reasonable from a practical perspective; in particular, if one assumes that the BS transmits a beacon signal before the users start transmitting then the last assumption follows because of reciprocity between the downlink and uplink.

Our goal now is to specify a MUD algorithm for this asynchronous CDRA scheme that returns an estimate $\hat{\mathcal{I}}$ of the set of active users $\mathcal{I} \overset{\text{def}}{=} \{i : \delta_i = 1\}$ from the $(M+\tau)$-dimensional vector $y$ without knowledge of the set of delays $\{\tau_i\}$ or the set of channel coefficients $\{h_i\}$ at the BS. Note that a benchmark for any such algorithm is synchronous, orthogonal signaling-based random access, which dictates the relationship $k \leq N \leq M$. Therefore, the primary objective of our algorithm must be to successfully manage an average number of active users that is almost linear in $M$, but also service a total number of users in the uplink that is significantly larger than $M$.

In addition to this primary objective, we are also interested in specifying probability of error, $P_{\text{err}} \overset{\text{def}}{=} \mathbb{P}(\hat{\mathcal{I}} \neq \mathcal{I})$, and providing a low-complexity implementation of the MUD algorithm. In the next section, we propose an algorithm that explicitly takes advantage of the random user activity in the network to successfully meet all these objectives.

### 4.3 Multiuser Detection Using The Lasso

In this section, we propose a MUD algorithm for asynchronous CDRA that is based on the mixed-norm convex optimization program known as the lasso [48]. The lasso
was first proposed in the statistics literature for linear regression in underdetermined settings. In [42], the lasso has been suggested as a potential method for MUD in *synchronous* on–off RACs. However, extending the ideas of [42] to the asynchronous case using the standard lasso formulation seems very difficult. In contrast, while the MUD algorithm proposed in this section is based on the lasso, we present a rather nonconventional usage of the lasso that is specific to the problem at hand. One of our major contributions indeed is establishing that this formulation is guaranteed to yield successful MUD with high probability. The fact that further differentiates our work from [42] is that we relate the performance of the proposed MUD algorithm for both arbitrary and random delays to two simple metrics of the set of user codewords, which enables us to construct specialized codewords for different applications. The analysis carried out in this regard might also be of independent interest to researchers working on configuration (neighbor discovery) in ad-hoc wireless networks and sensor networks. These results also have connections with the area of sparse signal recovery, as noted in Section 4.4.1 and Section 4.5.2.

### 4.3.1 Main Results

In order to make use of the lasso for MUD in asynchronous on-off RACs, we first rewrite (4.4) as

\[
y = \begin{bmatrix} \tilde{x}_1 & \tilde{x}_2 & \ldots & \tilde{x}_N \end{bmatrix} \tilde{\beta} + \mathbf{w},
\]

where the \(i\)-th entry of the vector \(\tilde{\beta} \in \mathbb{R}^N\) is described as \(\tilde{\beta}_i \overset{\text{def}}{=} h_i \delta_i \sqrt{\mathcal{E}_i}\). While (4.6) appears superficially similar to the standard lasso formulation, we cannot use the lasso to obtain an estimate of the set of active users \(\mathcal{I}\) from (4.6) since the \((M + \tau) \times N\) matrix \(\tilde{\mathbf{X}}\) in (4.6) is unknown due to the asynchronous nature of the problem. In
order to overcome this obstacle, we first define the $(M + \tau) \times (\tau + 1)$ Toeplitz matrices $X_i$ as

$$X_i = \begin{bmatrix} x_i & 0_{\tau} \\ \vdots & \ddots & \ddots & \vdots \\ 0_{\tau} & \cdots & x_i \\ \end{bmatrix}, \quad i = 1, \ldots, N,$$

and observe that we can equivalently write (4.6) in the form

$$y = \left[ X_1 \ X_2 \ \cdots \ X_N \right] \left[ \beta_1^T \ \beta_2^T \ \cdots \ \beta_N^T \right]^T + w,$$

where $X$ is now a $(M + \tau) \times N(\tau + 1)$ known matrix, which we term the expanded codebook. The vector $\beta \in \mathbb{R}^{N(\tau + 1)}$ is a concatenation of $N$ vectors, each of length $(\tau + 1)$, whose entries are given by $\beta_{i,j} = \tilde{\beta}_{\pi}1_{\{\tau' = j-1\}}, \quad i = 1, \ldots, N, \quad j = 1, \ldots, \tau + 1$.

We make use of this notation to describe the proposed lasso-based MUD algorithm for asynchronous CDRA in Algorithm 4.1.\footnote{Algorithm 4.1 acts as a hybrid between the standard lasso and the group lasso [49]. Specifically, it is clear from the problem formulation that the group lasso is ill-suited for the specified MUD problem since each of the sub-vectors $\{\beta_i\}$ in (4.8) has at most one non-zero entry. On the other hand, we are only interested in detecting the active users and need not estimate their delays; hence, the group nature of the detection criterion in the definition of $\hat{I}$. We next state the main results of this section, which bound the probability of error of Algorithm 4.1. Here we present MUD guarantees for arbitrary codebooks, parametrized by two metrics of the expanded codebook $X$. The first is the worst-case coherence of the expanded codebook, defined earlier in Section 3.8.1 and in this context is given by

$$\mu(X) \stackrel{def}{=} \max_{(i,j) \neq (i',j')} \left| \langle x_{i,j}, x_{i',j'} \rangle \right|$$

where $x_{i,j}$ denotes the $j$-th column of the Toeplitz matrix $X_i$. In words, the worst-case coherence is the largest inner product between any two codewords with arbitrary
Algorithm 4.1 Multiuser Detection in Asynchronous On–Off Random Access Channels Using the Lasso

**Inputs**
1) The chip-rate sampled vector \( y \)
2) Set of \( M \)-dimensional codewords \( \{ x_i \}_{i=1}^N \)
3) Maximum discrete delay \( \tau \) in the uplink
4) A regularization parameter \( \lambda \) for the lasso

**Construct** the expanded codebook \( X \) described in (4.8) using \( \{ x_i \} \) and \( \tau \)

\[
\hat{\beta} \leftarrow \arg \min_{b \in \mathbb{R}^{N(\tau+1)}} \frac{1}{2} \| y - Xb \|_2^2 + \lambda \| b \|_1
\]  

(LASSO)

\[
\hat{I} \leftarrow \{ i : \| \hat{\beta}_i \|_0 > 0 \}
\]

**Return** \( \hat{I} \) as an estimate of the set of active users \( I \)

shifts. The second metric is the spectral norm of the expanded codebook: \( \| X \|_2 \overset{\text{def}}{=} \sqrt{\lambda_{\text{max}}(X^TX)} \).

**Theorem 4.1.** Suppose that users in the network become active according to independent and identically distributed (iid) Bernoulli random variables such that \( P(\delta_i = 1) = k/N \), and the users have transmit powers satisfying

\[
\mathcal{E}_i > \frac{128 \log (N \sqrt{\tau + 1})}{|h_i|^2}, \quad i \in \mathcal{I}.
\]  

(4.10)

Then, with \( \lambda = 2\sqrt{2\log(N \sqrt{\tau + 1})} \), Algorithm 4.1 successfully carries out multiuser detection with \( P_{\text{err}} \leq 2N^{-1} \left( 2\pi \log(N \sqrt{\tau + 1}) \right)^{-1/2} + 5(N(\tau + 1))^{-2\log^2} + 3N^{-2\log^2} \)

when

\[
N \leq \exp \left( \frac{(c(\tau + 1)\mu(X))^{-1}}{\tau + 1} \right)
\]

and

\[
k \leq \frac{N}{c \log(N(\tau + 1))\|X\|_2^2}.
\]  

(4.11)

(4.12)

Here, the constant \( c > 0 \) is independent of the problem parameters.
**Remark 4.1.** Notice that Theorem 4.1 requires the transmit powers of all active users to satisfy (4.10). This could lead to unrealistic demands on the transmit powers of users with very small fading coefficients. There is, however, a straightforward extension of Theorem 4.1 that handles such situations by requiring users with small-enough fading coefficients to remain inactive. Since the required analysis in that case can be carried out by using well-known techniques for computing outage probabilities, we have chosen to forgo a detailed discussion of this issue for brevity of exposition.

The proof of this theorem is provided in Appendix 4.A. From (4.11) we see that to accommodate a large number of total users $N$, we need codewords that result in an expanded codebook with a small worst-case coherence $\mu(X)$. Similarly, (4.12) shows that codewords that result in small spectral norm $\|X\|_2$ allow the value of $k$ to be large. While Theorem 4.1 is general, it may be a bit opaque to some readers. Once applied to specific codewords in Theorems 4.3 and 4.4, however, favorable scaling relations between $k, N$ and $M$ become apparent.

Note that Theorem 4.1 considers recovery in the presence of an arbitrary set of delays $\{\tau_i\}$. Specifically, the result describes average-case behavior for user activity and worst-case behavior for the set of users’ delays. This is desirable since fixing a probability distribution on the delays restricts the applicability of our results to only certain classes of networks. Nonetheless, considering random delays can be desirable in certain cases. To this effect, we now place a probability model on the set of delays and derive result analogous to that of Theorem 4.1. Explicitly, for each $i$ with $\delta_i = 1$, we consider $\tau_i$ selected uniformly at random from the set $\{0, 1, \ldots, \tau\}$. In doing so, we find that the requirement on $k$ scales more favorably with respect to the maximum delay $\tau$ when one considers this typical-case analysis of $\{\tau_i\}$. Note that while any probability model on the delays reduces applicability of the corresponding results to certain network settings, the uniform distribution of delays is mainly an illustrative model that is also amenable to analysis.
**Theorem 4.2.** Suppose that users in the network become active according to iid Bernoulli random variables such that \( P(\delta_i = 1) = k/N \). Further, suppose that the delays of active users \( \{\tau_i : i \in I\} \) are drawn uniformly at random from \( \{0, 1, \ldots, \tau\} \) and the transmit powers of users satisfy (4.10). Then, with \( \lambda = 2\sqrt{2 \log(N\sqrt{\tau + 1})} \), Algorithm 4.1 successfully carries out multiuser detection with \( P_{err} \leq 2N^{-1}(2\pi \log(N\sqrt{\tau + 1}))^{-1/2} + 7(N(\tau + 1))^{-2\log^2} \) when

\[
N \leq \frac{\exp((c'(\tau + 1)\mu(X))^{-1})}{\tau + 1} \quad \text{and} \quad \quad \quad (4.13)
\]

\[
k \leq \frac{N(\tau + 1)}{c' \log(N(\tau + 1))\|X\|_2^2} \quad \quad \quad (4.14)
\]

Here, the constant \( c' > 0 \) is independent of the problem parameters.

The proof of this theorem is provided in Appendix 4.B. Compared with Theorem 4.1, we note that the bound on the number of active users in this case scales with an additional factor of \( \tau + 1 \). When we specialize this theorem to random codewords in Section 4.4.1 and a deterministic codeword construction in Section 4.4.2, this translates to a mere log-order dependence on \( \tau \).

### 4.3.2 Computational Complexity

Theorem 4.1 characterizes the performance of Algorithm 4.1 for MUD in asynchronous on–off RACs but fails to shed any light on its computational complexity. However, the lasso is a well-studied program in the statistics literature and—thanks to its convex nature—there exist a number of extremely fast (polynomial-time) implementations of the optimization program specified in (lasso); see, e.g., [50].

In this regard, computational complexity of the implementations of (lasso) such as SpaRSA [50] is determined—to a large extent—by the complexity of the matrix–vector multiplications \( Xb \) and \( X^T y \). It therefore seems that Algorithm 4.1 increases the computational complexity of the matrix–vector multiplications from \( O(MN) \),
corresponding to the case of perfectly-known user delays [cf. (4.6)], to $O(MN(\tau + 1))$. This observation, however, ignores the fact that $X$ in (4.8) has a Toeplitz-block structure. Specifically, if we write $b \in \mathbb{R}^{N(\tau + 1)}$ as $b = \begin{bmatrix} b_1^T & \ldots & b_N^T \end{bmatrix}^T$ then it follows from elementary signal processing that

$$Xb = \sum_{i=1}^{N} F_{M+\tau}^{-1}\left(F_{M+\tau}(x_i) \odot F_{M+\tau}(b_i)\right), \quad (4.15)$$

where $F_n(\cdot)$ and $F_n^{-1}(\cdot)$ denote the FFT implementation of the $n$-point discrete Fourier transform (DFT) and the $n$-point inverse DFT of a sequence, respectively, while $\odot$ denotes pointwise multiplication. Similarly, if we use $(\cdot)[n_1 : n_2]$ to denote the $n_1$-th to $n_2$-th elements of a vector and $(\cdot)^-$ to denote the time-reversed version of a vector, then it follows from routine calculations that $\forall i = 1, \ldots, N$, we have

$$X^Ty[i(\tau + 1) - \tau : i(\tau + 1)] = F_{2M+\tau-1}^{-1}\left(F_{2M+\tau-1}(x_i^-) \odot F_{2M+\tau-1}(y)\right)[M : M + \tau].$$

It therefore follows from the complexity of the FFT that the matrix–vector multiplications $Xb$ and $X^Ty$ in Algorithm 4.1 can in fact be carried out using only $O(MN \log(M + \tau))$ operations as opposed to $O(MN(\tau + 1))$ operations. This suggests that the computational complexity of Algorithm 4.1 at worst differs by a factor of $\log(M + \tau)$ from an oracle-based scheme that has perfect knowledge of $\tilde{X}$.

This conclusion is also justified numerically from the results of several numerical experiments reported in Section 4.5. Table 4.1 shows typical computation times of Algorithm 4.1 in Matlab for various values of $\tau$. The standard SpaRSA recovery is faster at low values of $\tau$ due to Matlab’s optimized matrix multiplications. However,
for $\tau \geq 100$, the advantage of the FFT-based implementation becomes apparent. The non-monotonicity of recovery times in the FFT augmented numerical experiments is due to the complex interaction between padding in Matlab’s FFT implementation, numerical accuracy, and additional SpaRSA iterations, a detailed discussion of which is beyond the scope of this work. Of course, for practical applications, optimizations are required beyond a Matlab implementation.

4.4 Codewords for Multiuser Detection

In this section, we consider two sets of user codewords for asynchronous CDRA using Algorithm 4.1. The first is a random construction with normalized iid $\pm 1$ user codewords and the second is a deterministic construction based on cyclic codes. The deterministic construction has the advantage that user codewords can be more efficiently stored and generated. However, the random construction is more flexible with regard to codeword length and number of codewords available. Using Theorems 4.1 and 4.2, we show that (ignoring $\tau$) both sets of codewords allow the recovery of $I$ when $k \lesssim M/\log N$. Furthermore, the random codewords allow $N$ to be super-polynomial in $M$ while the deterministic codewords allow $N$ to be polynomial in $M$.

4.4.1 Random Rademacher Codewords: Guarantees

Communication theory often uses random codewords for optimality. Furthermore, random measurement matrices are frequently used in sparse signal recovery. These examples inspire us to analyze randomly generated codewords in the context of Theorems 4.1 and 4.2 for MUD in RACs.

In the following, we assign each user a codeword of length $M$ that is independently generated from a $\text{binary}(\pm 1/\sqrt{M}, I_M)$ distribution. We seek to quantify $\mu(X)$ and
∥X∥₂ of the expanded codebook to specialize Theorems 4.1 and 4.2 to these random codewords. This is accomplished in the following lemmas.

**Lemma 4.1.** Given any fixed \( \varsigma > 0 \), the expanded codebook \( X \) of random Rademacher codewords satisfies \( \mu(X) \leq \varsigma \) with probability exceeding \( 1 - 2N^2(\tau + 1)^2e^{-\frac{M\varsigma^2}{4}} \).

*Proof.* The proof of this lemma is a consequence of the bound on the worst-case coherence \( \mu \) of random Toeplitz matrices [33, Theorem 3.5] and the Hoeffding inequality [51]. Specifically, we can write

\[
\mu(X) = \max \left\{ \max_{j \neq j'} |\langle x_{i,j}, x_{i,j'} \rangle|, \max_{i \neq i'} |\langle x_{i,j}, x_{i',j'} \rangle| \right\}.
\]

Furthermore, the proof of Theorem 3.5 in [33] implies that \( |\langle x_{i,j}, x_{i,j'} \rangle| \leq \varsigma \) with probability exceeding \( 1 - 4e^{-\frac{M\varsigma^2}{4}} \) for any \( j \neq j' \). Finally, since the product of two independent binary random variables is again a binary random variable, it can also be shown using the Hoeffding inequality that \( |\langle x_{i,j}, x_{i',j'} \rangle| \leq \varsigma \) with probability exceeding \( 1 - 2e^{-\frac{M\varsigma^2}{4}} \) for any \( i \neq i' \). It therefore follows from the union bound that \( \mu(X) \leq \varsigma \) with probability exceeding \( 1 - 2N^2(\tau + 1)^2e^{-\frac{M\varsigma^2}{4}} \).

**Lemma 4.2.** The spectral norm of the expanded codebook \( X \) of random Rademacher codewords satisfies

\[
\|X\|_2 \leq 26\sqrt{\tau + 1} \left( 1 + \sqrt{\frac{N}{M}} \right)
\]

(4.16)

with probability exceeding \( 1 - e^{-\sqrt{MN}} \).

*Proof.* We first recall that the spectral norm is invariant under column-interchange operations. Now define \( \Phi \overset{\text{def}}{=} \begin{bmatrix} x_1 & \ldots & x_N \end{bmatrix} \) and \( \Psi \overset{\text{def}}{=} \begin{bmatrix} \Phi_0 & \Phi_1 & \ldots & \Phi_\tau \end{bmatrix} \), where each block \( \Phi_i \) is an \( (M + \tau) \times N \) matrix that is constructed by prepending and appending \( \Phi \) with \( i \) rows and \( (\tau - i) \) rows of all zeros, respectively. It is then easy
to see that $\|X\|_2 = \|\Psi\|_2$ and $\|\Phi_0\|_2 = \cdots = \|\Phi_\tau\|_2 = \|\Phi\|_2$. Furthermore, we can write for any $N(\tau + 1)$-dimensional vector $z = \begin{bmatrix} z_0^T & z_1^T & \ldots & z_\tau^T \end{bmatrix}^T$

\[
\frac{\|\Psi z\|_2}{\|z\|_2} \leq \frac{\sum_{i=0}^{\tau} \|\Phi_i z_i\|_2}{\|z\|_2} \leq \frac{\|\Phi\|_2 \sum_{i=0}^{\tau} \|z_i\|_2}{\|z\|_2} \leq \frac{\sqrt{\tau + 1} \|\Phi\|_2}{\|z\|_2} = \sqrt{\tau + 1} \|\Phi\|_2, \tag{4.17}
\]

where (a) follows from the definition of $\Psi$ and the triangle inequality, while (b) follows from the Cauchy–Schwarz inequality. It therefore follows from the previous discussion and (4.17) that $\|X\|_2 \leq \sqrt{\tau + 1} \|\Phi\|_2$.

In order to complete the proof, notice that $\Phi$ is an $M \times N$ random matrix whose entries are independently distributed as binary($\pm 1/\sqrt{M}$). It can therefore be established, similar to [52, Proposition 2.4], that $\|\Phi\|_2 \leq 26 \left(1 + \sqrt{\frac{N}{M}}\right)$ with probability exceeding $1 - e^{-\frac{MN}{8}}$.

We now want to apply these lemmas to specialize Theorems 4.1 and 4.2 to the expanded codebook matrix of random codewords. We begin by noting that with $\varsigma$ of Lemma 4.1 chosen appropriately, the event

\[
G_1 = \left\{ \mu(X) \leq \sqrt{\frac{12 \log(N(\tau + 1))}{M}} \right\} \tag{4.18}
\]

holds with probability exceeding $1 - 2(N(\tau + 1))^{-1}$. Furthermore, Lemma 4.2 implies that the event

\[
G_2 = \left\{ \|X\|_2 \leq 52 \sqrt{\frac{N(\tau + 1)}{M}} \right\} \tag{4.19}
\]

holds with probability exceeding $1 - e^{-\frac{MN}{8}}$. 

75
Since we assume that the random codewords are assigned independently of the set of active users $I$, we can substitute (4.18) and (4.19) into (4.11) and (4.12) while adding the failure probabilities $\mathbb{P}(G_1^c)$ and $\mathbb{P}(G_2^c)$ to the probabilities of error in Theorems 4.1 and 4.2 via the union bound. This results in the following theorem.

**Theorem 4.3.** Suppose that the $N$ codewords $\{x_i \in \mathbb{R}^M\}_{i=1}^N$ are drawn independently from a binary $(\pm 1/\sqrt{M}, I_M)$ distribution. Furthermore, let $\lambda$ and $\mathcal{E}_i$ satisfy the conditions in Theorem 4.1 and let $N$ satisfy

$$N \leq \frac{\exp(c_{1,r}(\tau + 1)^{-2/3}M^{1/3})}{\tau + 1}. \quad (4.20)$$

(a) For an arbitrary set of user delays, if

$$k \leq \frac{c_{2,r}M}{(\tau + 1) \log \left( N(\tau + 1) \right)}, \quad (4.21)$$

then Algorithm 4.1 successfully carries out multiuser detection with $P_{err} \leq 2N^{-1} \left( 2\pi \log(\sqrt{\tau}) \right)^{-1/2} + 5 \left( N(\tau + 1) \right)^{-2\log^2} + 3N^{-2\log^2} + 2(N(\tau + 1))^{-1} + e^{\sqrt{MN}/8}$.

(b) For a set of user delays distributed uniformly at random, when

$$k \leq \frac{c_{3,r}M}{\log \left( N(\tau + 1) \right)}, \quad (4.22)$$

then Algorithm 4.1 successfully carries out multiuser detection with $P_{err} \leq 2N^{-1} \left( 2\pi \log(\sqrt{\tau}) \right)^{-1/2} + 7 \left( N(\tau + 1) \right)^{-2\log^2} + 2(N(\tau + 1))^{-1} + e^{\sqrt{MN}/8}$.

Here, the constants $c_{1,r}, c_{2,r}, c_{3,r} > 0$ are independent of the problem parameters.

**Remark 4.2.** It is important to note here that, instead of relying upon Theorem 4.1, if one were to directly analyze the MUD performance of Algorithm 4.1 for random codewords then it is possible to achieve the scaling $k \lesssim M/\log^5 \left( N(\tau + 1) \right)$ in the
case of arbitrary delays by using the results of [47] for the “invertibility condition” in Appendix 4.A. Specifically, the work in [47] considers random Toeplitz-block matrices in a similarly structured problem and achieves only a poly-logarithmic dependence on \( \tau \) in the case of arbitrary delays. The analysis in [47], however, is not extendable to arbitrary Toeplitz-block matrices and further the proof techniques used in there introduce some complications related to noise folding. In contrast, our focus in here is to provide conditions applicable to arbitrary codewords via the metrics \( \|X\|_2 \) and \( \mu(X) \), and our results for random matrices/codewords are primarily meant to be a demonstration of our more general results. Nonetheless, we believe that [47] provides unique insights for Algorithm 4.1 in the case of random codewords and arbitrary delays.

4.4.2 Deterministic Codewords: Construction and Guarantees

Though random codewords allow our proposed scheme to service a large number of users (with respect to \( M \)), deterministic codewords can have significant advantages. In particular, they tend to be much easier to generate and store. We will consider one such codeword construction in this section.

Our deterministic construction uses codewords derived from algebraic error correcting codes. In particular, we consider a cyclic code for which the codebook is closed under circular shifts of codewords. We use a cyclic code since we use cyclic shifts as approximations of delayed user codewords. As such, the full cyclic code is closely related to the expanded codebook matrix \( X \) which contains the delayed user codewords. To construct this relationship, we will select a subset of our cyclic code for assignment to users. In order to remove ambiguity when discussing both the full cyclic code and this subset, we will call the complete cyclic code the ambient code,
while the subset assigned to users will be called the user codebook (i.e., the user codebook is the set \( \{ x_i \} \)).

Our construction is parametrized by two positive integers, \( m \) and \( 2 \leq t < m/2 \). We will operate in the Galois finite field of size \( 2^m \), which we denote as \( \mathbb{F}_{2^m} \). Our code is constructed via the trace function \( \text{Tr} : \mathbb{F}_{2^m} \to \mathbb{F}_2 \) ([53, Ch. 4.8]) defined by

\[
\text{Tr}(a) = a + a^2 + \cdots + a^{2^{m-1}} = \sum_{j=0}^{m-1} a^{2^j}.
\]

Taking \( z \) as a primitive element of \( \mathbb{F}_{2^m} \) we define the \( j \)th element of a codeword in the ambient code as

\[
C_j^\alpha = \frac{1}{\sqrt{2^m - 1}}(-1)^{\text{Tr}\left[\alpha_0 z^j + \sum_{i=1}^{t} \alpha_j z^{(2^i+1)}\right]}, \quad j = 0, 1, 2, \ldots, 2^m - 2 \quad (4.23)
\]

where the vector \( \alpha = [\alpha_0 \quad \alpha_1 \quad \ldots \quad \alpha_t] \) with \( t + 1 \) elements in \( \mathbb{F}_{2^m} \) indexes the codeword.

Since \( z \) is primitive, \( \{ z^j : j = 0, 1, \ldots, 2^m - 2 \} \) is simply the set of all non-zero elements in \( \mathbb{F}_{2^m} \), which we denote \( \mathbb{F}^*_{2^m} \). Thus, we can equivalently enumerate the elements by \( x \in \mathbb{F}^*_{2^m} \) as

\[
C_x^\alpha = \frac{1}{\sqrt{2^m - 1}}(-1)^{\text{Tr}\left[\alpha_0 x + \sum_{i=1}^{t} \alpha_i x^{2^i+1}\right]}, \quad x \in \mathbb{F}^*_{2^m} \quad (4.24)
\]

The above construction produces codewords of length \( 2^m - 1 \) and, since each of \( \alpha_i, i = 0, \ldots, t \) can be any value in \( \mathbb{F}_{2^m} \), there are \( 2^{m(t+1)} \) codewords in the ambient code.

We use a subset of the ambient code as the user codebook. We will require two conditions on the selected subset. The first condition restricts us to a subset where no codeword in the subset is a cyclic shift of another. Such a restriction is necessary since, in bounding \( \mu(X) \), we will link cyclic shifts with different user delays. We
will call such a set a cyclic restricted subset. There are many ways to create a cyclic restricted subset. Consider that, under the element enumeration of (4.23), a codeword and its cyclic shift by \( T \) elements are related as \( C_{\alpha}^{j+T} = C_{\alpha'}^j \) where \( \alpha = [\alpha_0, \ldots, \alpha_t] \) and \( \alpha' = [z^T \alpha_0, z^{2T} \alpha_1, \ldots, z^{(2^t+1)T} \alpha_t] \). If, for example, we required that all codewords in our subset had \( \alpha_0 = c \) for \( c \in \mathbb{F}_{2^m}^* \), no codewords would be the shift of another. Explicitly, the codewords indexed by \( \{ \alpha \in \mathbb{F}_{2^m}^t : \alpha_0 = c \} \) form a cyclic restricted subset for any \( c \neq 0 \in \mathbb{F}_{2^m} \). Since we need only restrict a single entry in the vector \( \alpha \), enumerating over the remaining entries allows us to have a cyclic restricted subset of size \( 2^{mt} \) from the full \( 2^{m(t+1)} \) ambient code. We may choose to use a smaller set, affording flexibility in choosing the value of \( N \), and the set would remain a cyclic restricted subset.

The second condition on selecting the user codebook as a subset of the ambient code is used to ensure we can appropriately bound \( \|X\|_2 \). As above, our condition will be on the set of vectors \( \alpha \) which enumerate the codewords in the user codebook. To describe the condition we first define a wildcard index of the user codebook. We call \( w \) a wildcard index of the user codebook if, for each vector \( \alpha = [\alpha_0, \ldots, \alpha_w, \ldots, \alpha_t] \) that indexes a user codeword, the vector \([\alpha_0, \ldots, c, \ldots, \alpha_t]\) (i.e., the vector \( \alpha \) with \( \alpha_w \) replaced by \( c \)) also indexes a user codeword for each \( c \in \mathbb{F}_{2^m}^* \). We require that the user codebook have a wildcard index \( w \) such that \( 2^w + 1 \) does not divide \( 2^m - 1 \) (denoted \( 2^w + 1 \nmid 2^m - 1 \)). A consequence of requiring the existence of a wildcard index is that the user codebook must be a multiple of \( 2^m \) in size.

To summarize our construction, we assign to users codewords of the form (4.23). We require that the user codebook satisfy two conditions. The first is that it forms a cyclic restricted subset of the ambient code of all possible codewords. The second is that the user codebook contain a wildcard index \( w \) with \( 2^w + 1 \nmid 2^m - 1 \). This construction allows us to have \( M = 2^m - 1 \) while \( N \) may be a multiple of \( 2^m \) up to \( 2^{mt} \).
We now seek to apply Theorems 4.1 and 4.2 to this construction which requires us to bound the two metrics $\mu(X)$ and $\|X\|_2$. We consider each of the two metrics in turn. The metric $\mu(X)$ bounds inner products between any two shifted codewords in the user codebook. As discussed earlier, we will be relating the set of shifted codewords to the ambient code. As a result, our first goal is to bound the inner product of any two codewords in the ambient code. The bound can be obtained easily by exploiting properties of the ambient code. By the linearity of the trace [53, p.116], we find that the element-wise product of two codewords satisfies $C^j_\alpha C^j_{\alpha'} = C^j_{\alpha+\alpha'}$. As a result, the inner product between two non-identical codewords is simply the sum of the entries of a different codeword. This leads to an equivalent goal of bounding the sum of an arbitrary non-trivial codeword. That is, allowing $\{\alpha_i \in \mathbb{F}_{2^m}\}_{i=0}^t$ to be arbitrary with $\alpha \neq 0$, we attempt to bound the sum

$$S = \sum_{x \in \mathbb{F}_{2^m}} (-1)^{\text{Tr}[\alpha_0 x + \sum_{i=1}^t \alpha_i x^{2^i+1}]}.$$  \tag{4.25}$$

**Lemma 4.3.** The sum given in (4.25) satisfies $|S| \leq 2^m + t + 1/2$ for any codeword.

This lemma is proved in Appendix 4.C. We leverage this result to provide a bound on $\mu(X)$.

**Lemma 4.4.** Let the user codebook $\{x_i\}$ be a cyclic restricted subset of the ambient code defined by (4.23). Then the worst-case coherence is bounded by

$$\mu(X) \leq \frac{2^{t+1/2+m/2} + \tau}{2^m - 1}.$$ 

**Proof.** We are interested in bounding $|\langle x_{i,j}, x_{i',j'} \rangle|$. We will do so by relating this inner product to one in the ambient code. Using the fact that $x_{i',j'}$ is of the form (4.5), we can replace vectors of zeros above and below the codeword with shifted copies of $x_{i'}$.

---

6This is a simple reformulation of the fact that the non-exponentiated version of the code in (4.23) is linear.
in order to make the periodic vector $\hat{x}_{i',j'}$ with period $M$ on its $M + \tau$ length; we call $\hat{x}_{i',j'}$ the periodic extension of $x_{i',j'}$. Let $\Delta = x_{i',j'} - \hat{x}_{i',j'}$ be its difference from the original. By the triangle inequality,

$$|\langle x_{i,j}, x_{i',j'} \rangle| \leq |\langle x_{i,j}, \hat{x}_{i',j'} \rangle| + |\langle x_{i,j}, \Delta \rangle|. $$

The periodic extension has converted the shift $j'$ into a cyclic shift on the support of $x_{i,j}$. Furthermore, since the two unshifted codewords $x_i$ and $x_i'$ come from a cyclic restricted subset, they are guaranteed to be different on the support. Thus, for the first term we use the bound of Lemma 4.3 divided by $2^m - 1$, thereby accounting for the normalization of the users’ codewords. The second term is bounded by the fact that the support of $\Delta$ overlaps with that of $x_{i,j}$ with at most $\tau$ elements of value $\pm 1/\sqrt{2^m - 1}$.

Having bounded $\mu(X)$, we now turn to the second metric $\|X\|_2$ and bound it using the following lemma.

**Lemma 4.5.** Let the user codebook of cardinality $N$ selected from the ambient code have a wildcard index $w$ such that $2^w + 1 \nmid 2^m - 1$. Then the spectral norm of the expanded codebook $X$ is bounded by $\|X\|_2 \leq \sqrt{\frac{N}{2^m - 1}}(\tau + 1)$.

**Proof.** We begin the proof similarly to Lemma 4.2. Let $\Phi$ be an $M \times N$ matrix of the user codewords and recall from Lemma 4.2 that $\|X\|_2 \leq \sqrt{\tau + 1}\|\Phi\|_2$. We will now show that the rows of $\Phi$ are orthogonal, such that $\Phi\Phi^T = \frac{N}{2^m - 1}I$, which is sufficient for the proof since $\lambda_{\max}(\Phi\Phi^T) = \lambda_{\max}(\Phi^T\Phi)$. 

81
Using (4.24), the inner product between two rows, indexed by \( x \) and \( y \) in \( \mathbb{F}_2^m \), is the following sum over the vectors \( \alpha \) indexing the user codebook:

\[
\frac{1}{2^m - 1} \sum_{\alpha} C^{(x)}(\alpha) C^{(y)}(\alpha) = \frac{1}{2^m - 1} \sum_{\alpha} (-1)^{\text{Tr}\left[\alpha_0(x+y) + \sum_{i=1}^{t} \alpha_i(x^{2^i+1}+y^{2^i+1})\right]} \tag{4.26a}
\]

\[
= \frac{1}{2^m - 1} \sum_{\alpha \in \mathbb{F}_2^m} (-1)^{\text{Tr}\left[\alpha_w(x^{2^w+1}+y^{2^w+1})\right]} \sum_{\alpha \setminus \{\alpha_w\}} (-1)^{\text{Tr}\left[\alpha_0(x+y) + \sum_{i=1, i \neq w}^{t} \alpha_i(x^{2^i+1}+y^{2^i+1})\right]} \tag{4.26b}
\]

where in (4.26b) we have separated the wildcard index element \( \alpha_w \), which takes every value in \( \mathbb{F}_2^m \), into a separate sum. For all \( a \in \mathbb{F}_2^m \) we have \( a + a = 0 \). Thus, from the sum in (4.26a), when \( x = y \) each term takes unit value and their sum equals \( N \), the number of \( \alpha \). In the case when \( x \neq y \), we examine (4.26b). By our wildcard index condition we are guaranteed that \( x^{2^w+1} \neq y^{2^w+1} \) so that \( \alpha_w \) has a non-zero coefficient in the trace. By Proposition 4.5(a) in Appendix 4.C, precisely half the terms of the sum over \( \alpha_w \) are \(-1\) and thus the whole sum evaluates to 0. Therefore, the inner product between the two rows evaluates to \( N/(2^m - 1) \) when \( x = y \) and to 0 otherwise. This completes the proof of the lemma. \( \square \)

Having bound both \( \mu(X) \) and \( \|X\|_2 \), we are able to apply Theorems 4.1 and 4.2 to this deterministic construction and give the following recovery guarantees.

**Theorem 4.4.** Let \( m \) and \( t \) be positive integers and let \( M = 2^m - 1 \). Suppose that the \( N \) codewords \( \{x_i \in \mathbb{R}^M\}_{i=1}^N \) are chosen from the code defined by (4.23) such that they form a cyclic restricted subset and have a wildcard index \( w \) with \( 2^w + 1 \nmid 2^m - 1 \). Furthermore, let \( \lambda \) and \( \{E_i\} \) satisfy the conditions in Theorem 4.1 and let \( N \) satisfy,

\[
N \leq \frac{\exp\left(c_1.d^{(\tau+1)(2^m/2^{(\tau+1)/2})}\right)}{\tau + 1}. \tag{4.27}
\]

82
(a) For an arbitrary set of user delays, if
\[ k \leq \frac{c_2 d (2^m - 1)}{(\tau + 1) \log (N(\tau + 1))}, \tag{4.28} \]
then Algorithm 4.1 successfully carries out multiuser detection with \( P_{err} \leq 2N^{-1}(2\pi \log(N\sqrt{\tau + 1}))^{-1/2} + 5(N(\tau + 1))^{-2\log^2} + 3N^{-2\log^2}. \)

(b) For a set of user delays distributed uniformly at random, if
\[ k \leq \frac{c_3 d (2^m - 1)}{\log (N(\tau + 1))}, \tag{4.29} \]
then Algorithm 4.1 successfully carries out multiuser detection with \( P_{err} \leq 2N^{-1}(2\pi \log(N\sqrt{\tau + 1}))^{-1/2} + 7(N(\tau + 1))^{-2\log^2}. \)

Here, the constants \( c_1, d, c_2, d, c_3, d > 0 \) are independent of the problem parameters.

It is important to note here that although (4.27) appears to allow a super-polynomial number of users \( N \), our construction restricts us to at most \( 2^{mt} \) codewords to be assigned to users. For small values of \( t \), this restriction on \( N \) dominates the one in (4.27). However, as \( t \) approaches \( \frac{m-1}{2} \), (4.27) becomes the relevant bound on \( N \).

In general, comparing our deterministic construction of codewords to the randomly generated ones in Section 4.4.1, we find that the proposed deterministic codewords have advantages in storage and generation while randomly generated codes have the advantages that \( M \) is arbitrary and that \( N \) can be super-polynomial in \( M \).

### 4.5 Numerical Results and Discussion

#### 4.5.1 Monte Carlo Experiments

To verify and illustrate the results presented in this chapter for MUD in asynchronous RACs, we make use of Monte Carlo trials. Our numerical experiments assume a total
Figure 4.1: User support recovery error rate as a function of the expected number of active users \( k \).

of \( N = 3072 \) users communicating to the BS using codewords of length \( M = 1023 \). We report the MUD results for both the random codewords of Section 4.4.1 and the deterministic construction of Section 4.4.2. For the deterministic construction, a code generated with \( m = 10 \) and \( t = 2 \) is used with a subset of the ambient code of size \( N = 3(2^m) \) assigned to users. Random user activity is generated using independent 0–1 Bernoulli random variables \( \{\delta_i\} \) such that \( \mathbb{P}(\delta_i = 1) = k/N \) for a given \( k \). Furthermore, for a given maximum delay \( \tau \), the individual user delays \( \{\tau_i\} \) are generated once at random for each experiment and then fixed for the remainder of the experiment. The implementation of Algorithm 4.1 uses the SpaRSA package [50] in order to solve (lasso) and includes the modifications described in Section 4.3.2 for speeding up the matrix–vector multiplications \( \mathbf{Xb} \) and \( \mathbf{X}^T\mathbf{y} \). In all the numerical plots, results for random codewords are displayed using solid lines while those for deterministic codewords are displayed using dashed lines.

The numerical experiments correspond to the ability of the MUD scheme proposed in Algorithm 4.1 to correctly recover the active user set \( \mathcal{I} \) for varying values of the average number of active users \( k \) and maximum delay \( \tau \). The results of these
Figure 4.2: Normalized per user error as a function of the expected number of active users $k$.

Figure 4.3: Normalized per user error as a function of the received user power with $\tau = 50$. 

experiments are reported in Figure 4.1, which shows that when $k$ is below a certain threshold, $\mathcal{I}$ is exactly recovered (i.e., $\hat{\mathcal{I}} = \mathcal{I}$) in the vast majority of Monte Carlo trials. Beyond the threshold of $k \approx 50$, the fraction of Monte Carlo trials in error quickly approaches one. Figure 4.1 also shows that codewords generated as described in Section 4.4.2 perform nearly identically in performance to those randomly generated.

In order to compare our MUD results with some of the traditional SUD approaches, we have included numerical results corresponding to the performance of a matched filter receiver for the case of random codewords. We assume that the SUD receiver has access to the outputs of the matched filters for all the $N(\tau + 1)$ user codewords and shifts as well as an oracle knowledge of $|\mathcal{I}|$ (which is more specific than knowledge of $k = \mathbb{E}|\mathcal{I}|$). Consequently, the receiver declares the users corresponding to the $|\mathcal{I}|$ largest matched filter responses to be active. Note that, in general, any practical SUD receiver that detects using a fixed threshold for the matched filter responses is expected to perform worse than this oracle-like SUD. Despite this, we find that our proposed MUD algorithm significantly outperforms the traditional SUD receiver based on matched filtering ideas.

Note that the results in Figure 4.1 are reminiscent of Theorem 4.2 and the related sections of Theorems 4.3 and 4.4, rather than Theorem 4.1. This is because results for the worst-case algorithmic performance are difficult to verify experimentally. While guarantees for arbitrary user delays are desirable, verifying this numerically would require generating all $(\tau + 1)^k$ possible combinations of $\{\tau_i\}$ in each Monte Carlo trial. The worst-case analyses of (4.21) and (4.28) suggest that the threshold should be inversely proportional to $(\tau + 1)$. On the other hand, Figure 4.1 does not exhibit this behavior since our numerical experiments correspond to a random generation of the delays $\{\tau_i\}$. Rather, they show that the recovery threshold of $k$ for a typical set
of \( \{\tau_i\} \) is not a strong function of \( \tau \). This corresponds with the results for randomly distributed \( \{\tau_i\} \) in (4.22) and (4.29).

The recovery metric in Figure 4.1 matches that of the theorems and declares a trial to be in error when \( \hat{I} \neq I \). However, it is also useful in many cases to consider how far the estimate \( \hat{I} \) is from the correct set. Therefore, in Figure 4.2 we use the performance metric of \textit{average fraction of detection errors}, which corresponds to \( \frac{|(X \setminus \hat{X}) \cup (\hat{X} \setminus X)|}{k} \) and describes the number of errors in the estimated set of active users as a fraction of the average number of active users. With this metric, we see that Algorithm 4.1 fails gracefully as \( k \) increases.

We also use the recovery metric of average fraction of detection errors to describe the power requirements of the active users in Figure 4.3. This figure shows that the power requirement, \( E_i \) described in Theorems 4.1 and 4.2, is overly restrictive. Specifically, the rightmost point of the horizontal axis at \( E_i |h_i|^2 = 31 \text{dB} \) provides the reference point as the power seen at the receiver as required by (4.10). The figure shows much less power is needed for recovering \( I \). It also shows that the required power is not a function of \( k \) which is exactly in line with the results of our theory.

\subsection*{4.5.2 Discussion}

In order to place our results in context, we note that \( k \lesssim M/\log N \) scaling has also been suggested in [42] for the case of MUD in synchronous on–off RACs using the lasso and random Gaussian codewords. Here, however, we provide non-asymptotic results for the more general asynchronous case, in contrast to the asymptotic results in [42]. Furthermore, we provide guarantees that can be applied to arbitrary user codewords. For the codewords studied in Theorems 4.3 and 4.4 we have established that the MUD scheme for asynchronous on–off RACs has the ability to achieve roughly the same (non-asymptotic) scaling of the system parameters \( k, M, \) and \( N \) as that reported in [42] for the ideal case of synchronous channels.
With regard to the deterministic codewords introduced in Section 4.4.2, our construction is representative of a larger class of deterministic matrices derived from cyclic codes. We consider a particular cyclic code where the codewords are obtained by evaluating quadratic forms at elements of the field $\mathbb{F}_{2^m}$. The worst case coherence $\mu(X)$ of the expanded codebook matrix is determined by the minimum weight of the code and we bound this quantity by elementary methods in Appendix 4.C. We note that Yu and Gong [54] have calculated the exact weight distribution of a very similar code using more sophisticated methods from symplectic geometry.

Beyond the application to MUD for RACs, our results can also be related to work on model selection. Most directly, our work builds on the model selection theory of Candès and Plan [46] for the lasso. As in this work, [46] provides guarantees for lasso that are based on worst-case coherence $\mu(X)$ and spectral norm $\|X\|_2$. However, a key assumption in [46] requires that the vector $\beta$ be “generic” in the sense that its support is uniform over its $(\tau + 1)N$ elements. In this chapter, however, we assume a much different model: the support of $\beta$ is uniformly random over blocks of elements. In this light, the work here is related to recent work on block-sparse signals such as [55] which considers block-sparse signal recovery using a variant of orthogonal matching pursuit as opposed to the lasso. However, as work in the context of signal recovery rather than model selection, the work in [55] is not directly concerned with estimating $I$ and cannot be applied to the MUD problem in RACs.

As a study of sparse signal recovery using a structured measurement matrix, this work relates to that of Romberg and Neelamani [47]. Though [47] considers a different application and is concerned with signal recovery rather than estimating $I$, it studies Toeplitz-block matrices that are similar in structure to $X$. The approach in [47], however, differs from ours since they provide recovery guarantees based on the restricted isometry property (RIP) of the matrix. By working with the RIP, the analysis is particular to randomly generated Toeplitz columns. In contrast, here we provide
guarantees for any matrix $X$ with sufficiently small $\mu(X)$ and $\|X\|_2$. Subsequently, we give both randomly generated and deterministic codeword designs satisfying the requirements. Furthermore, our work provides support set recovery guarantees—in the spirit of model selection [56]—rather than bounds on recovered signal error guaranteed by RIP.

Finally, in terms of the application of our theory in the real-world, we note that Theorems 4.1–4.4 provide non-asymptotic bounds on $k$ and $N$ that guarantee recovery of the set of active users. However, we have not shown that these bounds are tight. Indeed, numerical experiments show that the bounds are somewhat loose in practice. Nonetheless, the theory provides useful scaling relationships with the metrics $\mu(X)$ and $\|X\|_2$ which, as we have demonstrated, can guide non-orthogonal codeword designs in practical systems.

We conclude this section by pointing out three key directions of future work in the context of random access within asynchronous network settings. One of these directions involves modifying Algorithm 4.1 to allow for a small fraction of missed detections at the expense of reducing the fraction of false positives. The second direction involves investigating tight converses of Theorems 4.1 and 4.2 in terms of $k$, $M$, and $N$. The last direction involves extending Theorems 4.1 and 4.2 under the assumption of multipath in the uplink. Given the structured nature of the problem discussed in here, all three of these directions present some unique analytical challenges and we expect to address those challenges in a sequel to this work.

### 4.6 Conclusion

In this chapter, we described a novel scheme for MUD in RACs that allows for the user codewords to be received asynchronously at the receiver. We leveraged and generalized sparse signal theory to provide recovery guarantees for a lasso-based al-
algorithm to find the set of active users. While our results are general and applicable to arbitrary sets of codewords, we specialized them to two specific sets of codewords, random binary codewords and specially constructed algebraic codewords.

The implications of the scaling behavior outlined in the pairs of inequalities in Theorems 4.3 and 4.4 are quite positive in the important special case of fixed-bandwidth spread spectrum waveforms and a BS serving a bounded geographic region. Specifically, they signify that—for any fixed number of temporal signal space dimensions $M$ and maximum delay $\tau$ in the system—the proposed MUD scheme can accommodate $N \lesssim \exp(O(M^{1/3}))$ total users in the case of random signaling and $N$ polynomial in $M$ when using our algebraic code design. Both sets of codewords allow $k \lesssim M/\log N$ active users in the system. This is a significant improvement over the $k \leq N \lesssim M$ scaling suggested by the use of classical matched filtering-based approaches to MUD employing orthogonal signaling.

4.A Proof of the Main Result: Arbitrary Delays

In this appendix, we provide a proof of Theorem 4.1. Before proceeding further, however, let us develop some notation to facilitate the forthcoming analysis. Throughout this appendix, we use $X_S$ to denote the block subdictionary of $X$ obtained by collecting the Toeplitz blocks of $X$ corresponding to the indices of the active users: $X_S \overset{\text{def}}{=} \{X_i : i \in \mathcal{I}\}$. In addition, we use $X_S$ to denote the $(M + \tau) \times |\mathcal{I}|$ submatrix obtained by collecting the columns of $X$ corresponding to the nonzero entries of $\beta$, while we use $\beta_S$ to denote the $|\mathcal{I}|$-dimensional vector comprising of the nonzero entries of $\beta$. Finally, we use $\text{sgn}(\cdot)$ for elementwise signum function: $\text{sgn}(z) \overset{\text{def}}{=} z/|z|$ for any $z \in \mathbb{R}$.

The basic idea behind the proof of Theorem 4.1 follows from the proof of [46, Theorem 1.3]. Specifically, using $\mathcal{S} \subset \{1, \ldots, N(\tau + 1)\}$ to denote the set of the
locations of the nonzero entries of $\beta$, we have from [46, Lemma 3.4] that the lasso solution $\hat{\beta} \overset{\text{def}}{=} \beta + h$ satisfies $h_{S^c} = 0$ and

$$h_S = (X_S^T X_S)^{-1}[X_S^T w - \lambda \text{sgn}(\beta_S)] \quad (4.30)$$

if $\min_{i \in S} |\beta_i| > 4\lambda$ and the following five conditions are met:

- $C_1$ – Invertibility condition: $\|(X_S^T X_S)^{-1}\|_2 \leq 2$.
- $C_2$ – Noise stability: $\|(X_S^T X_S)^{-1}X_S^T w\|_\infty \leq \lambda$.
- $C_3$ – Complementary noise stability: $\|X_S^T (I - X_S (X_S^T X_S)^{-1}X_S^T) w\|_\infty \leq \frac{\lambda}{\sqrt{2}}$.
- $C_4$ – Size condition: $\|(X_S^T X_S)^{-1}\text{sgn}(\beta_S)\|_\infty \leq 3$.
- $C_5$ – Complementary size condition: $\|X_S^T X_S (X_S^T X_S)^{-1}\text{sgn}(\beta_S)\|_\infty \leq \frac{1}{4}$.

Furthermore, it trivially follows in this case that the set of non-zero elements of $\hat{\beta}$ is $S$, which guarantees that $\hat{I} = I$. Our goal then is to consider the probability of each one of these conditions not being met under the assumptions of Theorem 4.1 and the proof of the theorem would then simply follow from the union bound.

### 4.A.1 Invertibility Condition

In order to establish the invertibility condition, we will make use of the following proposition from [57].

**Proposition 4.1** ([57]). Fix $q = 2 \log (N(\tau + 1))$ and define the block coherence

$$\mu_B(X) \overset{\text{def}}{=} \max_{1 \leq i,j \leq N} \|X_i^T X_j - 1_{\{i=j\}} I\|_2. \quad (4.31)$$
Then, for $Eq \overset{\text{def}}{=} [E|Z|^q]^{1/q}$ and $\delta \overset{\text{def}}{=} k/N$, we have the following bound

$$E_q\|X^T_B X_B - I\|_2 \leq 20\mu_B(X) \log (N(\tau+1)) + \delta\|X\|_2^2 + 9\sqrt{\delta \log (N(\tau + 1)) (1 + \tau \mu(X))}\|X\|_2.$$  

(4.32)

We would like to bound (4.32) via bounds on $\mu_B(X)$, $\mu(X)$ and $\|X\|_2$. First, we can use the linear algebra fact $\|\cdot\|_2 \leq \sqrt{\|\cdot\|_1 \|\cdot\|_\infty}$ [58] on (4.31) to show that $\mu_B(X) \leq (\tau + 1)\mu(X)$. Thus, we can rearrange the inequalities of (4.11) and (4.12) to obtain

$$\mu(X) \leq \frac{1}{c(\tau + 1) \log (N(\tau + 1))},$$  

(4.33)

$$\|X\|_2^2 \leq \frac{1}{c\delta \log (N(\tau + 1))}, \quad \text{and}$$  

(4.34)

$$\mu_B(X) \leq \frac{1}{c\log (N(\tau + 1))}.$$  

(4.35)

Substituting these inequalities into (4.32) and choosing $c$ appropriately large yields

$$E_q\|X^T_B X_B - I\|_2 < \frac{1}{4}.$$  

Finally, notice that $X_S$ is a submatrix of $X_B$ and therefore we trivially have $\|X^T_S X_S - I\|_2 \leq \|X^T_B X_B - I\|_2$. It can then be easily seen from the Markov inequality that

$$\mathbb{P}(\|X^T_S X_S - I\|_2 > 1/2) \leq 2^q(E_q\|X^T_B X_B - I\|_2)^q \overset{(a)}{\leq} (N(\tau + 1))^{-2\log^2_2}$$  

(4.36)

where $(a)$ follows from the fact that $E_q\|X^T_B X_B - I\|_2 < \frac{1}{4}$. We have now established that $\|X^T_S X_S\|_2 \in (1/2, 3/2)$ with high probability, which implies that

$$\mathbb{P}(C_1^c) \leq (N(\tau + 1))^{-2\log^2_2}.$$  

(4.37)
4.A.2 Noise Stability

In order to establish the noise-stability condition, we first condition on \( C_1 \) (the invertibility condition). Next, we denote the \( j \)-th column of \( X_S(X_T^T X_S)^{-1} \) by \( z_j \) and note that

\[
\| (X_T^T X_S)^{-1} X_T^T w \|_{\infty} = \max_{1 \leq j \leq |S|} |\langle z_j, w \rangle|.
\]  \hspace{1cm} (4.38)

Furthermore, since the noise vector \( w \) is distributed as \( \mathcal{N}(0, I) \), we also have that \( \langle z_j, w \rangle \sim \mathcal{N}(0, \|z_j\|_2^2) \). Finally, note that conditioned on \( C_1 \), we have the upper bound

\[
\|z_j\|_2 \leq \|X_S(X_T^T X_S)^{-1}\|_2 \leq \sqrt{2}.
\]

where the second inequality can be seen by considering the singular value decomposition of \( X_S \) along with the bound on the singular values from \( C_1 \).

The rest of the argument now follows easily from bounds on the maximum of a collection of arbitrary Gaussian random variables. Specifically, it can be seen from the previous discussion and a real-valued version of [56, Lemma 6] that

\[
P\left( \| (X_T^T X_S)^{-1} X_T^T w \|_{\infty} \geq \sqrt{2} t |C_1| \right) \leq \frac{2Ne^{-t^2/2}}{\sqrt{2\pi t}}.
\]

We substitute \( t = \lambda/\sqrt{2} \) in the above expression to obtain

\[
\frac{2Ne^{-\lambda^2/4}}{\sqrt{\pi}\lambda} = \frac{1}{N(\tau + 1) \sqrt{2\pi \log(N\sqrt{\tau + 1})}}.
\]
Summarizing, we have that the noise stability condition satisfies

\[ \mathbb{P}(C_2^c | C_1) \leq \frac{1}{N(\tau + 1) \sqrt{2\pi \log(N\sqrt{\tau + 1})}}. \]  

(4.39)

### 4.A.3 Complementary Noise Stability

In order to establish the complementary noise-stability condition, we use ideas similar to the ones used in the previous section. To begin with, we again condition on the event \( C_1 \) and use \( P_{X_S} \triangleq X_S(X_S^T X_S)^{-1} X_S^T \) to denote the orthogonal projector onto the column span of \( X_S \). Next, we use \( z_j \) to denote the \( j \)-th column of \( (I - P_{X_S})X_{S^c} \) and note that

\[ \|X_{S^c}^T (I - P_{X_S})w\|_\infty = \max_{1 \leq j \leq |S^c|} |\langle z_j, w \rangle|. \]  

(4.40)

Finally, given that \( P_{X_S} \) is a projection matrix and the columns of \( X \) have unit norm, we have that

\[ \|z_j\|_2 = \|(I - P_{X_S})X_{S^c}e_j\|_2 \leq 1, \]  

(4.41)

where \( e_j \) denotes the \( j \)-th canonical basis vector.

It is now easy to see that, since \( \langle z_j, w \rangle \) is also distributed as \( \mathcal{N}(0, \|z_j\|_2^2) \), we can make use of [56, Lemma 6] to obtain

\[ \mathbb{P}(\|X_{S^c}^T (I - P_{X_S})w\|_\infty \geq t | C_1) \leq \frac{2N(\tau + 1)e^{-t^2/2}}{\sqrt{2\pi t}}. \]

We substitute \( t = \lambda/\sqrt{2} \) in the above expression to obtain

\[ \frac{1}{N \sqrt{2\pi \log(N\sqrt{\tau + 1})}} \leq \frac{2N(\tau + 1)e^{-\lambda^2/4}}{\sqrt{\pi \lambda}}. \]
Summarizing, we have that the complementary noise stability condition satisfies
\[
P(C_3^c | C_1) \leq \frac{1}{N \sqrt{2\pi \log(N \sqrt{\tau + 1})}}.
\]  

(4.42)

4.A.4 Size Condition

In order to establish the size condition, we first write
\[
\|(X_S^T X_S)^{-1} \text{sgn}(\beta_S)\|_\infty \overset{(a)}{\leq} \|(X_S^T X_S)^{-1} - I\| \text{sgn}(\beta_S)\|_\infty + \|\text{sgn}(\beta_S)\|_\infty \\
= \|(X_S^T X_S)^{-1} - I\| \text{sgn}(\beta_S)\|_\infty + 1 \\
= \max_{1 \leq j \leq |S|} |\langle z_j, \text{sgn}(\beta_S) \rangle| + 1
\]  

(4.43)

(4.44)

where (a) follows from the triangle inequality and we once again use \(z_j\) to denote the \(j\)-th column of \((X_S^T X_S)^{-1} - I\). Now define \(A = (X_S^T X_S - I)\) and condition on the event \(C_1\). Then it follows from the Neumann series (cf. [46, p. 2171]) that \(\|z_j\|_2 \leq 2\|A e_j\|_2\).

Furthermore, since \(X_S\) is a submatrix of \(X_B\), we have \(\|A e_j\|_2 \leq \|(X_B^T X_B - I) e_{j'}\|_2\), where \(j'\) is such that the \(j'\)-th column of \(X_B\) matches the \(j\)-th column of \(X_S\).

Finally, define the diagonal matrix \(Q \overset{\text{def}}{=} \text{diag}(\delta_1, \ldots, \delta_N)\) with the “random activation variables” \(\{\delta_i\}\) on the diagonal and define a new matrix \(R = Q \otimes I_{\tau + 1}\), where \(\otimes\) denotes the Kronecker product. Next, use the notation \(H \overset{\text{def}}{=} (X^T X - I)\) and notice that \(\|(X_B^T X_B - I) e_{j''}\|_2 = \|R H e_{j''}\|_2\), where \(j''\) is such that the \(j''\)-th column of \(X\) matches the \(j\)-th column of \(X_S\). In addition, note that \(H = \begin{bmatrix} H_1 & H_2 & \ldots & H_N \end{bmatrix}\) has a block structure that can be expressed as

\[
H = \begin{bmatrix} H_{1,1} & H_{1,2} & \ldots & H_{1,N} \\
H_{2,1} & H_{2,2} & \ldots & H_{2,N} \\
\vdots & \vdots & \ddots & \vdots \\
H_{N,1} & H_{N,2} & \ldots & H_{N,N} \end{bmatrix},
\]  

(4.45)
where $H_{i,j} = X_i^T X_j - 1_{i=j}I$, $1 \leq i, j \leq N$, and $H_i = [H_{1,i}^T \ldots H_{N,i}^T]^T$. We now define two blockwise norms on $H$ as follows: $\|H\|_{B,1} \overset{\text{def}}{=} \max_{1 \leq i \leq N} \|H_i\|_2$, and $\|H\|_{B,2} \overset{\text{def}}{=} \max_{1 \leq i, j \leq N} \|H_{i,j}\|_2$.

Then it follows from the preceding discussion and the structure of the block matrix $H$ that

$$\|z_j\|_2 \leq 2\|Ae_j\|_2 \leq 2\|RHe_j''\|_2 \leq 2\|RH\|_{B,1}. \quad \text{(4.46)}$$

Our next goal then is to provide a bound on $\|RH\|_{B,1}$ and for this we resort to [57, Lemma 5].

**Proposition 4.2** ([57]). For $q \geq 2\log N$ and $\delta = k/N$, we have that

$$E_q\|RH\|_{B,1} \leq 2^{1.5} \sqrt{q}\|H\|_{B,2} + \sqrt{\delta}\|H\|_{B,1}. \quad \text{(4.47)}$$

Now notice from the definition of $H$ and $\|\cdot\|_{B,2}$ that $\|H\|_{B,2} \equiv \mu_B(X) \leq (\tau + 1)\mu(X)$. In addition, we have from the definition of $H$ and $\|\cdot\|_{B,1}$ that

$$\|H\|_{B,1} \overset{(b)}{\leq} \max_{1 \leq i \leq N} \|X_i^T X_i\|_2 + \|I_t+1\|_2 \overset{(c)}{\leq} \sqrt{1 + \tau\mu(X)}\|X\|_2 + 1 \leq 2\sqrt{1 + \tau\mu(X)}\|X\|_2, \quad \text{(4.48)}$$

where $(b)$ follows from the definition of the spectral norm and the triangle inequality, while $(c)$ mainly follows from the fact that $\|X_i\|_2 \leq \sqrt{1 + \tau\mu(X)}$ because of the Geršgorin disc theorem [58]. We can now fix $q = 2\log N$ and make use of the above bounds to conclude from Proposition 4.2 that

$$E_q\|RH\|_{B,1} \leq 4(\tau + 1)\mu(X)\sqrt{\log N} + 2\sqrt{\delta(1 + \tau\mu(X))}\|X\|_2. \quad \text{(4.49)}$$
We can now substitute (4.33) and (4.34) into the above expression to obtain
\[ E_q \| RH \|_{B,1} \leq \gamma_0 \] with
\[ \gamma_0 \overset{\text{def}}{=} \frac{4}{c\sqrt{\log(N(\tau + 1))}} + \frac{2}{\sqrt{c\log(N(\tau + 1))}} \sqrt{1 + \frac{1}{c\log(N(\tau + 1))}}. \] (4.50)

In order to establish the size condition, we now define the event \( \mathcal{E} = \{ \max_{1 \leq j \leq |S|} \| z_j \|_2 < \gamma \} \) and make use of the Markov inequality along with (4.46) and the preceding discussion to obtain
\[ P(\mathcal{E}^c) \leq \gamma^{-q} \left[ E_q \max_{1 \leq j \leq |S|} \| z_j \|_2 \right]^q \leq \left( \frac{2}{\gamma} E_q \| RH \|_{B,1} \right)^q \leq \left( \frac{2\gamma_0}{\gamma} \right)^q. \]

Finally, we use \( Z \overset{\text{def}}{=} \max_{1 \leq j \leq |S|} | \langle z_j, \text{sgn}(\beta_S) \rangle | \) and conclude that
\[ P(Z \geq t) \leq P(Z \geq t \mid \mathcal{E}) + P(\mathcal{E}^c)^{(d)} \leq 2Ne^{-t^2/(2\gamma^2)} + (2\gamma_0/\gamma)^q, \] (4.51)
where \((d)\) is a consequence of the Hoeffding inequality and the union bound. The condition is now established from (4.43) by setting \( t = 2 \) in the above expression. Furthermore, set
\[ \gamma = \sqrt{\frac{2}{(1 + 2\log 2) \log N}}, \] (4.52)
which leads to \( 2Ne^{-2/\gamma^2} \leq 2N^{-2\log 2} \) and
\[ \frac{\gamma_0}{\gamma} \leq \frac{2(\sqrt{1 + c + 2})}{0.9155c} < 1/4. \] (4.53)

Therefore, we obtain that \( P(\mathcal{E}^c) \leq (1/2)^q \leq N^{-2\log 2} \) and thus we have that the size condition does not hold with probability at most
\[ P(C_c^4 \mid C_1) \leq 3N^{-2\log 2}. \] (4.54)
4.A.5 Complementary Size Condition

In order to establish the complementary size condition, we proceed similar to the case of the “size condition” and define \( z_j \) as

\[
 z_j \overset{\text{def}}{=} (X_S^T X_S)^{-1} X_S^T X_S e_j.
\]

It can then be easily seen that

\[
 \|X_S^T X_S e_j\|_2 \leq 2 \|X_S^T X_S e_j\|_2, \quad j = 1, \ldots, |S_c|.
\]

We now define \( X_{B^c} \overset{\text{def}}{=} \{ X_i : i \in I^c \} \) and consider the set of indices \( T_1 \overset{\text{def}}{=} \{ j' : X_S e_{j'} \text{ is a column in } X_{B^c} \} \). It is then easy to argue by making use of the notation developed in Section 4.A.4 that if \( j \in T_1 \) then

\[
 \|X^T S X_S c e_j\|_2 \leq \max_{i' \in I^c} \|X_{B^c}^T X_i\|_2 \overset{(a)}{=} \|RH\|_{B,1}, \quad \tag{4.55}
\]

where \((a)\) follows from the fact that \( X_{B^c}^T X_{B^c} \) is a submatrix of \( RH \). We therefore have from the discussion following Proposition 4.2 and the Markov inequality that \( \forall j \in T_1 \) and for \( q = 2 \log N \) and \( \gamma > 0 \)

\[
 \mathbb{P}(\|X^T S X_S e_j\|_2 > \gamma) \leq \frac{[\mathbb{E}_q\|RH\|_{B,1}]^q}{\gamma^q} \leq \left( \frac{\gamma_0}{\gamma} \right)^q. \quad \tag{4.56}
\]

Finally, the argument involving \( j \in T_1^c \) is a little more involved but follows along similar lines. Specifically, fix any \( j \in T_1^c \) and define \( i' \in I \) to be such that \( X_S e_j \) is a column of \( X_i \). Next, define \( \tilde{x}_{S \cap i'} \) to be the column of \( X_S \) that lies within the Toeplitz block \( X_i \) and \( \tilde{X}_{S \setminus i'} \) to be the submatrix constructed by removing the column \( \tilde{x}_{S \cap i'} \) from \( X_S \). Then, if we use the notation \( X_{B \setminus \{ i' \}} \overset{\text{def}}{=} \{ X_i : i \in B \setminus \{ i' \} \} \), it can be verified that for any \( j \in T_1^c \) we have

\[
 \|X^T S X_{B \setminus i'} e_j\|_2 \leq \max_{i' \in I} \|X_{B \setminus i'}^T X_{i'}\|_2 + \|\tilde{x}_{S \cap i'}^T X_S e_j\|^2 \leq \|RH\|^2_{B,1} + \mu^2(X). \quad \tag{4.57}
\]
where (b) again makes use of the fact that the spectral norm of a matrix is an upper bound for the spectral norm of any of its submatrices. We therefore once again obtain from the discussion following Proposition 4.2 and the Markov inequality that $\forall j \in T_1^c$ and for $q = 2\log N$ and $\gamma > 0$

$$\mathbb{P}(\|X_S^T X_S e_j\|_2 > \gamma) \leq \mathbb{P}\left(\|R_H\|_{B,1} > \sqrt{\gamma^2 - \mu^2(X)}\right) \leq \left(\frac{\gamma_0}{\sqrt{\gamma^2 - \mu^2(X)}}\right)^q. \quad (4.58)$$

We can now define the event $E = \{\|X_S^T X_S e_j\|_2 \leq \gamma\}$ and use the notation $Z \equiv \max_{1 \leq j \leq |S_c|} |\langle z_j, \text{sgn}(\beta_S) \rangle|$ to conclude from (4.56) and (4.58) that

$$\mathbb{P}(Z \geq t) \leq \mathbb{P}(Z \geq t|E) + \mathbb{P}(E^c) \leq 2N(\tau + 1)e^{-t^2/2\gamma^2} + (\gamma_0/\gamma)^q + (\gamma_0/\sqrt{\gamma^2 - \mu^2(X)})^q, \quad (4.59)$$

where (c) follows from the Hoeffding inequality and the union bound. The condition is now established by setting $t = \frac{1}{4}$ in the above expression. Furthermore, set

$$\gamma = \frac{1}{\sqrt{32(1 + 2\log 2)\log(N(\tau + 1))}}, \quad (4.60)$$

which yields $2N(\tau + 1)e^{-1/32\gamma^2} \leq 2(N(\tau + 1))^{-2\log 2}$ and $\frac{\gamma_0}{\sqrt{\gamma^2 - \mu^2}} \leq \frac{2\sqrt{1/\gamma^2} + \frac{4}{\sqrt{0.1144^2 - 1/e^2}}}{\sqrt{0.1144^2 - 1/e^2}} < 1/2$. Therefore, we obtain that $\mathbb{P}(E^c) \leq 2(\gamma_0/\sqrt{\gamma^2 - \mu^2})^q \leq 2(1/2)^q \leq 2(N(\tau + 1))^{-2\log 2}$ and thus we have that the size condition satisfies

$$\mathbb{P}(C_5^c|C_1) \leq 4(N(\tau + 1))^{-2\log 2}. \quad (4.61)$$

### 4.A.6 Proof of Theorem 4.1

The proof of Theorem 4.1 follows from the preceding discussion by taking a union bound over all the respective conditions and removing the conditionings: $\mathbb{P}(\mathcal{C}_1 \cap \mathcal{C}_2 \cap \ldots \cap \mathcal{C}_n)$
\( C_3 \cap C_4 \cap C_5 \cap C \leq P(C_1) + P(C_2 | C_1) + P(C_3 | C_1) + P(C_4 | C_1) + P(C_5 | C_1). \) Consequently, we obtain that the probability of error is upper bounded by
\[ 2N^{-1} \left( 2\pi \log(N\sqrt{\tau + 1}) \right)^{-1/2} + 5(N(\tau + 1))^{-2\log^2} + 3N^{-2\log^2}. \]

### 4.B Proof of the Main Result: Random Delays

In this appendix, we provide a proof of Theorem 4.2. The proof parallels that of Theorem 4.1, thus the definitions and notation in Appendix 4.A are reused. Key to the proof is the distribution and generation of the support set \( S \), which we examine first.

As described in Section 4.2, here we consider the case when \( \tau_i \) are uniformly selected from \( \{0, \ldots, \tau\} \) at random. Translating the notions of users and delays to the block structure of \( X \), the set \( S \) can be viewed as generated by a two step procedure: (1) blocks are activated with probability \( \delta = k/N \); (2) within each active block, a delay/column is selected uniformly at random. We call this the conventional activation procedure (CAP). However, to prove Theorem 4.2, it is useful to examine a different activation procedure of \( S \) as follows:

1. Let \( \{\tilde{\delta}_i\}_{i=1}^{N(\tau + 1)} \) be a set of Bernoulli random variables with \( \mathbb{P}(\tilde{\delta}_i = 1) = \frac{kp}{N(\tau + 1)} \). Set \( \tilde{S} \) to be the set \( \{i : \tilde{\delta}_i = 1\} \).

2. Mapping indices to the block structure on \( X \), prune \( \tilde{S} \) to \( S \): For each block with more than one active element in \( \tilde{S} \), select a single element uniformly at random among the active elements in the block.

We call this the equivalent activation procedure (EAP) and we now argue that, with an appropriate value of \( \rho \), the set \( S \) is distributed identically to that generated using the conventional procedure. Of particular utility will be the set \( \tilde{S} \) since \( \tilde{S} \supset S \) and \( \tilde{S} \) is generated simply from iid Bernoulli variables. We further define \( \tilde{S}_i \) to be \( \tilde{S} \) restricted to elements in block \( i \).
The value of $\rho$ needed can be calculated by requiring the probability of block activity to be equal under the CAP and the EAP. That is, for any $i = 1, \ldots, N$,

$$\mathbb{P}[\tilde{S}_i > 1] = 1 - \mathbb{P}[\tilde{S}_i = 0] = 1 - \left(1 - \frac{k\rho}{N(\tau + 1)}\right)^{\tau + 1} = \frac{k}{N},$$

(4.62)

where the last equality links the two procedures. Solving for $\rho$ gives

$$\rho = \left(1 - \left(1 - \frac{k}{N}\right)^{\frac{1}{\tau + 1}}\right) \frac{N(\tau + 1)}{k}.$$ 

(4.63)

When $k \ll N$, $(1 - \frac{k}{N})^{1/(\tau+1)} \approx 1 - \frac{k}{N(\tau + 1)}$ and $\rho \approx 1$ as expected. This approximation will be made more explicit later in (4.67).

To prove equivalence in distribution between the two methods, it remains to show independence of blocks and uniformity among columns in blocks in the EAP. Independence of blocks is inherited from the independence of column activation in Step 1 (since the blocks are disjoint sets). We now make a symmetry argument to show a uniform selection of columns.

Let $(i,j)$ be an arbitrary column/block pair. Let $\mathcal{Y}$ be the event that $(i,j)$ is activated in Step 1 and let $\mathcal{X}$ be the event that $(i,j)$ is selected in Step 2. Since the events satisfy $\mathcal{X} \subset \mathcal{Y} \subset \{|\tilde{S}_j| > 0\}$, we can write

$$\mathbb{P}[\mathcal{X}] = \mathbb{P}[\mathcal{X} \cap \mathcal{Y} \cap \{|\tilde{S}_j| > 0\}]$$

$$= \sum_{n=1}^{\tau + 1} \mathbb{P}[\mathcal{X} \cap \mathcal{Y} \cap \{|\tilde{S}_j| = n\}]$$

$$= \sum_{n=1}^{\tau + 1} \mathbb{P}[\mathcal{X} \mid \mathcal{Y} \cap \{|\tilde{S}_j| = n\}] \mathbb{P}[\mathcal{Y} \cap \{|\tilde{S}_j| = n\}]$$

$$= \sum_{n=1}^{\tau + 1} \frac{1}{n} \mathbb{P}[\mathcal{Y} \cap \{|\tilde{S}_j| = n\}],$$

(4.64)
where the last equality is due to the uniform selection in Step 2. Now, for \( n = 1, \ldots, \tau + 1 \), we have

\[
\mathbb{P}[\mathcal{Y} \cap \{|\tilde{S}_j| = n\}] = \mathbb{P}\left[\{|\tilde{S}_j| = n\} \mid \mathcal{Y}\right] \mathbb{P}[\mathcal{Y}]
\]

\[
= \left( \binom{\tau}{n-1} \right) \left( 1 - \frac{k\rho}{(\tau+1)N} \right)^{\tau-n+1} \left( \frac{k\rho}{(\tau+1)N} \right)^{n-1} \left[ \frac{k\rho}{(\tau+1)N} \right]^{-n}
\]

\[
= \left( \binom{\tau}{n-1} \right) \left( 1 - \frac{k\rho}{(\tau+1)N} \right)^{\tau-n+1} \left( \frac{k\rho}{(\tau+1)N} \right)^n,
\]

(4.65)

where the first factor is Binomial over the \( \tau \) remaining columns given that \( i, j \) was selected in Step 1. At this point, it is sufficient to note that (4.65) is not a function of our choice of \((i, j)\). Thus, by symmetry, the probability is equal for all columns. Nonetheless, we complete the calculation to show it takes the anticipated value. Returning to (4.64), we have,

\[
\mathbb{P}[\mathcal{X}] = \sum_{n=1}^{\tau+1} \frac{1}{n} \left( \binom{\tau}{n} \right) \left( 1 - \frac{k\rho}{(\tau+1)N} \right)^{\tau-n+1} \left( \frac{k\rho}{(\tau+1)N} \right)^n
\]

\[
= \sum_{n=1}^{\tau+1} \frac{1}{\tau+1} \left( \binom{\tau+1}{n} \right) \left( 1 - \frac{k\rho}{(\tau+1)N} \right)^{\tau-n+1} \left( \frac{k\rho}{(\tau+1)N} \right)^n
\]

\[
= \frac{1}{\tau+1} \left[ 1 - \left( 1 - \frac{k\rho}{(\tau+1)N} \right)^{\tau+1} \right]
\]

\[
= \frac{1}{\tau+1} \frac{k}{N},
\]

where in the second equality we use a simple identity on Binomial coefficient, in the third equality we note the sum is nearly complete over a Binomial distribution function and lastly, we use (4.62).
Having shown the equivalence between CAP and EAP, we are ready to prove the five conditions $C_1$ through $C_5$ that guarantee recovery. While our model on the users corresponds to the CAP, we will use the EAP in the remainder of the proof. Since $\hat{S}$ is formed from iid random variables, we are able to follow a proof technique similar to that of [46]. We include our proof for completeness since our theory is based on slightly different assumptions and aspects of our proof use different methods.

4.B.1 Invertibility Condition

To bound $\|\left(X_S^T X_S\right)^{-1}\|_2$ we consider $S$ and $\hat{S}$ as generated from EAP. Since $\hat{S}$ is uniformly distributed over possible column selections, we can bound $\|\left(X_S^T X_S\right)^{-1}\|_2$ using methods of [46] where, using [59] and $q = 2 \log \left(N(\tau + 1)\right)$, we have

$$
E_q\|X_S^T X_S - I\|_2 \leq 30 \mu(X) \log(N(\tau + 1)) + 13 \sqrt{\frac{2k\rho\|X\|_2^2 \log(N(\tau + 1))}{N(\tau + 1)}}.
$$

(4.66)

We would like to translate this into conditions similar to (4.33) and (4.34). To do so, we make the approximation noted below (4.63) explicit. We will assume $k/N \leq 1/4$ here, which follows trivially from the condition (4.14) in the theorem. This assumption allows us to make the following approximation.

$$
1 - (1 - k/N)^{1/(\tau + 1)} \leq \frac{k}{N(\tau + 1)}(1 - 1/4)^{1/(\tau + 1)} \leq \frac{k}{N(\tau + 1)}\frac{4}{3}.
$$

(4.67)

The first inequality is an application of Taylor’s remainder theorem on the function $f(x) = 1 - (1-x)^{1/(\tau + 1)}$, while the second inequality is due to the fact that $(1 - \epsilon)^{1/(\tau + 1)} \leq 1$ for $\epsilon \geq 0$. Applying this approximation to (4.63) yields $\rho \leq 4/3$. 

103
Thus, if
\[
\mu(X) \leq \frac{1}{c' \log(N(\tau + 1))} \quad \text{and} \\
\|X\|_2^2 \leq \frac{N(\tau + 1)}{c'k \log(N(\tau + 1))} \leq \frac{N(\tau + 1)}{c''k \rho \log(N(\tau + 1))},
\]
then \(\mathbb{E}_q\|X_S^T X_{\tilde{S}} - I\|_2 \leq 1/4\). Above, \(c'\) and \(c''\) are appropriately chosen constants independent of the problem parameters. Since \(S \subset \tilde{S}\), we have \(\|X_S^T X_S - I\|_2 \leq \|X_{\tilde{S}}^T X_{\tilde{S}} - I\|_2\) and therefore \(\mathbb{E}_q\|X_S^T X_S - I\|_2 \leq 1/4\). Following the calculations in Appendix 4.A.1, cf. (4.36)-(4.37), this gives
\[
\mathbb{P}(C^c_1) \leq (N(\tau + 1))^{-2\log^2 2}.
\]

### 4.B.2 Noise Stability and Complementary Noise Stability

Conditions \(C_2\) and \(C_3\) follow when conditioned on \(C_1\) in an identical manner to Appendix 4.A.2 and Appendix 4.A.3 with probabilities (4.39) and (4.42), respectively.

### 4.B.3 Size Condition

For \(C_4\) we begin as in Appendix 4.A.4 with the following upper bound
\[
\|(X_S^T X_S)^{-1} \text{sgn}(\beta_S)\|_\infty \leq \max_{1 \leq j \leq |S|} |\langle z_j, \text{sgn}(\beta_S) \rangle| + 1,
\]
where \(z_j\) denotes the \(j\)-th column of \((X_S^T X_S)^{-1} - I\). Using the definitions from Section 4.A.4 and additionally defining \(\tilde{A} = X_{\tilde{S}}^T X_{\tilde{S}} - I\), and conditioning on \(C_1\), we have
\[
\|z_j\|_2 \leq 2\|A e_j\|_2 \leq 2\|\tilde{A} e_j\|_2,
\]
then 104
where the first inequality is due to an application of the Neumann series and $C_1$. The second inequality is due to $A$ being a sub-matrix of $\tilde{A}$ and the choice of $j'$ such that the column $X_{S}e_j$ is the same as $X_{S}e_{j'}$.

Next we define $\tilde{R} \overset{\text{def}}{=} \text{diag}(\tilde{\delta}_1, \ldots, \tilde{\delta}_{N(\tau+1)})$ as a selection matrix for the EAP similar to $R$ so that $X_{S} = \tilde{R}X$ (conforming to the first step in the EAP). With this definition we have

$$\|\tilde{A}e_{j'}\|_2 \leq \|\tilde{R}H\|_{1 \rightarrow 2}, \quad (4.72)$$

where $\| \cdot \|_{1 \rightarrow 2}$ denotes the maximal column norm as defined in [59].

Since $\{\tilde{\delta}_i\}$ are iid Bernoulli random variables, we can apply [59, Theorem 3.2] which gives, for $q = 2 \log(N(\tau + 1))$,

$$E_q\|\tilde{R}H\|_{1 \rightarrow 2} \leq 2^{1.75} \sqrt{\log(N(\tau + 1))}\mu(X) + \sqrt{\delta}\|H\|_{1 \rightarrow 2}$$

We can bound $\|H\|_{1 \rightarrow 2}$ as follows:

$$\|H\|_{1 \rightarrow 2} = \max_{1 \leq i \leq N(\tau+1)} \|(X^TX - I)e_i\|_2 \leq \max_{1 \leq i \leq N(\tau+1)} \|(X^T\tilde{x}_i)\|_2 + 1$$

$$\leq \|X\|_2 + 1 \leq 2\|X\|_2$$

where we use $1 \leq \|X\|_2$ since the columns have unit norm. This gives

$$E_q\|\tilde{R}H\|_{1 \rightarrow 2} \leq 2^{1.75} \sqrt{\log(N(\tau + 1))}\mu(X) + 2\sqrt{\delta}\|X\|_2.$$ 

Upon substituting in the values from (4.33) and (4.34), we obtain

$$E_q\|\tilde{R}H\|_{1 \rightarrow 2} \leq \tilde{\gamma}_0$$

(4.73)

with

$$\tilde{\gamma}_0 = \frac{1}{\sqrt{\log(N(\tau + 1))}} \left[ \frac{2^{1.75}}{c} + \frac{2}{\sqrt{c}} \right]$$

(4.74)
In order to establish the size condition, we now define the event \( E = \{ \max_{1 \leq j \leq |S|} \| z_j \|_2 < \gamma \} \) and make use of the Markov inequality along with (4.71), (4.72) and the preceding discussion to obtain

\[
P(E^c) \leq \gamma^{-q} \left[ \mathbb{E}_q \max_{1 \leq j \leq |S|} \| z_j \|_2 \right]^q \leq \left( \frac{2}{\gamma} \mathbb{E}_q \| \tilde{R}H \|_{1 \to 2} \right)^q \leq \left( \frac{2\tilde{\gamma}_0}{\gamma} \right)^q.
\]

Finally, we use \( Z \overset{\text{def}}{=} \max_{1 \leq j \leq |S|} |\langle z_j, \text{sgn}(\beta_S) \rangle| \) and conclude that

\[
P(Z \geq t) \leq P(Z \geq t | E) + P(E^c) \overset{(a)}{=} 2N(\tau + 1)e^{-t^2/2\gamma^2} + (2\tilde{\gamma}_0/\gamma)^q,
\]

where \((a)\) is a consequence of the Hoeffding inequality and the union bound. The condition is now established from (4.43) by setting \( t = 2 \) in the above expression.

Furthermore, set

\[
\gamma = \sqrt{\frac{2}{(1 + 2 \log 2) \log(N(\tau + 1))}},
\]

which leads to \( 2N(\tau + 1)e^{-2/\gamma^2} \leq 2(N(\tau + 1))^{-2\log \gamma^2} \) and

\[
\frac{\tilde{\gamma}_0}{\gamma} \leq \frac{(2^{1.75} + 2\sqrt{c})(1 + 2 \log 2)}{c} < 1/4.
\]

Therefore, we obtain that \( P(E^c) \leq (1/2)^q \leq (N(\tau + 1))^{-2\log \gamma^2} \) and thus we have that the size condition does not hold with probability at most

\[
P(C^c_4 | C_1) \leq 3 \left( N(\tau + 1) \right)^{-2 \log \gamma^2}.
\]
4.B.4 Complementary Size Condition

As in Appendix 4.A.5, we define $z_j$ as $z_j \overset{def}{=} (X_S^TX_S)^{-1}X_S^TX_se_j$. It can then be easily seen that $\|X_S^T(X_S^TX_S)^{-1}sgn(\beta_S)\|_{\infty} = \max_{1 \leq j \leq |S^c|} |\langle z_j, sgn(\beta_S) \rangle|$. Now condition on the event $C_1$ and notice that $\|z_j\|_2 \leq 2\|X_S^TX_Se_j\|_2$, $j = 1, \ldots, |S^c|$.

We then see that

$$\|X_S^TX_Se_j\|_2 \leq \|X_S^TX_S\|_{1 \rightarrow 2} \overset{(a)}{\leq} \|\tilde{R}H\|_{1 \rightarrow 2},$$

where $(a)$ follows from the fact that $X_S^TX_S$ is a submatrix of $\tilde{R}H$. As in the previous subsection, by redefining the event $E = \{\max_{1 \leq j \leq |S^c|} \|z_j\|_2 < \gamma\}$ and $Z = \max_{1 \leq j \leq |S^c|} |\langle z_j, sgn(\beta_S) \rangle|$ and using (4.73), (4.75) and (4.76) hold once again.

In accordance with the complementary size condition, we take $t = 1/4$ and set

$$\gamma = \sqrt{\frac{1}{32(1 + 2 \log 2) \log(N(\tau + 1))}}$$

so that $2N(\tau + 1) e^{-\frac{t^2}{2\tau^2}} = 2(N(\tau + 1))^{-2\log 2}$ and $\tilde{\gamma}_0 / \gamma \leq 1/4$. This gives us

$$\mathbb{P}(C_5^c | C_1) \leq 3(N(\tau + 1))^{-2\log 2}.$$

4.B.5 Proof of Theorem 4.2

The proof of Theorem 4.2 follows from the preceding discussion by taking a union bound over all the respective conditions and removing the conditionings: $\mathbb{P}((C_1 \cap C_2 \cap C_3 \cap C_4 \cap C_5)^c) \leq \mathbb{P}(C_1^c) + \mathbb{P}(C_2^c | C_1) + \mathbb{P}(C_3^c | C_1) + \mathbb{P}(C_4^c | C_1) + \mathbb{P}(C_5^c | C_1)$. Consequently, we obtain that the probability of error is upper bounded by $2N^{-1}(2\pi \log(N\sqrt{\tau + 1}))^{-1/2} + 7(N(\tau + 1))^{-2\log 2}$. 

107
4.C Proof of Lemma 4.3

In order to bound the sum of (4.25), we will use the following propositions.

**Proposition 4.3.** For \( x, y \in \text{GF}(2^m) \) and \( i = 1, 2, \ldots \)

\[
x^{2^i+1} + y^{2^i+1} = (x + y)^{2^i+1} + \sum_{j=0}^{i-1} (xy)^{2^j} (x + y)^{2^{i-2^j+1}+1}
\]

**Proof.** We prove this by induction and application of \((x + y)^{2^i} = x^{2^i} + y^{2^i}\). That is, we first note that the lemma holds for \(i = 1\) and, assuming true for \(i\), we have

\[
(x + y)^{2^{i+1}+1} = (x + y)^{2^i} (x + y)^{2^{i+1}}
\]

\[
= (x + y)^{2^i} [x^{2^{i+1}} + y^{2^{i+1}} + \sum_{j=0}^{i-1} (xy)^{2^j} (x + y)^{2^{i-2^j+1}+1}]
\]

\[
= (x^{2^i} + y^{2^i})(x^{2^{i+1}} + y^{2^{i+1}}) + \sum_{j=0}^{i-1} (xy)^{2^j} (x + y)^{2^{i+1-2^{i+1}+1}}
\]

\[
= x^{2^{i+1}+1} + y^{2^{i+1}+1} + (xy)^{2^i} (x + y) + \sum_{j=0}^{i-1} (xy)^{2^j} (x + y)^{2^{i+1-2^{i+1}+1}}.
\]

Incorporating the middle term in the sum completes the proof. \(\Box\)

**Proposition 4.4** ([53, pp. 278–9]). The quadratic polynomial \(x^2 + fx + g\) with coefficients in \(\text{GF}(2^m)\) and \(f \neq 0\) has two distinct roots in \(\text{GF}(2^m)\) if \(\text{Tr}(g/f^2) = 0\) and no roots in \(\text{GF}(2^m)\) if \(\text{Tr}(g/f^2) = 1\).

**Proposition 4.5.** (a) The cardinality of \(\{g \in \text{GF}(2^m) : \text{Tr}(\alpha g) = c_1\}\) is \(2^{m-1}\) for \(\alpha \in \text{GF}(2^m), \alpha \neq 0 \) and \(c_1 \in \text{GF}(2)\). (b) The cardinality of \(\{g \in \text{GF}(2^m) : \text{Tr}(\alpha g) = c_1, \text{Tr}(\beta g) = c_2\}\) is \(2^{m-2}\) for \(\beta \in \text{GF}(2^m), \beta \neq \alpha, \beta \neq 0 \) and \(c_2 \in \text{GF}(2)\).

**Proof.** Let \(\{\eta_i\}_{i=1}^m\) and \(\{\lambda_i\}_{i=1}^m\) be dual bases of \(\text{GF}(2^m)\) [53, p. 117] and consider \(\alpha\) and \(g\) in these bases respectively as \(\alpha = a_1\eta_1 + \cdots + a_m\eta_m\) and \(g = \gamma_1\lambda_1 + \cdots + \gamma_m\lambda_m\) for \(a_i, \gamma_i \in \text{GF}(2)\). Then \(\text{Tr}(\alpha g) = a_1\gamma_1 + \cdots + a_m\gamma_m = c_1\) is a restriction of a single
degree of freedom in selecting \( \{ \gamma_i \}_{i=1}^m \). Similarly, \( \text{Tr}(\beta g) = c_2 \) restricts an additional degree of freedom.

We are now ready to bound the sum given by (4.25). We will begin with a simple (yet required) case which illustrates our use of Proposition 4.5. Suppose the non-zero vector \( \alpha \) is zero everywhere but at \( \alpha_0 \). In this case we have

\[
S = \sum_{x \in \text{GF}^*(2^m)} (-1)^{\text{Tr}(\alpha_0 x)} = \sum_{x \in \text{GF}(2^m)} (-1)^{\text{Tr}(\alpha_0 x)} - 1
\]

where we’ve completed the sum to be over all of \( \text{GF}(2^m) \). By Proposition 4.5 (a), \( \text{Tr}(\alpha_0 x) = 1 \) for precisely precisely half the \( 2^m \) terms of the sum. Thus, the sum is 0 and \( |S| = 1 \). For the remainder of the proof, we will assume that \( \alpha_i \) is non-zero for some \( i \geq 1 \).

Considering the square of (4.25), by using the linearity of the trace we have

\[
S^2 = \sum_{x \in \text{GF}^*(2^m)} \sum_{y \in \text{GF}^*(2^m)} (-1)^{\text{Tr}[\alpha_0 (x+y)+\sum_{i=1}^t \alpha_i (x^{2^i+1}+y^{2^i+1})]}
\]

\[
= 2^m - 1 + \sum_{x \in \text{GF}^*(2^m)} \sum_{y \in \text{GF}^*(2^m)} (-1)^{\text{Tr}[\alpha_0 (x+y)+\sum_{i=1}^t \alpha_i (x^{2^i+1}+y^{2^i+1})]}
\]

\[
= 2^m - 1 + \sum_{x \in \text{GF}^*(2^m)} \sum_{y \in \text{GF}^*(2^m)} (-1)^{\text{Tr}[\alpha_0 (x+y)+\sum_{i=1}^t \alpha_i ((x+y)^{2^i+1}+\sum_{j=0}^{i-1} (xy)^{2^j}(x+y)^{2^i-2^j+1})]}
\]

In the last equality we have used Proposition 4.3 so that we may apply the change of variables given by \( f = x + y \) and \( g = xy \). To justify this substitution we note that

\[
\{(x+y, xy) : x \in \text{GF}^*(2^m), y \in \text{GF}^*(2^m), y \neq x \}
\]

\[
= \{(f, g) : f \in \text{GF}^*(2^m), g \in \text{GF}^*(2^m), \text{Tr}(g/f^2) = 0 \}.
\]

To see this, consider quadratics \((z+x)(z+y) = z^2 + fz + g\) with non-zero roots. The first set generates all quadratics with two solutions by enumerating the roots while,
by Proposition 4.4, the second set generates the same quadratics by enumerating the coefficients. Since with this substitution both \((x, y)\) and \((y, x)\) map to \((f, g)\) we account for the extra factor of 2 below. We now have

\[
S^2 = 2^m - 1 + 2 \sum_{f, g \in GF^*(2^m) \atop \Tr(g/f^2) = 0} (-1)^{\Tr\left[\alpha_0 f + \sum_{i=1}^t \alpha_i f^{2^i+1}\right]} \sum_{g \in GF^*(2^m) \atop \Tr(g/f^2) = 0} (-1)^{\Tr\left(\sum_{i=1}^t \sum_{j=0}^{i-1} \alpha_i g^{2^j} f^{2^{2^j+i}+1}\right)}
\]

\[
= 2^m - 1 + 2 \sum_{f \in GF^*(2^m)} (-1)^{\Tr\left(\alpha_0 f + \sum_{i=1}^t \alpha_i f^{2^i+1}\right)} \sum_{g \in GF(2^m) \atop \Tr(g/f^2) = 0} (-1)^{\Tr\left(\sum_{i=1}^t \sum_{j=0}^{i-1} \alpha_i g^{2^j} f^{2^{2^j+i}+1}\right)} - 1
\]

\[
\leq 3(2^m - 1) + 2 \sum_{f \in GF^*(2^m)} (-1)^{\Tr\left(\alpha_0 f + \sum_{i=1}^t \alpha_i f^{2^i+1}\right)} \sum_{g \in GF(2^m) \atop \Tr(g/f^2) = 0} (-1)^{\Tr\left(\sum_{i=1}^t \sum_{j=0}^{i-1} \alpha_i g^{2^j} f^{2^{2^j+i}+1}\right)}
\]

\((4.81)\)

where, in the third equality, we’ve completed the sum in \(g\) to include \(g = 0\). The resulting subtraction creates a sum over \(f\) which we trivially bound by \(2^m - 1\). Turning our attention to the innermost sum over \(g\), we will show that the sum is either 0 or \(2^m - 1\). Further, we will bound the number of \(f\) for which it is not zero.

To separate \(g\), we can use the linearity of the trace and \(\Tr(x) = \Tr(x^{2^j})\) for each \(j\) and rewrite the exponent of the inner sum as \(\Tr[(\sum_{i=1}^t \sum_{j=0}^{i-1} \alpha_i^{2^j} f^{2^{2^j+i}+2^j} - 2^j - 2^j - 2)]\) = \(\Tr(\Gamma_f g)\) where we’ve introduced \(\Gamma_f \in GF(2^m)\) to simplify notation. Suppose, for a fixed \(f\), there exists some \(g\) with \(\Tr(g/f^2) = 0\) such that \((-1)^{\Tr(\Gamma_f g)} = -1\). Then we must have \(\Gamma_f \neq 1/f^2\) and \(\Gamma_f \neq 0\). In this case, Proposition 4.5 tells us that the inner sum of \((4.81)\) evaluates to 0 since part (a) gives the size of the sum while (b) shows precisely half the terms take value \((-1)\). Thus, we are interested in when \((-1)^{\Tr(\Gamma_f g)}\) maps all of the subset \(\{g \in GF(2^m) : \Tr(g/f^2) = 0\}\) to 1.

When \((-1)^{\Tr(\Gamma_f g)}\) is a trivial map of the subset, we have \(\{g : \Tr(\Gamma_f g) = 0\} \supset \{g : \Tr(g/f^2) = 0\}\) which provides two cases. The first is that \(\Gamma_f = 0\) and the above
inclusion is strict. In second case, when $\Gamma_f \neq 0$, the sets have same cardinality by Proposition 4.5 and, thus, the two sets are equal. In this case, by $\text{Tr}[(\Gamma_f + 1/f^2)g] = 0 \ \forall g$, the non-degeneracy of the trace [60, Proposition 28.87] tells us $\Gamma_f = 1/f^2$. In both cases, Proposition 4.5 gives the size of the inner sum of (4.81) as $2^{m-1}$. The task now becomes to bound the number of $f$ for which each of these cases occur.

$\Gamma_f = 0$ defines the following polynomial in $f$:

$$0 = \sum_{i=1}^{t} \sum_{j=0}^{i-1} \alpha_i^{2^t-j} f^{2^t-j+2^t-j-2} = \sum_{i=1}^{t} \sum_{j=0}^{i-1} \alpha_i^{2^t-j-1} f^{2^{t+j-1}+2^t-j-1-2^t}$$

where, in the second equality, we’ve used $(x+y)^{2^t-1} = x^{2^t-1} + y^{2^t-1}$ to ensure the powers of $f$ are positive integers. The degree of this polynomial is at most $2^{2t-1} + 2t - 1 - 2^t$ and thus we at most $2^{2t-1} + 2t - 1 - 2^t$ roots at which $\Gamma_f = 0$.

The case for $f^2 \Gamma_f = 1$ is similar and follows the same steps on a slightly different polynomial. In this case we find there are at most $2^{2t-1} + 2t - 1$ values of $f$ for which $\Gamma_f = 1/f^2$. Combining the two cases, we find that there are at most $2^{2t}$ values of $f$ for which $(-1)^{\text{Tr}(\Gamma_f g)}$ is a trivial map over the sum. Returning to (4.81), we’ve found the sum in $f$ have terms with values of either 0 or $\pm 2^{m-1}$ with the non-zeros terms occurring at most $2^{2t}$ times. Thus,

$$S^2 \leq 3(2^m - 1) + 2 \times 2^{2t} \times 2^{m-1} \leq 2^{m+2t+1}$$

Taking the root gives the result.
Chapter 5

Codes for Random Access with Virtual Full Duplex

Communication networks conventionally operate with half-duplex methods and interference avoiding schemes to manage multiple transceivers. Here we consider a method in which nodes transmit and receive in concert to achieve full duplex communication without transmitter coordination. We build on a recent framework for full-duplex communication in ad-hoc wireless networks recently proposed by Zhang, Luo and Guo. An individual node in the wireless network either transmits or it listens to transmissions from other nodes but it cannot do both at the same time. There might be as many nodes as there are MAC addresses but we assume that only a small subset of nodes contribute to the superposition received at any given node in the network. We develop deterministic algebraic coding methods that allow simultaneous communication across the entire network. We call such codes choir codes. Users are assigned subspaces of $\mathbb{F}_{2^m}$ to define their transmit and listen times. Codewords on these subspaces are designed and proven to adhere to bounds on worst-case coherence and the associated matrix spectral norm. This in turn provides guarantees for multiuser detection using convex optimization. Further, we show that matrices
for each receiver’s listening times can be related by permutations, thus guarantee-
ing fairness between receivers. Compared with earlier work using random codes, our
methods have significant improvements including reduced decoding/detection error
and non-asymptotic results. Simulation results verify that, as a method to manage
interference, our scheme has significant advantages over seeking to eliminate or align
interference through extensive exchange of fine-grained channel state information.

5.1 Introduction

In Chapter 4, we introduced the notation of random access networks where wireless
nodes communicate in an uncoordinated fashion. Users are not allocated channels or
time-slots and they independently choose when to transmit their data. Examples of
wireless random access networks include control channels of cellular systems as well as
certain ad hoc and sensor networks. Conventionally, nodes in these scenarios compete
for channel resources using contention resolution schemes such as ALOHA or CSMA.
In these schemes, nodes either wait for acknowledgements and retransmit collided
data or preemptively detect activity on the channel to avoid collisions. However, such
schemes can introduce significant delays and waste channel resources. Addressing this
problem, a new approach has developed which uses the fact that typically only a few
users simultaneously compete for the channel.

Since the set of active users is small, if they simultaneously transmit their code-
words, the signal at a receiver is, in a sense, sparse. This was recognized in [42]
where recent advances in sparse signal recovery were applied to random access uplink
communication. It was shown that data from multiple users could be detected simul-
taneously, without interference avoidance or coordination. This was strengthened and
extended to asynchronous uplink communication in Chapter 4 and [62]. Recently, [63]
introduced a novel scheme in which similar ideas were applied beyond the uplink. It
was shown that network-wide virtual full-duplex communication could be achieved with half-duplex hardware. By switching radios between transmitting and receiving on user specific intervals, nodes could simultaneously recover data from neighbors while transmitting data themselves. This was further developed in [44] and [45]. It is upon the scheme in [63] – [45] which the work in this chapter is based. In this chapter, we consider a code design for the virtual full-duplex framework of [44] for random access wireless networks.

In [44] and [45], randomly generated codewords, defining both the receive periods and transmitted symbols, were considered. It was proved that, using a group testing or message-passing algorithm with the random codewords, data from transmitting nodes can be recovered with high probability. Further, in [44] they simulated the use of deterministic second-order Reed-Muller codewords with random erasures defining receive periods. Using the chirp decoding algorithm of [64] they empirically showed successful recovery. In this chapter, we develop a fully deterministic code with several advantageous properties. Firstly, as a fully deterministic code, storage and generation is relatively simple. Second, by proving bounds on metrics of the codebook, we are able to give data recovery guarantees when a variety of algorithms are used. Finally, we show that the recovery problems exposed to each receiver are equivalent in a manner that ensures fairness between them.

The remainder of the chapter is organized as follows. In Section 5.2 we describe the virtual full-duplex system and its model along with our underlying assumptions. Section 5.3 is the bulk of the chapter and contains the description of our deterministic codes. In the subsections we analyze the code’s properties in the context of three metrics; the worst-case coherence, the average coherence, and the spectral norm. In Section 5.4 we apply our analysis to results from the literature to provide recovery guarantees from the code. Results from simulations are described in Section 5.5 and we conclude our discussion in Section 5.6.
5.2 System Model

We consider a framework like that of [44] in which nodes in a wireless network simultaneously transmit codewords. While the nodes are assumed to be half-duplex devices on the symbol time-scale, we exploit the fact that hardware can rapidly switch between transmission and reception. Codewords are designed to describe not only the transmitted signal but also time periods during which the node is set to receive. In effect, full-duplex is achieved on the codeword time-scale. In this section we describe our model of code transmission and reception while assuming a set of known codewords. In Section 5.3, we make our code explicit.

Let $\mathcal{U}$ be an indexing set for the codewords/users. For each $a \in \mathcal{U}$, we associate a codeword $x_a$ as a vector with elements in $\{-1, +1, 0\}$. We collect these vectors together and define the full codebook matrix as

$$\tilde{X} = \begin{bmatrix} x_1 & x_2 & \cdots & x_{M+1} \end{bmatrix}$$

(5.1)

where, for clarity, we have assumed $\mathcal{U} = \{1, \ldots, M + 1\}$. Using a simple random access model, for each $a \in \mathcal{U}$, we define an independent and identically distributed (iid) Binomial random variable $I_a$ with $\mathbb{P}[I_a = 1] = p_t$. The probability $p_t$ is assumed to be small. The set of active users is defined as $\mathcal{I} = \{a \in \mathcal{U} : I_a = 1\}$ which is a small subset of $\mathcal{U}$. While the codebook is known to users, the set $\mathcal{I}$ is not.

We model the communication as follows. Each user $a \in \mathcal{I}$ simultaneously transmits their codeword $x_a$ by modulating the non-zero elements of $x_a$ on the channel while switching their radio to receive during the 0 elements. Restricted to the time slots of the 0 elements, users receive a truncated version of other users’ codewords. For a receiver $a \in \mathcal{U}$, we define the collapsed codewords of the remaining users as $\{x_b^{(a)}\}_{b \in \mathcal{U} \setminus \{a\}}$ where $x_b^{(a)}$ is the vector $x_b$ with the rows corresponding to the non-zero elements of $x_a$ removed. Correspondingly we define the collapsed measurement matrix

115
as
\[
X_n = \begin{bmatrix}
    x_{b_1}^{(a)} & x_{b_2}^{(a)} & \cdots & x_{b_M}^{(a)}
\end{bmatrix}
\]  \hspace{1cm} (5.2)

where we have indexed \( U/a \) by \( b_1, \ldots, b_M \). \( X_a \) is a sub-matrix of \( \tilde{X} \) with rows and one column removed.

We can model the signal received by user \( a \) as
\[
y_a = X_a \beta + n \hspace{1cm} (5.3)
\]

where \( y_a \) is the vector of samples received during the user’s receiving time slots and \( n \) is vector of noise distributed as \( \mathcal{N}(0, \sigma I) \). The vector \( \beta \) has non-zero elements corresponding to \( \mathcal{I} \) with values determined by the fading and power modulation of transmitting users. Since \( \mathcal{I} \) is a small portion of \( \mathcal{U} \), \( \beta \) has few non-zero components and is a sparse vector.

The goal of user \( a \) is to recover \( \mathcal{I} \) from the support of the vector \( \beta \) to decode the data. Devoid of the communication network context, this problem is known as model selection and, since \( \beta \) is sparse, recent work in sparse recovery and compressed sensing suggest recovery is possible [56], [46]. Indeed, there is a large body work providing recovery conditions and methods for formulations similar to (5.3) when considering a single user. However, unique to this problem is that codewords jointly generate a family of recovery problems (one for each user). In the original paper by Zhang, Luo and Guo [44], vectors \( x_a \) generated at random or partially at random are shown to work in this framework with high probability. In this chapter, we develop a fully deterministic construction.
5.3 A Small Choir Code

In this section we provide a code designed to operate in the framework described in Section 5.2. We construct a code deterministically by operating in the finite field $\mathbb{F}_{2^m}$ with $m$ odd. In particular we take $\mathcal{U} = \mathbb{F}_{2^m}^*$ and enumerate codeword symbols by $\mathbb{F}_{2^m}^*$, where we use $\mathbb{F}_{2^m}^*$ to denote the multiplicative group of the field.

The code makes extensive use of the field trace operator denoted as $\text{Tr}(\cdot)$. As a review, the field trace has the following relevant properties for $a, b \in \mathbb{F}_{2^m}$.

(i) $\text{Tr} : \mathbb{F}_{2^m} \to \mathbb{F}_2$

(ii) $\text{Tr}(a^2) = \text{Tr}(a)$

(iii) $\text{Tr}(a + b) = \text{Tr}(a) + \text{Tr}(b)$

Letting $m$ be odd and $x \in \mathbb{F}_{2^m}^*$ enumerate the elements of a codeword, we define the 

choir code as

\[ [x_a]_x = (-1)^{\text{Tr}(a^3 x^3)} \delta_{\text{Tr}(ax),0} \]

where $\delta$ denotes the Kronecker delta so that elements are only non-zero when $\text{Tr}(ax) = 0$. These elements correspond to the transmission symbols of the codewords. When $\text{Tr}(ax) = 1$, the user $a$ switches its radio to receive signals. To begin, we investigate some properties of the sets of $x$ during which $\text{Tr}(ax) = 0$.

Considering $\mathbb{F}_{2^m}$ as a vector space and using property (iii) of the trace, each user $a$ is associated with a subspace we denote $\mathcal{N}_a = \{ x \in \mathbb{F}_{2^m} : \text{Tr}(ax) = 0 \}$. These subspaces correspond to the transmission times for each user and the complementary sets, denoted $\mathcal{N}_a^c$, define the receiving times. The subspaces have the following useful property.

**Fact 5.1.** For $a_1, \ldots, a_l$ as linearly independent elements in $\mathbb{F}_{2^m}^*$ the cardinality of the set $\mathcal{N}_{a_1} \cap \mathcal{N}_{a_2} \cap \cdots \cap \mathcal{N}_{a_l}$ is $2^{m-l}$. 

117
This fact can be proved using dual bases, for example see Proposition 4.5.

Since each subspace is of size $|\mathcal{N}_a| = 2^{m-1}$, users transmit during approximately half of their length $2^m - 1$ codeword. Further, intersections of subspaces are of size $2^{m-2}$ meaning users are able to receive approximately half of every other users’ transmission. Assigning subspaces of $\mathbb{F}_{2^m}$ to users this way ensures that no user’s transmissions completely mask any other user. While these non-overlapping transmission supports are necessary to allow recovery, they are not sufficient to ensure recovery. Below we consider properties of the matrices $X_a$ which we later apply to recovery guarantees in Section 5.4.

### 5.3.1 Worst-Case Coherence

Worst-case coherence is a common metric found in sparse recovery literature. The worst-case coherence is the largest magnitude of inner-products between columns defined earlier in Section 3.8.1 and in this context is given by

$$\mu(X_a) = \frac{1}{2^{m-2}} \max_{b,c \in U \setminus \{a\}, b \neq c} |\langle x_b^{(a)}, x_c^{(a)} \rangle|$$

where we have added a normalization factor of $2^{-(m-2)}$ to account for the fact that worst-case coherence is customarily applied to unit-normed columns. Using (5.4), we can write the inner-product of columns as the sum

$$\langle x_b^{(a)}, x_c^{(a)} \rangle = \sum_{x \in \mathbb{F}_{2^m}} (-1)^{\text{Tr}((b^3 + c^3)x^3)}$$

where the sum is over all of $\mathbb{F}_{2^m}$ since 0 is already excluded from $\mathcal{N}_a^c$. First, note that if $a = b + c$ the sum is 0 since $\mathcal{N}_a^c \subset (\mathcal{N}_b \cap \mathcal{N}_c)^c$. For the non-trivial case, we have the following lemma.

**Lemma 5.1.** For the linearly independent elements $a, b, c \in \mathbb{F}_{2^m}$, $\langle x_b^{(a)}, x_c^{(a)} \rangle^2 \leq 2^{m+1}$. 

118
Proof. Defining $g = (b^3 + c^3)$ for brevity, we can write the squared sum of (5.6) as

$$\langle x^{(a)}_b, x^{(a)}_c \rangle^2 = \sum_{x, y \in \mathbb{F}_{2^m}} (-1)^{Tr(g(x^3 + y^3))}$$

$$= \sum_{x, y \in \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c} (-1)^{Tr(g((x+y)^3 + xy(x+y)))}$$

(5.7)

$$= \sum_{z, y \in \mathbb{F}_{2^m}} (-1)^{Tr(g(z^3 + zy(z+y)))}$$

where in the last equality we have used the change of variables $z = x + y$. Using the linearity of the trace, we know that $Tr(ax) = 1$ with $Tr(ay) = 1$ implies $Tr(a(x+y)) = 0$. Thus, for all $y \in \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c$, we have $\mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c + y = \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c$. We can therefore factor the above sum as

$$\langle x^{(a)}_b, x^{(a)}_c \rangle^2 = \sum_{z \in \mathbb{F}_{2^m}} (-1)^{Tr(gz^3)} \times$$

$$\sum_{y \in \mathbb{F}_{2^m}} (-1)^{Tr(g(z^2y + y^2z))}$$

(5.8)

$$= \sum_{z \in \mathbb{F}_{2^m}} (-1)^{Tr(gz^3)} \times$$

$$\sum_{y \in \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c} (-1)^{Tr((gz^2 + \sqrt{g}z)y)}$$

where for the final equality we have used the properties (ii) and (iiiA) of the trace in the inner sum. Focusing on the inner sum, in the exponent we have a linear function of $y$ with the null space $\mathcal{N}_{gz^2 + \sqrt{g}z}$. For most values of $z$, the inner sum is 0 since, by Fact 5.1, precisely half the summands are $(-1)$. However, if $\mathcal{N}_{gz^2 + \sqrt{g}z} \supset \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c$, all the summands are equal and the inner sum evaluates to $\pm |\mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c| = \pm 2^{m-3}$. In what follows, we bound the number of $z$ in the outer sum for which this occurs.
The condition $\mathcal{N}_{g^2 + \sqrt{g}} \supset \mathcal{N}_a \cap \mathcal{N}_b \cap \mathcal{N}_c$ is equivalent to $gz^2 + g\sqrt{g} = s$ for $s \in \text{Span}\{a, b, c\}$. By Proposition 5.1 in the Appendix, this equation is linear with two solutions to the homogeneous equation. Therefore, there are at most $2 \times |\text{Span}\{a, b, c\}| = 2^4$ values of $z$ for which the inner sum of (5.8) evaluates to $\pm 2^{m-3}$. Applying the triangle inequality on the outer sum yields the result. \hfill \Box

An application of the above lemma to the definition in (5.5) yields the following theorem.

**Theorem 5.1.** For any $a \in \mathbb{F}_{2^m}$, $\mu(X_a) \leq 2^{-(m+2)}$.

### 5.3.2 Average Coherence

Where the worst-case coherence considers the magnitude of inner-products pairwise, the average coherence considers the inner-product with the average received codeword. As a metric, it is useful for guaranteeing support recovery using one-step thresholding [56]. The average coherence is defined as

$$\nu(X_a) = \frac{1}{2^{m-2} |\mathcal{U}|} \frac{1}{2} \max_{b \in \mathcal{U}\backslash\{a\}} \left| \sum_{c \in \mathcal{U}\backslash\{a, b\}} \langle x_a^{(a)} b, x_a^{(a)} c \rangle \right| \tag{5.9}$$

where, once again, we have added an additional factor of $2^{-(m-2)}$ to account for codeword normalization.

To bound the average coherence, we begin with a simple fact about the full code matrix $\tilde{X}$.

**Lemma 5.2.** With appropriate enumeration of the users and codeword elements, the matrix $\tilde{X}$ is circulant.

**Proof.** Let $z$ be a generator for the multiplicative group $\mathbb{F}^{*}_{2^m}$ and consider the following enumeration of the codeword elements and users. Let the $i$th element be indexed by
$x_i = z^i$ and the $j$th user be indexed by $a_j = z^{-j}$. Thus, elements of the matrix $\tilde{X}$ are given by

$$[\tilde{X}]_{i,j} = (-1)^{\text{Tr}(a_j^3 x_i^j)} = (-1)^{\text{Tr}(z^{3(i-j)})}$$

(5.10)

which is a function of $(i - j) \mod 2^m - 1$. □

Since $\tilde{X}$ is circulant, the row sums of $\tilde{X}$, (and $X_a$) are constant. We denote this value $R$ and can say the following.

**Lemma 5.3.** The row sum of $\tilde{X}$ is $R = \pm 2^{m-1} - 1$.

**Proof.** Completing the row with a element of value 1 to sum over $\mathbb{F}_{2^m}$ rather than $\mathbb{F}_{2^m}^*$ and taking our arbitrary row to be $x = 1$, we have

$$R + 1 = \sum_{a \in \mathbb{F}_{2^m}} (-1)^{\text{Tr}(a^3)}. \quad (5.11)$$

Squaring the sum gives

$$(R + 1)^2 = \sum_{a,b \in N_1} (-1)^{\text{Tr}(a^3 + b^3)}$$

$$= \sum_{a,b \in N_1} (-1)^{\text{Tr}((a+b)^3 + ab(a+b))}$$

$$= \sum_{w \in N_1} (-1)^{\text{Tr}(w^3)} \sum_{b \in N_1} (-1)^{\text{Tr}(w^2 b + b^2 w)}$$

$$= \sum_{w \in N_1} (-1)^{\text{Tr}(w^3)} \sum_{b \in N_1} (-1)^{\text{Tr}(w^2 + \sqrt{w}) b}$$

(5.12)

where in the final two equalities we first made the substitution $w = a + b$ and second, as we did in (5.8), used properties of the trace to produce a linear function of $b$ in the exponent. The inner sum is precisely zero unless $w^2 + \sqrt{w} = 0$ or 1 and we consider these two cases in turn.

Using the fact that the field has characteristic 2, solutions to $w^2 + \sqrt{w} = 1$ are also solutions to $w^4 + w + 1 = 0$. This is an irreducible polynomial and therefore its
solutions are in $\mathbb{F}_4$. However, since we take $m$ to be odd, $\mathbb{F}_4$ is not a sub-field. Thus, no solutions exists to $w^2 + \sqrt{w} = 1$ in $\mathbb{F}_{2^m}$ for $m$ odd.

By Proposition 5.1 in the Appendix, the solutions to $w^2 + \sqrt{w} = 0$ are 0 and 1. However, only $w = 0$ is an element of $\mathcal{N}_1$. Thus, the outer sum has only one non-trivial term with the value $(R + 1)^2 = |\mathcal{N}_1| = 2^{m-1}$.

In addition to the row sum, we also require a bound on sums of arbitrary columns of the collapsed matrix $X_a$. The following lemma provides such a bound.

**Lemma 5.4.** For any $b \neq a \in \mathbb{F}_{2^m}^*$, the sum of a collapsed codeword obeys $\langle x_b^{(a)}, 1 \rangle \leq 2^{m+1}$, where $1$ is the vector of all 1.

**Proof.** The proof is similar to that of Lemma 5.1 since the column sum has the form

$$\langle x_b^{(a)}, 1 \rangle = \sum_{x \in \mathcal{N}_a \cap \mathcal{N}_b} (-1)^{\text{Tr}(b^3x^3)}$$  \hspace{1cm} \text{(5.13)}$$

which differs from (5.6) only in the subspaces of the index and the coefficient of $x^3$. Letting $g = b^3$ and following similar manipulations yields an equation identical to (5.8), though with the outer and inner sum indices as $\mathcal{N}_a \cap \mathcal{N}_b$ and $\mathcal{N}_a^c \cap \mathcal{N}_b$, respectively. In this case, $|\mathcal{N}_a^c \cap \mathcal{N}_b| = 2^{m-2}$. Further $gz^2 + \sqrt{gz} \in \text{Span}\{a, b\}$ has at most $2^3$ solutions for $z$. As a result, applying the triangle inequality yields

$$\langle x_b^{(a)}, 1 \rangle^2 \leq 2^3 \times 2^{m-2}.$$  \hspace{1cm} \text{(5.14)}$$

Using the results of Lemmas 5.3 and 5.4 we can derive the following bound on average coherence.

**Theorem 5.2.** For any $a \in \mathbb{F}_{2^m}^*$, $\nu(X_a) \leq 5/(2^m - 3)$. 

122
Proof. Using the fact established in Lemma 5.2 that the row sums of $X_a$ are constant, we can write
\[
\sum_{c \in \mathcal{U}/\{a,b\}} x_c^{(a)} = R1 - x_b^{(a)}.
\] (5.15)

Applying this to the inner-product of (5.9) gives
\[
\sum_{c \in \mathcal{U}/\{a,b\}} \langle x_c^{(a)}, x_b^{(a)} \rangle = R\langle x_b^{(a)}, 1 \rangle - \langle x_b^{(a)}, x_b^{(a)} \rangle 
\leq 2^m \left( \frac{3}{4} + 2^{1-m} \right) \] (5.16)

where we have used Lemmas 5.3 and 5.4 with the fact that $\langle x_b^{(a)}, x_b^{(a)} \rangle = 2^{m-2}$. Taking $m \geq 3$ to bound the second term in the brackets by $1/2$ and including the additional multiplicative factors in (5.9) gives the result. \hfill \Box

5.3.3 Spectral Norm

Here, we investigate the spectral norm (or induced $\ell_2$ norm) $\|X_a\|_2$. This norm can be used to give recovery guarantees for the lasso algorithm. We begin with the simple fact that, since $X_a$ is a sub-matrix of $\tilde{X}$, $\|X_a\|_2 \leq \|\tilde{X}\|_2$. Thus, the bulk of this subsection is devoted to bounding the spectral norm of the full codebook matrix.

The spectral norm of $\tilde{X}$ is determined by the eigenvalues of the Gram matrix $\tilde{X}^T\tilde{X}$. We first investigate the off diagonal entries of the Gram matrix.

**Lemma 5.5.** For $x \neq y \in \mathbb{F}_{2^m}$ indexing a row and column, the element $[\tilde{X}^T\tilde{X}]_{x,y}$ is either $-1$ or $\pm 2^{m-1} - 1$.

**Proof.** Let $S$ be the element at location $x, y$. Its value is given by the sum
\[
(S + 1) = \sum_{a \in N_x \cap N_y} (-1)^{\text{Tr}(ax^3 + ay^3)}
\]
(5.17)
\[
= \sum_{a \in N_x \cap N_y} (-1)^{\text{Tr}(ga^3)}
\]
where we have added 1 to allow a sum over $\mathbb{F}_{2^m}$ rather than $\mathbb{F}_{2^m}^*$ and defined $g = x^3 + y^3$.

Following similar steps to (5.7) and (5.8) we have

$$(S + 1)^2 = \sum_{a,b \in \mathcal{N}_x \cap \mathcal{N}_y} (-1)^{\text{Tr}(g((a+b)^3 + ab(a+b)))}$$

$$= \sum_{z \in \mathcal{N}_x \cap \mathcal{N}_y} (-1)^{\text{Tr}(gz^3)} \times \sum_{b \in \mathcal{N}_x \cap \mathcal{N}_y} (-1)^{\text{Tr}((gz^2 + \sqrt{g}z)b)}$$

(5.18)

where we have used the change of variables $z = a + b$. As we have found in earlier sections, unless $gz^2 + \sqrt{g}z \in \text{Span}\{x, y\}$, the inner sum vanishes which motivates the definition of the set $\mathcal{Z} = \{z \in \mathbb{F}_{2^m} : gz^2 + \sqrt{g}z \in \text{Span}\{x, y\}\}$. Unlike in earlier sections, we can take advantage of not indexing over a complementary set such as $\mathcal{N}_a^c$. Here, the inner sum is identically $|\mathcal{N}_x \cap \mathcal{N}_y| = 2^{m-2}$ for $z \in \mathcal{Z}$ (i.e., we know the sign is not negative). As result, we have

$$\frac{(S + 1)^2}{2^{m-2}} = \sum_{z \in \mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}} (-1)^{\text{Tr}(gz^3)}$$

(5.19)

where we have included $\mathcal{Z}$ in the index to reflect that they are the only terms that remain in the outer sum. By a simple application of Proposition 5.1 in the Appendix, $\mathcal{Z}$ and consequently $\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}$ is a subspace of $\mathbb{F}_{2^m}$. Therefore, seeing that (5.19)
is similar to (5.17), we can apply the same transformations in (5.18) and find

$$\left(\frac{(S + 1)^2}{2^{m-2}}\right)^2 = \sum_{z \in \mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}} (-1)^{\text{Tr}(g^3)} \times \sum_{b \in \mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}} (-1)^{\text{Tr}(g^2 + \sqrt{\mathcal{Z}}b)}$$

$$= \sum_{z \in \mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}} (-1)^{\text{Tr}(g^3)} \times |\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}| \cdot \frac{(S + 1)^2}{2^{m-2}}$$

(5.20)

where, in the second equality we use the fact that outer sum is restricted to making the inner sum constant for every $z$. In the last equality, we use that the sum in (5.19) has reemerged. Thus, we have a quadratic equation for $(S + 1)^2$ which has two solutions, $(S + 1)^2 = 0$ or

$$(S + 1)^2 = |\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}| \times 2^{m-2}. \quad (5.21)$$

Focusing on the second case, we first note this can occur only if all the terms in (5.19) are 1. Considering that $z = g^{-1/3} \in \mathcal{Z}$ has the violating property $\text{Tr}(g^3) = \text{Tr}(1) = 1$, we find $\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}$ is a strict subset of $\mathcal{Z}$. Next, since $|\text{Span}\{x, y\}| = 4$, using Proposition 5.1, $|\mathcal{Z}|$ is at most 8. However, by definition, $(S + 1)$ must be an integer. Therefore, to make (5.21) a perfect square, $|\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}|$ must be an odd power of 2. The only possibility is $|\mathcal{N}_x \cap \mathcal{N}_y \cap \mathcal{Z}| = 2$ which gives the result. □

Having characterized the entries of $[\tilde{X}^T \tilde{X}]_{x,y}$ we can bound the norm of $X_a$ with the following theorem. We state the result in terms of $\frac{1}{2^{m-2}}\|X_a\|_2^2$, since conventionally the spectral norm is calculated with normalized columns.

**Theorem 5.3.** For arbitrary $a \in \mathbb{F}_2^m$, $\frac{1}{2^{m-2}}\|X_a\|_2^2 \leq 2^{m-5}$

**Proof.** The diagonal entries of $\tilde{X}^T \tilde{X}$ are all $2^m - 1$ while the off-diagonal entries are, by Lemma 5.5, bounded in magnitude by $2^{\frac{m-1}{2}} + 1$. Thus, by Gershgorin’s circle...
Figure 5.1: Comparison of spectral norm bounds for $\mathbf{X}_a$ for various values of $m$ in a semi-log plot. Normalizations are applied for comparison with Theorem 5.3.

theorem [38],

$$\|\tilde{\mathbf{X}}\|_2^2 = \lambda_{\text{max}}(\tilde{\mathbf{X}}^T \tilde{\mathbf{X}})$$

$$\leq (2^{m-1} - 1) + (2^{m-1} + 1)(2^m - 2)$$

$$\leq 2^{3m+1}$$

(5.22)

Since $\mathbf{X}_a$ is a sub-matrix of $\tilde{\mathbf{X}}$, it also obeys this bound. Normalizing by $\frac{1}{2^m - 2}$ gives the result.

We concede that the bound in Theorem 5.3 is not tight due to the crude application of Gershgorin circle theorem. However, it is possible to efficiently calculate a tighter bound on the spectral norm. In particular, from Lemma 5.2 we know that $\tilde{\mathbf{X}}$ is diagonalized by the Fourier matrix. Therefore, $\|\tilde{\mathbf{X}}\|_2$ can be calculated with the computationally efficient Fast Fourier Transform (FFT) applied on any codeword $\mathbf{x}_a$. This in turn bounds $\|\mathbf{X}_a\|_2$. In Figure 5.1 we display computations of this bound. For comparison, we include direct computations of $\frac{1}{\sqrt{2^m - 2}}\|\mathbf{X}_a\|_2$ for small values of $m$. We also include a the lower bound $\sqrt{\frac{2^m - 2}{2^m - 1}}$ provided by a tight frame of unit normed columns with dimensions of $\mathbf{X}_a$. Figure 5.1 shows that the Theorem 5.3 is loose but
a tighter bound is computable. Further, we conjecture that \( \frac{1}{\sqrt{2^m}} \|X_a\|_2 \) scales like a tight frame. Methods extending those of Lemma 5.5 to give this result are in progress.

### 5.3.4 Receiver Fairness

While the above results for \( \mu(X_a), \nu(X_a) \), and \( \|X_a\|_2 \) apply to every user, they are upper bounds and apply to sufficiency conditions in Section 5.4. As such, it may be the case that some collapsed measurement matrices perform better than others. In this section we show that this is not the case. We do so by showing that the collapsed measurement matrices are related by row and column permutations.

We define the following matrices to aid in our discussion. For \( a \in \mathbb{F}_{2^m}^* \) let \( C_a \) be the \((2^m - 1) \times (2^m - 1)\) matrix which, when multiplied on the left, “zeros” rows indexed by \( x \in \mathbb{F}_{2^m}^* \) that satisfy \( \text{Tr}(ax) = 0 \). That is, \( C_a \) is the identity matrix with the defined rows/columns set to zero. Further, for \( \alpha \in \mathbb{F}_{2^m}^* \), define \( T_\alpha \) as the \((2^m - 1) \times (2^m - 1)\) permutation matrix which, when multiplied on the right, permutes columns such that column \( a \) is moved to column \( a\alpha \). From the basic facts of permutation matrices \( T_\alpha \) is also a row permutation matrix which, when multiplied on the left, permutes rows such that row \( x \) is moved to row \( x\alpha^{-1} \).

\( C_a \tilde{X} \) is very closely related to \( X_a \). \( X_a \) is a sub-matrix of \( C_a \tilde{X} \) since the latter merely contains extra rows of zeros and one extra column of zeros. We will show that \( C_b \tilde{X} \) is formed from a permutation of rows and columns of \( C_a \tilde{X} \). As sub-matrices, \( X_a \) and \( X_b \) are like-wise related by permutations. We begin with the following lemmas.

**Lemma 5.6.**

\[
C_b T_{ba^{-1}} = T_{ba^{-1}} C_a \tag{5.23}
\]

**Proof.** The following compound operations are equivalent:

- moving row \( x \) to \( b^{-1}ax \) then setting it to zero if \( \text{Tr}(bb^{-1}ax) = \text{Tr}(ax) = 0 \)
• setting row $x$ to zero if $\text{Tr}(ax) = 0$ then moving to $b^{-1}ax$

Lemma 5.7.

$$T_\alpha \tilde{X} = \tilde{X} T_{\alpha^{-1}}$$ (5.24)

Proof. By a fact of permutation matrices, $T_{\alpha}^{-1} = T_{\alpha^{-1}}$. Thus, here we equivalently show $T_{\alpha} \tilde{X} T_{\alpha} = \tilde{X}$. The element at location $(x,a)$ is moved to $(a\alpha, x\alpha^{-1})$ by the pre and post multiplication of $T_{\alpha}$. The value at $(a\alpha, x\alpha^{-1})$ is $(-1)^{\text{Tr}(a^3 \alpha^{-3} x^3)} \delta(\text{Tr}(a\alpha x) = 0)$ which is equal to the value at $(x,a)$. □

Theorem 5.4. For $a \neq b \in \mathbb{F}_{2^m}$, $X_b$ is a permutation of the rows and columns of $X_a$.

Proof. As noted above, since $X_b$ and $X_a$ are sub-matrices, it is enough to show that $C_b \tilde{X}$ is a permutation of $C_a \tilde{X}$.

$$C_b \tilde{X} = T_{ba^{-1}} C_a T_{b^{-1}a} \tilde{X} = T_{ba^{-1}} C_a \tilde{X} T_{ba^{-1}}$$ (5.25)

where the first equality is due to Lemma 5.6 and the second equality is due to Lemma 5.7. □

5.4 Performance with Recovery Methods

In this section, we take the Theorems 5.1–5.3 of Section 5.3 and apply them to known results in the literature which can guarantee the recovery of $\beta$ or $I$ for the problem posed in Section 5.2.
5.4.1 Restricted Isometry Property

Perhaps the best known recovery guarantees for sparse signals are those based on the restricted isometry property (RIP). For example, in [7], the RIP is used to provide sparse signal recovery guarantees using a linear program. Recall from Section 3.3, we say a matrix $A$ with unit-normed columns satisfies the RIP of order $S$ with parameter $\delta_S$ is if

$$
(1 - \delta_S)\|v\|_2^2 \leq \|A_Sv\|_2^2 \leq (1 + \delta_S)\|v\|_2^2
$$

(5.26)

for all $v \in \mathbb{R}^S$ and for all sub-matrices $A_S$ of $A$ constructed by selecting $S$ columns. While the choir codes were not designed with the RIP in mind, due to its prolific nature we characterize the RIP of the matrix $X_a$ using $\mu(X_a)$.

The condition in (5.26) is equivalently a bound on the eigenvalues of $A_S^TA_S$. Using the methods of [10] any eigenvalue $\lambda$ of $A_S^TA_S$ satisfies $|\lambda - 1| \leq (S - 1)\mu(A)$. Applied to $X_a$ we have that $X_a$ satisfies the RIP for all $\delta_S$ and $S$ satisfying

$$
\delta_S \geq 2^{-\frac{m + 5}{2}}(S - 1).
$$

(5.27)

This RIP result can be applied to a wide variety of recovery methods. For example, applied to the Dantzig selector [26], gives a guarantee that $\beta$ is estimated accurately when

$$
|\mathcal{I}| \leq C_d 2^{\frac{m}{2}}
$$

(5.28)

for a known constant $C_d$. This result is somewhat weak due to the reliance on $\mu(X_a)$ to prove the RIP.

Recent advances, however, have provided recovery guarantees that depend directly on the metrics proved in Section 5.3 rather than the RIP. Further, they consider the support recovery problem directly and address estimates of $\mathcal{I}$ rather than estimates of $\beta$. We consider two of these results in the subsections below.
5.4.2 One-Step Thresholding

The model selection problem of estimating $\mathcal{I}$ from $y_a$ using one-step thresholding is studied in [56]. One-step thresholding is the simple algorithm of back-projecting $y_a$ onto $X_a^T$ and thresholding the resulting vector. The two conditions

$$
\mu(X_a) \leq \frac{0.1}{\sqrt{2 \log(2^m - 1)}} \quad \text{and} \quad (5.29)
$$

$$
\nu(X_a) \leq \frac{\mu(X_a)}{\sqrt{2^{m-1} - 1}} \quad (5.30)
$$

are proven to allow one-step thresholding to recover $\mathcal{I}$ with high probability. Using Theorems 5.1 and 5.2, choir codes satisfy both conditions. As a result, [56] gives the following guarantee. With an appropriately chosen threshold, one-step threshold recovers $\mathcal{I}$ with high probability when

$$
|\mathcal{I}| \leq C_O \frac{2^{m-1} - 1}{m \log 2} \quad (5.31)
$$

for a known constant $C_O$ dependent on the noise [56, Theorem 1]. Compared with (5.28), we find that with one-step thresholding a large set $\mathcal{I}$ can be guaranteed to be recovered.

5.4.3 The Lasso

We introduced the lasso earlier in Section 4.3. Using the variables of interest in this chapter, the lasso minimization given by

$$
\hat{\beta} = \arg \min_b \frac{1}{2} \|y_a - X_a b\|_2^2 - \lambda \sigma \|b\|_1 \quad (5.32)
$$

is an estimation technique for the sparse signal $\beta$. It is studied in [46] in the context of model selection, whereby an estimate of $\mathcal{I}$ is formed from the support of $\hat{\beta}$. In [46],
the following two conditions are given.

\[ \mu(X_a) \leq \frac{C_{L_0}}{\log(2^m - 1)} \quad (5.33) \]

\[ |\beta|_a > 8\sigma \sqrt{2 \log(2^m - 1)} \quad \forall a \in I \quad (5.34) \]

where \( C_{L_0} \) is a known constant. For the choir code, (5.33) is satisfied by Theorem 5.1, while (5.34) is a mild requirement on the received signal power. When satisfied, \( I \) is successfully recovered with high probability as long as \( |I| \leq C_{L_1} \frac{2^{m-1}}{m^2 \log(2^m - 1)} \) for a known constant \( C_{L_1} \) [46, Theorem 1.3]. Applying Theorem 5.3 and assuming received powers satisfy (5.34), recovery of \( I \) is assured with high probability if

\[ |I| \leq C_{L_2} \frac{2^m}{m \log 2} \quad (5.35) \]

for a known constant \( C_{L_2} \). This scales slightly worse than (5.28). However, the bound on \( \|X_a\|_2 \) calculable using the FFT can be used. As shown in Figure 5.1 and discussed in Section 5.3.3, evidence shows that \( \|X_a\|_2 \) scales as a tight frame. In this case,

\[ |I| \leq C_{L_3} \frac{2^m}{m \log 2} \quad (5.36) \]

guarantees recovery. This scales as (5.31).

In Section 5.5 we simulate the use of the choir code with lasso recovery.

## 5.5 Receiver Simulations

To verify the results presented in this chapter, we use simulations of a receiver in a network using choir codes. In our experiments, we take \( m = 11 \) and select an arbitrary user \( a \) as a receiver. The active user set \( I \) and Gaussian noise \( n \) is generated at random in Monte Carlo iterations. We populate the vector \( \beta \) on the support corresponding to
I with the value 1 and choose the noise parameter $\sigma$ such that the SNR is 20dB. 500 Monte Carlo trials are used for each experiment. We choose the lasso as our recovery method and use the SpaRSA [50] package as a solver.

For comparison, we also simulate a receiver in a network using randomly generated codewords. The random codewords are generated with iid symbols. In expectation, half the symbols are 0. The transmitted symbols are $\pm 1$ with equal probability. The random codebook is generated once per experiment.

Results of these simulations are shown in Figure 5.2. The experiments show the average quality of recovery of $I$ as a function of the sparsity $E[|I|]$. The sparsity level is adjusted via activation probability $p_t$. The quality of recovery is measured as the average size of the error set $(I \cap \hat{I}^c) \cup (I^c \cap \hat{I})$ (i.e., the average number of missed detections and false positives). We see that we are able to recover the active user set with few errors when the average number of users is less than 70. By comparison, we find that the randomly generated code begins to show significant errors with a smaller active user set sizes.
5.6 Conclusion

In this chapter, we introduced choir codes for use in random access wireless networks. The code’s intent is to allow fullduplex communication in the network by including 0 symbols indicating sampling periods during which a user’s radio is set to receive. We allocate the 0 symbols to users by assigning subspaces of $\mathbb{F}_{2^m}$, the field upon which the code is defined. This ensures each user can receive a sufficient portion of other users’ transmitted codewords when restricted to the listening symbols. The set of codewords creates a family of estimation problems where each user must recover data from sets of collapsed codewords of other users. On the sets of these collapsed codewords, we calculate bounds on three important metrics: worst-case coherence, average coherence and spectral norm. We draw these metrics from literature on sparse signal recovery and model selection. By bounding them, we provide guarantees that users can recover transmitted data when receivers use one-step thresholding, the lasso or various other algorithms. Further, we show that no user is at a disadvantage to another by proving that the recovery problems are equivalent via permutations.

Compared to past work, choir codes have several advantages. Firstly, the code is a purely deterministic construction. As such, allocation, storage and retrieval of codewords and the matrices $X_a$ is relatively simple. The code also exhibits performance benefits. As shown in Figure 5.2, the performance of the code exceeds that of the randomly generated codewords of [44] when using the lasso. We expect that this extends to other recovery methods as well. Further, the results in Section 5.3 and their subsequent application in Section 5.4 are non-asymptotic which gives designers more insight into parameter selection.

These codes represent a movement away from avoiding interference to managing interference. Conventional peer-to-peer random access wireless networks operate with orthogonal signaling or collision detection and avoidance mechanisms. This can be
costly in delays or pre-communication coordination. Choir codewords, on the other hand, work in harmony to provide simultaneous network-wide communication.

5.A Appendix

**Proposition 5.1.** For $m$ odd and $g \in \mathbb{F}_{2^m}$, let $f : x \mapsto gx^2 + \sqrt{gx}$. Then $f$ is linear and $f = 0$ has two solutions given by $x = 0$ and $x = g^{-1/3}$.

*Proof.* Since $\mathbb{F}_{2^m}$ has characteristic 2, $(x + y)^2 = x^2 + y^2$. Further, $\sqrt{x} = x^{2^{(m-1)}}$. The linearity of $f$ comes from these two facts.

Solutions to $f = 0$ also satisfy $f^2 = 0$ which factors as $gx(gx^3 + 1) = 0$. Thus, the two solutions are $x = 0$ and $x = g^{-1/3}$. The cubed root is well defined since, by Proposition 5.2, $(\frac{a}{b})^3 = 1$ has the unique solution $a = b$. Therefore, $a \mapsto a^3$ is a bijection. \qed

**Proposition 5.2.** For $m$ odd, there are no non-trivial cubed roots of unity in $\mathbb{F}_{2^m}$.

*Proof.* Assume a non-trivial cubed root exists. Since the order of an element must divide the order of $\mathbb{F}_{2^m}^*$, we must have $3 \mid 2^m - 1$. However, by [65, Theorem 2.3], $\gcd(3, 2^m - 1) = 2^{\gcd(2, m)} - 1 = 1$, where for the last equality we use the fact that $m$ is odd. \qed
Chapter 6

Summary and Future Directions

In this dissertation we have developed novel compressed sensing theory and considered its applications to wireless communication. Compressed sensing is a recent development allowing sparse signals to be reconstructed from fewer measurements than preceding methods imply. Measurements in compressed sensing are generalized as inner-products and form a measurement matrix applied to the data. A particular focus in this thesis has been how measurement matrices benefit from coded design and are influenced by application.

In Chapter 2, we considered a measurement matrix design for recovery speed. We developed a compressed sensing measurement matrix jointly with an algorithm that allowed recovery in $O(M \log M)$ computational time for $M$ measurements. The design was based on discrete chirp signals which, after manipulation, allowed for sparse recovery using the FFT. Recovery using the “chirp codes” outperforms matched pursuit with Gaussian random measurements, a standard recovery method for compressed sensing, while requiring much less computation.

Chapter 3 investigated OFDM channel estimation for sparse channels. In the chapter, we described how channel estimation can be posed as a compressed sensing problem with the measurement matrix formed as rows of a Fourier matrix. Further,
we defined a selection procedure of pilot tones which formed matrices that permitted recovery with standard compressed sensing methods.

A novel multiple user detection scheme was developed Chapter 4. The asynchronous channel model used imposed additional structure on the matrix of receivable signals, acting as a measurement matrix. Compressed sensing theory was extended to address the particulars of the multiuser detection problem and the asynchronous channel. Further, a coding scheme was designed to fit the particular recovery problem.

Lastly, in Chapter 5 we showed how a virtual full duplex system could be implemented with nodes using only half-duplex hardware. Codewords were designed which included symbols to indicate switching hardware to receive. Further, they were designed to ensure that each node was presented with a unique compressed sensing problem to recover the random access network data.

These chapters show that the coding of measurements, both for compressed sensing generally as well as for applications in wireless communication, is fruitful. They do not, however, exhaust possibilities.

From a coding perspective, the designs presented here occupy a discrete set of points in the space of possibilities. Consider, as an analogy, algebraic channel codes. Each family of channel codes occupies a set of points in the space of error probability in noise vs rate. Similarly, in this thesis we have designed deterministic compressed sensing measurements which occupy points in the space of recoverable sparsity $K$, measurements $M$ and data dimension $N$. Other measurement matrix designs can trade-off these metrics differently and give systems designers more flexibility in choosing operating points. In the context of the various chapters, distinct code designs correspond to supporting a larger variety of signal types, channel types and network characteristics.

Beyond the applications outlined in the chapters, many other problems require structured measurement designs. Examples within communication technologies are
network monitoring and anomaly detection [66] as well as sensor network data aggregation [67]. Other examples include seismic imaging, magnetic resonance imaging (MRI) and coded apertures. These problems each impose a unique structure on the types of measurements that are possible and benefit from tailored deterministic measurement designs. While compressed sensing is a versatile paradigm, practical systems require designed measurements as well as adherence to constraints. As a result there are many opportunities for extending results and techniques developed in thesis to other problems.

Finally, as shown in Chapter 5, novel systems and methods are possible with coded compressed sensing. New possibilities have opened which can fundamentally change how transfer and glean information. Examples include new analog-to-digital converters [68] and experimental design. In these new systems, specialized compressed sensing theory and measurements constructions will be required to reap the potential gains. The work in this thesis is illustrative that such efforts are worthwhile.
Bibliography


