Nonadaptive Lossy Encoding of Sparse Signals

by

Ruby J. Pai

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August 2006
[September 2006]

Author ....
Department of Electrical Engineering and Computer Science
August 18, 2006

Certified by ....
Vivek K Goyal
Associate Professor
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Nonadaptive Lossy Encoding of Sparse Signals

by

Ruby J. Pai

Submitted to the Department of Electrical Engineering and Computer Science
on August 18, 2006, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

At high rate, a sparse signal is optimally encoded through an adaptive strategy that finds and encodes the signal's representation in the sparsity-inducing basis. This thesis examines how much the distortion rate ($D(R)$) performance of a nonadaptive encoder, one that is not allowed to explicitly specify the sparsity pattern, can approach that of an adaptive encoder. Two methods are studied: first, optimizing the number of nonadaptive measurements that must be encoded and second, using a binned quantization strategy. Both methods are applicable to a setting in which the decoder knows the sparsity basis and the sparsity level. Through small problem size simulations, it is shown that a considerable performance gain can be achieved and that the number of measurements controls a tradeoff between decoding complexity and achievable $D(R)$.

Thesis Supervisor: Vivek K Goyal
Title: Associate Professor

Acknowledgments

My parents, for their love and support. My thesis advisor, Vivek Goyal, for his patience, encouragement, knowledge, and being generally cool.

Contents

Key to Notation **11**

1 Introduction **13**

2 Background **17**

2.1 Compressed Sensing 17

2.2 Source Coding Basics 23

3 Problem Setup **27**

4 Nonadaptive Encoding with Standard Quantization **33**

4.1 Minimizing M in the Lossless Case 33

4.2 Standard Quantization 41

5 Nonadaptive Encoding with Binned Quantization **47**

6 Conclusion **61**

6.1 Summary 61

6.2 Possible Design Improvements 62

6.3 Extensions 63

A Distribution of y_i for $K = 2$ **65**

List of Figures

2-1 Compressed sensing setup 18

2-2 Toy problem illustration of compressed sensing idea 20

2-3 Two types of uniform scalar quantization 25

3-1 Generalized compressed sensing setup 28

3-2 Complete nonadaptive lossy source coding setup 29

3-3 Comparison of adaptive and nonadaptive (quantization-aware basis pursuit recovery) $D(R)$ 31

4-1 Basis pursuit ordered search recovery performance 36

4-2 Probability of sparsity pattern recovery by truncated basis pursuit ordered search for $N = 16$, $K = 2$ 38

4-3 Probability of sparsity pattern recovery by truncated basis pursuit ordered search for $N = 16$, $K = 3$ 39

4-4 Probability of sparsity pattern recovery by truncated basis pursuit ordered search for $N = 16$, $K = 4$ 40

4-5 Worst-case basis pursuit ordered search complexity 41

4-6 Nonadaptive $D(R)$ achieved by an ordered search recovery method . . 42

4-7 Complexity of ordered search recovery method 43

4-8 Probability of sparsity pattern recovery from quantized measurements achieved by an ordered search recovery method 45

5-1 Two complementary multiple description quantizers with binned quantizer cells 48

5-2 Encoder/decoder with binned quantization 49

5-3 Toy problem illustration of binned quantization 50

5-4 Sample binned quantizer pattern 51

5-5 Entropy of binned quantizer output for different values of binning parameters L and B 53

5-6 Performance comparison for different values of B at $(M = 9, L = 2)$. 55

5-7 Nonadaptive $D(R)$ for optimal values of (M, L) 56

5-8 Nonadaptive $D(R)$ for individual values of $M \in [4 : 7]$ and different values of L. 59

5-9 Nonadaptive $D(R)$ for individual values of $M \in [8 : 10]$ and different values of L. 60

A-1 Distribution of y_i for $K = 2$ 65

Key to Notation

Signal parameters

Symbol	*Dimension*	*Definition*
N	–	signal space dimension
K	–	sparsity level
x	$N \times 1$	signal to be encoded
Φ	$N \times N$	orthonormal sparsity basis, $x = \Phi\theta$
ϕ_j	$N \times 1$	sparsity basis vectors (columns of Φ)
θ	$N \times 1$	sparsity basis representation of x
$\tilde{\theta}$	$K \times 1$	nonzero coefficients of θ

Measurement parameters

Symbol	*Dimension*	*Definition*
M	–	number of nonadaptive measurements
F	$M \times N$	nonadaptive measurement matrix
$f_{*,j}$	$M \times 1$	columns of F
$f_{i,*}$	$1 \times N$	rows of F
$\tilde{F}$	$M \times K$	columns of F corresponding to a given sparsity pattern

Quantization parameters

Symbol	*Dimension*	*Definition*
Δ	–	uniform scalar quantizer step size
L	–	# of (scalar) quantizer cells in a (scalar) bin
B	–	# of (scalar) quantizer cells between cells in same bin
R	–	rate in bits per source component (bpsc)
D	–	total mean squared error (MSE) distortion

Other Notation

Symbol	*Dimension*	*Definition*
$H(z)$	–	entropy of discrete random variable z (bits)
$\text{supp}(z)$	–	support of random variable z
$\|z\|_0$	–	number of nonzero coefficients of vector z
I_N	$N \times N$	identity matrix
m	–	max. # of allowed iterations in truncated BPOS (Ch. 4.1)

Also note that for any variable z, $\hat{z}$ is either its quantized version or its reconstruction, depending on context.

Chapter 1

Introduction

Recent enthusiasm about sparsity stems from two major areas of study. First, the existence of good heuristics for solving a sparse approximation problem given a dictionary and a signal to be approximated has been shown [14], [17], [4], [8], [18]. Second, there has been a flurry of activity around the concept of "compressed sensing" for sparse signals [2], [7], [3], by which this thesis is inspired.

In reality, signals are rarely exactly sparse, but in many cases of interest can be well approximated as such. For example, piecewise smooth signals have good sparse approximations in wavelet bases and this extends empirically to natural images. The power of nonlinear approximation in sparsifying bases explains the success of wavelets in image transform coding [6], [13].

In source coding, one wishes to represent a signal as accurately and as efficiently as possible, two requirements which are at odds with one another. If a transform concentrates the essential features of a class of signals in a few coefficients, encoding only the significant coefficients in the transform domain may allow one to spend more of the available bits on what is important. There are subtleties involved, however, due to the nonlinearity of sparse approximations. Nonlinear means that instead of a fixed set of coefficients which are optimal on average, the coefficients which participate in the approximation are adapted to each signal realization. An important consequence in the source coding context is that the positions of these signal-dependent significant coefficients must be encoded as well [20], [19].

In this work, we study *nonadaptive* lossy encoding of exactly sparse signals. "Lossy" simply refers to quantization. The key word is "nonadaptive": we study the encoding of a signal which has an exact transform domain representation with a small number of terms, *but in a context where we cannot use this representation.*

To be precise, consider a signal $x \in \mathbb{R}^N$ which has a sparse representation in an orthonormal basis Φ: $x = \Phi\theta$, $\Phi \in \mathbb{R}^{N \times N}$ is an orthogonal matrix, and $\|\theta\|_0 = K \ll N$.[1] At high rate, an adaptive encoding strategy is optimal: transform x to its sparsity basis representation θ, spend $\log_2 \binom{N}{K}$ bits to losslessly encode the sparsity pattern (the nonzero positions of θ), and spend the remaining bits on encoding the values of the K nonzero coefficients. We will be studying nonadaptive encoding of sparse signals, where by nonadaptive we mean that the encoder is not allowed to specify the sparsity pattern. We assume in addition that the encoder is Φ-blind, meaning it does not use the sparsity basis, though this is not required by the definition of nonadaptive. We assume that Φ is known to and can be used by the decoder.

Our nonadaptive encoder leans on compressed sensing theory, which states that such a sparse signal x is recoverable from $M \sim O(K \log N)$ random measurements (linear projections onto random vectors) with high probability using a tractable recovery algorithm. However, in the same way that applying conventional approximation theory to compression has its subtleties, so does applying the idea of compressed sensing to compression. In a source coding framework, instead of counting measurements, one must consider rate, and instead of probability of recovering the correct sparsity pattern, one must consider some appropriate distortion metric. In particular, the goal of this thesis is to explore how much the performance of nonadaptive encoding can approach that of adaptive encoding. By performance, we mean not only the fidelity of the reconstruction but the number of bits required to achieve that level of fidelity. At first glance, the $\log N$ multiplicative penalty in number of measurements is discouraging; we will see that finding a way to minimize M greatly improves the performance of nonadaptive encoding.

[1] The ℓ_0 quasi-norm just counts the number of nonzero coefficients.

An outline of this thesis is as follows: Chapter 2 gives background on compressed sensing and reviews some source coding basics. Chapter 3 discuss the problem setup in detail. Chapters 4 and 5 present the main ideas and results of this work. Finally, Chapter 6 discusses open questions and concludes.

Chapter 2

Background

2.1 Compressed Sensing

Theory. Consider a signal $x \in \mathbb{R}^N$ such that $x = \Phi\theta$, where $\|\theta\|_0 = K$ and Φ is an orthogonal matrix. In a nutshell, compressed sensing (CS) theory states that such a signal can be recovered with high probability from $M \sim O(K \log N)$ random measurements (linear projections onto random vectors) using a tractable recovery algorithm [2], [7], [3].

Compressed sensing results have their roots in generalizations of discrete time uncertainty principles which state that a signal cannot be simultaneously localized in time and frequency. The intuition is that if a signal is highly sparse in the time domain, it cannot also be highly sparse in the frequency domain, and taking a large enough subset of frequency samples should "see" enough of the signal to allow reconstruction. In [1], the canonical time and frequency bases were studied, and it was shown that for N prime and $\Phi = I_N$, x could be exactly recovered from any M frequency measurements so long as $M \geq 2K$. However if $M \geq 2(K-1)$, $M < N$, then M frequency measurements no longer guarantee exact recovery. Though this theorem only holds for N prime, [1] argues that for nonprime N it holds with high probability for sparsity patterns and frequency samples chosen uniformly at random. Moreover, recovery continues to occur with high probability using a recovery heuristic discussed below if $O(K \log N)$ measurements are taken.

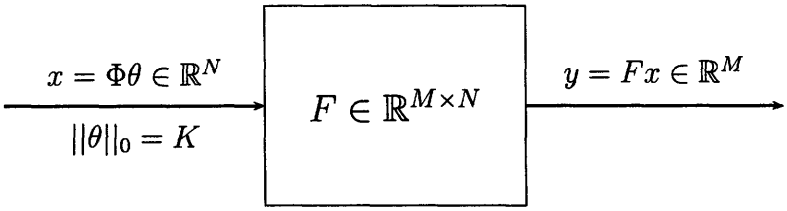

$x = \Phi\theta \in \mathbb{R}^N$
$||\theta||_0 = K$
$F \in \mathbb{R}^{M \times N}$
$y = Fx \in \mathbb{R}^M$

Figure 2-1: Compressed sensing setup.

This last result was subsequently generalized to any pair of mutually incoherent sparsity and measurement bases. The mutual coherence between two sets of vectors is the largest magnitude inner product between vectors in these sets. Requiring the mutual coherence between the measurement vectors $\{f_{i,*}\}_{i=1}^M$ and sparsity basis vectors $\{\phi_j\}_{j=1}^N$ to be small essentially just says that the measurement vectors must not "look like" the vectors of the sparsity basis. Note that here "small" depends on the sparsity level K. To meet this requirement, randomness rather than explicit design is the solution. In practice, independent, identically distributed (i.i.d.) Gaussian or i.i.d. Bernoulli (± 1) measurement matrices work well.

The compressed sensing "encoding" strategy is depicted in Figure 2-1, in which the measurement values have been stacked into a vector $y \in \mathbb{R}^M$ and the corresponding measurement vectors into a matrix $F \in \mathbb{R}^{M \times N}$, so that $y = Fx$.

To recover x from knowledge of y and F, one uses the sparsity model to combat what is otherwise an underdetermined problem. That is to say, in theory one would like to solve

$$\hat{x} = \arg\min_v ||\Phi^T v||_0 \quad \text{such that} \quad y = Fv. \tag{2.1}$$

In words, find the sparsest solution that is consistent with the observations y and F. This is a sparse approximation problem: $y \in \mathbb{R}^M$ is a K-sparse signal with respect to the N-element dictionary (overcomplete representation) for $\mathbb{R}^M$ formed by the columns of $F\Phi$. For large problem sizes and unstructured F, solving (2.1) is not computationally feasible and one resorts to heuristics. There are two flavors of such: greedy matching pursuit [14], [17] and convex relaxation to ℓ_1 minimization, also known as basis pursuit [4], [8], [18]. Initial compressed sensing results focused on basis pursuit. Instead of (2.1), one solves

$$\hat{x} = \arg\min_{v} \|\Phi^T v\|_1 \quad \text{such that} \quad y = Fv. \tag{2.2}$$

Results in sparse approximation theory give conditions for when a sparse representation of a signal with respect to a dictionary $\mathcal{D}$ will be the unique sparsest representation with respect to $\mathcal{D}$ (i.e., the unique solution to (2.1)), and when basis pursuit will find it (i.e., also the unique solution to (2.2)). These conditions involve the sparsity level K and coherence $\mu(\mathcal{D})$ of the dictionary. In particular, if a signal has a K-term representation with respect to $\mathcal{D}$, and $K < \frac{1}{2}(1 + \mu(\mathcal{D})^{-1})$, then this representation is the unique sparsest representation with respect to $\mathcal{D}$, and basis pursuit will find it [17]. These conditions are sufficient but not necessary.

To summarize, basis pursuit recovers a signal with sparsity level K from $M \sim O(K \log N)$ random measurements with probability close to 1. It bears emphasizing that the $\log N$ multiplicative penalty in number of measurements is the price paid for the tractability of solving (2.2) instead of (2.1).

Toy Problem Illustration. Let us consider a toy problem which gives insight into the compressed sensing idea. Let $N = 3$, $K = 1$, and $M = 2$. Then $x \in \mathbb{R}^3$ lies on one of three lines, and we propose to recover it from its projection onto two vectors in $\mathbb{R}^3$.

Assume that the measurement vectors $f_{1,*}$ and $f_{2,*}$ are linearly independent. Then the span of $f_{i,*}$ define a plane in the signal space $\mathbb{R}^N$, the measurement subspace, as depicted in Figure 2-2a. For ease of illustration, we have assumed that the realizations of $f_{i,*}$ are such that this plane coincides with the e_1-e_2 plane, i.e. $\text{span}(f_{1,*}, f_{2,*}) = \text{span}(e_1, e_2)$.

There are two perspectives from which to regard the problem. From the point of view of the signal space $\mathbb{R}^N$, x is in one of $\binom{N}{K}$ K-dimensional sparsity subspaces (in

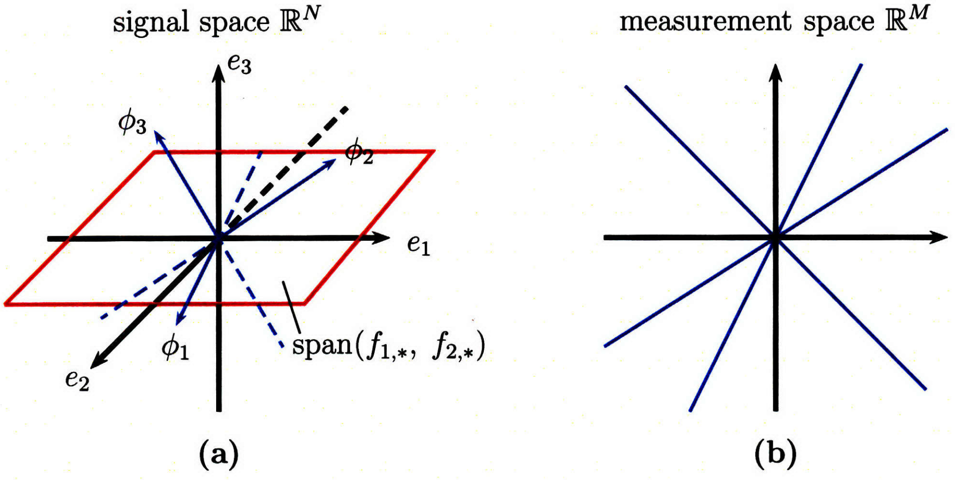

signal space $\mathbb{R}^N$
e_3
ϕ_3
ϕ_2
e_1
ϕ_1
span$(f_{1,*},\ f_{2,*})$
e_2
(a)
measurement space $\mathbb{R}^M$
(b)

Figure 2-2: Toy problem illustration of compressed sensing idea.

this case on one of three lines), and we are projecting one of these subspaces onto the measurement subspace, a random M-dimensional subspace of $\mathbb{R}^N$. The measurements y_i, $i = 1, \ldots, M$, are the coefficients of this projection. The second perspective arises from considering the problem from the point of view of the measurement space $\mathbb{R}^M$. As previously explained, $y \in \mathbb{R}^M$ is synthesized from K elements of the N-element dictionary for $\mathbb{R}^M$ formed by the columns of $F\Phi$. Thus y lies in one of $\binom{N}{K}$ K-dimensional subspaces of $\mathbb{R}^M$.

Without the sparsity prior on x, two measurements leave the dimension orthogonal to the measurement subspace unspecified. In particular, each measurement defines a plane in $\mathbb{R}^N$: $f_{i,1}x_1 + f_{i,2}x_2 + f_{i,3}x_3 = y_i$. The intersection of these two planes specifies that x lies on a line which is parallel to the unspecified dimension e_3. With the sparsity prior, however, x can be uniquely determined by two measurements, since this line is likely to intersect one of the sparsity subspaces at just one point.

Figure 2-2b depicts this from the perspective of the measurement space $\mathbb{R}^M$. The three lines are the representations of the sparsity subspaces in the measurement space (not quite their projections onto the measurement space, but what they would synthesize in the measurement space). With this geometry, the transformation from x to y is an invertible mapping.

A Fun Example. Compressed sensing is also the idea behind a brain teaser the author was confronted with at an interview, which is modified here for entertainment and analogy-drawing purposes. Suppose, for Bob knows what reason,[1] there are ten people who are each obligated to bring you ten pieces of chocolate.[2] Each piece is supposed to weigh 100 grams. Suppose you know that one person is cheating: his chocolate pieces are either all 99 grams or all 98 grams. Being a chocolaholic, you are determined to find out who and how much he owes you. You have a scale which gives a digital readout in grams of whatever quantity you might choose to measure. Consider the compressed sensing approach to this problem: you would like to take much less than ten measurements. In addition, the measurement vectors—the number of pieces taken from each person—are to be drawn at random from some distribution. That is, the general class of measurement vectors can be specified beforehand, but not the specific realizations that will be used.

This being a brain teaser, we can use quizmanship to infer that the key to selecting the measurement vector class lies in the fact that each of the ten people brings ten chocolate pieces: each measurement should take a different number of pieces from each person and weigh the resulting combination. A more careful analysis shows that this strategy will catch the culprit with "high probability." For ease of discussion consider the measurement values y_i to be the amount by which the scale readout falls short of what it should have been ($y_i = 5500 -$ the scale readout), and let θ be the amount by which each of the offending pieces is deficient ($\theta = 1$ or 2). Essentially what we have is a length 10, 1-sparse signal where the nonzero coefficient takes one of two values. In addition to finding this value, we must find the identity of the culprit, which is the same as finding the sparsity pattern. The measurement vectors to be used are permutations of [1:10].

There are two cases in which the answer can be immediately determined by one measurement alone. If the value of the first measurement y_1 is odd, then $\theta = 1$ and the culprit is the person from whom you took y_1 pieces for the first measurement.

[1] See Douglas Adams' *Mostly Harmless* for an introduction to Bob.

[2] Dedicated to the many such which perished during the writing of this document.

If y_1 is even and $y_1 > 10$, then $\theta = 2$ and the culprit is the person from whom you took $\frac{y_1}{2}$ pieces for the first measurement. Only when y_1 is even and $y_1 \leq 10$ is the answer unclear from the first measurement alone. For any such value of y_1, there are two possibilities: $\theta = 1$ and the culprit contributed y_1 pieces or $\theta = 2$ and the culprit contributed $\frac{y_1}{2}$ pieces. However, with high probability a second measurement will distinguish between these two cases. Indeed, only when the second measurement takes the same number of pieces from the two suspects as the first measurement will the second measurement fail to resolve the answer. Thus if the two measurement vectors are drawn uniformly at random from all the possible permutations of [1:10], then the probability that two measurements will *not* resolve the answer is loosely bounded by $\frac{8!}{10!} = \frac{1}{90}$. Adding a third measurement decreases this bound to $\left(\frac{1}{90}\right)^2$, and so on.

The adaptive analogy in this problem is if you knew in advance who was cheating. Then you would simply weigh his contributions alone to determine the value of θ.

In the above, slightly silly example, a considerable amount of prior information makes much fewer measurements than would be needed in the most general case possible (if every single person's chocolate pieces were allowed to be deficient one or two grams, then to find the weight corresponding to each person, there is no other way but to take ten measurements). Note also how the desired information is immediately obvious without error from the one adaptive measurement, whereas some processing is required in the nonadaptive case, which still contains a nonzero, though very small, probability of error.

For the rest of this report, we assume for simplicity and without loss of generality that x is sparse in the standard basis ($\Phi = I_N$). We make this assumption for the conceptual convenience of having F operate directly on the K-sparse θ. This assumption can be made without contradicting the assumption that the encoder does not use the sparsity basis because it is the same as having a general Φ and the Φ-aware decoder considering the effective measurement matrix to be $F_{\text{eff}} = F\Phi$.

2.2 Source Coding Basics

Entropy. The following is a very brief summary of the relevant material found in [5]. Let X be a discrete random variable taking values on an alphabet $\mathcal{X}$, and let $p(x)$, $x \in \mathcal{X}$, be the probability distribution of X. The entropy of X,

$$H(X) = -\sum_{x\in\mathcal{X}} p(x) \cdot \log p(x), \tag{2.3}$$

is a measure of the amount of uncertainty, or information, in X. Unless otherwise stated, logarithms in this report are base 2 and thus $H(X)$ is in bits. $H(X)$ is the minimum achievable rate for lossless encoding of a source which emits an infinitely long sequence of independent realizations of X, where rate is defined as the expected number of bits for encoding one realization. This is an asymptotic result; to approach this rate in practice, one would use a variable length code in which more probable elements of $\mathcal{X}$ are encoded with shorter codewords. There are systematic ways of constructing lossless variable length codes with rate no larger than $H(X) + 1$. It is conventional (and convenient) to use $H(X)$ as a slightly optimistic estimate of the rate of an entropy code.

Now consider a pair of correlated discrete random variables X and Y, drawn from a distribution $p(x, y)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$. One can consider the Cartesian product of X and Y to be one random variable taking values on the alphabet $\mathcal{X} \times \mathcal{Y}$; then the joint entropy $H(X, Y)$ of X and Y is defined to be

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y) \cdot \log p(x,y). \tag{2.4}$$

The conditional entropy of Y given X, $H(Y|X)$, describes the amount of uncertainty left in Y when one knows X. It is given as

$$H(Y|X) = -\sum_{x\in\mathcal{X}} p(x) \sum_{y\in\mathcal{Y}} p(y|x) \cdot \log p(y|x) \tag{2.5}$$

and is the minimum achievable rate for lossless encoding of Y given knowledge of X.

Note that

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y), \tag{2.6}$$

as expected. The Slepian-Wolf theorem states that in a scenario in which encoders for X and Y are separated but X and Y are to be jointly decoded, lossless encoding is achievable so long as the rate of the X-encoder is at least $H(X|Y)$, the rate of the Y-encoder is at least $H(Y|X)$ and the total rate of both encoders is at least $H(X,Y)$ [16].

For a more detailed discussion of entropy and lossless source coding, the reader is referred to [5].

Quantization. The following summarizes the relevant material in [11]. Consider now a continuous random variable Z taking values on support supp(Z). In order to encode Z, it is necessary to apply some form of quantization. That is to say, a discrete set of reproduction values (also called levels or points) must be chosen, and any realization z of Z must be represented with one of these values. A quantizer is defined by the set of reproduction values and the partition which specifies the mapping of supp(Z) onto this set. The set of all values which are quantized to the same reproduction level is called a quantization "cell". If we are quantizing the output of a source which emits an infinite sequence of i.i.d. realizations of Z, the resulting discrete random variable $\hat{Z} = Q(Z)$ can then be entropy coded.

In this brief introduction, we have limited our attention to scalar quantization. The most simple form of scalar quantization is uniform scalar quantization in which the real line is partitioned into cells of the same step size Δ, each of which is quantized to the center of the cell. Uniform scalar quantization comes in two common flavors, midstep quantization and midrise quantization, as depicted in Figure 2-3.

With small step size Δ (in the high rate regime), uniform scalar quantization results in rate

$$H(\hat{Z}) \approx h(Z) - \log \Delta \tag{2.7}$$

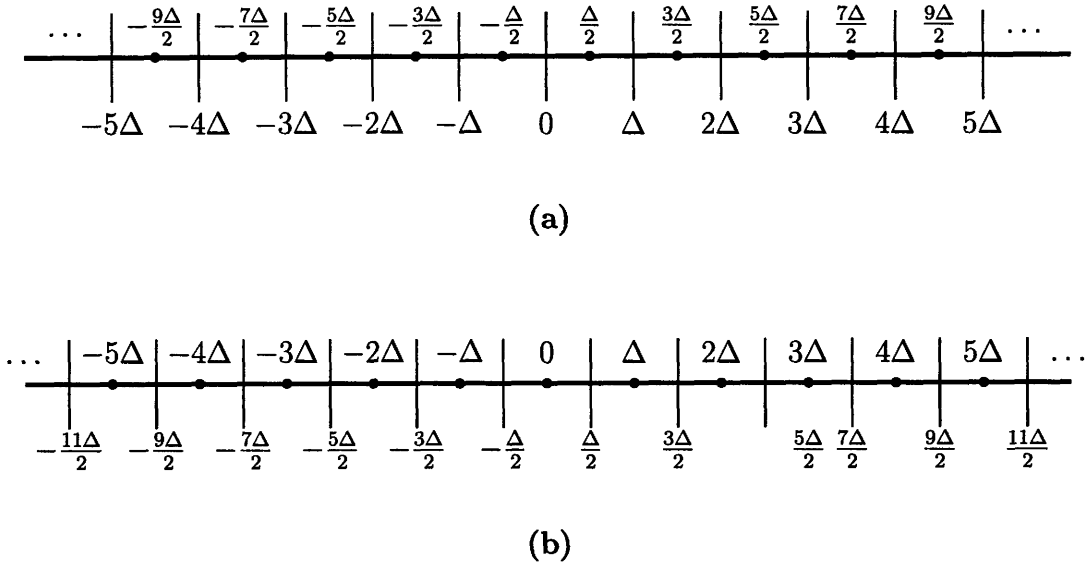


Figure 2-3: Two types of uniform scalar quantization: (a) midrise (b) midstep.

where

$$h(Z) = -\int_{z \in \mathrm{supp}(Z)} p(z) \cdot \log p(z)\, dz \tag{2.8}$$

is the differential entropy of Z.

Quantization results in distortion; the most commonly used distortion metric is mean squared error (MSE)

$$D(Z, \hat{Z}) = E[(Z - \hat{Z})^2]. \tag{2.9}$$

The question of interest boils down to this: for a given source and a given type of quantization, what is the distortion-rate ($D(R)$) behavior? For a given source distribution, what is the best achievable $D(R)$ over all possible quantization schemes? With increasing rate, the optimal entropy-constrained quantizer approaches uniform, and at high rate entropy-constrained uniform quantization results in the "6 dB per bit" rule:

$$D(R) \approx \frac{1}{12} \cdot 2^{2h(Z)} \cdot 2^{-2R}. \tag{2.10}$$

For more details, the reader is referred to [11].

Chapter 3

Problem Setup

Recall we are designing a nonadaptive encoder for a signal $x \in \mathbb{R}^N$ that has an exact K-term representation with respect to a fixed orthonormal basis Φ. By "nonadaptive," we mean the encoder does not use this K-term representation. We assume the decoder knows and uses Φ. Our nonadaptive encoding scheme builds on the compressed sensing paradigm of representing x with $M < N$ random linear measurements. The overarching aim is to explore how much nonadaptive $D(R)$ performance can approach the $D(R)$ curve achieved by adaptive encoding.

Let us step back for a moment and consider the problem setup from a broader perspective, which will allow us to then clarify the specific parameters on which we intend to focus. Consider the classic compressed sensing scenario in which the decoder has lossless access to the measurements y, but with the following generalizations:

- The only restriction on the measurement vectors is that $M < N$. In particular, measurement vectors are not necessarily random. Denote the type of measurement vectors by type(F).

- The recovery algorithm attempts to solve the problem

$$\hat{x} = \underset{v}{\operatorname{argmin}} \, \|\Phi^T v\|_0 \quad \text{such that} \quad y = Fv, \tag{3.1}$$

  but it may do so in any way (for example, combinatorial search through all $\binom{N}{K}$

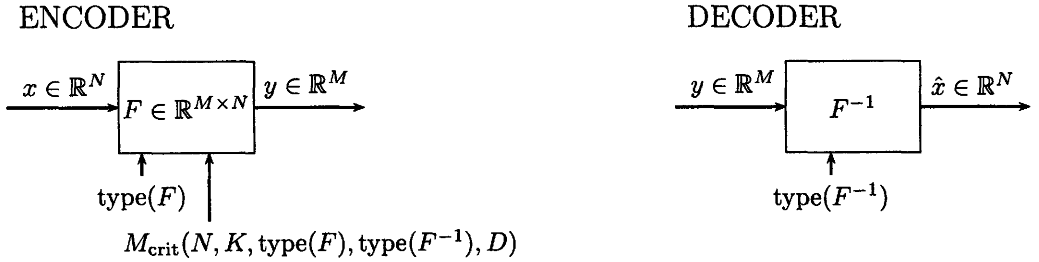

ENCODER
$x \in \mathbb{R}^N$
$F \in \mathbb{R}^{M\times N}$
$y \in \mathbb{R}^M$
type(F)
$M_{\text{crit}}(N, K, \text{type}(F), \text{type}(F^{-1}), D)$
DECODER
$y \in \mathbb{R}^M$
F^{-1}
$\hat{x} \in \mathbb{R}^N$
type(F^{-1})

Figure 3-1: Generalized compressed sensing setup, where the only constraint is $M < N$. In particular, type(F) is not restricted to certain classes of random matrices and type(F^{-1}) is not restricted to basis pursuit. M_{crit} is the minimal number of measurements required to achieve distortion no greater than D.

sparsity patterns, convex relaxation, matching pursuit, maximum likelihood estimation of the sparsity pattern). Denote the recovery strategy by type(F^{-1}).

This "generalized compressed sensing" setup discards the $M \sim O(K \log N)$ basis pursuit requirement and retains only the idea that it may be possible to recover a sparse signal from $M < N$ linear measurements, using the sparsity model to solve an otherwise underdetermined system of equations. This setup is depicted in Figure 3-1. Here the goal is to find and use M_{crit}, the smallest number of measurements which results in distortion[1] no greater than the allowed distortion level D. Note that M_{crit} may depend on type(F) and type(F^{-1}).

The complete nonadaptive lossy source coding setup is depicted in Figure 3-2, in which an encoder and decoder for y have been added. The y-encoder is responsible for turning y into bits; the y-decoder is responsible for taking these bits and producing a reconstruction of y, $\hat{y} = y + \eta$. The main component of interest in the y-encoder box is the quantizer design; in the y-decoder, the associated recovery algorithm. From the point of view of compressed sensing theory, the y-encoder and y-decoder can be encompassed in a "black box" which simply reproduces y with some bounded additive noise η. Note that the encoder may or may not have knowledge of Φ. For a fixed problem size (N, K), fixed type(F), fixed type(F^{-1}), fixed target distortion D, and

[1] As defined in Section 2.2.

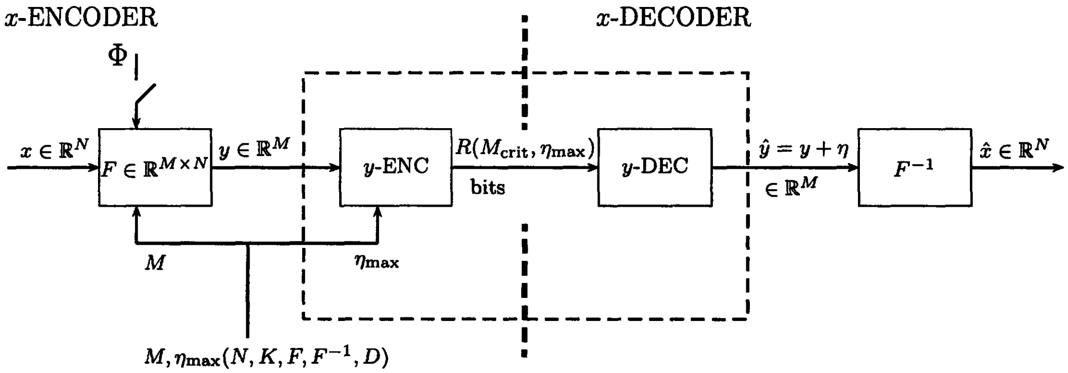


Figure 3-2: Complete nonadaptive lossy source coding setup, assuming fixed problem size (N, K), fixed type(F), fixed type(F^{-1}), and fixed target distortion D.

depending on whether the encoder is Φ-aware, one might imagine there are different allowable pairs of the parameters $(M, \eta_{\max})$. In general, the goal is to find the pair that results in the lowest rate R. If we in addition fix some allowable $(M, \eta_{\max})$, then the goal of the "black box" is even more clear-cut: encode and decode M numbers y_i such that $\hat{y} = y + \eta$, with $\eta < \eta_{\max}$ and R minimal.

Some points of information which will stay fixed throughout this work. Throughout this work, we assume the encoder to be Φ-blind and we fix type(F) to be Gaussian. That is, the entries of F are i.i.d. zero mean, unit variance Gaussian random variables, $f_{i,j} \sim N(0, 1)$. Our goal is to study choice of M coupled with quantizer design within the encoder box and an associated decoding scheme in the decoder box for optimal $D(R)$ behavior. We do so through small problem size simulations. Unless otherwise noted, we take as an example $N = 16$, $K = 2$. Each data point corresponds to 1000 trials. For each trial, x is generated by drawing a sparsity pattern uniformly at random from all $\binom{N}{K}$ possible sparsity patterns. The K nonzero coefficients are i.i.d. $N(0, 1)$. (Recall that in Section 2.1 we assumed without loss of generality that $\Phi = I_N$.) For each x realization, a different F is generated.[2] For each problem size, different encoding experiments are run on the same set of x and corresponding F realizations. Throughout this work, we stay within the framework of encoding each

[2] An associated assumption is that the encoder and decoder share a common seed, so that F is known to both for each encoded signal.

measurement y_i separately. At the encoder each measurement is scalar quantized, then the quantizer outputs are individually losslessly entropy coded. Not only is this a simple, practical design that allows distributed encoding of the measurements, but it is justified by the fact that the measurements are unconditionally independent because of the randomness of F and of the sparsity pattern. Finally, uniform scalar quantization is always midrise.

Consider applying uniform scalar quantization with step size Δ to each measurement y_i. This is a reasonable starting point; we are mainly interested in comparing the performance achievable by nonadaptive encoding (in the framework of the above assumptions) to that of adaptive encoding in the high rate region, as that is where adaptive encoding is optimal. As discussed in Section 2.2, at high rate entropy-coded uniform quantization is optimal. In addition, in the compressed sensing framework, all measurements have equal importance (or unimportance), so there is no reason for any one measurement dimension to be quantized more finely or coarsely than another.

Thus the information that the x-encoder sends is that the representation of x in the measurement space $\mathbb{R}^M$ lies within an M-dimensional Δ-hypercube. We will also refer to this hypercube as the "quantizer cell," trusting that the difference between a scalar quantizer cell and the resulting cell in $\mathbb{R}^M$ will be clear from context.

At the decoder, reconstruction from quantizer cell knowledge will use the simple yet powerful concept of consistency: picking a reconstruction which agrees with the available information about the original signal [10]. To do so, the decoder solves the optimization

$$\hat{x} = \underset{v}{\operatorname{argmin}} \, \|v\|_1 \quad \text{such that} \quad (Fv)_i \in \left[\hat{y}_i - \frac{\Delta}{2},\ \hat{y}_i + \frac{\Delta}{2}\right], \quad i = 1, \ldots, M. \tag{3.2}$$

This quantization-aware version of basis pursuit (QABP) searches for a solution within the quantizer cell instead of, for example, setting the constraint to be $Fv = \hat{y}$, where $\hat{y}$ is the center of the cell. This facilitates picking a consistent reconstruction because the center of the quantizer cell may not coincide with any of the $\binom{N}{K}$ possible sparsity patterns.

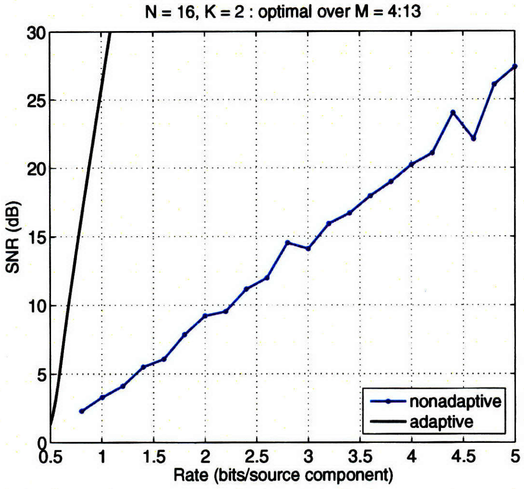


Figure 3-3: Comparison of adaptive and nonadaptive $D(R)$, where the nonadaptive decoder uses quantization-aware basis pursuit and the optimal value of M has been chosen for each rate.

Figure 3-3 compares the $D(R)$ performance achieved by adaptive encoding and this nonadaptive scheme. For nonadaptive, quantization-aware basis pursuit is used at the decoder and the optimal value of M has been chosen at each rate. Note that rate is given in bits per source component (bpsc) and distortion is given as signal to noise ratio (SNR):

$$R = \frac{M}{N} \cdot H(Q(y_i)) \quad \text{bpsc} \tag{3.3}$$

$$\text{SNR} = 10 \log_{10} \left(\frac{K}{\text{MSE}} \right) \quad \text{dB} \tag{3.4}$$

$$\text{MSE} = E \left[\sum_{i=1}^{N} (x_i - \hat{x}_i)^2 \right] \tag{3.5}$$

As seen in Figure 3-3, there is a huge gap between the performance of the adaptive and nonadaptive encoding schemes. In the adaptive case, the sparsity pattern is losslessly encoded, whereas in the nonadaptive case, there is a nonzero probability of failing to recover the sparsity pattern. In the adaptive case, all the available bits except the $\log_2 \binom{N}{K}$ allotted to the sparsity pattern are being spent to encode K coefficients, which is considerably less than the M coefficients that the nonadaptive scheme must encode.

In order to improve nonadaptive performance, it is desirable to minimize M while maintaining a high probability of sparsity pattern recovery. A method for doing so is described in the following chapter. This method takes advantage of the fact that in this problem, there are two forms of consistency: quantizer-cell consistency and sparsity-consistency.

Chapter 4

Nonadaptive Encoding with Standard Quantization

4.1 Minimizing M in the Lossless Case

Chapter 3 concluded by explaining that for improved nonadaptive $D(R)$ behavior, it is desirable to minimize the number M of nonadaptive measurements while still maintaining a high probability of recovering the sparsity pattern (SP). In theory, when the measurements are losslessly known to the decoder, a brute force combinatorial search through all $\binom{N}{K}$ possibilities will recover the correct sparsity pattern from $M = K + 1$ measurements for all K-sparse signals but a set of measure zero. Of course, in practice this is computationally prohibitive. In this chapter we consider a method for dealing with a known, fixed sparsity level K. In order to focus on how much M can be minimized, in this chapter we remove the quantization component of the problem. If a choice of M performs poorly in the lossless case for some problem size (N, K), then it will not perform well when the measurements are quantized.

Before going further, a few definitions. Define $\tilde{\theta} \in \mathbb{R}^K$ to be the "collapsed" version of θ, that is, the vector containing only the nonzero coefficients of the sparsity basis representation of x.[1] For a given sparsity pattern $\{j_k\}_{k=1}^{K}$, define $\tilde{F}$ to be the matrix containing the columns f_{*,j_k} of F. The method we consider is an ordered

[1] Recall we have assumed without loss of generality that $\Phi = I_N$, and therefore $\theta = x$.

search through the possible sparsity patterns and is given in the following.

1. Run the standard basis pursuit recovery algorithm. That is to say, solve

$$w = \operatorname*{argmin}_{v} \|v\|_1 \quad \text{such that} \quad Fv = y. \tag{4.1}$$

   If M is large enough, $M > M_{\text{crit,BP}}$, w will be K-sparse with probability almost 1. Assume M is not "large enough". Then w has more than K nonzeros with high probability.

2. Generate an ordered list of the N possible nonzero positions by sorting $|w_i|$ in descending order.

3. Pick the first K positions from this list as a candidate sparsity pattern ($\widehat{\text{SP}}$). Call this the first iteration.

4. Given the sparsity pattern, knowledge of F and the M measurements y_i form an overdetermined representation of x (more precisely, an overdetermined representation of $\tilde{\theta}$). For the given $\widehat{\text{SP}}$, reconstruct the associated $\hat{\tilde{\theta}}$ by using the inverse frame operator:

$$\hat{\tilde{\theta}} = \hat{\tilde{F}}^{\dagger} y = (\hat{\tilde{F}}^{\mathrm{T}} \hat{\tilde{F}})^{-1} \hat{\tilde{F}}^{\mathrm{T}} \tilde{F} x. \tag{4.2}$$

5. Check if the sparsity pattern candidate is consistent with the known measurements by checking if $F\hat{\tilde{\theta}} \overset{?}{=} y$. If yes, declare that the correct sparsity pattern has been recovered.

6. If no, move on to the next iteration by picking the next position from the ordered list. At the second iteration, $\binom{K}{K-1}$ new SP candidates are generated. In general, at the nth iteration, there are $\binom{K+n-2}{K-1}$ new SP candidates. (Any new SP candidate must include the newly added position. This leaves $K-1$ positions to be chosen from the $K+n-2$ already active positions.)

7. Repeat steps 4 and 5 until a SP candidate consistent with the measurements has been found. Stop at the first one, since the probability that there is more than one is negligible.

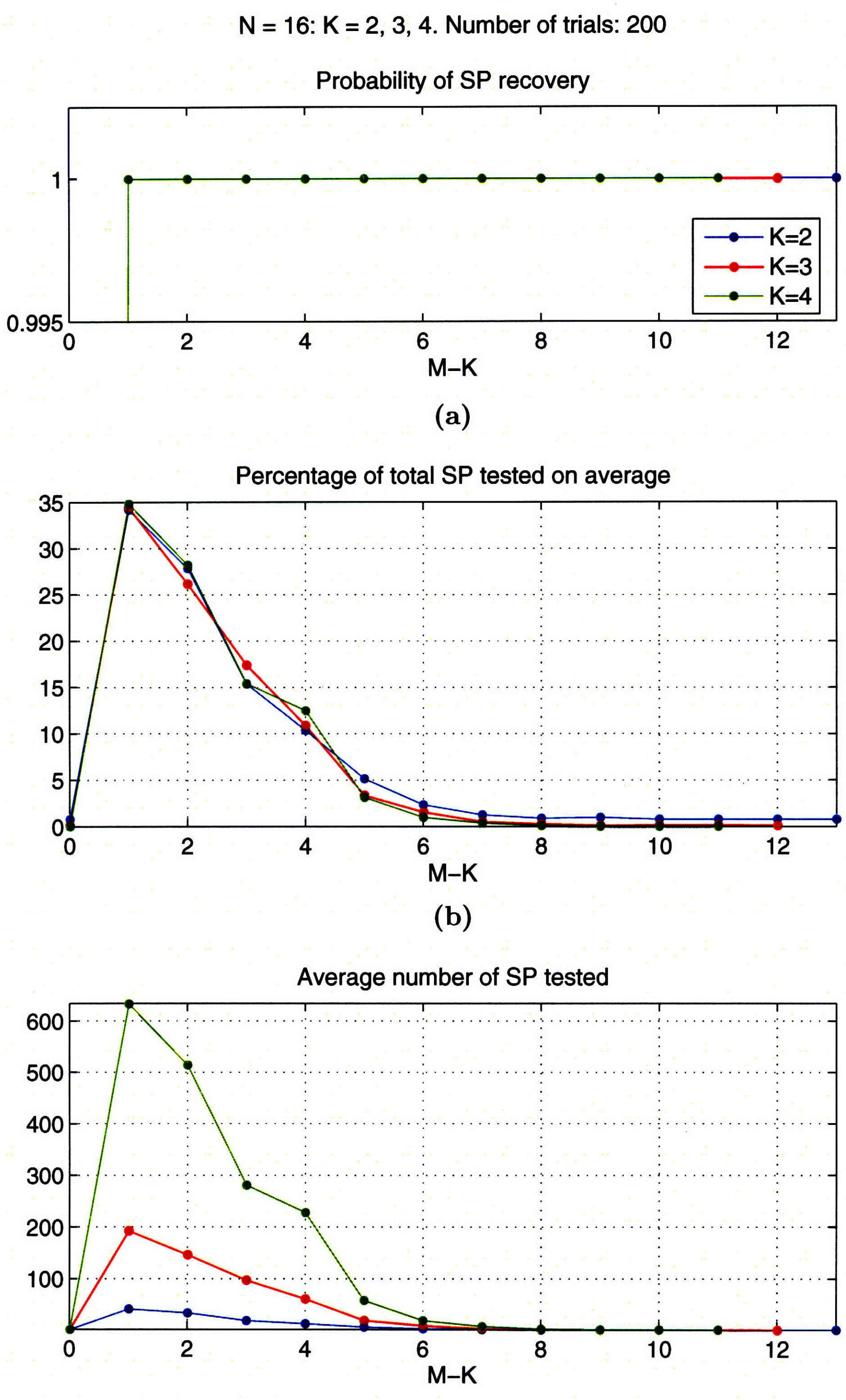
Figure 4-1 shows simulation results for this basis pursuit facilitated ordered search (BPOS) recovery method. Since BPOS is just an ordered combinatorial search, it is not surprising that SP recovery with probability 1 occurs at $M \sim K+1$. The issue at hand is computational feasibility. This is studied in parts (b) and (c) of Figure 4-1, which present the same information from two different perspectives; (c) is a reminder of how fast $\binom{N}{K}$ grows. Note that the range of M prescribed by compressed sensing theory corresponds to the range in which the average number of SP candidates tested by BPOS is close to 1.

It is interesting to note from the available data that, at $M = K + 1$, the average number of SP candidates tested before reaching the correct one is about a third of the total number of SP candidates. Of course, for all but small toy problem sizes, $\frac{1}{3}\binom{N}{K}$ is just as computationally prohibitive as $\binom{N}{K}$. However, the average number of SP candidates tested decreases as M increases. Thus, as long as $M \geq K + 1$, the sparsity pattern information is contained in M random measurements, but there is a tradeoff between number of measurements and complexity of recovery. This is in contrast with an unordered search, which has complexity independent of M.

One can also consider running BPOS with a specified maximum allowable number of iterations m, where m can take values from 1 to $N-K+1$, the maximum possible for a given problem size (N, K). With this truncated version of BPOS, a decrease in recovery complexity is obtained at the cost of a decrease in performance, since one no longer has 100% SP recovery.

Figures 4-2, 4-3, and 4-4 study the performance of truncated BPOS. Each data point corresponds to a different value of m, from $m = 1$ to the smallest value of m for probability 1 SP recovery. Percentage of total SP candidates tested is plotted instead of number of iterations, as each iteration adds a different number of new SP candidates.

To compare the average and worst case complexities of running untruncated BPOS



(a)

(b)

(c)

Figure 4-1: BPOS recovery performance. Note that the number of measurements is plotted as $M - K$.

for 100% SP recovery, compare Figure 4-1b with Figure 4-5. The worst case is very bad for a large range of M. Referring again to Figures 4-2–4-4, we see why this is the case. For a fixed M, consider two adjacent data points and denote their corresponding percentage of SP candidate tests by p_1 and p_2, where $p_2 > p_1$. Consider the y-axis difference between these data points, $\delta(p_1, p_2)$. When running untruncated BPOS, $\delta(p_1, p_2)$ percent of trials require p_2 percent of SP candidates to be tested. At large enough M, standard basis pursuit succeeds for almost all trials (only one iteration is required by BPOS), but a small percentage of trials require testing almost all $\binom{N}{K}$ SP candidates. Thus, the average computational complexity for this range of M is low, but the worst case is computationally prohibitive for larger problem sizes. For these unlucky cases, the performance of truncated BPOS just degenerates to that of standard basis pursuit. There are two conclusions to be drawn. First, for the range of M prescribed by compressed sensing theory, taking the very small hit in SP recovery probability incurred by just running standard basis pursuit and retaining the K largest magnitude coefficients is a tradeoff very much worth making. Second, truncated BPOS (with $m > 1$) is only a potentially useful idea for values of M much smaller than this range.

Finally, note that in a source coding setting, we can expect both the untruncated and truncated BPOS tradeoffs to translate into trading recovery complexity for improved $D(R)$ performance. For a fixed $\Delta > 0$, plots with the same basic trend as Figures 4-2–4-4 should be obtained, except that probability 1 SP recovery is no longer the upper bound on performance. (With increasing Δ, the curves for each value of M should shift downwards; for Δ large enough, some values of M will be too small to be viable for any percentage of SP tests.) Since M and Δ together determine rate, at a fixed Δ decreasing M decreases rate. The greater complexity of performing a full search at the decoder will allow untruncated BPOS to achieve a given probability of SP recovery with less measurements than needed by standard quantization-aware basis pursuit, decreasing rate while possibly maintaining a comparable level of distortion. For truncated BPOS, at a fixed M and Δ, allowing more SP candidates to be tested should decrease distortion. The trend in Figures 4-2–4-4 provides a nice il-

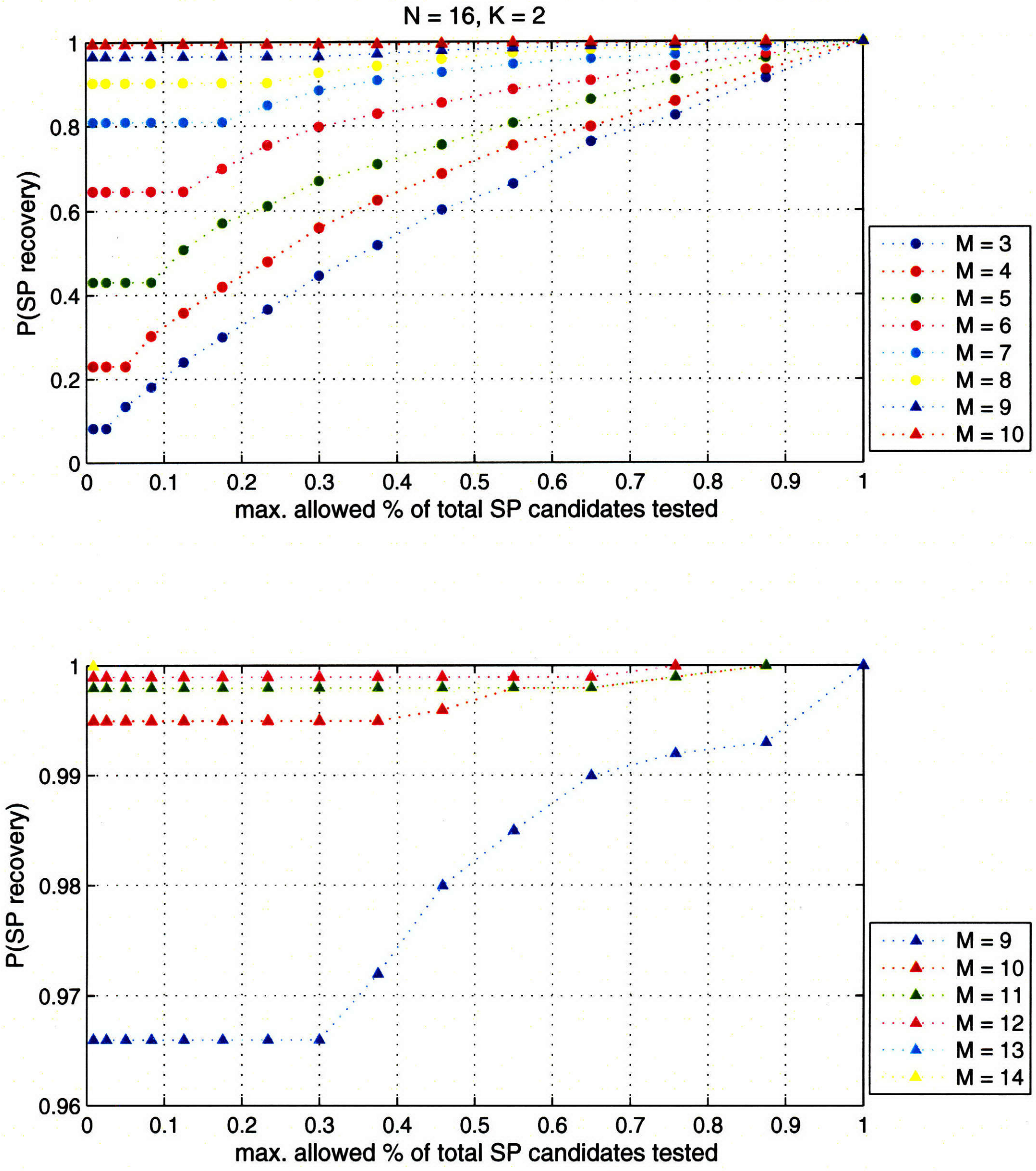


Figure 4-2: Probability of SP recovery by truncated BPOS for $N = 16$, $K = 2$. Each data point corresponds to a different number of maximum allowed iterations, from 1 to the smallest value for 100% SP recovery.

lustration of this main theme. Essentially what we have is a recovery complexity-rate tuner: for the best possible reconstruction fidelity at a fixed value of Δ, the options range from on one end using a large value of M with low recovery complexity to the opposite end, using a very small value of M with high recovery complexity.

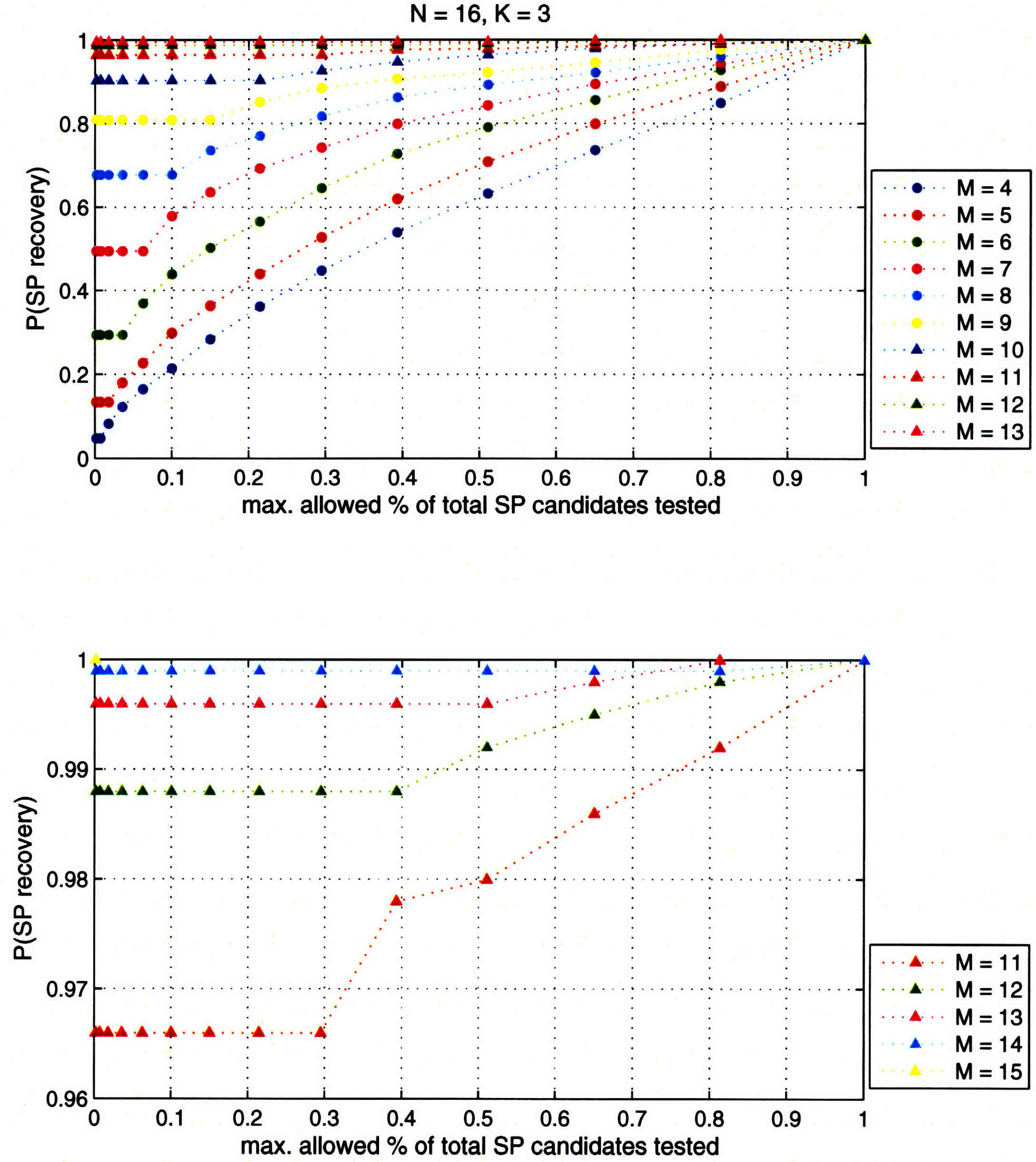
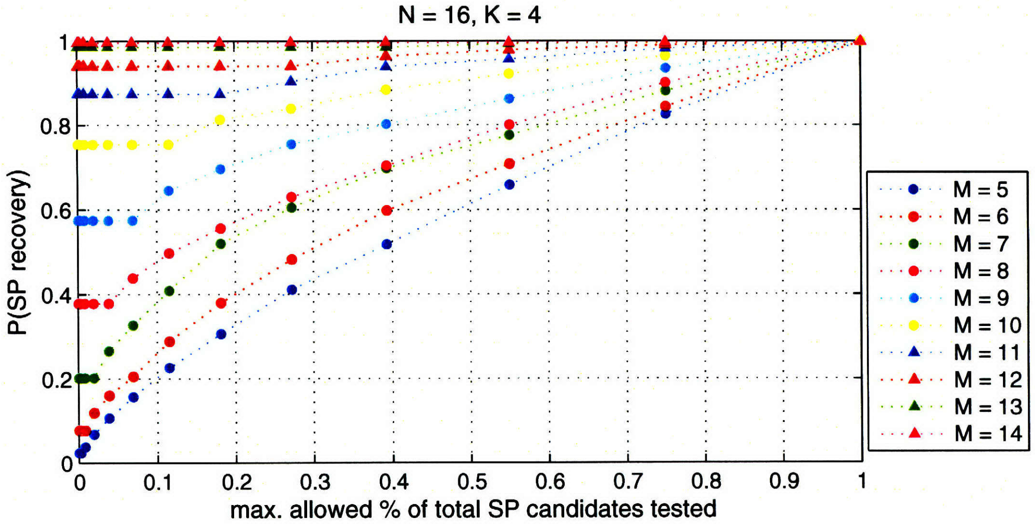


Figure 4-3: Probability of SP recovery by truncated BPOS for $N = 16$, $K = 3$.



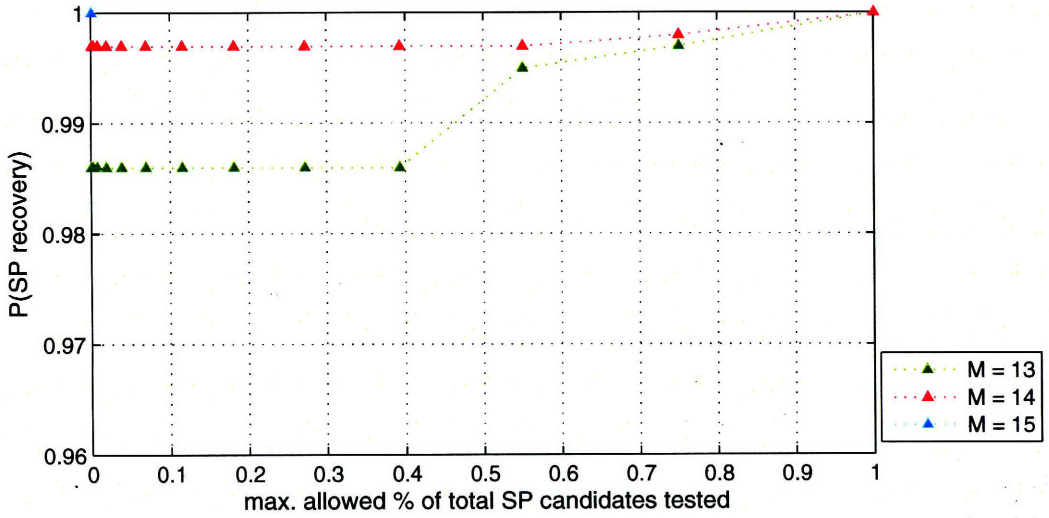
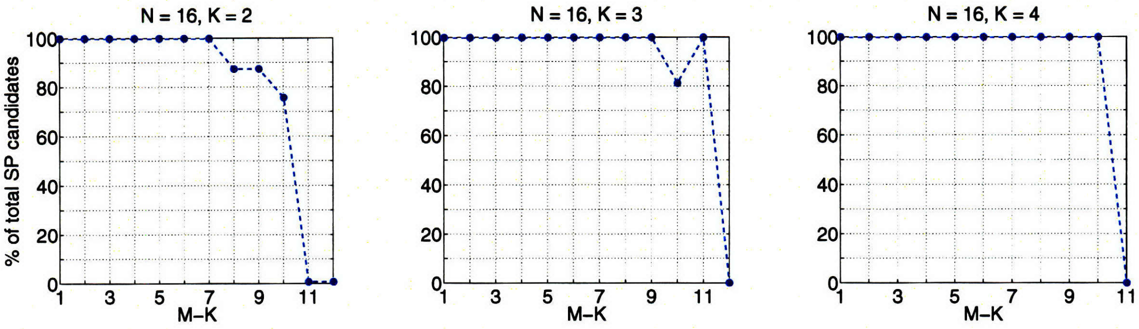


Figure 4-4: Probability of SP recovery by truncated BPOS for $N = 16$, $K = 4$.



(a) (b) (c)

Figure 4-5: BPOS minimum m_{SP} needed for 100% SP recovery, where m_{SP} is the maximum allowed percent of SP candidate tests.

4.2 Standard Quantization

We now revisit the nonadaptive encoder in Chapter 3, which applies the same uniform step size Δ scalar quantizer to each of the M measurements y_i. However, instead of merely using quantization-aware basis pursuit for signal recovery as in Chapter 3, we now incorporate the ordered search method of Section 4.1. With the addition of quantization, steps 1, 4 and 5 of the recovery procedure must be altered as follows:

1. Run quantization-aware basis pursuit:

$$w = \operatorname*{argmin}_{v} \|v\|_1 \quad \text{such that} \quad (Fv)_i \in \left[\hat{y}_i - \frac{\Delta}{2},\ \hat{y}_i + \frac{\Delta}{2}\right], \quad i = 1, \ldots, M. \tag{4.3}$$

   For given values of (N,K), when $M < M_{\text{crit,BP}}$, w is even less likely to be K-sparse than in the lossless case.

2. As in the lossless case, generate an ordered list of the N possible nonzero positions by sorting $|w_i|$ in descending order.

3. As in the lossless case, for the first iteration, pick the first K positions from this list as a candidate sparsity pattern.

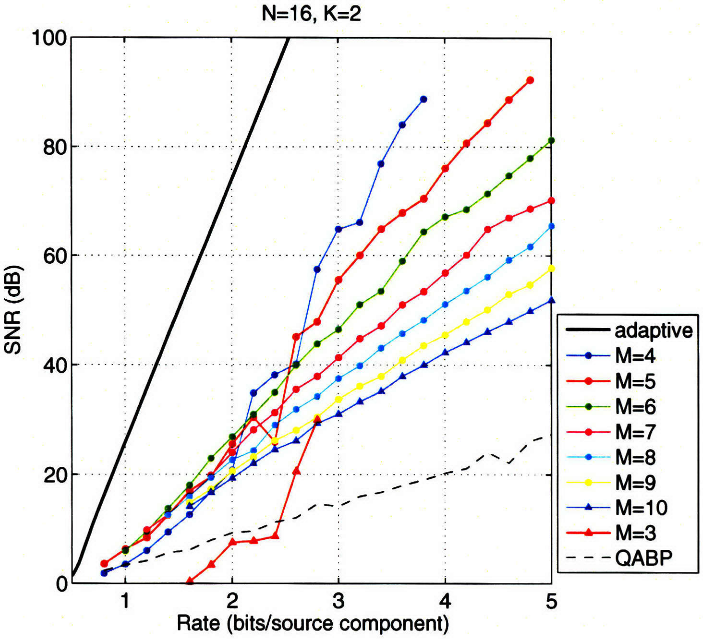


Figure 4-6: Nonadaptive $D(R)$ achieved by an ordered search recovery method.

4. Given a sparsity pattern candidate, in the lossy case, access to the quantized measurements $\hat{y}$ tells us that $F\tilde{\theta} \in [\hat{y}-\frac{\Delta}{2}, \hat{y}+\frac{\Delta}{2}]$. If we attempt to reconstruct to the center of the quantization cell in $\mathbb{R}^M$ defined by $[\hat{y}-\frac{\Delta}{2}, \hat{y}+\frac{\Delta}{2}]$, reconstruction of $\tilde{\theta}$ becomes:

$$\hat{\tilde{\theta}} = \underset{v}{\text{argmax}}\ \|d\|_1 \quad \text{such that} \quad \tilde{F}v + d \in \left[\hat{y} - \frac{\Delta}{2},\ \hat{y} + \frac{\Delta}{2}\right] \text{ and } d_i \in \left[0,\ \frac{\Delta}{2}\right] \tag{4.4}$$

5. In the lossy case, there may be no solution to (4.4), in which case the candidate sparsity pattern cannot be the true sparsity pattern. This is the quantization generalization of the measurement-consistency check.

6. As in the lossless case, if the existing SP candidate(s) are not consistent, move to the next iteration by picking the next position from the ordered list; at the nth iteration, this step generates $\binom{K+n-2}{K-1}$ new candidate sparsity patterns.

7. As in the lossless case, repeat steps 4 and 5 until there is a solution to (4.4).

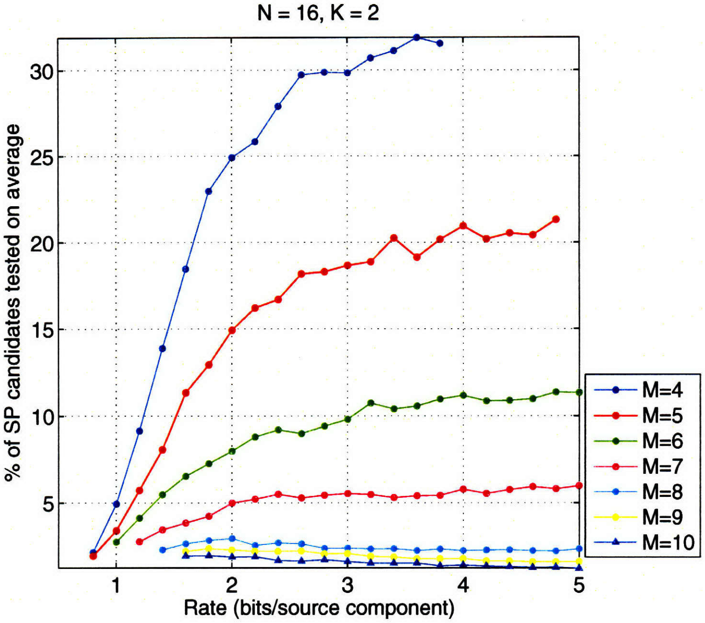


Figure 4-7: Complexity of ordered search recovery method.

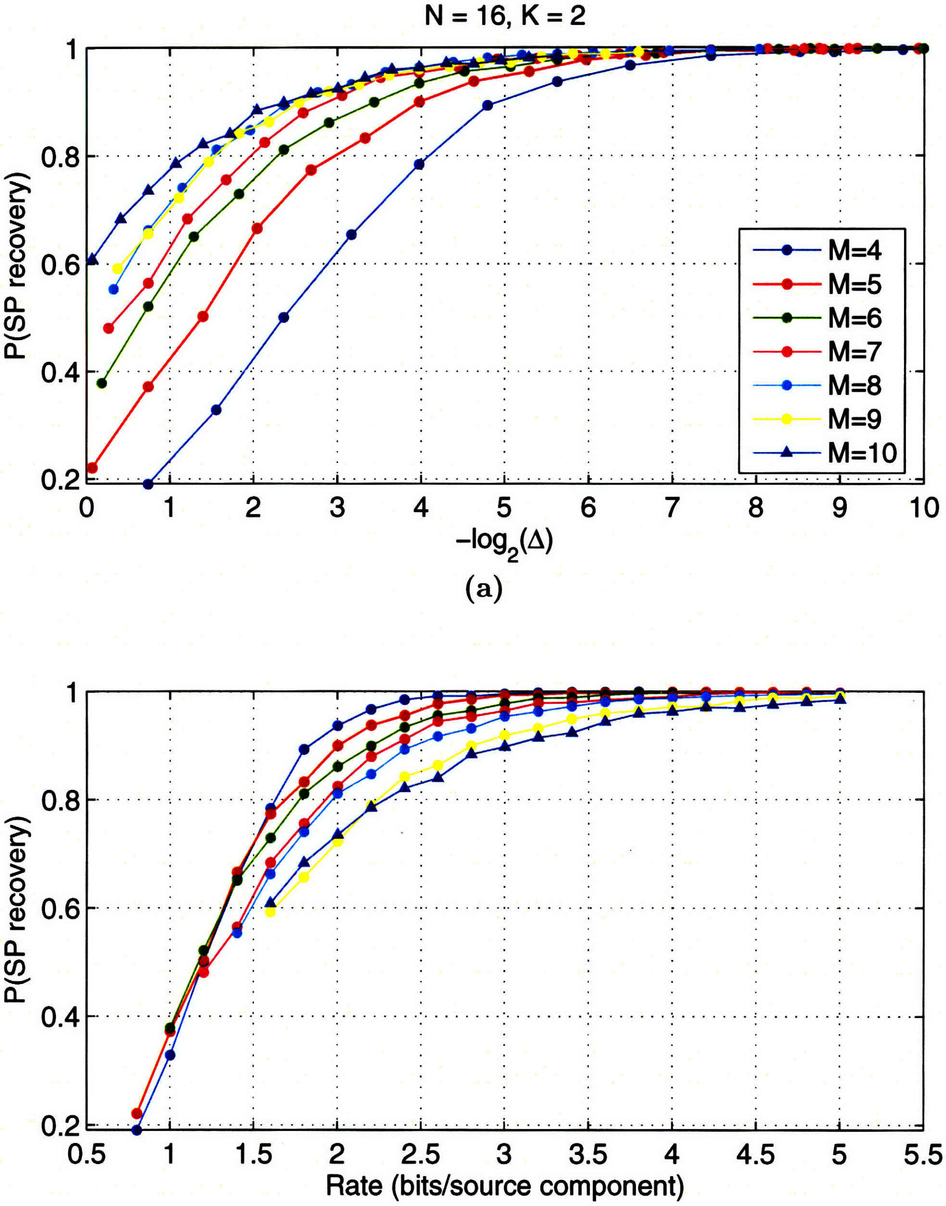
Figure 4-6 shows the resulting $D(R)$ curves for a range of M. Note that the data is for values of $\Delta \in [10^{-4}, 1]$ so that for different values of M, different ranges of R are obtained. An ordered search recovery procedure can correctly recover the sparsity pattern under conditions in which quantization-aware basis pursuit will fail. In addition, the enforcement of K-sparsity by the search procedure means that when the sparsity pattern is correctly recovered, one has an overcomplete representation of $\tilde{\theta}$. Thus there is a large $D(R)$ performance improvement across all values of M. In particular, values of M too small to be admitted by compressed sensing theory become, not only viable, but outperform larger values of M at high rate.

At high rate $M = 4$ clearly yields optimal $D(R)$. However, Figure 4-7 shows that there is a price paid for using such a small number of measurements in terms of computation at the decoder. Figures 4-6 and 4-7 together illustrate the tradeoff between achievable $D(R)$ performance and recovery complexity introduced end of Section 4.1. The value of M controls this tradeoff. Here it is of interest to note that, for the larger values of M ($M \in [8 : 10]$, well within the range prescribed by compressed sensing) testing no more than about two percent of the total SP candidates on average results in a considerable $D(R)$ improvement over that achieved by quantization-aware basis pursuit alone.

Finally, consider probability of SP recovery as depicted in Figure 4-8. Figure 4-8a shows that for a given value of M, decreasing Δ increases the probability of SP recovery. For a given value of Δ, increasing M also increases the probability of SP recovery. Both these trends are straightforward to understand from the compressed sensing background given in Section 2.1. Figure 4-8b also plots probability of SP recovery, but as a function of rate instead of quantizer step size. Increasing rate is the same as decreasing Δ. The interesting point is that above a certain rate, the trend across M is the opposite from that at a fixed Δ. This underscores the difference between compressed sensing in the presence of bounded noise in the conventional approximation setting and in a rate-distortion source coding setting. At any given rate, the value of M constrains the finest possible resolution at which each measurement is quantized, so that there is a choice to make between a larger number of more coarsely quantized measurements or a smaller number of more finely quantized measurements. Figure 4-8b shows that, above about 1.6 bpsc, $M = 4$ is also optimal in terms of SP recovery—fewer, more finely quantized measurements wins over a larger number of more coarsely quantized measurements when using an ordered search recovery. This is not at all surprising considering the results of Section 4.1.

The optimality of $M = 4$ for SP recovery at high rate partly accounts for the observed optimal $D(R)$ behavior as recovering the sparsity pattern correctly is a large component of nonadaptive encoding performance. However, it is worthwhile to note that the mean squared error distortion metric is not exactly the same as probability of SP recovery. This can be seen in the difference between Figure 4-6 and Figure 4-8b, as $M = 4$ is optimal over different ranges of R for the two different criteria. At rates where the optimal value of M for SP recovery is not optimal from the $D(R)$ point of view, it must be that the MSE conditioned on incorrect SP recovery for this value of M is larger than for the value of M that results in lowest distortion.

The question now arises: can we do better than this, over any, or all, ranges of R? In the next chapter, we explore a quantization strategy for improving $D(R)$ performance.



(a)

(b)

Figure 4-8: Probability of sparsity pattern recovery from quantized measurements achieved by an ordered search recovery method.

Chapter 5

Nonadaptive Encoding with Binned Quantization

In a multiple description (MD) source coding scenario, the encoder produces many different encodings or "descriptions" of the signal to be communicated, and the key assumption is that the encoder does not know whether the decoder will receive all or only a subset of these descriptions. Thus it is desirable that each encoding alone should produce an acceptable quality reproduction, while receiving and decoding more than one description should give a better quality reproduction [9]. A quantization strategy in MD coding is to bin disjoint quantizer cells. To illustrate, we take an example from [9]. Suppose we wished to produce two MD encodings of a random variable z uniformly distributed on $[-1, 1]$. We use the two binned quantizers depicted in Figure 5-1. Say $z = \frac{5}{16}$, so that the first quantizer produces the index '100' and the second quantizer the index '011.' If the decoder only receives the first description, then it only knows that $z \in [\frac{1}{4}, \frac{3}{8}] \cup [\frac{1}{2}, \frac{3}{4}]$. If the second description is also received, then it refines the existing information about z, narrowing down the interval in which z lies to $[\frac{1}{4}, \frac{3}{8}]$. Receiving both descriptions results in the effective reconstruction quality of a 4-bit uniform quantizer with step size $\frac{1}{8}$. This effect can also be achieved with two uniform quantizers with step size $\frac{1}{4}$ and overlapping quantization cells, but each quantizer would have a rate of three bits. The disjoint quantizer cells allow the rates of the individual quantizers to be reduced.

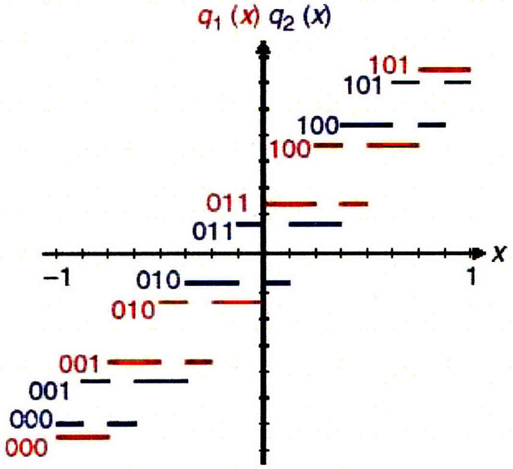


Figure 5-1: Two complementary multiple description quantizers with binned quantizer cells. Taken from [9].

Binning is a concept which generalizes to any source coding scenario in which auxiliary information about the signal being encoded is available at the decoder. For example, practical Slepian-Wolf codes for binary correlated sources (say X and Y) use binning in the form of cosets of linear channel codes to achieve lossless distributed encoding of X and Y at the joint entropy $H(X, Y)$ [15].

Returning to the problem at hand, one can consider the sparsity model prior on the signal x as side information which is definitely available at the decoder. Consider then binning disjoint quantizer cells, and relying on the sparsity model to select the correct cell at the decoder. In particular, in the previous chapter, uniform step size Δ scalar quantization of each measurement y_i produced M-cube quantization cells in $\mathbb{R}^M$. What we propose now is to group many such cells together in a bin and to send as the description of x the index of the bin which contains the quantization cell of y, as shown in Figure 5-2. This strategy attempts to improve $D(R)$ performance by reducing rate while keeping the same level of distortion. Ideally, because of the restrictiveness of the sparsity model, all but the correct cell within the bin will be inconsistent with the sparsity prior. If the decoder can take advantage of this to recover the correct cell, then binned quantization achieves the performance of standard uniform Δ quantization at a lower rate. For an intuitive picture of why this should be possible,

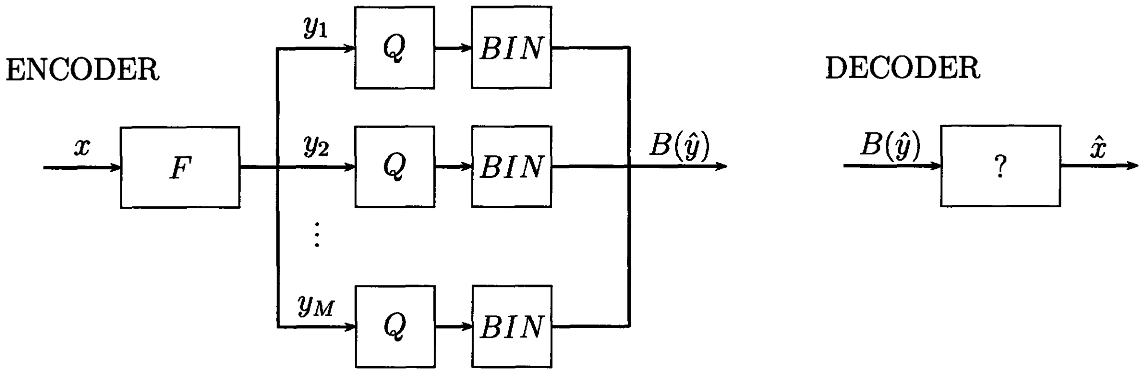


Figure 5-2: Encoder/decoder with binned quantization.

recall the toy problem of Section 2.1, in which $N = 3$, $K = 1$, and $M = 2$. Figure 5-3 illustrates binning for this example. In this chapter, we study the performance of binned quantization for nonadaptive encoding of sparse x.

To keep things simple at the encoder, we restrict our design to the scalar quantization framework of encoding each measurement separately. After scalar quantizing each measurement, the same binning pattern is applied across all measurement dimensions. In particular: (a) the number of quantizer cells binned together in a dimension, denoted by L, is the same for all measurement dimensions; and (b) the number of quantizer cells between cells in the same bin, denoted by B, stays constant within and across measurement dimensions. Thus there are two parameters involved in our design, L and B. Figure 5-4 shows a sample binning pattern.

In this setup, the encoder sends M scalar bin indices, one for each measurement, which are each scalar entropy coded. At the decoder, this information defines L^M possible quantization cells in $\mathbb{R}^M$. The decoder will attempt to jointly recover the quantization cell and sparsity pattern by finding the "intersection" between the set of possible quantization cells and the $\binom{N}{K}$ possible sparsity patterns.

Consider the specifics of signal recovery at the decoder. The brute force approach would be to perform $L^M \cdot \binom{N}{K}$ consistency tests, one for every possible sparsity pattern and quantization cell combination. Even with toy problem sizes, this is intractable. Our solution is to adapt the ordered search recovery to this binned quan-

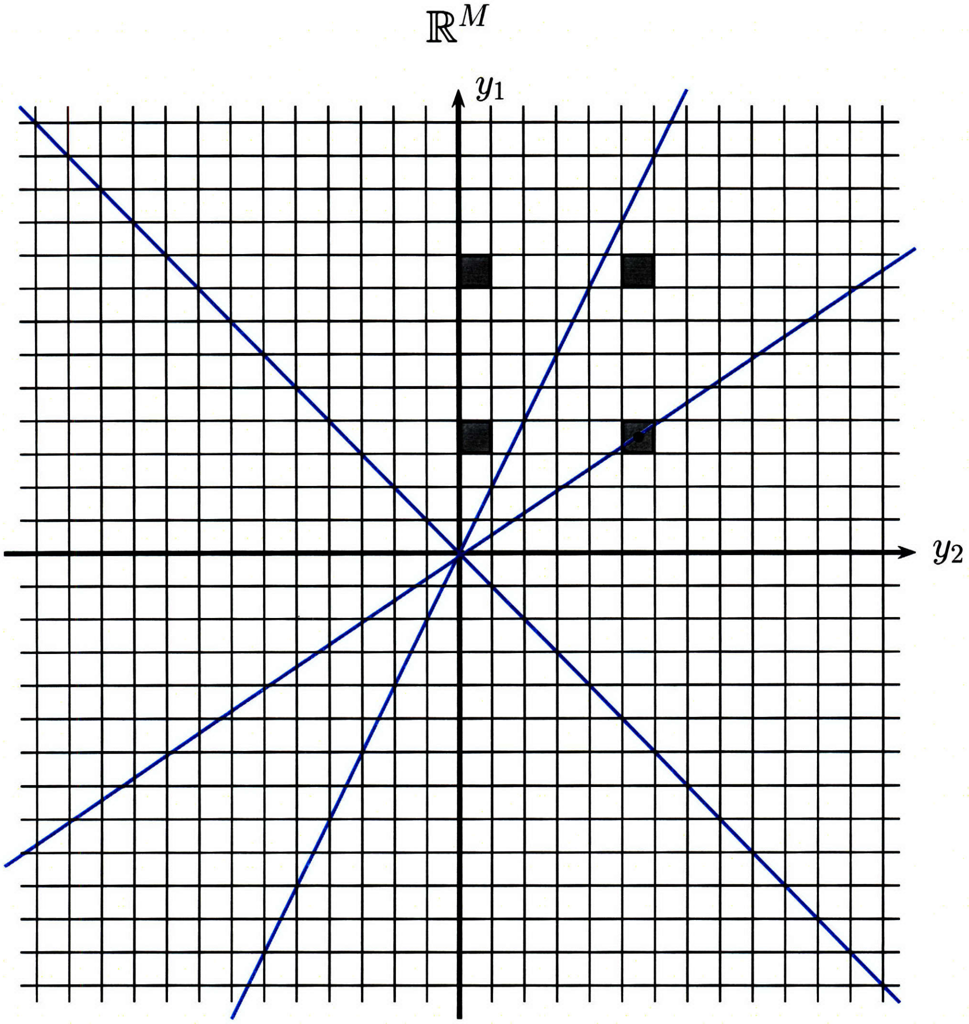


Figure 5-3: Toy problem illustration of binned quantization. $x \in \mathbb{R}^3$ is 1-sparse and two measurements are taken. The blue lines depict the measurement space representation of the sparsity subspaces for one particular realization of F. The shaded quantization cells are the cells in the active bin.

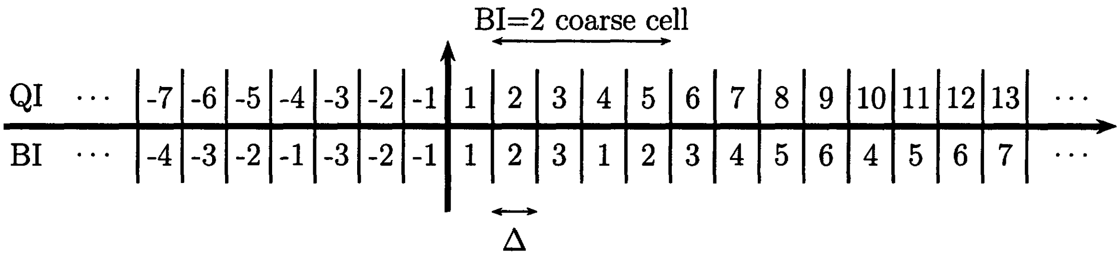

BI=2 coarse cell
QI ··· -7 -6 -5 -4 -3 -2 -1 1 2 3 4 5 6 7 8 9 10 11 12 13 ···
BI ··· -4 -3 -2 -1 -3 -2 -1 1 2 3 1 2 3 4 5 6 4 5 6 7 ···
Δ

Figure 5-4: Sample binned quantizer pattern. $L = 2$, $B = 2$. QI : quantizer cell index, BI : bin index.

tization setting. In $\mathbb{R}^M$, all the cells in a given bin lie within the coarse cell with side $[(L-1)\cdot(B+1)+1]\cdot\Delta$, as shown in Figure 5-4. Our approach is to modify BPOS by running basis pursuit with the *coarse cell* to generate the ordered list of sparsity patterns. Then test the candidate sparsity patterns in order for consistency with the actual cells within the bin. Stop at the first valid quantizer cell-sparsity pattern pair, and declare the resulting reconstruction to be $\hat{x}$. Again, as in the previous chapter, what we are proposing is to search through the possible sparsity patterns for the given values of N and K, but in an ordered fashion. The search in this case involves much more computation because for each candidate SP tested, the decoder must also search through the L^M cells specified by the M bin indices. In practice, because most of those cells (ideally, all but one) are inconsistent, the number of consistency tests needed can be greatly reduced by using a group testing approach.

Before presenting the $D(R)$ curves that result from binned quantization, consider in more detail the issues in choosing the binning parameters L and B. L should be made as large as possible in order to decrease the rate as much as possible. However, since the number of cells in a bin in $\mathbb{R}^M$ grows as L^M, with increasing L, there is also an increasing probability that greater than one cell in a bin will be consistent, and so an increasing probability of recovering the wrong quantization cell. This is particularly bad when greater than one sparsity pattern intersects the active cells, so that recovering the wrong cell could mean recovering the wrong sparsity pattern. It is not at all surprising, that decreasing rate should decrease performance. For the extra computation at the decoder that it incurs, binning is a good strategy if the effect of the former is greater than the effect of the latter.

Now consider fixing L and Δ. Then the only remaining design parameter is B, or the distance in each measurement dimension between cells in a bin, $B \cdot \Delta$. B must be chosen such that binning actually occurs. In particular, B must not be so large that some of the cells in some (or all) bins are outside the likely support of y_i. Thus the acceptable range of B depends on the distribution of y_i, L, and Δ. To illustrate, consider Figure 5-5a, which shows the entropy of the binned quantizer output for a single measurement dimension, $H(B(Q(y_i)))$, as a function of quantizer step size Δ for $K = 2$, $L = 2$ and different values of B. $H(B(Q(y_i)))$ was tabulated from the distribution of y_i, which, in our problem setup, is the sum of K products of independent $N(0, 1)$ random variables:

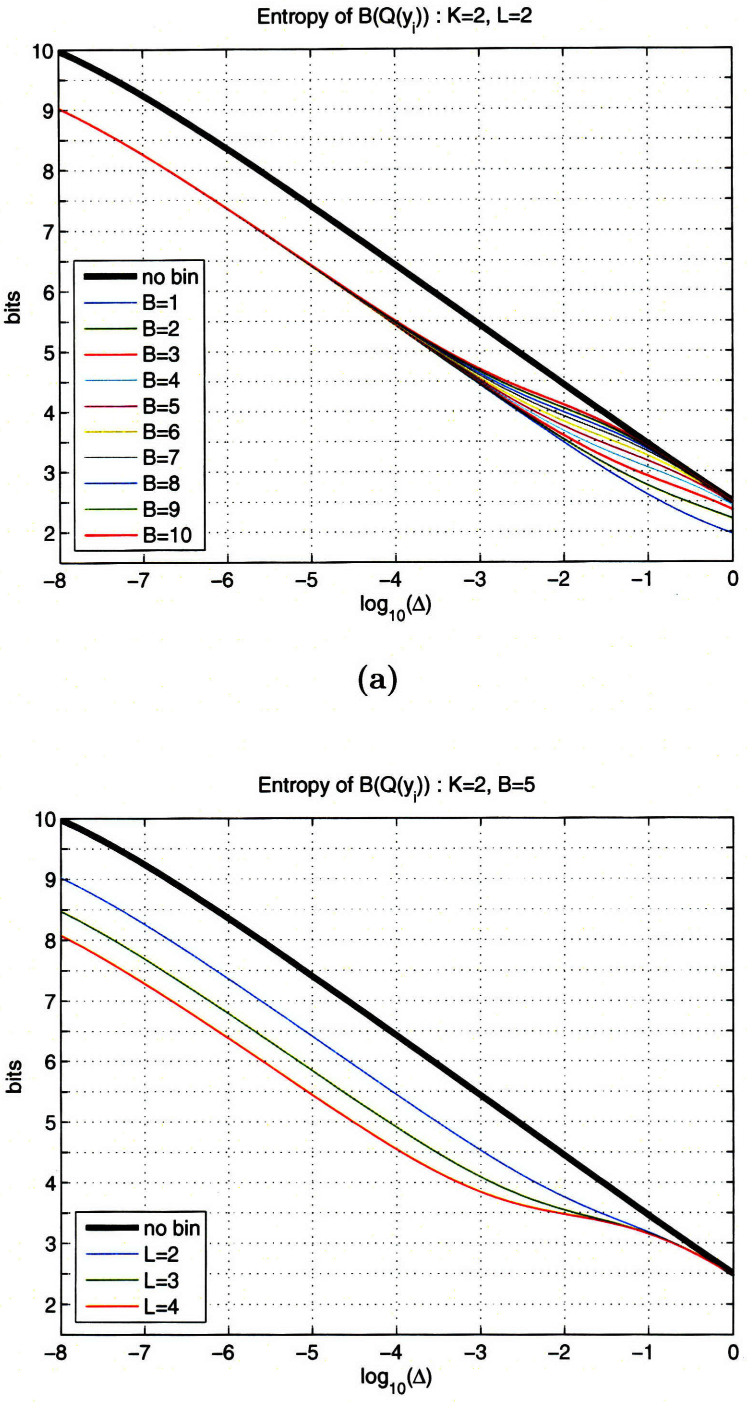
$$y_i = \sum_{j=1}^{K} \tilde{f}_{i,j} \cdot \tilde{\theta}_j. \tag{5.1}$$

For $L = 2$, an allowable binning pattern should reduce $H(B(Q(y_i)))$ by nearly one full bit.[1] One sees that for $\Delta = 1$, even $B = 1$ does not result in fully effective binning. Instead, it results in an entropy reduction of about half a bit, which makes sense since, for $K = 2$, $\text{supp}(y_i) \approx [-2.5,\ 2.5]$.[2] For small enough Δ, all values of B plotted produce the expected 1 bit decrease in $H(B(Q(y_i)))$.

Besides being "small enough," B also must be "large enough" to allow effective binning. Besides the obvious constraint that B be at least 1, it was found experimentally that B must be larger than some critical value. To see this, consider how performance for one specific choice of (M, L) varies with B, as shown in Figure 5-6. For $B \leq 4$, increasing B improves performance. A possible explanation for this behavior is that when B is too small, instead of the resolution of the individual cells within each bin, in some scenarios there is only the resolution of the coarse cell, since incorrect sparsity patterns whose representations in the measurement space intersect

[1] A decrease of exactly one bit is only possible if y_i is uniformly distributed.

[2] See Appendix A for plot of $P(y_i)$.



(a)

(b)

Figure 5-5: (a) Entropy of binned quantizer output for different values of B when $L = 2$. (b) For different values of L when $B = 5$.

the coarse cell are also likely to intersect the individual quantization cells.

So long as B is within the allowable range, for a given value of Δ, the choice of B does not affect performance, as seen in Figure 5-6b. This behavior can be explained by the fact that there is an element of randomness in how well any set of binned quantization parameters (Δ, L, B) perform for any given realizations of x and F. Since F is random, the representations of the sparsity subspaces in the measurement space will be random. In the measurement space, representations y_1 and y_2 of two different sparse signals x_1 and x_2 will be at arbitrary orientations with respect to each other, regardless of their relative orientations in the actual signal space. Thus there is no way to design B, besides picking a value that allows binning to actually occur. In the simulations that follow, B is fixed to be 5. Figure 5-5b plots $H(B(Q(y_i)))$ as a function of Δ for different values of L when $B = 5$.

Note that Figures 5-5a and 5-6 break the effect of binning on $D(R)$ into its component effects on rate and distortion, respectively. They are the first clue that binning may work for some values of (M, L).

We pause to note that, within the acceptable range, a smaller value of B results in a smaller coarse cell, which translates to the ordered search being more effective for reducing computation. In particular, a smaller number of inconsistent SP candidates will be tested before reaching a consistent SP, where by "consistent" we now mean consistency with the individual quantization cells in the given bin. That is to say, there will be a smaller number of sparsity patterns whose representations in the measurement space intersect the coarse cell but not the cells in the active bin. However, this only affects computation and not $D(R)$ performance.

Figure 5-7 presents the main result of this work: optimal $D(R)$ behavior of non-adaptive encoding over the two methods studied in this work, standard quantization ($L = 1$) with an ordered search in the recovery algorithm and binned quantization with its modified ordered search recovery. Each curve compares different ranges of M for $L = 1, 2, 3,$ and 4.

At high rate, (M=4, L=1) by far outperforms any other (M, L) pair. Denote $M = 4$ by M_{opt}. In general, at high rate the smallest value of $M \geq M_{\text{opt}}$ under

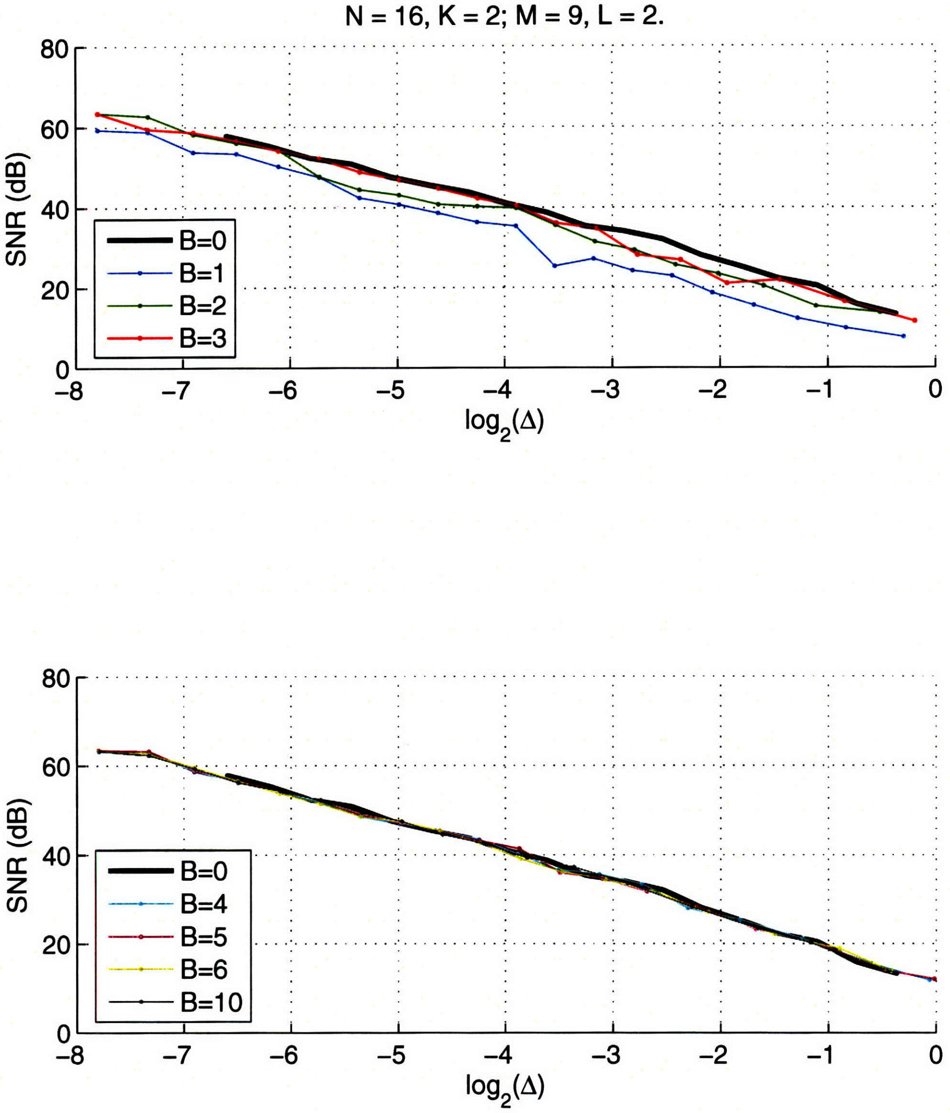


Figure 5-6: Performance comparison for different values of B at $(M = 9, L = 2)$.

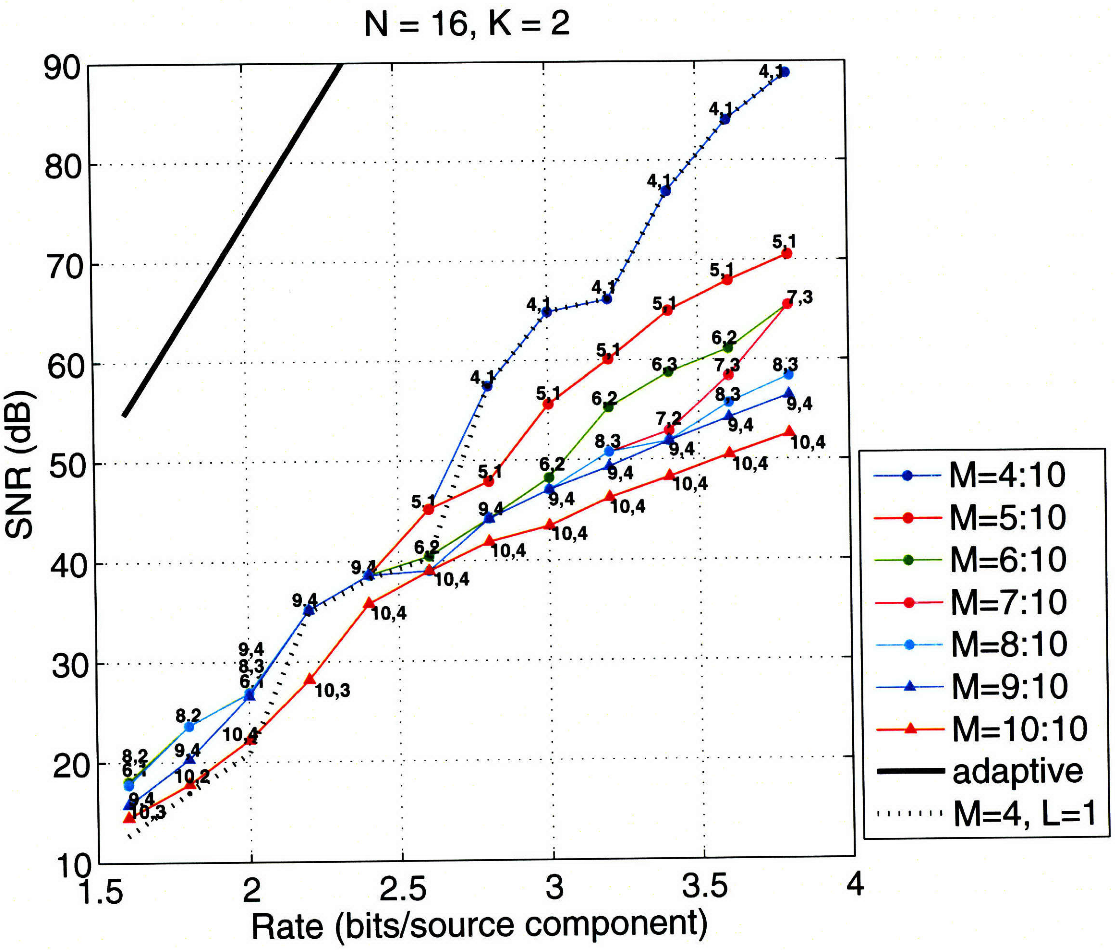


Figure 5-7: Nonadaptive $D(R)$ for optimal values of (M, L).

comparison produces optimal $D(R)$ behavior. The corresponding optimal value of L generally increases with M.

For a more in depth understanding of the results, consider the individual $D(R)$ curves for different values of L at each value of M as shown in Figures 5-8 and 5-9. At $M = M_{\text{opt}}$, binning performs worse than no binning. As M increases, binning starts to perform better over the range of R large enough for Δ to allow effective binning. The value of M at which binning starts to consistently outperform no binning increases with L. In the transition from M_{opt} to M large enough for consistent binning performance, binning sporadically outperforms no binning. Because of the erratic behavior for this range of (M, L), the optimal $D(R)$ data points which correspond to these (M, L) are not necessarily reliable. The erratic performance of binning in this intermediate range is the reason the middle optimal $D(R)$ curves contain a fluctuation of (M, L) pairs, before settling down to (9,4) and (10,4) for the last two curves.

When (M, L) is such that binning consistently outperforms no binning, the effect of binned quantization is to shift the $D(R)$ curve to the left, as expected. Recall that for given values of L and B and a given y_i distribution, there is a range of Δ small enough for the binning rate reduction to be fully effective. For a fixed Δ in this range, every factor of 2 in L will result in a rate reduction of 1 bit per measurement. This translates to a $\frac{L}{2} \cdot \frac{M}{N}$ bpsc decrease in R. If binning is completely successful at a particular value of M, the same SNR will be achieved by binning at a value of R which is $\frac{L}{2} \cdot \frac{M}{N}$ bpsc smaller than that needed by $L = 1$ to achieve the same SNR. For example, at $M = 10$ we see this behavior exactly for $L = 2, 3,$ and 4.

It is not surprising that M must be at least some $M_{\min}(L)$ for binning to be consistently successful over valid ranges of Δ. Larger M means the representations of the $\binom{N}{K}$ sparsity patterns in the measurement space are more likely to be further apart at each fixed distance from the origin. At a fixed (M, L), binning will shift the no binning $D(R)$ curve to the left by a full $\frac{L}{2} \cdot \frac{M}{N}$ bpsc if it is highly improbable that the quantization cells in a bin contain more than one sparsity pattern representation. For these values of (M, L), the binned quantization scheme can be thought of as a form of Slepian-Wolf code of $\{\hat{y}_i\}_{i=1}^{M}$ whose design is inferred from the geometry of the sparsity model.

Thus binning is fully successful for large M. However, when $M \gg M_{\text{opt}}$, the penalty for overly large M outweighs the binning gain; at high rate the (9,4) and (10,4) curves do not even approach the (4,1) curve. Note also that for (9,4) and (10,4), Figure 5-9 and Figure 4-6 show that the low rate data points in the optimal $D(R)$ plots of Figure 5-7 are misleading in that binning for the most part gets a negligible gain over any no binning $D(R)$ curve with $M > M_{\text{opt}}$.

To summarize, binning can significantly improve $D(R)$ performance for a fixed, large value of M, but this is by far not the global optimum. An encoder which does not employ binning but uses M_{opt} measurements at high rate and any $M > M_{\text{opt}}$ at low rate will achieve optimal $D(R)$.

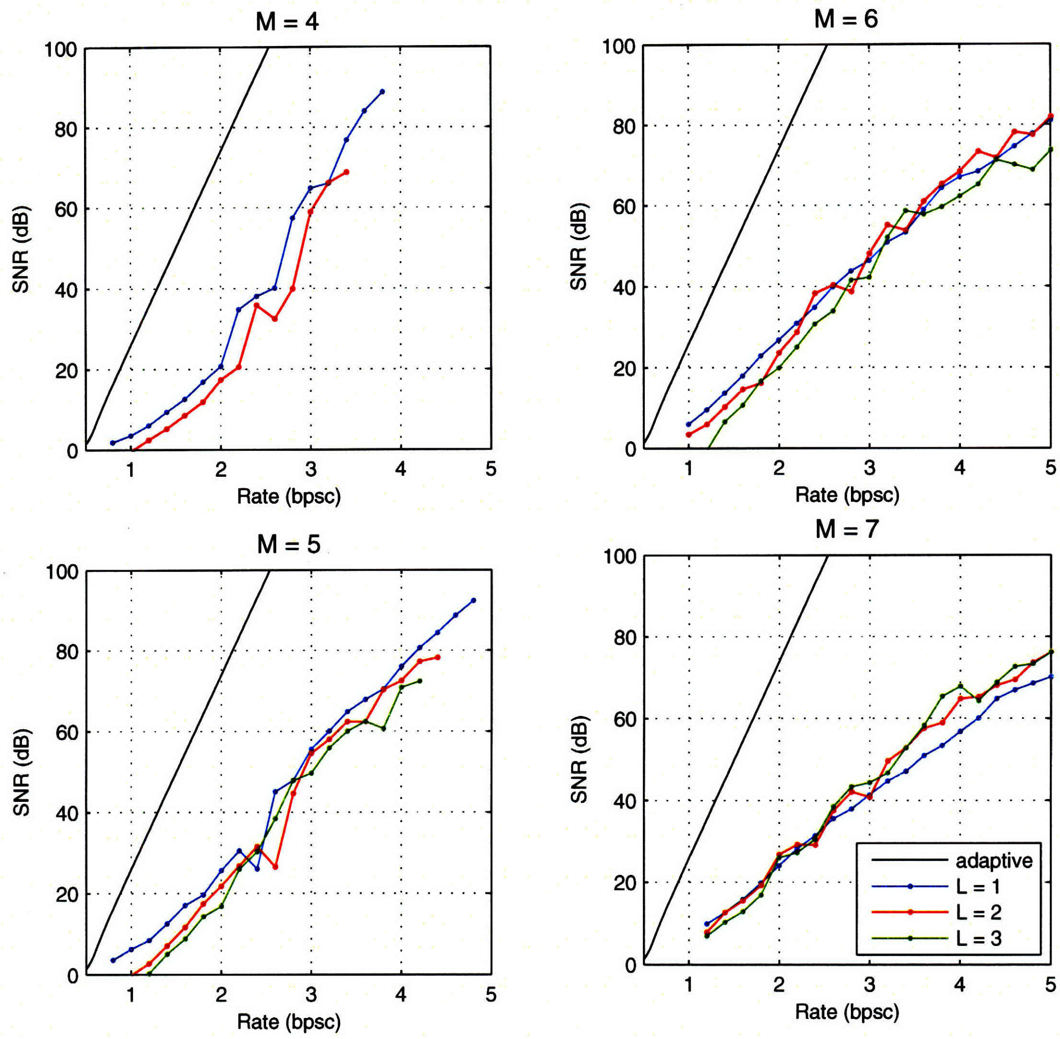


Figure 5-8: Nonadaptive $D(R)$ for individual values of $M \in [4:7]$ and different values of L.

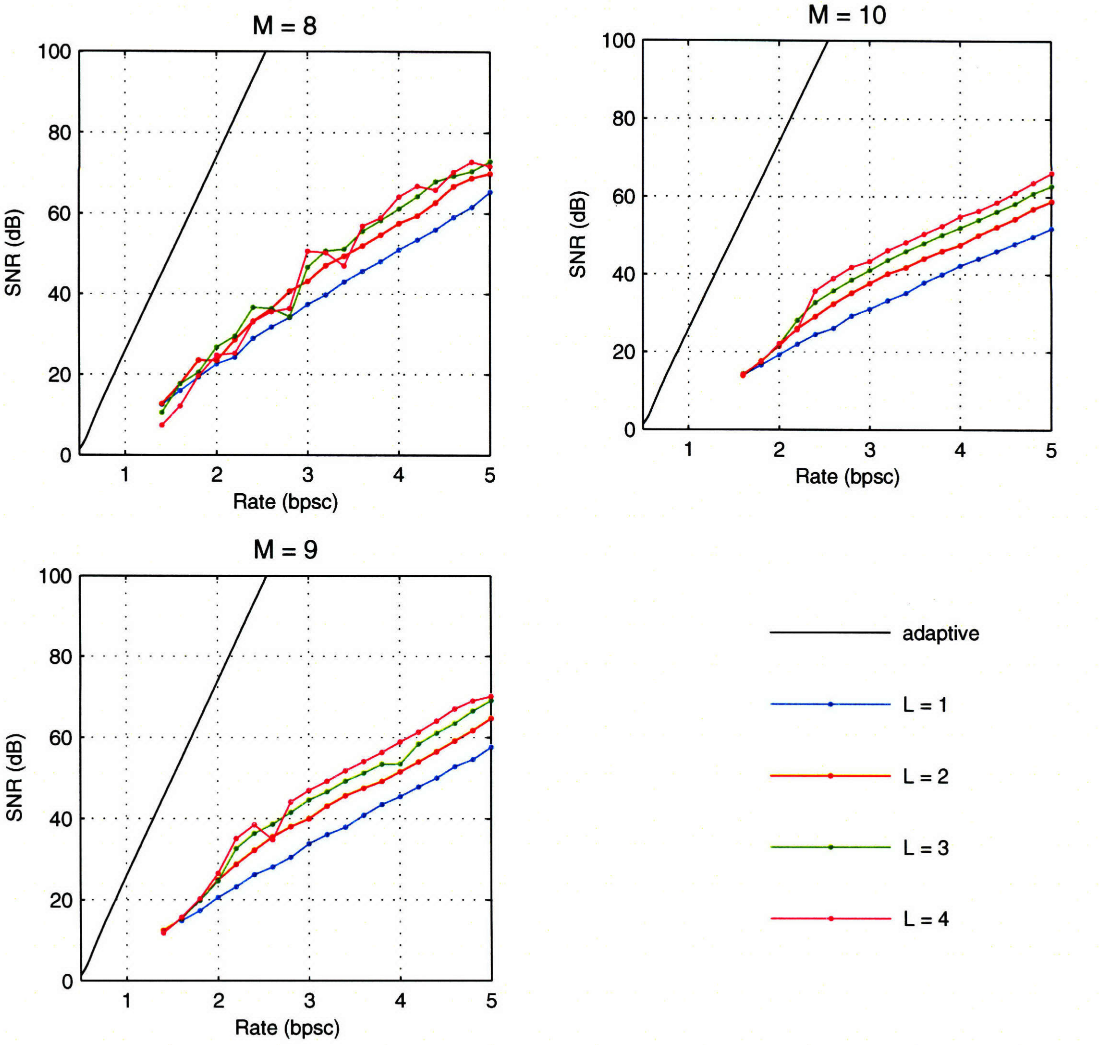


Figure 5-9: Nonadaptive $D(R)$ for individual values of $M \in [8 : 10]$ and different values of L.

Chapter 6

Conclusion

6.1 Summary

This work has studied how much a nonadaptive encoder for sparse signals can approach the $D(R)$ performance of an adaptive encoder through increased complexity at the decoder. We have considered two strategies for nonadaptive encoding applicable to a setting where the sparsity basis Φ and sparsity level K are known to the decoder. The first strategy increases complexity at the decoder in the form of an ordered search through possible sparsity patterns. This allows the number of nonadaptive measurements to be reduced while maintaining a high level of SP recovery, resulting in considerable $D(R)$ improvement. Using an ordered search provides two advantages over a brute force unordered search: one can tune the average computational complexity of the search through the choice of M, and it is possible to recognize worst case scenarios and terminate early. The second strategy involves binning the scalar quantizer output to reduce rate for a given quantizer step size Δ and taking advantage of the restrictiveness of the sparsity model to maintain a reconstruction fidelity level comparable to that of standard quantization. The corresponding recovery utilizes a modified ordered search through possible sparsity patterns.

Through small problem size simulations, we have shown that the encoding parameters for optimal $D(R)$ are a small number of measurements M_{opt} with no binning. At $M = M_{\text{opt}}$, binning performs worse than no binning, across all rates. For $M \gg M_{\text{opt}}$, binning consistently outperforms no binning, but cannot make up the large $D(R)$ penalty incurred for using such a large value of M. However, the choice of $M = M_{\text{opt}}$ only takes into account achievable $D(R)$ and not the amount of computational burden placed on the decoder. Using standard quantization with an increased the number of measurements worsens $D(R)$ performance but decreases the amount of decoding computation.

This work differs from the "classical" compressed sensing theory for sparse signals in which y is available losslessly to the recovery algorithm and the performance criterion is probability of sparsity pattern recovery. It also differs from the extension of CS theory studied by [12], in which y is corrupted by unbounded, random noise, since quantization adds bounded, signal-dependent noise. There are aspects of the problem we have studied which are particular to the source coding context. In compressed sensing, larger M can only mean better performance, because the measurements are likely to "see" more of the signal. In a $D(R)$ context, however, at a fixed rate, there is a tradeoff between number of measurements and the amount of resolution with which each measurement can be represented. In the case of a few finely quantized measurements versus a larger number of more coarsely quantized measurements, the verdict is that the former wins. Besides the difference between counting measurements and having to account for rate, there is also the difference between MSE and strict sparsity pattern recovery performance criterions. If $|\tilde{\theta}_i|$ is small (relative to the expected value of $|\tilde{\theta}_i|$, say), then the MSE penalty for incorrectly reconstructing it may be relatively small, as opposed to the binary correct or incorrect SP criterion.

6.2 Possible Design Improvements

We have studied a very simple binned quantization design in this work. Whether there are improvements to this design that would result in performance gains is yet to be explored. In the encoding of any single measurement y_i, there are two components: the scalar quantizer and the binning pattern design. Throughout the simulation results presented, a midrise quantizer was used. At low rates, a midstep quantizer might be better; at high rates it should make no difference. There is, however, a possible improvement to the binning pattern design. While the relative orientations of the sparsity pattern representations in $\mathbb{R}^M$ are random, they are closer together near the origin and farther apart farther from the origin, irrespective of their relative orientations (see Figure 5-3). For a fixed quantizer step size Δ, a possible improvement might be to slightly vary B as a function of distance from the origin: make B larger near the origin, and smaller far from the origin. At the end of Chapter 5, we mentioned that for M large enough for successful binning, one could consider binned quantization as a form of Slepian-Wolf code for $\{\hat{y}_i\}_{i=1}^{M}$. If the joint entropy of the quantized measurements, $H(\hat{y}_1, \ldots, \hat{y_M}) = H(y)$, could be calculated, it should give a bound on binning performance.

6.3 Extensions

We have used small problem size simulations in order to study how much increased complexity at the decoder can fill in the gap between nonadaptive and adaptive encoding $D(R)$ performance. For real world problem sizes, the ordered search, though "smarter" than a straightforward search, would still be intractable. In Chapter 4.1 the idea of a truncated search was introduced. The resulting $D(R)$ behavior has yet to be studied.

In this work we have studied nonadaptive Φ-blind encoding. However, the former characteristic does not necessarily imply the latter, and there might be performance gains that would result from the encoder using Φ. For example, a Φ-aware nonadaptive encoder could choose F such that the columns of $F_{\text{eff}} = F\Phi$ form a Grassmannian (minimal maximum coherence) packing of $\mathbb{R}^M$. Synthesizing y from vectors that are as far apart in the measurement space as possible should improve probability of sparsity pattern recovery from quantized measurements.

Most importantly, the ordered search recovery method requires exact K-sparsity, with known K. In practice, however, a signal is more likely to be *compressible* than exactly sparse. That is to say, its Φ representation coefficients ordered by decreasing magnitude will have a fast decay. Perhaps the most significant extension to this work is to adapt the ordered search method to compressible signals. A compressible signal can be well-approximated by a sparse signal. Recall our toy problem illustration from Section 2.1, in which we considered taking two measurements of a 1-sparse signal in $\mathbb{R}^3$. For an exactly 1-sparse x, y lies on one of three lines in $\mathbb{R}^2$. If we have instead a compressible x, y would be likely to be in an area in $\mathbb{R}^2$ immediately surrounding these three lines. An adaptive encoder would use Φ to determine the K largest magnitude coefficients in the compressibility basis, losslessly encode their positions, and spend the remaining available bits on their values. (The encoder would choose K in some appropriate fashion.) A possible, as yet untried strategy for adapting our method to a compressible signal is to pretend that the signal is K-sparse and use the same recovery algorithm at the decoder, but with a larger Δ than actually used at the encoder when testing candidate sparsity patterns for quantization cell consistency. The hope would be that the measurement space representation of the optimal K-term approximation sparsity pattern would intersect the enlarged quantization cell. In that case, the decoder would compute a reconstruction with the same compressibility basis support as that of the optimal approximation that would have been found by an adaptive encoder.

Appendix A

Distribution of y_i for $K = 2$

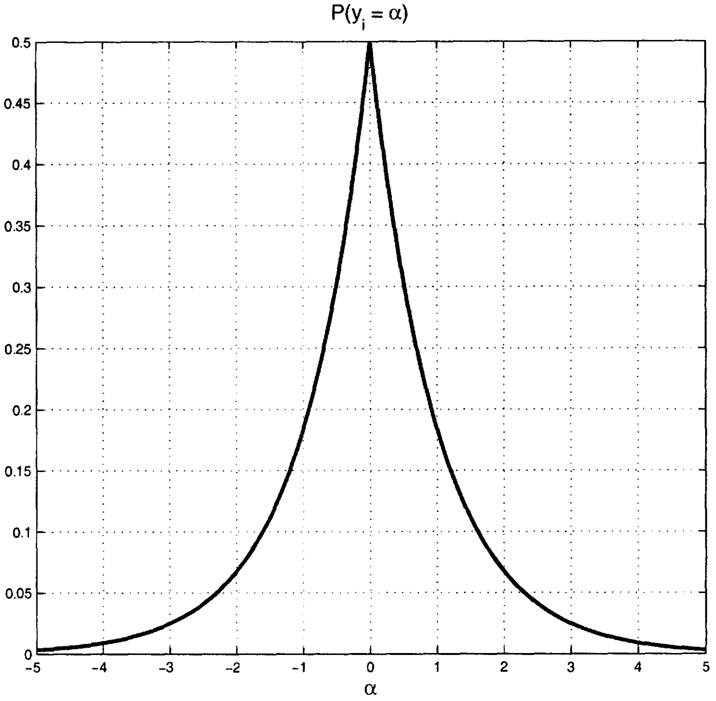


Figure A-1: Distribution of y_i for $K = 2$, $\tilde{\theta} \sim N(0, 1)$ and $f_{i,j} \sim N(0, 1)$

Bibliography

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

[2] E. Candès and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, June 2004. Submitted.

[3] E. J. Candès, J. K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, August 2006.

[4] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal of Scientific Computing*, 20(1):33–61, 1998.

[5] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Wiley and Sons, Inc., 1991.

[6] R. A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

[7] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306, April 2006.

[8] D. L. Donoho, M. Elad, and V. N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1):6–18, January 2006.

[9] V. K Goyal. Multiple description coding: Compression meets the network. *IEEE Signal Processing Magazine*, 18:74–93, September 2001.

[10] V. K Goyal, M. Vetterli, and N. T. Thao. Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis, and algorithms. *IEEE Transactions on Information Theory*, 44:16–31, January 1998.

[11] R. M. Gray and D. L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44:2325–2383, October 1998.

[12] J. Haupt and R. Nowak. Signal reconstruction from noisy random projections. *IEEE Transactions on Information Theory*, 2006. To appear.

[13] S. Mallat and F. Falzon. Analysis of low bit rate image transform coding. *IEEE Transactions on Signal Processing*, 46:1027–1042, April 1998.

[14] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, December 1993.

[15] S. S. Pradhan and K. Ramchandran. Distributed source coding using syndromes (DISCUS): Design and construction. *IEEE Transactions on Information Theory*, 49:626–643, March 2003.

[16] D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, IT–19:471–480, July 1973.

[17] J. A. Tropp. Greed is good: algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, October 2004.

[18] J. A. Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 52(3):1030–1051, March 2006.

[19] C. Weidmann. *Oligoquantization in Low-Rate Lossy Source Coding.* PhD thesis, Ecole Polytechnique Fédérale de Lausanne (EPFL), 2000.

[20] C. Weidmann and M. Vetterli. Rate-distortion analysis of spike processes. In *Proceedings of IEEE Data Compression Conference*, pages 82–91, Snowbird, Utah, March 1999.