Multiple Description Coding for Distributed Compressed Sensing

BY

DIEGO VALSESIA
B.S. Politecnico di Torino, Turin, Italy, 2010
M.S. Politecnico di Torino, Turin, Italy, 2012

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Chicago, 2013

Chicago, Illinois

Defense Committee:

Rashid Ansari, Chair and Advisor
Enrico Magli, Politecnico di Torino
Daniela Radakovic, Motorola Corp.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

CHAPTER **PAGE**

LIST OF FIGURES

LIST OF FIGURES (continued)

FIGURE **PAGE**

LIST OF FIGURES (continued)

FIGURE **PAGE**

LIST OF ABBREVIATIONS

BP	Basis Pursuit
BPDN	Basis Pursuit DeNoising
BPDQ	Basis Pursuit DeQuantizing
CS	Compressed Sensing
DCT	Discrete Cosine Transform
JSM	Joint Sparsity Model
MD	Multiple Description
MDC	Multiple Description Coding
MDSQ	Multiple Description Scalar Quantizer
MMV	Multiple Measurement Vectors
MSE	Mean Square Error
NSP	Null Space Property
OMP	Orthogonal Matching Pursuit
RD	Rate-Distortion
RIP	Restricted Isometry Property
SNR	Signal to Noise Ratio
QC	Quantization Consistency

SUMMARY

This thesis is devoted to the study of a multiple description framework for compressed sensing, with particular focus on a distributed application of compressed sensing. Compressed sensing is a novel theory for signal acquisition, that enables to directly acquire a compressed representation of a sparse or compressible signal, regardless of what is the basis in which the signal is actually sparse (compressible). The low complexity of an acquisition stage adopting compressed sensing raised interests about adopting it in sensor networks. In this distributed scenario compressed sensing is also able to exploit inter-correlation, in the form of joint sparsity, among different signals to improve coding efficiency without demanding any complex operation.

Our work proposes the CS-SPLIT scheme to generate two or more descriptions from measurements acquired through compressed sensing. This scheme proved experimentally superior to another classic method of obtaining multiple descriptions, that is using a multiple description scalar quantizer on the measurements (CS-MDSQ). An analytic treatment of the two methods in terms of rate-distortion performance has also been given, using the current results in the theory of compressed sensing.

CS-SPLIT can be readily used in sensor networks thanks to its extreme simplicity. We developed two new joint reconstruction algorithms (*Difference* and *Texas Difference*) that significantly improve over existing algorithms for the JSM-1 model, when the number of measurements is limited. This is relevant to multiple descriptions because it allows to get a better quality in the reconstruction of the single descriptions of CS-SPLIT when joint decoding is not possible.

CHAPTER 1

INTRODUCTION

The framework of *compressed sensing* (CS, also known as *compressive sampling*) has emerged in recent years from the ground breaking work of Candès, Romberg, Tao and Donoho that set the foundations of the theory and derived many of the fundamental results. In order to understand why compressed sensing has attracted so much interest in areas of applied mathematics, electrical engineering and computer science, we must realize how it goes past the traditional limits set by the sampling theory.

Over the past decades, the Nyquist-Shannon sampling theorem enabled the digital revolution by proving that a continuous-time band-limited signal could be perfectly reconstructed from uniformly spaced samples taken at a rate greater than twice the maximum frequency in the signal (the Nyquist rate). The implications of this result turned out to be immense since processing signals, now vectors, in the digital domain rather than in their analog form allowed the development of new and complex systems, not to mention that the digital counterpart of already available analog systems are more robust, flexible and cheaper.

However the sampling theorem has a fundamental limitation in that the acquisition rate required to satisfy the theorem may be so high for some applications that one finds himself with far too many samples or unable to build a device that reaches that sampling rate. There are applications in which the amount of data produced by acquisition must be reduced before being able to do anything useful on them. As an example think of imaging systems: still images that

can be used for medical purposes or, even more challenging, video are highly dimensional signals that must resort to some kind of compression to make them more manageable. Compression may be lossless, when it is possible to recover exactly the signal from its concise representation, or lossy, when some pieces of information are discarded in order to achieve higher compression ratios.

One of the most popular methods for signal compression is *transform coding*. Transform coding relies on finding an equivalent representation of the signal under a different basis that unveils some kind of hidden structure. It is known that many signals of interest (e.g. natural images, music, ...) have a very compact representation under certain bases. This means that it is possible to represent those signals with just few coefficients when you look at them under the proper basis. A signal with n entries may have just $k \ll n$ coefficients that are significantly nonzero when represented with the right basis, so transform coding just keeps those few nonzero coefficients to obtain a good, compact approximation of the original signal.

This discovery of an hidden structure in many signals raises an interesting point, whether it is really useful to acquire so many samples of a signal as dictated by the sampling theorem, when the signal itself actually carries a more limited amount of information.

Beyond Shannon

Compressed sensing focuses on the particular set of signal that have a sparse or compressible representation in some basis. In the literature we can find the terms *sparse* to indicate a signal that has exactly k nonzero entries and *compressible* when it is well approximated by the k largest nonzero entries. The core results of CS show that it is possible recover a signal from a

small set of linear, non-adaptive measurements, a situation that would be regarded as a vastly undersampled problem in the classical framework. Most notably, exact recovery is possible for sparse signals under certain assumptions, while some guarantees on the recovery error have been derived for compressible signals. Before going into further details about the theory of CS, it is worth noticing how the framework of CS, that is explored by the majority of the literature on the topic, slightly differs from the assumptions of classical sampling theory:

1. **Sampling - Sensing**

   Classic: continuous-time, infinite-length signals

   CS: vectors in $\mathbb{R}^n$

2. **Measurements**

   Classic: value of the signal at some point in time or space

   CS: inner product between signal and test function

3. **Recovery**

   Classic: sinc interpolation

   CS: nonlinear methods

Despite the fact that the foundations of the theory focus on applying the concepts of compressed sensing to discrete-time signals, i.e. vectors in $\mathbb{R}^n$, there is a lot of research going on in the direction of directly sensing continuous signals in a compressed manner. This has clear advantages in avoiding an intermediate step in which the signal is first sampled using the classical approach and then compressed sensing measurements are produced. A notable early work on hardware directly implementing CS acquisition is the single-pixel camera developed in [1].

Quick review of linear algebra

As the theory of CS relies on concepts of linear algebra it is relevant to recall a few basic notions. Most of the theory focuses on the $\mathbb{R}^n$ vector space and its subspaces.

- A normed vector space is a vector space endowed with a norm, i.e. a function $g : \mathbb{R}^n \to \mathbb{R}^+$ such that:
  - $g(\underline{x}) \geq 0$ and $g(\underline{x}) = 0$ if and only if $\underline{x} = 0$
  - $g(a\underline{x}) = |a|g(\underline{x})$ with $a \in \mathbb{R}$
  - $g(\underline{x} + \underline{y}) \leq g(\underline{x}) + g(\underline{y})$ (*triangle inequality*)
- For $\mathbb{R}^n$ the l_p norms are defined as follows:

$$\|\underline{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}} \quad \text{for } p \geq 1$$

$$\|\underline{x}\|_\infty = \max_{i=1,\ldots,n} |x_i|$$

- The concept can be extended to $p < 1$ obtaining a quasinorm.
- Important is the l_0 norm that tells the cardinality of the support of $\underline{x}$

$$\|\underline{x}\|_0 = |\text{supp}(\underline{x})|$$

- Inner product. For $\underline{x}$, $\underline{z} \in \mathbb{R}^n$

$$\langle \underline{x}, \underline{z} \rangle = \underline{x}^T \underline{z} = \sum_{i=1}^{n} x_i z_i$$

- $\left\{ \underline{\psi_i} \right\}_{i=1}^{n}$ is a basis (dictionary) for $\mathbb{R}^n$ if it spans $\mathbb{R}^n$ and $\underline{\psi_i}$ are linearly independent. Consequently any $\underline{x} \in \mathbb{R}^n$ can be written as

$$\underline{x} = \sum_{i=1}^{n} c_i \underline{\psi_i} = \Psi \underline{c}$$

where $\Psi = \left[\underline{\psi_1} | \underline{\psi_2} | \cdots | \underline{\psi_n} \right] \in \mathbb{R}^{nxn}$ and $\underline{c}$ is a column vector of coefficients.

A basis is orthonormal if $\left\langle \underline{\psi_i}, \underline{\psi_j} \right\rangle = \delta_{ij}$. Under this assumption the coefficients can be computed as

$$\underline{c} = \Psi^T \underline{x}$$

- $\left\{ \underline{\psi_i} \right\}_{i=1}^{n}$ is a frame (overcomplete dictionary) made of vectors $\underline{\psi_i} \in \mathbb{R}^d$ with $d < n$ if for all vectors $\underline{x} \in \mathbb{R}^n$:

$$\alpha \left\| \underline{x} \right\|_2^2 \leq \left\| \Psi^T \underline{x} \right\|_2^2 \leq \beta \left\| \underline{x} \right\| \qquad \text{with } 0 < \alpha \leq \beta < \infty$$

If α is the largest possible and β the smallest possible we have the optimal frame bounds.

α is the largest eigenvalue of Ψ and β is the smallest eigenvalue of Ψ.

The dual frame $\tilde{\Psi}$ is such that $\tilde{\Psi}\Psi^T = \Psi\tilde{\Psi}^T = I$. The canonical dual frame is the Moore-

Penrose pseudoinverse $\tilde{\Psi} = \left(\Psi^T \Psi\right)^{-1} \Psi$.

The coefficients can be obtained by

$$\underline{c} = \tilde{\Psi}^T \underline{x}$$

- spark(A) is the smallest number of linearly dependent columns of matrix A

1.1 Compressed Sensing

The framework of compressed sensing can be explained in the following manner. We have a signal $\underline{x}$ that is k-sparse (or compressible). $\underline{x}$ may be sparse by itself or because it is the representation of a signal $\underline{f}$ under some basis $\underline{f} = \Psi \underline{x}$. We will call the set of k-sparse signals $\Sigma_k = \{\underline{x} : \|\underline{x}\|_0 \leq k\}$. We will call $\sigma_k(\underline{x})_p = \min_{\hat{\underline{x}} \in \Sigma_k} \|\underline{x} - \hat{\underline{x}}\|_p$ the best k-term approximation error, i.e. the error, measured with an l_p norm, that is made by approximating a compressible signal with its k terms with largest magnitude.

The measurements $\underline{y} = \mathbb{R}^m$ are obtained by taking the inner product between the signal $\underline{x} \in \mathbb{R}^n$ and a sensing matrix $A \in \mathbb{R}^{m \times n}$.

$$\underline{y} = A\underline{x}$$

We have few measurements compared to the length of the signal ($m \ll n$), so we have a reduction in the dimension of the data we have to deal with. However, there are some questions that must be addressed for any of this to make sense. First, how should the sensing matrix be designed in order to get meaningful measurements, i.e. measurements that preserve the

information contained in $\underline{x}$? This problem is of primary concern for the success of the method and it is addressed by examining some properties of the sensing matrix A. Second, we need a procedure to stably recover the signal $\underline{x}$ from the measurements $\underline{y}$, if such procedure exists at all. It is interesting to note that the problem of recovering $\underline{x}$ from $\underline{y} = A\underline{x}$ requires solving an underdetermined linear system. This problem cannot be solved exactly in general as there are infinitely many valid solutions. However, compressed sensing exploits the underlying knowledge that $\underline{x}$ is sparse, to make this problem solvable.

1.1.1 Properties of the sensing matrix

The first intuitive reason why we want the sensing matrix A to satisfy some properties is that we want to obtain meaningful measurements in the sense that it is not possible that two different signals return the same measurements. Assuming $\underline{x}, \underline{x}' \in \Sigma_k$ and $\underline{z} = \underline{x} - \underline{x}'$, hence $\underline{z} \in \Sigma_{2k}$, we want:

$$\begin{aligned} & A\underline{x} \neq A\underline{x}' \\ \Rightarrow \quad & A\left(\underline{x} - \underline{x}'\right) \neq 0 \\ \Rightarrow \quad & A\underline{z} \neq 0 \end{aligned}$$

Therefore, we require that the null space of A contains no vector from the set of $2k$-sparse signals.

Theorem 1. *For any vector $\underline{y} \in \mathbb{R}^m$ there exists at most one signal $\underline{x} \in \Sigma_k$ such that $\underline{y} = A\underline{x}$ if and only if spark$(A) > 2k$*

A proof of this theorem is given in [2]. A direct consequence of this theorem is that $m \geq 2k$ since the maximum spark is equal to $m - 1$.

It must be remarked that, especially when dealing with compressible signals, we must require that the null space of A does not contain vectors that are too sparse, i.e. we would like the zero vector to be the only k-sparse vector in the null space of A. This is the meaning of the *Null Space Property.*

Definition 2. (Null Space Property) A matrix A satisfies the NSP of order k if there exists a constant $C > 0$ such that

$$\|\underline{h}_{\Lambda}\|_2 \leq C\frac{\|\underline{h}_{\Lambda^c}\|_1}{\sqrt{k}}$$

holds for all $\underline{h} \in \mathcal{N}(A)$ and for all Λ such that $|\Lambda| \leq k$.
The subscript Λ introduced with this notation is a set of indices and $\underline{h}_{\Lambda}$ is the vector $\underline{h}$ whose components not indexed by Λ have been set to zero.

The NSP is a sufficient and necessary condition to establish guarantees on signal reconstruction. Let's denote by $\Delta(A\underline{x})$ the reconstruction obtained through some algorithm of the original signal.

Theorem 3. *If the pair (A,Δ) satisfies the guarantee on reconstruction*

$$\|\Delta(A\underline{x}) - \underline{x}\|_2 \leq C\frac{\sigma_k(\underline{x})_1}{\sqrt{k}}$$

then A satisfies the NSP of order 2k.

A proof is given in [2].

It has also been shown that for the l_1 minimization algorithm that will be later introduced, satisfying the NSP establishes a guarantee of the kind reported in the previous theorem.

Theorem 4. *Let A be a matrix satisfying the NSP of order k under this form $\|\underline{h}_\Lambda\|_1 \leq \gamma \|\underline{h}_{\Lambda^c}\|_1$ with $0 < \gamma < 1$. Let $\underline{\hat{x}}$ be the reconstruction obtained through the l_1 minimization algorithm (basis pursuit). Then*

$$\|\underline{\hat{x}} - \underline{x}\|_1 \leq \frac{2(1+\gamma)}{1-\gamma}\sigma_k(\underline{x})_1$$

A proof is given in [3]

While the NSP is both necessary and sufficient condition to establish guarantees it doesn't consider noise. Another property is particularly desired when measurements may be contaminated with noise (it could be an error vector that sums to $A\underline{x}$ or quantization noise, ...) and it is the Restricted Isometry Property. The meaning of the Restricted Isometry Property is that the matrix A approximately preserves the Euclidean distance between any pair of k-sparse vectors. Many of the guarantees derived for different reconstruction methods use assumptions on A satisfying the RIP with certain bounds on the RIP constant.

Definition 5. (Restricted Isometry Property) *A matrix A satisfies the RIP of order k if there exists a $\delta_k \in (0,1)$ such that*

$$(1-\delta_k)\|\underline{x}\|_2^2 \leq \|A\underline{x}\|_2^2 \leq (1+\delta_k)\|\underline{x}\|_2^2$$

holds for all $\underline{x} \in \Sigma_k$

RIP and NSP are related. In fact, it can be shown that the RIP implies the NSP [3].

However proving that the RIP or the NSP hold for a particular sensing matrix may be computationally complex, so it is preferred to investigate another parameter, the coherence of A. The coherence can be related to the RIP, the NSP and the spark of A.

Definition 6. (Coherence)

$$\mu(A) = \max_{1 \leq i < j \leq n} \frac{\left|\left\langle \underline{a}_i, \underline{a}_j \right\rangle\right|}{\|\underline{a}_i\|_2 \left\|\underline{a}_j\right\|_2}$$

The coherence basically tells the maximum value of the inner product between columns of the sensing matrix. It can be shown [2] that $\text{spark}(A) \geq 1 + \frac{1}{\mu(A)}$ and this information can be used together with the fact that $\text{spark}(A) \geq 2k$ to obtain the following theorem.

Theorem 7. *If* $k < \frac{1}{2}\left(1 + \frac{1}{\mu(A)}\right)$ *then for each measurement vector* $\underline{y} \in \mathbb{R}^m$ *there exist at most one signal* $\underline{x} \in \Sigma_k$ *such that* $\underline{y} = A\underline{x}$.

This result immediately highlights the fact that we would like to have a sensing matrix with low coherence in order to be able to recover more sparse vectors.

1.1.2 Signal recovery

We are now going to address the second problem that we posed when introducing compressed sensing, that is whether a method to recover $\underline{x}$ from measurements $\underline{y}$ exists. The first intuition is to cast the problem in the form of "*look for the sparsest vector that satisfies the measurements*"

and recalling that the l_0 norm expresses the cardinality of the support of the signal, it is possible to formally write it as

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_0 \quad \text{subject to } \underline{z} \in \mathcal{B}\left(\underline{y}\right)$$

$\mathcal{B}\left(\underline{y}\right)$ can have many forms itself, depending on the type of problem that we are solving. e.g. $\mathcal{B}\left(\underline{y}\right) = \left\{\underline{z} : A\underline{z} = \underline{y}\right\}$ in case of a noise-free problem or $\mathcal{B}\left(\underline{y}\right) = \left\{\underline{z} : \left\|A\underline{z} - \underline{y}\right\|_2 < \epsilon\right\}$ in case of bounded noise, where ϵ is the expected norm of the noise. However, the problem of minimizing the l_0 norm is computationally complex because of its combinatorial nature (it requires an exhaustive search). It has been discussed in [4] that l_0-minimization contains the subset sum problem that is known to be NP-complete[1]. The computational intractability of the previous solution led to the study of alternative solutions. In particular, substituting the l_0 norm with the l_1 norm, turned out to be a major breakthrough as it an extremely successful method with limited complexity, being a convex optimization problem that can be solved by well known techniques. Even before the results concerning reconstruction by l_1 minimization were proven, there were reasons to think that the l_1 norm was a good candidate to replace the l_0 norm. A first intuition is related to the fact that l_p norms with a low value of p seem to promote sparse solutions to systems as hinted in [2]. Moreover the l_1 norm was already a successful tool in fields like seismology, dealing, in fact, with measurements of sparse signals [6] [7]. Some

---

[1]Quoting [5] "*No polynomial-time algorithm has yet been discovered for an NP-complete problem, nor has anyone yet been able to prove that no polynomial-time algorithm can exist for any of them. This so called $P \neq NP$ question [is] one of the deepest, most perplexing open research problems in theoretical computer science*"

common names to the l_1-minimization procedure can be found in the literature for different constraints.

$$\begin{aligned}
\underline{\hat{x}} &= \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } A\underline{z} = \underline{y} \qquad (basis\, pursuit) \\
\underline{\hat{x}} &= \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \left\|A\underline{z} - \underline{y}\right\|_2 < \epsilon \qquad (basis\, pursuit\, denoise) \\
\underline{\hat{x}} &= \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \left\|A^T\left(A\underline{z} - \underline{y}\right)\right\|_\infty \leq \lambda \qquad (Dantzig\, selector)
\end{aligned}$$

We are now reporting a few theorems establishing guarantees on signal reconstruction using l_1-minimization . Those theorems are formally proven in [2].

Theorem 8. *(Noise-free basis pursuit) Let A satisfy the RIP of order $2k$ with $\delta_{2k} < \sqrt{2} - 1$. Let the measurements be $\underline{y} = A\underline{x}$ and $\underline{\hat{x}} = \underset{\underline{z}}{argmin} \|\underline{z}\|_1$ subject to $A\underline{z} = \underline{y}$. Then*

$$\|\underline{\hat{x}} - \underline{x}\|_2 \leq c_0 \frac{\sigma_k\left(\underline{x}\right)_1}{\sqrt{k}}$$

with $c_0 = 2\frac{1-\left(1-\sqrt{2}\right)\delta_{2k}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}$.

Theorem 9. *(Bounded noise) Let A satisfy the RIP of order $2k$ with $\delta_{2k} < \sqrt{2} - 1$. Let the measurements be $\underline{y} = A\underline{x} + \underline{e}$ with $\|\underline{e}\|_2 \leq \epsilon$ and $\underline{\hat{x}} = \underset{\underline{z}}{argmin} \|\underline{z}\|_1$ subject to $\left\|A\underline{z} - \underline{y}\right\|_2 < \epsilon$. Then*

$$\|\underline{\hat{x}} - \underline{x}\|_2 \leq c_0 \frac{\sigma_k\left(\underline{x}\right)_1}{\sqrt{k}} + c_2\epsilon$$

with $c_0 = 2\frac{1-\left(1-\sqrt{2}\right)\delta_{2k}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}$ *and* $c_2 = 4\frac{\sqrt{1+\delta_{2k}}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}$.

Corollary 10. *Under the previous assumptions on* A *and* $\underline{y} = A\underline{x} + \underline{e}$ *with* $\underline{x} \in \Sigma_k$*, let* $\underline{e}$ *have i.i.d.* $\mathcal{N}\left(0, \sigma^2\right)$ *entries and let* $\hat{\underline{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1$ subject to $\left\|A\underline{z} - \underline{y}\right\|_2 < 2\sqrt{m}\sigma$. Then

$$\|\hat{\underline{x}} - \underline{x}\|_2 \leq c_2 2\sqrt{m}\sigma$$

with probability at least $1 - e^{-c_a m}$ where c_a is such that $P\left(\|\underline{e}\| \geq 2\sqrt{m}\sigma\right) \leq e^{-c_a m}$.

Theorem 11. *(Coherence-based bounded noise) Let* A *have coherence* μ *and* $\underline{x} \in \Sigma_k$ *with* $k < \frac{1}{4}\left(\frac{1}{\mu} + 1\right)$. *Let* $\underline{y} = A\underline{x} + \underline{e}$ *and let*

$\hat{\underline{x}} = \underset{\underline{z}}{argmin} \|\underline{z}\|_1$ *subject to* $\left\|A\underline{z} - \underline{y}\right\|_2 < \epsilon$. *Then*

$$\|\hat{\underline{x}} - \underline{x}\|_2 \leq \frac{\|\underline{e}\|_2 + \epsilon}{\sqrt{1 - \mu\left(4k - 1\right)}}$$

Theorem 12. *(Dantzig selector) Let* A *satisfy the RIP of order* $2k$ *with* $\delta_{2k} < \sqrt{2} - 1$. *Let the measurements be* $\underline{y} = A\underline{x} + \underline{e}$ *with* $\left\|A^T\underline{e}\right\|_\infty \leq \lambda$ *and* $\hat{\underline{x}} = \underset{\underline{z}}{argmin} \|\underline{z}\|_1$ *subject to* $\left\|A^T\left(A\underline{z} - \underline{y}\right)\right\|_\infty \leq \lambda$. *Then*

$$\|\hat{\underline{x}} - \underline{x}\|_2 \leq c_0 \frac{\sigma_k\left(\underline{x}\right)_1}{\sqrt{k}} + c_3\sqrt{k}\lambda$$

with $c_0 = 2\frac{1-\left(1-\sqrt{2}\right)\delta_{2k}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}$ *and* $c_3 = \frac{4\sqrt{2}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}$.

Corollary 13. *Under the previous assumptions on A and $\underline{y} = A\underline{x} + \underline{e}$ with $\underline{x} \in \Sigma_k$, let $\underline{e}$ have i.i.d. $\mathcal{N}\left(0, \sigma^2\right)$ entries and let $\hat{\underline{x}} = \underset{\underline{z}}{argmin} \|\underline{z}\|_1$ subject to $\left\|A^T\left(A\underline{z} - \underline{y}\right)\right\|_\infty \leq 2\sqrt{\log n}\sigma$. Then*

$$\|\hat{\underline{x}} - \underline{x}\|_2 \leq 4\sqrt{2}\frac{\sqrt{1+\delta_{2k}}}{1-\left(1+\sqrt{2}\right)\delta_{2k}}\sqrt{k\log n}\sigma$$

with probability at least $1 - \frac{1}{n}$.

Alternatives to the l1-minimization procedure exist. In particular, a popular greedy algorithm, that has been proved successful in the recovery task, is *Orthogonal Matching Pursuit* (OMP). In a nutshell, OMP works by estimating the sparsity support of the signal and then solving a least squares problem to recover the amplitude of the coefficients. More in detail the algorithm works as follows:

1. Initialize $\underline{r}_0 = \underline{y}$, $A(c_0) = \emptyset$, $i = 1$
2. Find the column $\underline{A_{t_i}}$ that maximizes $|\underline{A_t}^T \underline{r}_{i-1}|$ and add it to the set of columns $c_i = c_{i-1} \cup \{t_i\}$
3. Compute the residual from the projection onto the current space spanned by the set of columns indexed by c_i $\underline{r}_i = (I - A(c_i)[A(c_i)^T A(c_i)]^{-1} A(c_i)^T)\underline{y}$
4. If stopping criterion is met then stop, else $i = i + 1$, repeat from step 2

The stopping criterion plays an important role and should be tuned on the desired level of accuracy or estimate of the noise level. In case of bounded noise [8] suggest some stopping criteria. For example, in case of noise with bounded l_2 norm $\|\underline{n}\|_2 < \epsilon$ it is possible to use $\|\underline{r}_i\|_2 \leq \epsilon$ as stopping condition.

1.1.3 The problem of constructing the sensing matrix

So far we have introduced how it is possible to reconstruct the sparse signal $\underline{x}$ from the linear measurements $\underline{y}$ thanks to the l_1-minimization method, provided that the sensing matrix is well constructed. In fact, we have seen that the quality of the construction of A in terms of its RIP constant or coherence is crucial in establishing guarantees on the performance of the reconstruction algorithm. Also, it is interesting from a practical point of view to investigate how many measurements are actually needed to properly recover the original signal with a given sensing matrix. The first problem that is encountered when trying to construct a sensing matrix that satisfies the RIP, is that a deterministic matrix of size $m \times n$ requires m to be relatively large. A first example in this sense is reported in [3] where it is claimed that $A = [I|F]$ with $F_{x,y} = \frac{e^{j2\pi xy/m}}{\sqrt{m}}$ has small coherence and one needs $m \geq \text{const} \cdot k^2$. Another example is given by the deterministic matrix investigated in [9] that needs $m = O\left(kn^{o(1)}\right)$ measurements. In both cases the number of measurements is often too large for any practical application.

It is possible to overcome this limitation by forgetting deterministic matrices and, instead, using random sensing matrices. There are a number of random constructions that proved themselves to be viable solutions. Among them Bernoulli, Gaussian, subgaussian, random partial Fourier matrices are known to satisfy the RIP with overwhelming probability provided enough measurements [2] [3] [4].

1. *Bernoulli matrices*: entries are realizations of i.i.d. Bernoulli random variables that can take values $\pm\frac{1}{\sqrt{m}}$ with 0.5 probability. $m = O\left(k \log \frac{n}{k}\right)$

2. *Gaussian matrices*: entries are i.i.d. Gaussian random variables with zero mean and variance $\frac{1}{m}$. $m = O\left(k \log \frac{n}{k}\right)$

3. *Random partial Fourier*: A is derived taking m rows uniformly at random from the DFT matrix . $m = O\left(k \log^4 n\right)$

Another advantage of random matrices is universality, that is they work well regardless of the basis under which the original signal is sparse. For example, let Ψ be basis of choice and let A be a Gaussian sensing matrix then $A\Psi$ is Gaussian as well and will satisfy the RIP with high probability provided enough measurements.

CHAPTER 2

QUANTIZATION CONSISTENCY

Compressed sensing provides a way of obtaining a compressed representation of a sparse signal by acquiring a limited number of linear measurements in the form of random projections. Applications require these measurements to be quantized with a finite number of bits in order to meet the specifications in terms of bitrate of the system in use. This procedure is inevitably lossy, in the sense that some piece of information is lost during the quantization process. Therefore it is not possible to achieve a perfect reconstruction of the original signal in presence of quantized measurements.

A classic way of analysing the effects of quantization is to model the error introduced by the quantizer as additive noise.

$$\underline{y_q} = \underline{y} + \underline{n} \tag{2.1}$$

Models for $\underline{n}$ have been studied for different classes of quantizers. For instance, for the uniform scalar quantizer it is common practice to use a high resolution approximation that claims that the noise vector is made of independent and identically distributed entries having a uniform distribution inside the quantization interval. It follows that

$$E[\|\underline{n}\|_2^2] = \frac{\Delta^2}{12} m \tag{2.2}$$

where Δ is the quantization step and m the length of vector $\underline{\mathrm{n}}$.

The reconstruction algorithm must consider the fact that exact reconstruction is impossible. A possible reconstruction method is to solve the *basis pursuit denoising* (BPDN) convex optimization problem. Its formulation is

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \left\|A\underline{z} - \underline{y}\right\|_2 < \epsilon \tag{2.3}$$

We provide a bound on the average noise power by means of ϵ. e.g. $\epsilon = \sqrt{\frac{\Delta^2}{12}m}$. Another solution is to use a greedy algorithm like *Orthogonal Matching Pursuit* (OMP). It is possible to set its stopping condition, as showed in [8], to when the 2-norm of the residue falls below ϵ.

The presented reconstruction algorithms work well, but they raise an interesting theoretical point: they only set a bound on the average noise. Measuring the reconstructed signal $\underline{\hat{x}}$ with the original sensing matrix we obtain a new set of measurements $\underline{\hat{y}}$. The aforementioned reconstruction algorithms do not guarantee that the new measurements fall inside the original quantization intervals. Some experiments confirming this fact are shown in Figure 2.1 and Figure 2.2. The figures report the percentage of new measurements that fall outside the original quantization bins. Those numbers do not seem to be negligible, therefore it is interesting to enforce this condition on the reconstructed signal in order to analyse if there a significant gain in exploiting all the information at our disposal. This problem is often denoted as *quantization consistency* or *consistent reconstruction* and it has been considered in [10], [11], [12].

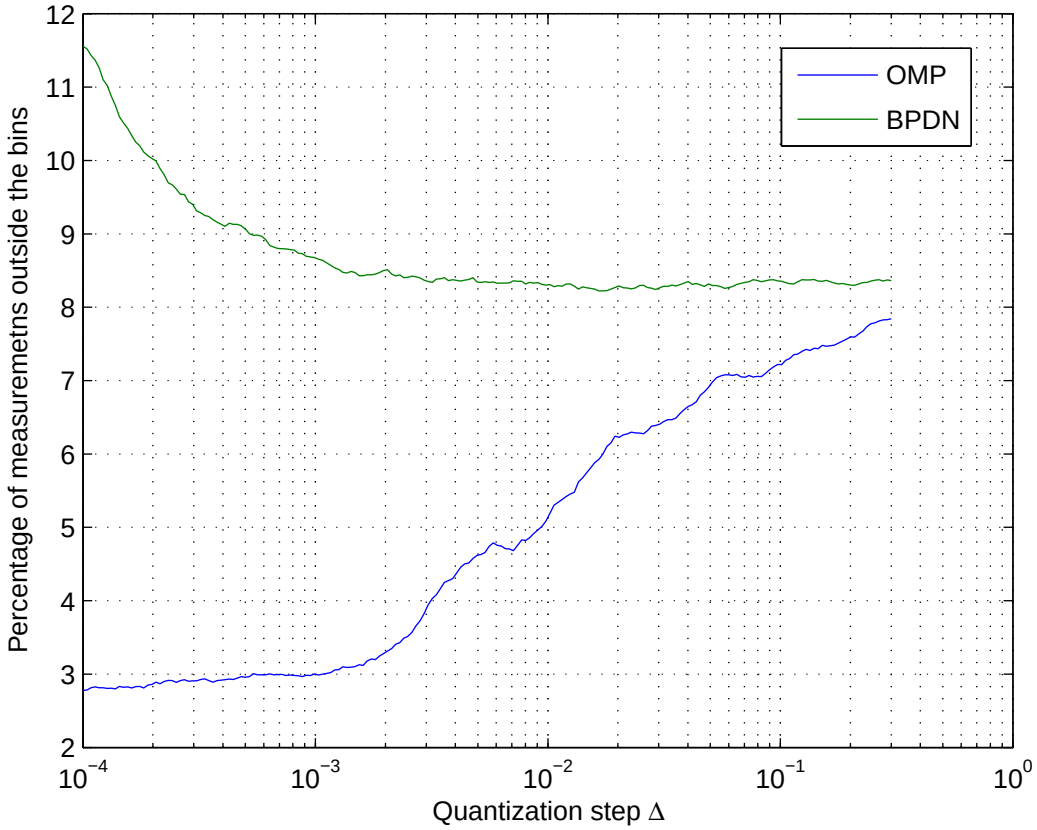

Figure 2.1. Percentage of measurements falling outside the quantization intervals. Original signal has length $n = 1024$, sparsity $k = 100$ in the identity basis and $m = 400$ measurements are taken with a Gaussian sensing matrix. SPGL1 [13] was used to solve the BPDN problem.

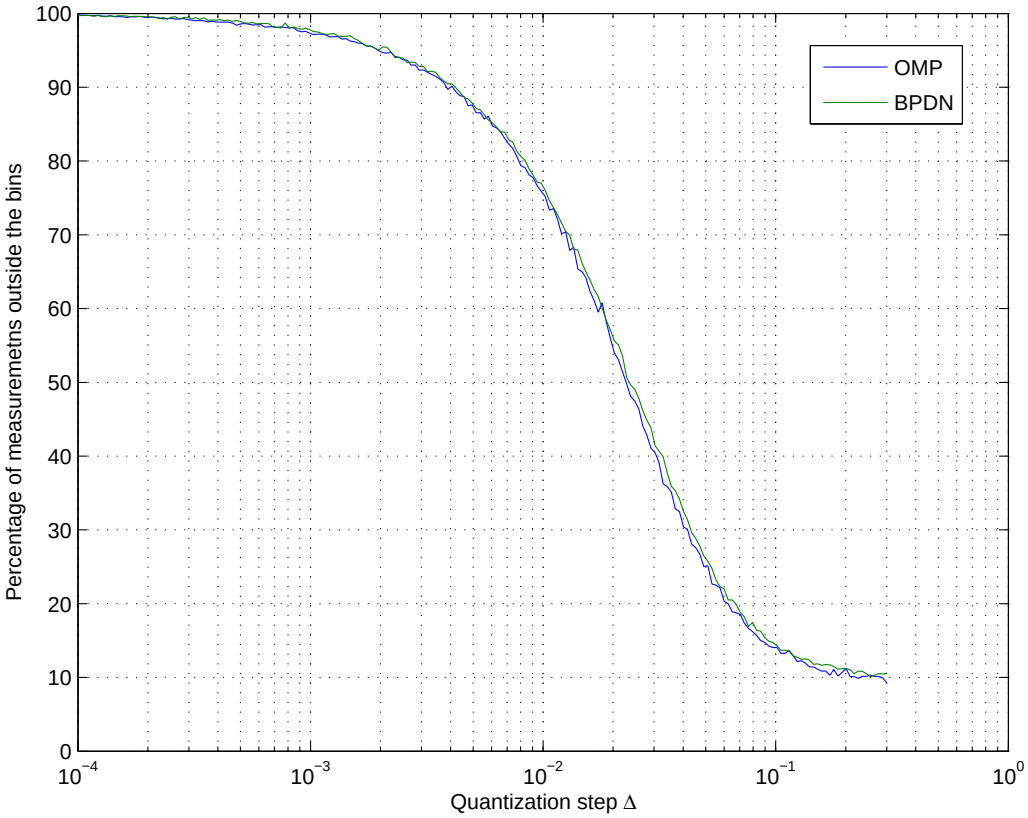

Figure 2.2. Percentage of measurements falling outside the quantization intervals. Original signal has length $n = 1024$, sparsity $k = 100$ in the DCT basis. i.i.d. Gaussian noise is added in the identity domain and $m = 400$ measurements are taken with a Gaussian sensing matrix.

In [10] the authors analyse an improvement to BPDN, called *Basis Pursuit DeQuantization* (BPDQ) by imposing a constraint on the noise using an l_p norm instead of l_2, and develop a method to solve

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \left\|A\underline{z} - \underline{y}\right\|_p < \epsilon \tag{2.4}$$

The analysis shows that there is a gain in using BPDQ over BPDN when the number of measurements is large with respect to the sparsity of the signal. Also they claim that an optimal value of the moment p exists and $2 < p < \infty$. The procedure relates to quantization consistency as the percentage of new measurements falling inside the original quantization intervals is increasing. For a uniform quantizer $p = \infty$ achieves perfect quantization consistency, i.e. all the measurements fall in the new bins, but this is not optimal in terms of reconstruction error.

[11] shows that imposing quantization consistency is a convex constraint, so the optimization problem can be solved by standard methods in convex optimization. However, a problem posed in the following form, where we call $\mathcal{R}_y$ the set of quantization regions for the entries of $\underline{y}$,

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \quad A\underline{\hat{x}} \in \mathcal{R}_y \tag{2.5}$$

is not optimal in terms of reconstruction error. The authors showed gains in the reconstruction SNR, by exploiting quantization consistency, when the number of measurements is large with respect to the sparsity of the signal. This result seems to be consistent with the experimental findings using BPDQ.

We experimentally investigated whether quantization consistency could be advantageous in a setting in which it is desirable to take as few measurements as possible, and the oversampling rates considered in [10], [11] cannot be reached. A generic solver for convex optimization problems (CVX [14]) was used to test different approaches.

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \left\|A\underline{z} - \underline{y}\right\|_\infty \leq \frac{\Delta}{2} \tag{2.6}$$

$$\underline{\hat{x}} = \underset{\underline{z}}{\operatorname{argmin}} \|\underline{z}\|_1 \text{ subject to } \begin{cases} \left\|A\underline{z} - \underline{y}\right\|_\infty \leq \frac{\Delta}{2} \\ \left\|A\underline{z} - \underline{y}\right\|_2 \leq \sqrt{\frac{\Delta^2}{12}m} \end{cases} \tag{2.7}$$

A uniform scalar quantizer was used in the tests. Equation (2.6) imposes only the quantization consistency constraint on the reconstruction, while (2.7) works as the BPDN algorithm but enforcing consistent reconstruction. The outcome of the tests shows that there is no, or very limited, practical gain in enforcing quantization consistency. In particular, Figure 2.3 shows that the performance of the reconstruction method that enforces both a constraint on the average noise and the consistency performs as the standard BPDN method. It can be also noticed that a reconstruction method that uses only the information on consistency is always outperformed by the other methods. Figure 2.4 reports a more complex signal that is almost sparse, i.e. it is sparse but corrupted by some additive noise. Moreover the signal is sparse in the DCT basis. The results are the same as in the previous case.

A further test used the BPDQ program developed in [10]. Figure 2.5, Figure 2.6 and Figure 2.7 show the relative reconstruction error as a function of the number of measurements.

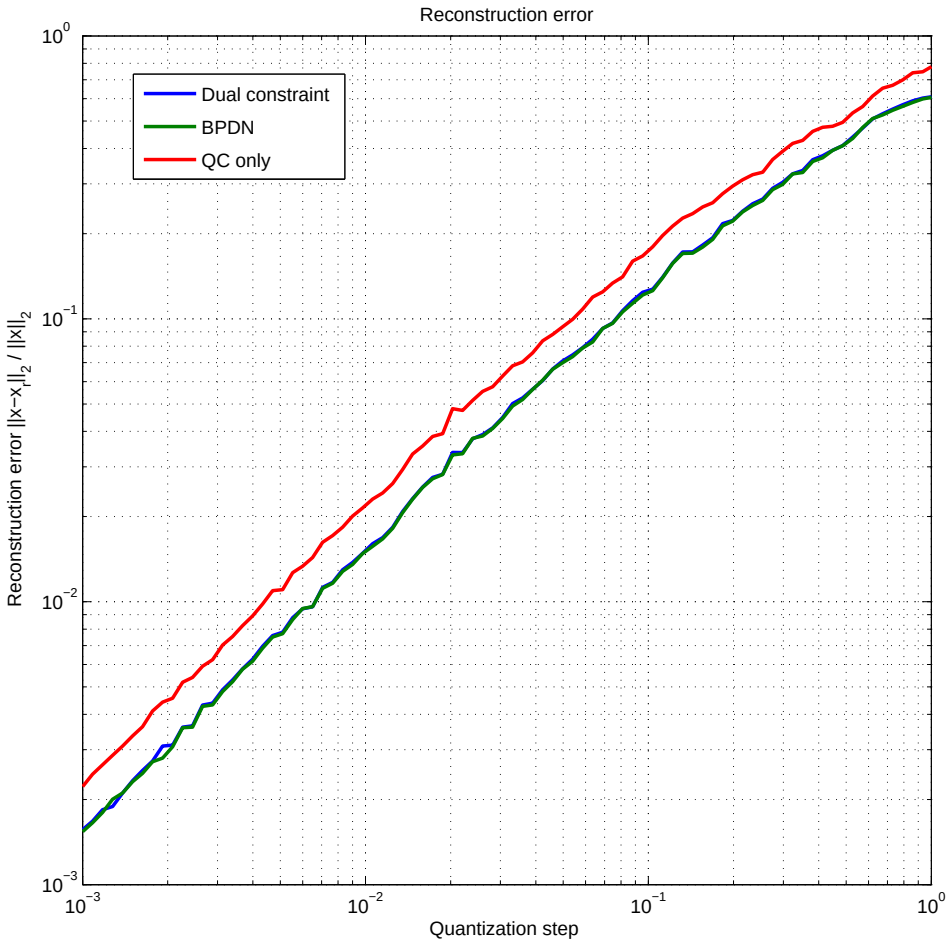

Figure 2.3. Original signal has length $n = 1024$, sparsity $k = 100$. It is sparse in the identity basis and $m = 400$ measurements are taken with a Gaussian sensing matrix.

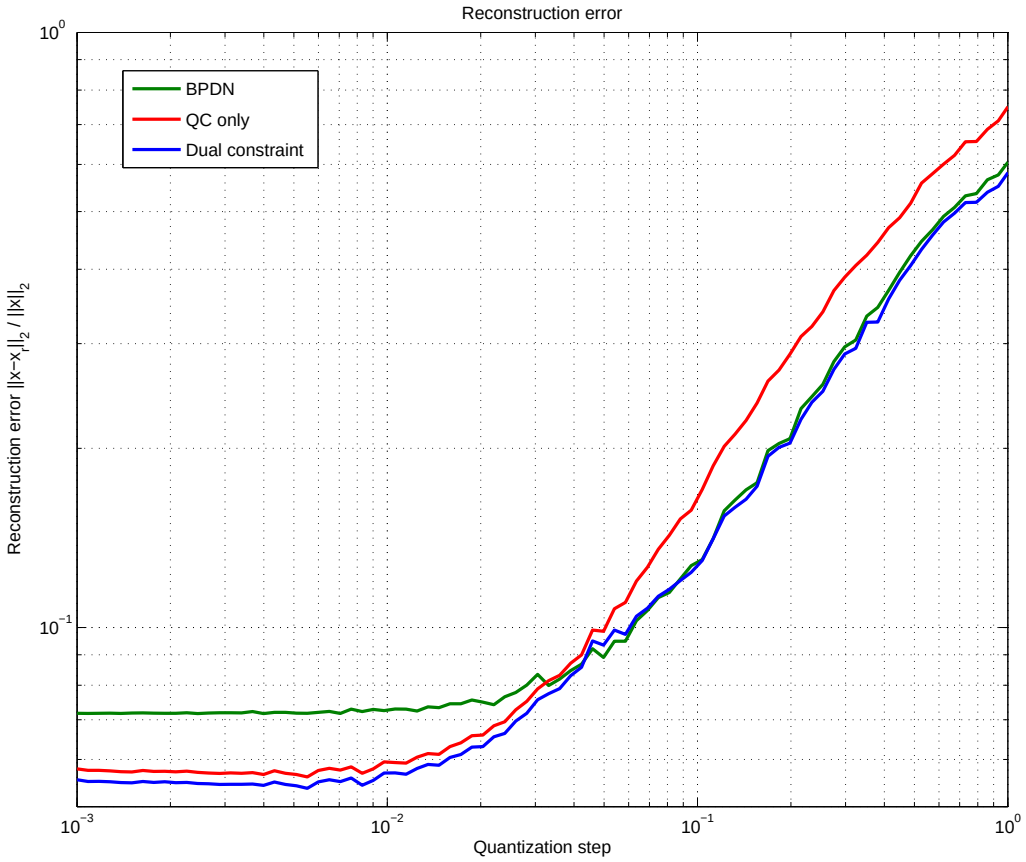

Figure 2.4. Original signal has length $n = 1024$, sparsity $k = 100$ in the DCT basis. i.i.d. Gaussian noise is added in the identity domain and $m = 400$ measurements are taken with a Gaussian sensing matrix.

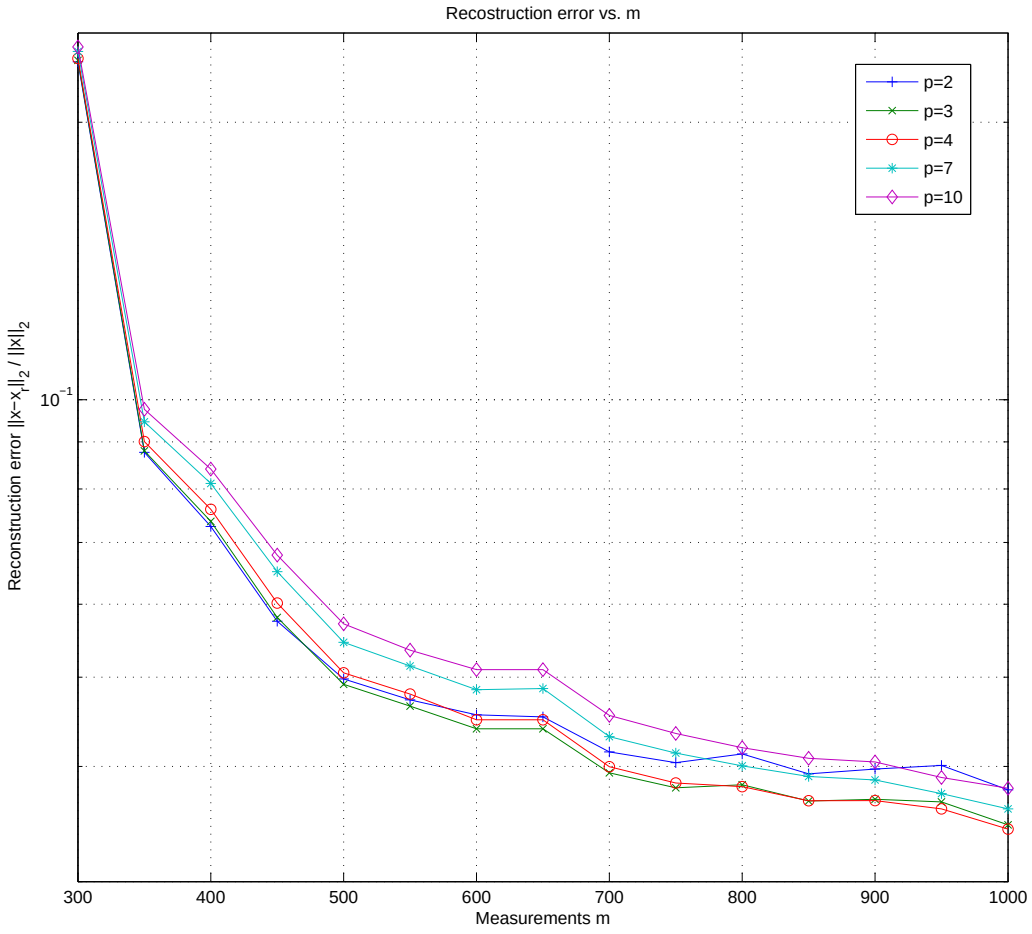

Figure 2.5. $BPDQ_p$ algorithm. Original signal has length $n = 1024$, sparsity $k = 100$. It is sparse in the identity basis. Quantization step of the measurements $\Delta = 0.05$.

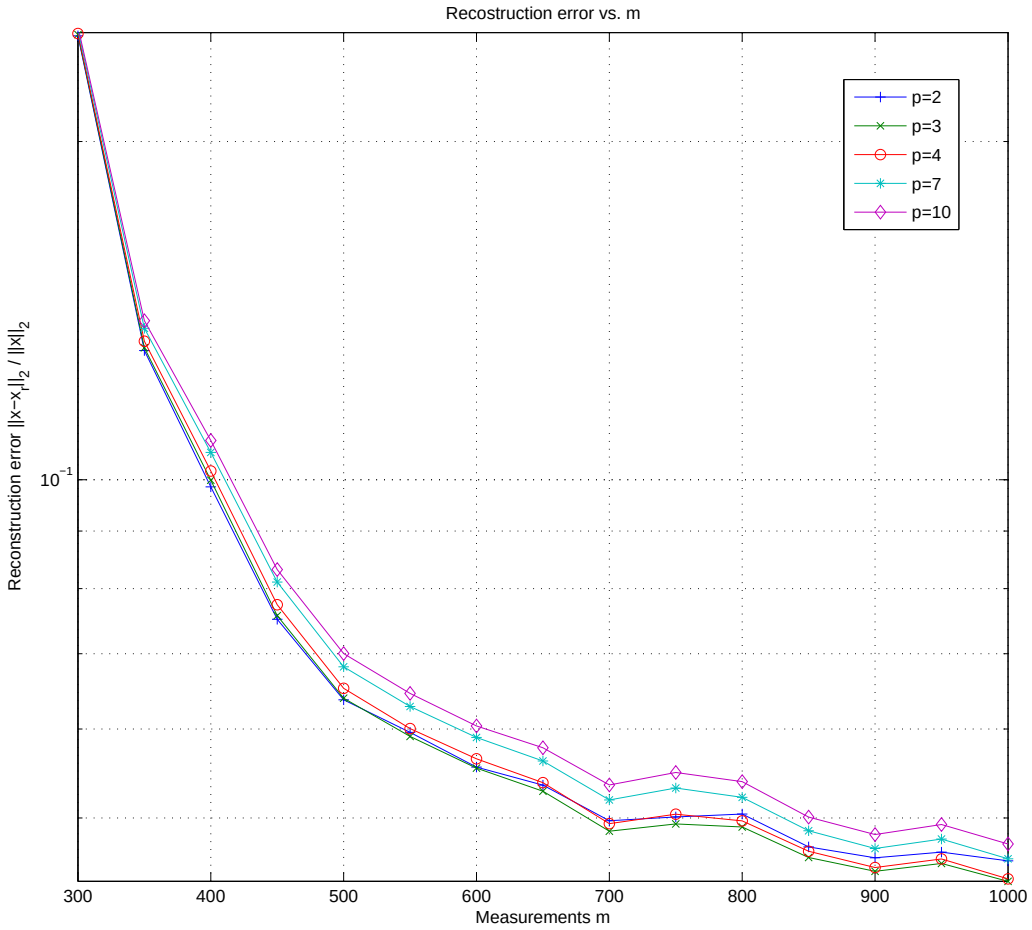

Figure 2.6. $BPDQ_p$ algorithm. Original signal has length $n = 1024$, sparsity $k = 100$ in the DCT basis. i.i.d. Gaussian noise is added in the identity domain. Quantization step of the measurements $\Delta = 0.05$.

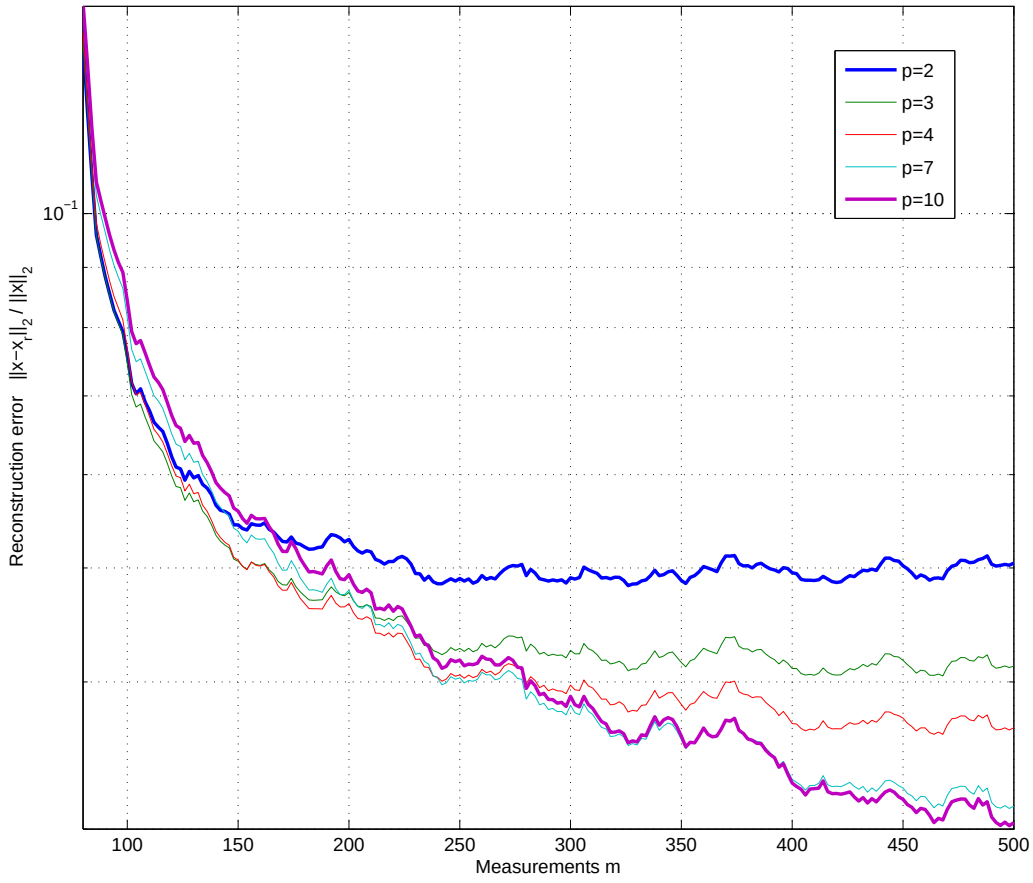

Figure 2.7. $BPDQ_p$ algorithm. Original signal has length $n = 256$, sparsity $k = 20$. It is sparse in the identity basis. Quantization step of the measurements $\Delta = 0.05$.

They support the claim that quantization consistency (here in the form of a higher moment p in the constraint of the reconstruction procedure) produces significant gains with respect to BPDN only when the oversampling factor increases, i.e. a large number of measurements is taken with respect to the sparsity level, many more than the minimum that is needed to ensure reconstruction with high probability.

CHAPTER 3

MULTIPLE DESCRIPTION CODING

Multiple description coding [15] is a way of coding an information source, that is resilient to transmission errors. Networks like the Internet may be unreliable due to their best-effort routing policy, so packet losses may happen at any time. The multiple description technique allows to create multiple correlated representations of the original set of data, each carrying enough information to decode the data with a certain fidelity. Losing a description will not make the data unusable since each description can be decoded separately, albeit with a limited quality. However, the best decoding quality is obtained when all the descriptions are available. Figure 3.1 shows a block diagram for a system working with two descriptions, that could be sent over separate channels. The decoding stage provides side decoders that are able to recover a low-quality version of the data when a single description is received, as well as a central decoder that is able to exploit both the descriptions to achieve the best decoding quality. Since we always speak of reduced or full quality, the multiple description paradigm makes sense only in a lossy coding environment, when it is acceptable for the end user to enjoy a variable degree of fidelity to the original source. Classic examples involve audio or speech signals, images, video, etc. *Side distortion* is the term used to refer to the extent of distortion, according to some fidelity criterion (e.g. mean square error), when a single description is available. *Central distortion* is the distortion that can be achieved when all the descriptions are received.

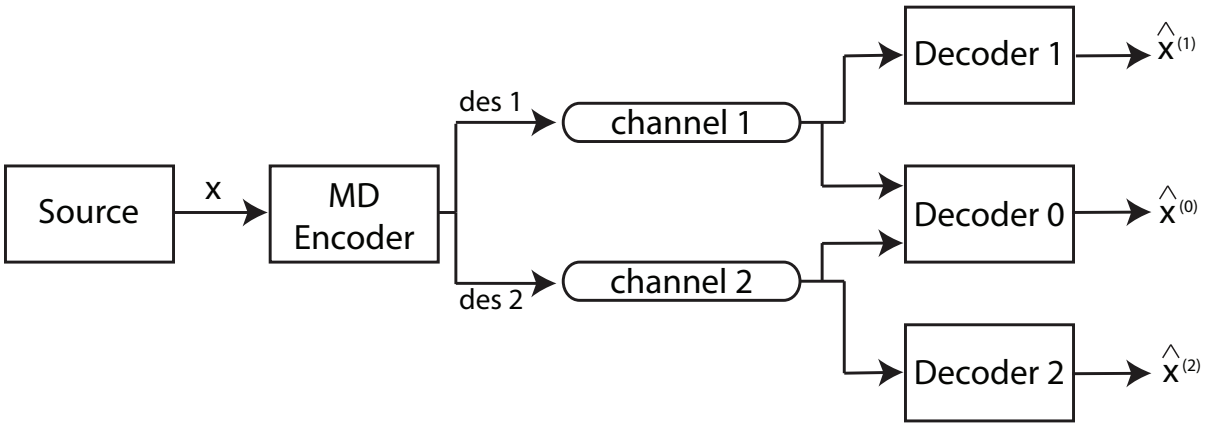

Figure 3.1. A two-description MD sytem

Resilience to description erasures is not for free and it comes at the price of a lower coding efficiency. There is an information-theoretic problem of determining the achievable rate-distortion region of a multiple description coding of a given source, i.e. the set of points identified by the rates of the descriptions and the corresponding side and central distortions that can be actually achieved for that specific source. This is in general a complex problem, but for a memoryless Gaussian source with variance σ^2 and the mean squared distortion metric, it is known [16] that the side distortions D_1, D_2 and the central distortion D_0 follow:

$$D_i \geq \sigma^2 2^{-2R_i} \;\; i = 1, 2 \tag{3.1}$$

$$D_0 \geq \sigma^2 2^{-2(R_1+R_2)} \cdot \gamma_D(R_1, R_2, D_1, D_2) \tag{3.2}$$

with

$$\gamma_D(R_1, R_2, D_1, D_2) = \frac{1}{1 - \left(\sqrt{\left(1 - \frac{D_1}{\sigma^2}\right)\left(1 - \frac{D_2}{\sigma^2}\right)} - \sqrt{\frac{D_1 D_2}{\sigma^4} - 2^{-2(R_1+R_2)}} \right)^2} \tag{3.3}$$

The practical implementation of the multiple description paradigm requires to find a smart way to generate the descriptions. In fact, one should try to get acceptable side distortions but at the same time the gain provided by central decoding should be worth receiving all the descriptions. If central decoding provides little improvement over the single descriptions, then it is not worth receiving all the available ones. Multiple descriptions can be produced in various stages of the transmission chain. Methods are known to generate descriptions by preprocessing the source (e.g. a trivial method is to separate even and odd samples), by applying a correlating transform, by using a special quantizer, or by applying unequal error protection to the transmitted data.

In the following we draw the attention to the *multiple description scalar quantizer* (MDSQ) outlined in [17], that is a way of producing the descriptions in the quantization process. Restricting ourselves to the two-description case, the basic idea behind the MDSQ is quantizing the data in such a way that each description can be coarsely dequantized, but a finer resolution can be achieved when both the descriptions are available. More formally, for each input $x \in \mathbb{R}$ two indices (i_1, i_2) are produced by a MDSQ with (M_1, M_2) levels, each of them representing a description of the input. The side decoder receives just one index and outputs an estimate of x based on the side codebook $\chi^{(i)} = \left\{ x_k^{(i)} | k \in \mathcal{I}_i \right\}$, $i = 1, 2$, where $\mathcal{I}_i = \{1, 2, ..., M_i\}$ for description i, and $\left\{ x_k^{(i)} \right\}$ is the set of possible output levels. The central decoder is able to use both

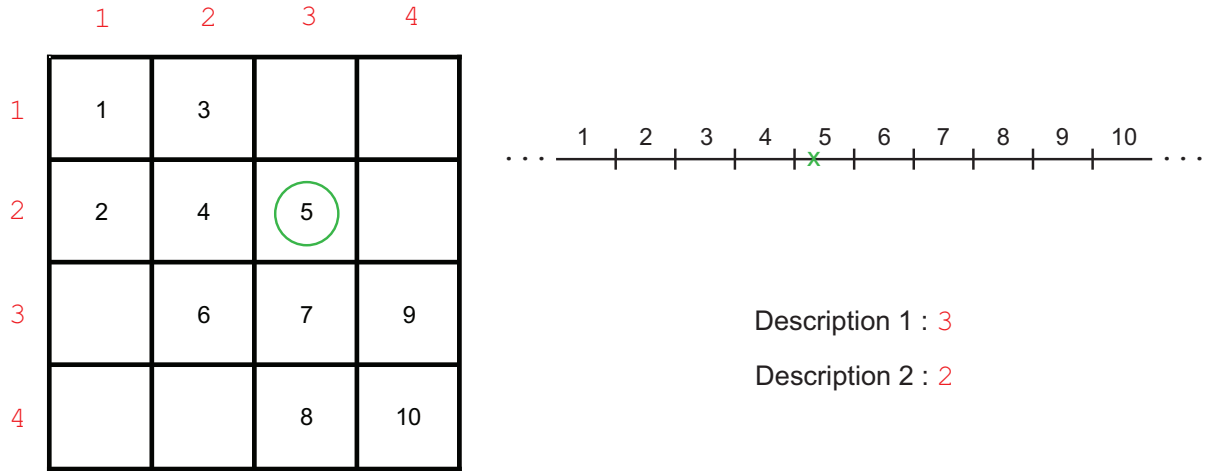

Figure 3.2. Example of MDSQ. The index assignment matrix uses nested assignment. The rate is $R = 2$ bits per description

descriptions to produce the estimate from the central codebook $\chi^{(0)} = \left\{x_{ij}^{(0)} | (i,j) \in \mathcal{I}_1 \times \mathcal{I}_2\right\}$ The encoder can be broken into two pieces: a standard quantizer defining the central partition and central codebook, and an index assignment problem that associates the pair (i_1, i_2) to each cell defined of the central partition. Optimal index assignment is a complex problem and no explicit solution is known but [17] gives some heuristic constructions. They are based on placing the cell numbers of the central partition in the *index assignment matrix* according to some heuristic criterion (e.g. nested assignment, or linear assignment). The row and column indices in the index assignment matrix are i_1 and i_2. Figure 3.2 shows an example index assignment matrix, filled using a procedure called *nested assignment* [17], and also shows how descriptions can be generated. In a balanced system, the matrix has 2^R rows and 2^R columns, where

R is the rate of the description, so at most 2^{2R} entries can be stored. However, there is a trade-off between how densely we can fill up the matrix and the achievable side distortions so one may want to fill just a few diagonals. The methods proposed by the author in [17] provide a parameter to control this filling density. Finally, optimization of the MDSQ can be done at various levels. The reference [17] proposes a procedure that is based on a Lagrangian cost function accounting for central and side distortion to be minimized by alternatively optimizing the central partition for given decoding functions and the decoders for a given central partition. As this procedure may be computationally expensive, simpler schemes are also considered:

- The encoder uses a simple uniform scalar quantizer defining the central codebook. The side decoders select, among the multiple cells, the one closest to the mean of the probability distribution of the source and the dequantized value is provided by the entry in the central codebook for that cell.
- For any encoder, the side decoders may recover an estimate of the original value by calculating the centroid of the support defined by the cells corresponding to the received row or column index.

3.1 CS and multiple descriptions

3.1.1 CS-SPLIT and CS-MDSQ

Our goal is to investigate a multiple description framework for compressed sensing. Besides the traditional ways of implementing MDC, compressed sensing enables a new technique to generate the descriptions that is inherited by the "democracy" of the measurements. It is commonly meant, by democratic measurements, that each measurement carries a little piece of information about the original signal and no measurement is better than the others. This fact implies that any measurement can be lost and the quality of the reconstruction is expected to degrade in the same manner. In our method, that we refer to as CS-SPLIT, descriptions can be created by partitioning the set of measurements into subsets. For example, a balanced two-descriptions system will have two descriptions containing $\frac{m}{2}$ measurements each, out of the original m. The method can be summarized by the following description. The transmitter side has to perform the following operations:

- Acquire the measurements
- Split the measurements into two groups. Due to the democracy of the measurements there is no loss of generality if the splitting selects the first $\frac{m}{2}$ and the last $\frac{m}{2}$ to form the two groups.
- Transmit the two groups separately

The receiver behaves differently whether it receives both descriptions or just a single one. If both descriptions are available, the decoder juxtaposes the second part with the first one and

runs the BPDN procedure (or any other recovery algorithm) with the original sensing matrix. If only one description is received the steps are:

- Renormalize the problem by multiplying sensing matrix and measurements by a factor $\sqrt{2}$. This improves performance because it yields a sensing matrix with unit norm columns. Having unit norm columns is needed to satisfy the Restricted Isometry Property with a better constant δ. Also see the Introduction chapter for the discussion on how the properties of the sensing matrix matter to the success of compressed sensing.
- Run the recovery procedure using the correct submatrix.

Also, Figure 3.3 graphically explains the method.

Before going deeper into the analysis of this method and comparing it with other techniques for MDC, we already state that CS-SPLIT requires a certain amount of oversampling to work properly. In fact each description should be decodable by itself with high probability. However, the method is so simple that it requires almost no processing capabilities, whereas traditional MDC techniques may be computationally complex to apply. This is particularly interesting for environments in which a small oversampling comes at low cost, while processing power is expensive. We focus on the comparison between CS-SPLIT (Figure 3.4) and CS-MDSQ, that uses a MDSQ applied to the measurements (Figure 3.5). Both the methods are used to produce descriptions with the same weight in terms of total number of bits, to make the comparison fair.

We experimentally compare the relative reconstruction error of the methods as a function of the number of bits per description. A first scenario supposes that we are free to choose

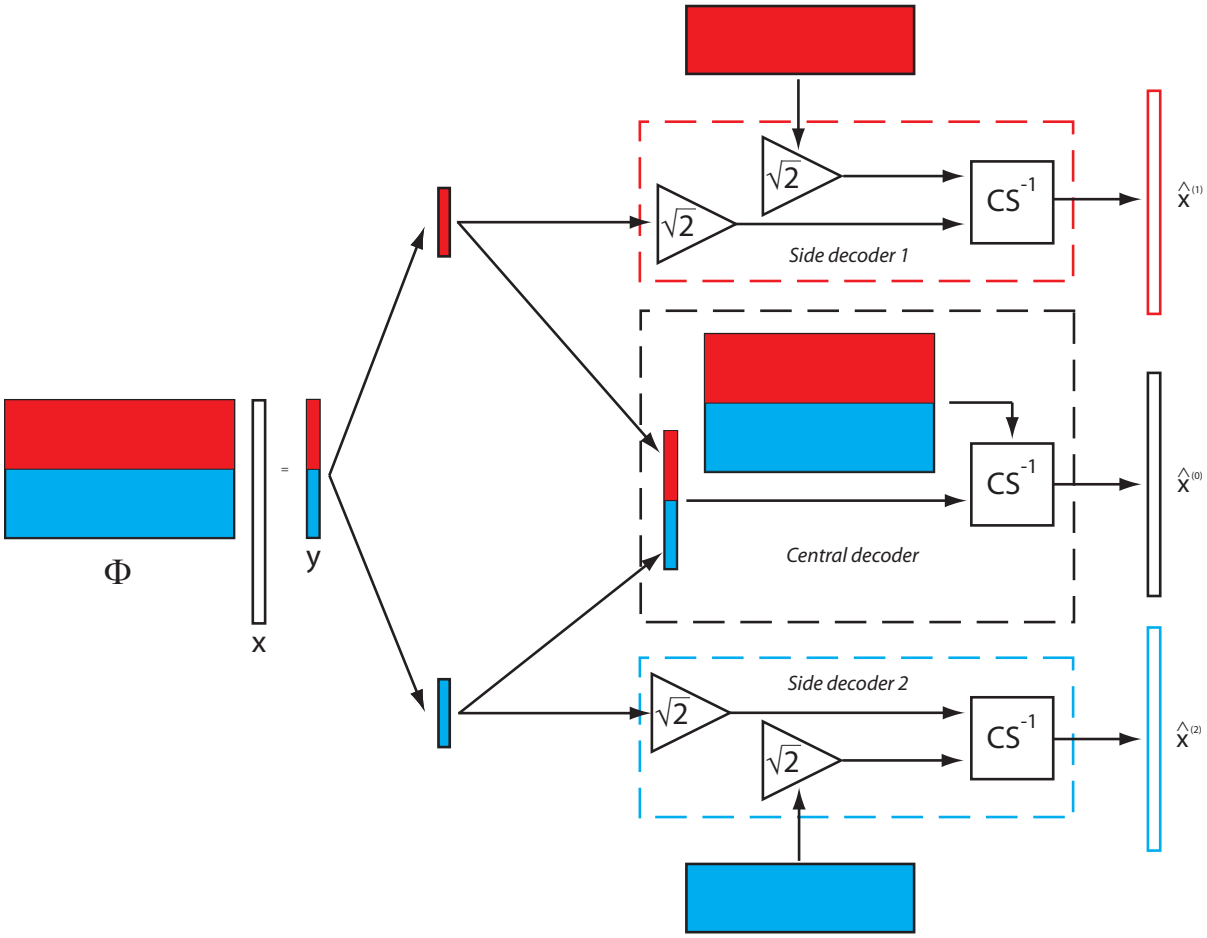

Figure 3.3. CS-SPLIT

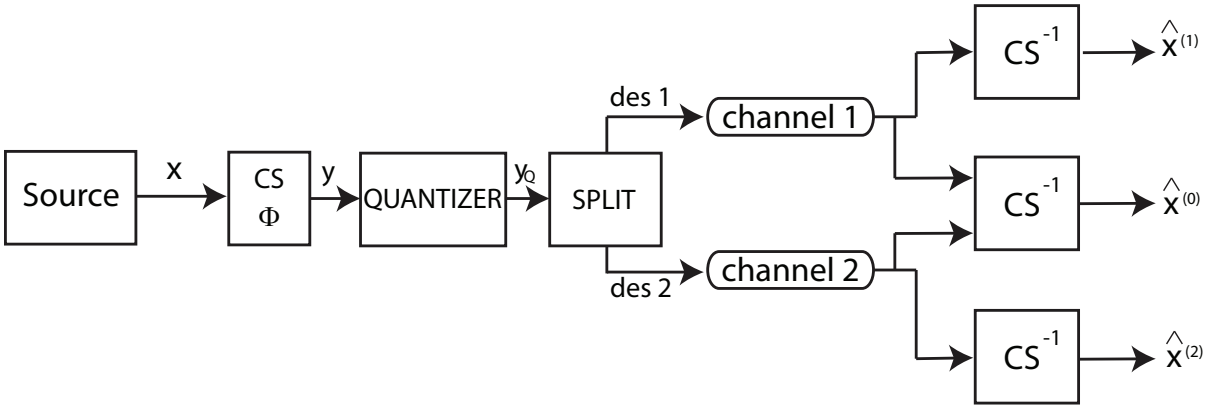

Figure 3.4. CS-SPLIT block diagram

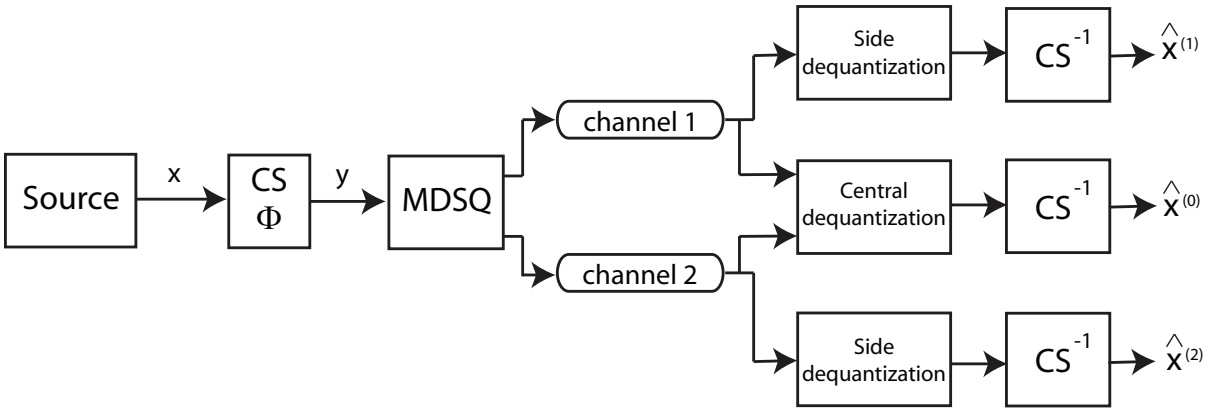

Figure 3.5. CS-MDSQ block diagram

how many CS measurements can be taken by the system. The best results for CS-SPLIT are obtained by oversampling the source by a certain factor, that is not necessarily two. There is a trade-off between the number of measurements and the resolution of the quantization process to achieve a target bitrate. In general, there is a quantization-limited regime in which there are enough measurements to potentially reach low error levels, but coarse quantization limits performance. On the other hand there is a measurement-limited regime in which measurements are too few to achieve good performance, no matter how many bits are spent on quantization. Figure 3.6 shows the relative error when a single description is received. It can be immediately noticed that CS-SPLIT failed with only 400 measurements, because 200 measurements per description are not enough to ensure reconstruction from a single description. The other curves show how a CS-SPLIT system that doubles the total number of measurements performs slightly worse than the MDSQ system, but reducing the oversampling factor to 1.5, i.e. taking only $m = 600$ measurements, performance significantly improve because it is possible to have a finer quantization. Figure 3.7 shows the corresponding central distortions.

An alternative scenario that can be considered, is when the system enforces a constraint on the total number of measurements. In this case we are not free to choose any oversampling ratio for CS-SPLIT. The comparison reported in Figure 3.8 and Figure 3.9 shows how CS-SPLIT typically performs better than MDSQ except in the region with few measurements, where side side-decoding is measurement-limited. However, CS-SPLIT always performs better if we consider the central distortion because the MDSQ machinery is not able to provide the same quantization resolution.

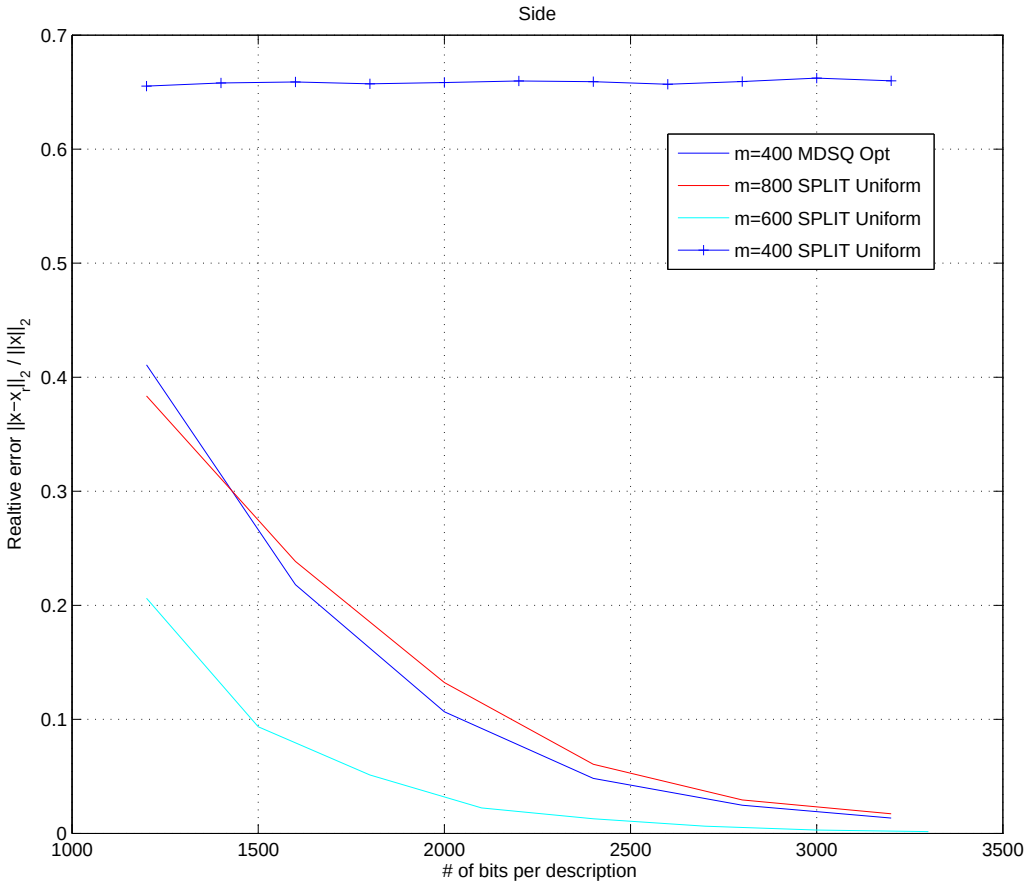

Figure 3.6. Side distortion. $n = 1024$, 100-sparse in identity domain. m is the total number of measurements, CS-SPLIT uses $\frac{m}{2}$ measurements per description

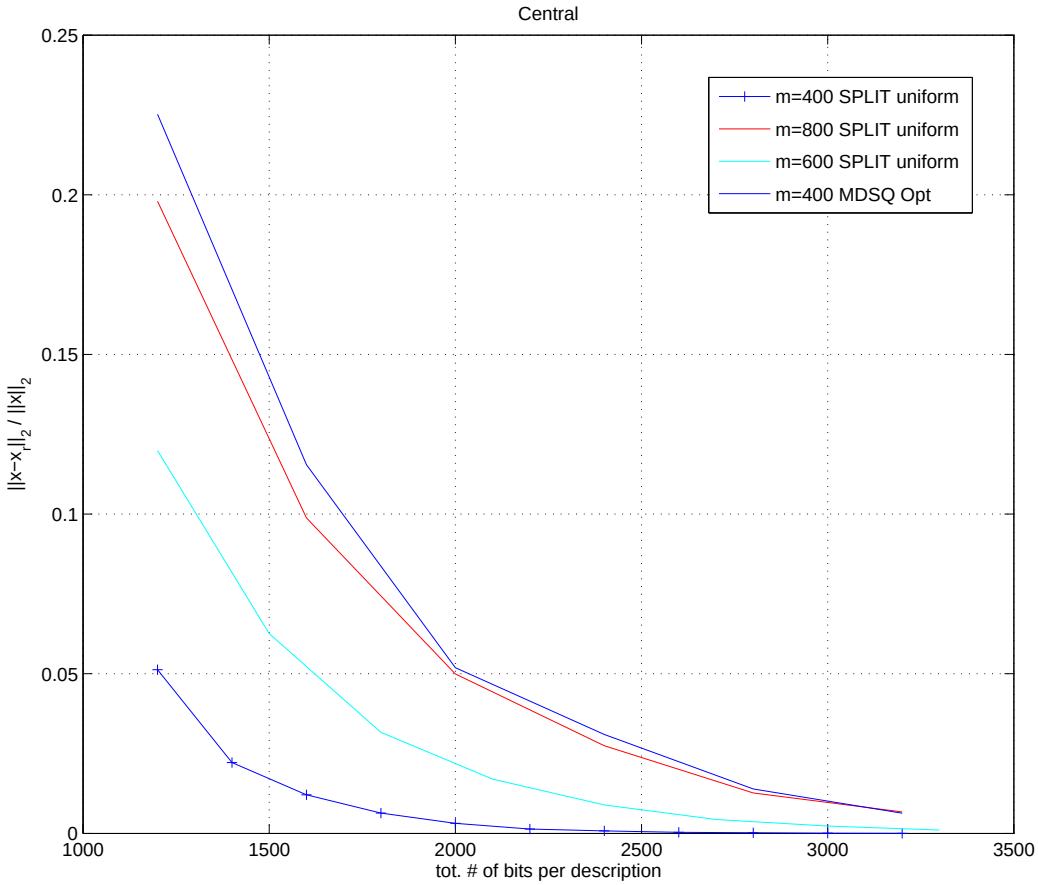

Figure 3.7. Central distortion. $n = 1024$, 100-sparse in identity domain. m is the total number of measurements, CS-SPLIT uses $\frac{m}{2}$ measurements per description

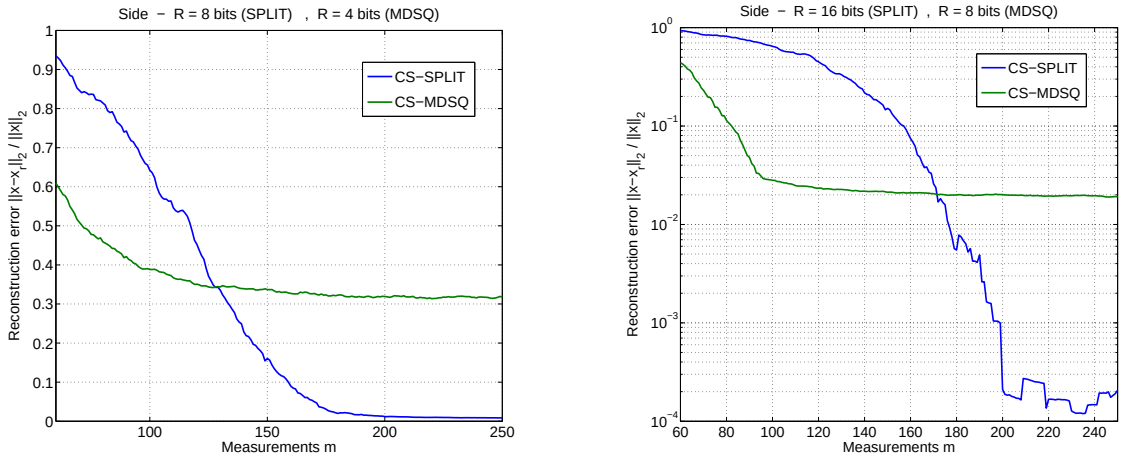

Figure 3.8. Side distortion for two different rates. Original signal has $n = 256$, $k = 20$. R is the number of bits used for each entry in the description

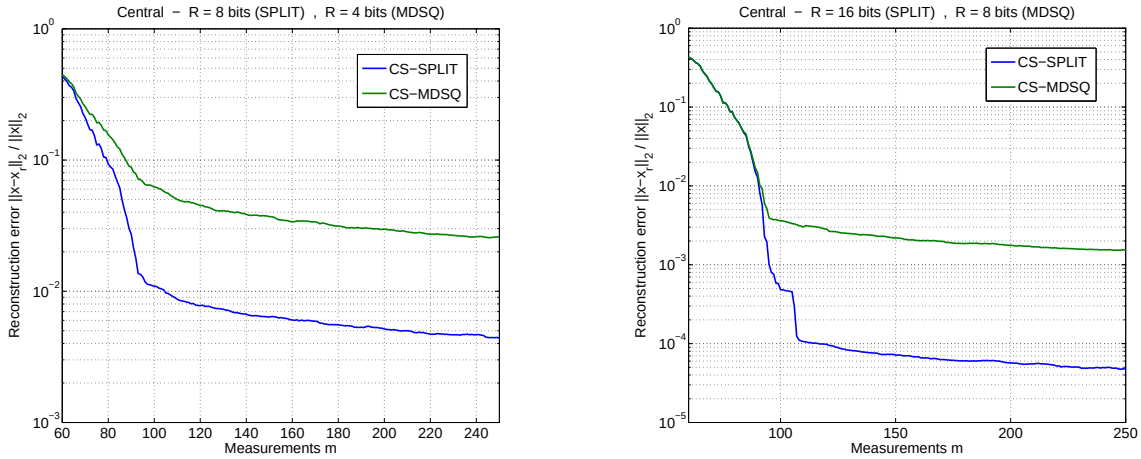

Figure 3.9. Central distortion for two different rates. Original signal has $n = 256$, $k = 20$. R is the number of bits used for each entry in the description

3.1.2 CS-SPLIT and MDSQ without CS

In this section we compare the reconstruction error achieved by a multiple-description system that uses MDSQ but no compressed sensing and another system that uses CS-SPLIT. When the input signal is sparse in some domain CS provides a very convenient sampling method, that, most importantly, is blind to the domain in which the signal is actually sparse. However, at least in the multiple description framework that we are considering, it may not always be the best solution just in terms of reconstruction error. We have run some experiments to determine which of the two systems provides a lower side and central distortion for the same number of bits per description. CS has a natural advantage when dealing with sparse signals and this shows up as a lower reconstruction error when the MDSQ system does not perform any entropy coding of the data. However, the results change when we insert an entropy coding block in the systems. The system without CS has a great benefit from entropy coding since the signal is mostly made of null entries. Instead, applying entropy coding to the CS measurements provides a limited gain due to their Gaussian distribution that makes their entropy relatively high. Figure 3.10 and Figure 3.11 show how the MDSQ with entropy coding always performs better than entropy-coded CS-SPLIT, at least for the signals that we considered. Anyway, it should be remarked that an entropy coder that is able to approach the entropy of the source adds significant complexity to system and, for the case without CS, the coefficients in the sparsifying basis must be known.

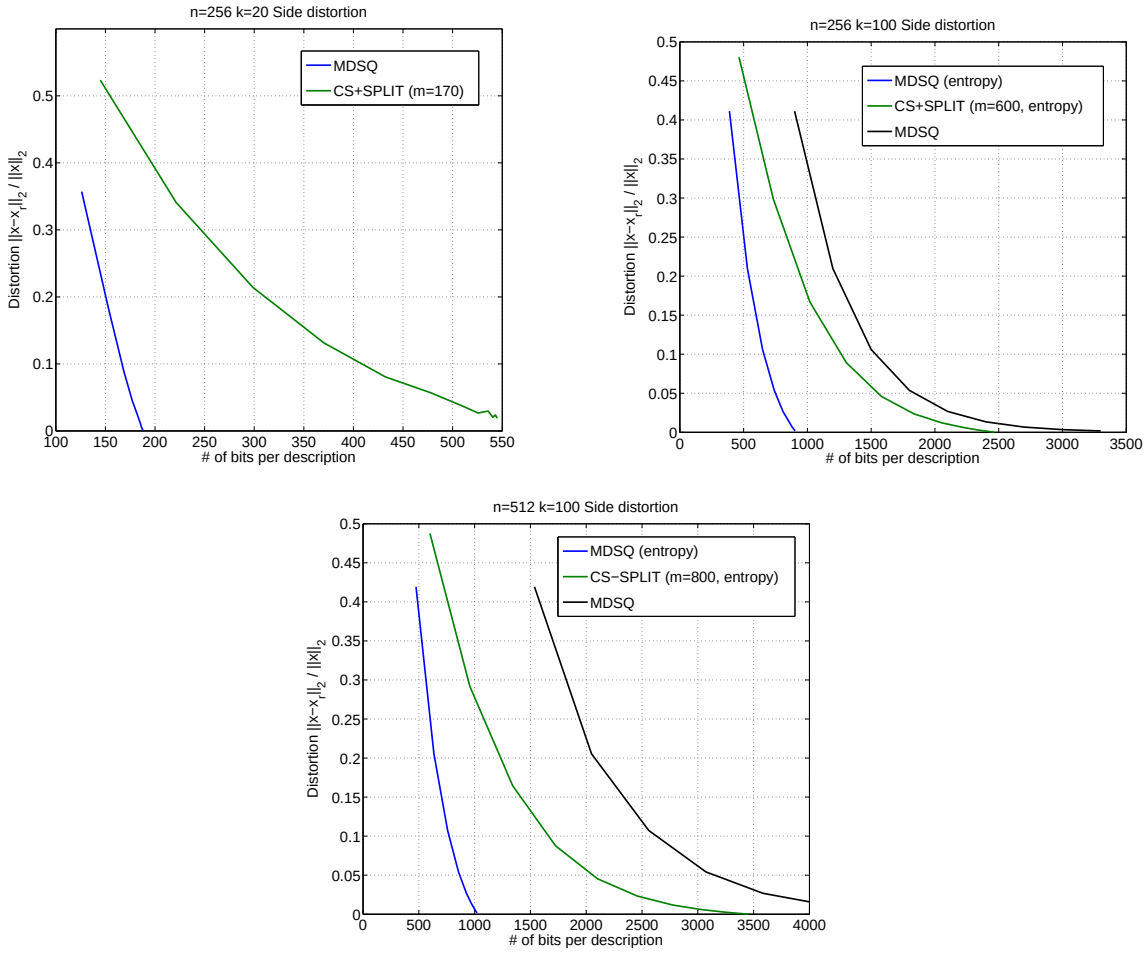

Figure 3.10. Side distortion. Original signal is k-sparse in the identity domain. For the CS-SPLIT method, m is the number of measurements.

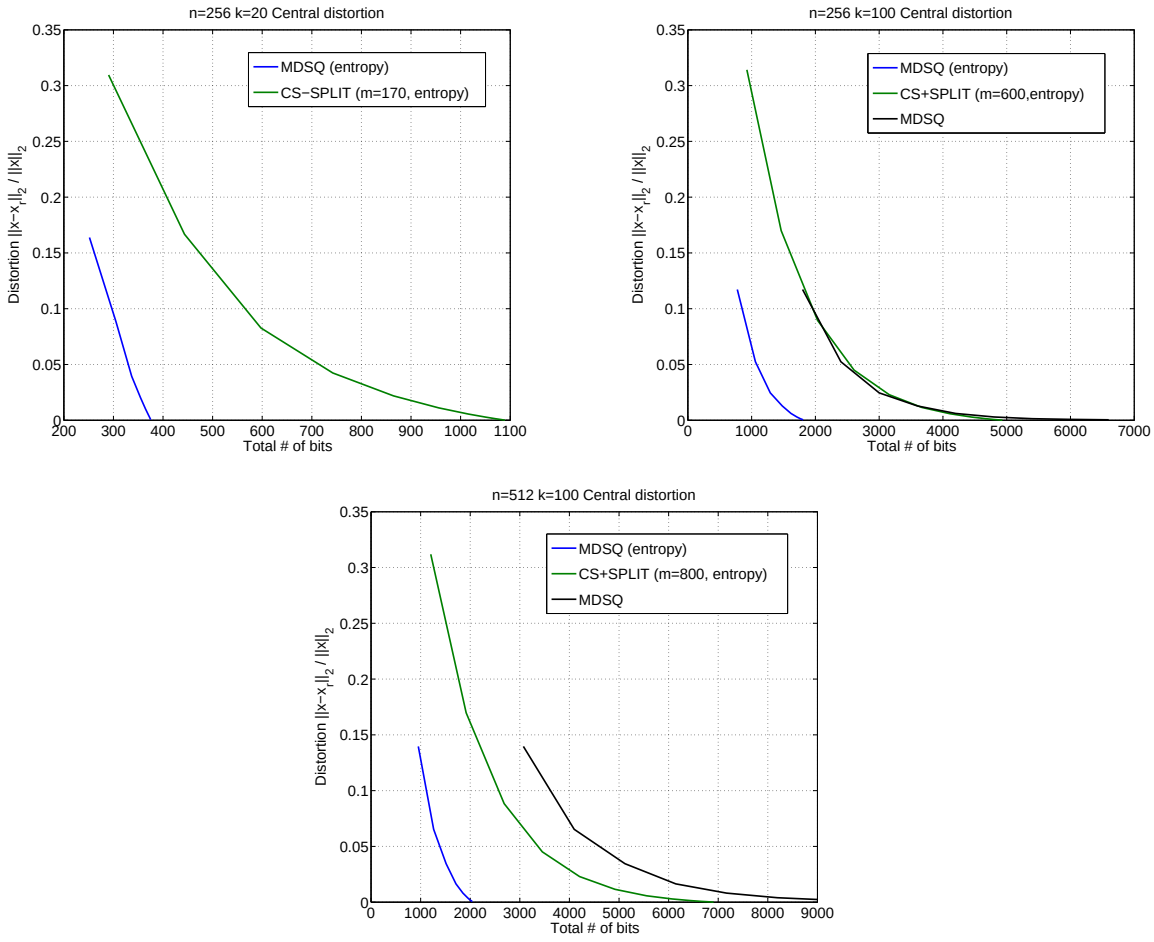

Figure 3.11. Central distortion. Original signal is k-sparse in the identity domain. For the CS-SPLIT method, m is the number of measurements.

3.2 A mathematical description of CS-SPLIT and CS-MDSQ

In the following we focus on deriving a mathematical framework aimed at characterizing the rate-distortion performance of the presented methods. We can anticipate that the analysis is able to produce a lower and an upper bound to the region where the rate-distortion curve sits with high probability, given the assumptions. The main issue that was encountered in the analysis is characterizing the performance of the reconstruction stage of compressed sensing. The current literature has showed some reconstruction guarantees for various algorithms, but, typically, they are not very sharp and the assumptions they require on the structure of the acquisition stage are somewhat uncommon in practice (e.g. the number of measurements and the dimensionality of the signal may be required to be very high compared to the actual sparsity). However, compressed sensing proved itself through applications to be much more reliable and the guarantees overly pessimistic.

3.2.1 RD performance of CS-SPLIT

Theorem 14. *Assume that the signal $\underline{x} \in \mathbb{R}^n$ is k-sparse and that the amplitude of the nonzero entries follows a distribution with zero mean and variance σ_x^2. Assume that measurements $\underline{y} = \Phi\underline{x}$, $\underline{y} \in \mathbb{R}^m$ are obtained using a sensing matrix Φ whose entries are i.i.d. Normal random variables with $\Phi_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ and such that $m > 60 \log n$, $k < \frac{1}{4}(\frac{1}{\mu}+1)$, where μ is the coherence of the matrix. Furthermore, assume that the BPDN algorithm is used for reconstruction. If the*

assumptions hold, then the distortion $D = \|\underline{x} - \underline{\hat{x}}\|_2^2$ *in the reconstructed signal as a function of rate* R *is bounded as follows, with high probability:*

$$2\frac{k^2}{m}\sigma_x^2 2^{-2R} \leq D_{side}(R) \leq \frac{4k\sigma_x^2 2^{-2R}}{1 - \sqrt{\frac{15\log n}{\frac{m}{2}}}(4k-1)} \tag{3.4}$$

$$\frac{k^2}{m}\sigma_x^2 2^{-2R} \leq D_{central}(R) \leq \frac{4k\sigma_x^2 2^{-2R}}{1 - \sqrt{\frac{15\log n}{m}}(4k-1)} \tag{3.5}$$

It can be noticed that the distribution of the measurements is Gaussian by the central limit theorem. Also, the measurements are supposed to be approximately independent. The mean value is

$$\mathbb{E}\left[\underline{y}\right] = \mathbb{E}[\Phi\underline{x}] = \mathbb{E}[\Phi]\,\mathbb{E}[\underline{x}] = 0 \tag{3.6}$$

The variance for the central case is

$$\begin{aligned}\sigma_{y,central}^2 &= \mathbb{E}\left[y_i^2\right] = \mathbb{E}\left[\left(\sum_{l=1}^{n}\Phi_{il}x_l\right)^2\right] = \mathbb{E}\left[\sum_{l=1}^{n}\Phi_{il}^2 x_l^2\right] + 2\,\mathbb{E}\left[\sum_{\substack{l,l'=1\\ l\neq l'}}^{n}\Phi_{il}\Phi_{il'}x_l x_{l'}\right] \\ &= \sum_{l=1}^{n}\mathbb{E}\left[\Phi_{il}^2\right]\mathbb{E}\left[x_l^2\right] = \frac{k}{m}\sigma_x^2\end{aligned} \tag{3.7}$$

The side case requires a renormalization by $\sqrt{2}$, so the variance is doubled

$$\sigma_{y,side}^2 = 2\frac{k}{m}\sigma_x^2 \tag{3.8}$$

The distortion-rate curve, i.e. the minimum distortion that can be reached for a given rate, of a memoryless Gaussian with the mean square error metric, is well know in literature. Hence we can derive the distortion on the single measurement, due to quantization, to be

$$D_{sm,central}(R) = \frac{k}{m}\sigma_x^2 2^{-2R} \tag{3.9}$$

$$D_{sm,side}(R) = 2\frac{k}{m}\sigma_x^2 2^{-2R} \tag{3.10}$$

D_{sm} is the mean square error $\mathbb{E}\left[(y_i - \bar{y}_i)^2\right]$. It is possible to compute the mean squared norm of the whole error vector by

$$D_{m,central}(R) = D_{m,side} = k\sigma_x^2 2^{-2R} \tag{3.11}$$

with $D_m = \mathbb{E}\left[\left\|\underline{y} - \underline{\bar{y}}\right\|_2^2\right]$. We call $\underline{\bar{y}}$ the distorted version of $\underline{y}$, e.g. due to quantization $\underline{\bar{y}} = \mathcal{Q}\left(\underline{y}\right)$.

The optimal reconstruction method, that can be used to establish an ultimate lower bound on the reconstruction error, is the oracle-based recovery. It is based on the assumption that an oracle tells us the sparsity support $\mathcal{S}$ of the signal, so that it does not have to be estimated from the measurements. Once the support is known, the best reconstruction is obtained by projecting the measurements onto the subspace spanned by the columns of the sensing matrix corresponding to the entries in the support. Mathematically, let's call Φ_S the sensing matrix restricted to the columns indexed by support S, i.e. with zeros in the other columns, then

$\underline{\hat{x}} = \left(\Phi_S^T \Phi_S\right)^{-1} \Phi_S^T \underline{\bar{y}}$. It is known from [18] that the oracle-assisted recovery produces an error bounded by

$$\frac{k}{1+\delta} D_{sm} \leq \|\underline{x} - \underline{\hat{x}}\|_2^2 \leq \frac{k}{1-\delta} D_{sm} \tag{3.12}$$

where δ is the RIP constant of matrix Φ. Computing the RIP constant of a matrix can be very complex. However, the best value that could be ideally achieved is $\delta = 0$, so we will consider this limit case in the analysis.

BPDN is known to reconstruct a sparse signal from noisy measurements with an error that is proportional to the noise power. The proportionality constant depends on the properties of the sensing matrix, in terms of RIP constant or coherence parameter. Davenport *et al.* report in [2] a reconstruction guarantee that explicitly expresses the constant as function of the coherence of the sensing matrix. Calling $\varepsilon^2 = \mathbb{E}\left[\|e\|_2^2\right]$ the power of noise, and assuming that the constraint in the BPDN optimization problem uses exactly ε, and that $k < \frac{1}{4}(\frac{1}{\mu} + 1)$, the reconstruction error is bounded by

$$\|\underline{x} - \underline{\hat{x}}\|_2 \leq \frac{2\varepsilon}{\sqrt{1 - \mu(4k-1)}} \tag{3.13}$$

where μ is the coherence of the sensing matrix.

We limit ourselves to the case in which the sensing matrix in made of Gaussian entries, which is one of the most common cases in practical applications and among the most studied from a theoretical standpoint. In fact Bajwa *et al.* have derived in [19] worst-case and average-case expressions for the coherence of a Gaussian matrix that hold with high probability. The

following worst-case bound is of particular interest to our analysis. Assume that $\Phi_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ and that $m > 60 \log n$, then

$$\mu \leq \sqrt{\frac{15 \log n}{m}} \tag{3.14}$$

with probability exceeding $1 - \frac{2}{m}$. A proof of this result is given in [19].

Now we have all the building blocks to prove (3.4) and (3.5). The lower bound is provided by the oracle inequality with $\delta = 0$, while the upper bound is given by the performance guarantee on BPDN. Concerning the lower bound it is immediate to derive

$$D_{oracle,side} = 2\frac{k^2}{m}\sigma_x^2 2^{-2R} \tag{3.15}$$

$$D_{oracle,central} = \frac{k^2}{m}\sigma_x^2 2^{-2R} \tag{3.16}$$

For the upper bound, it is sufficient to notice that equation (3.11) can be used as the expression of the noise norm ε in (3.26), and substituting the worst case coherence reported in (3.27) we can obtain

$$D_{upper,side} = \frac{4k\sigma_x^2 2^{-2R}}{1 - \sqrt{\frac{15 \log n}{\frac{m}{2}}}(4k-1)} \tag{3.17}$$

$$D_{upper,central} = \frac{4k\sigma_x^2 2^{-2R}}{1 - \sqrt{\frac{15 \log n}{m}}(4k-1)} \tag{3.18}$$

Figure 3.12 reports graphically the lower and upper bounds for the side and central cases.

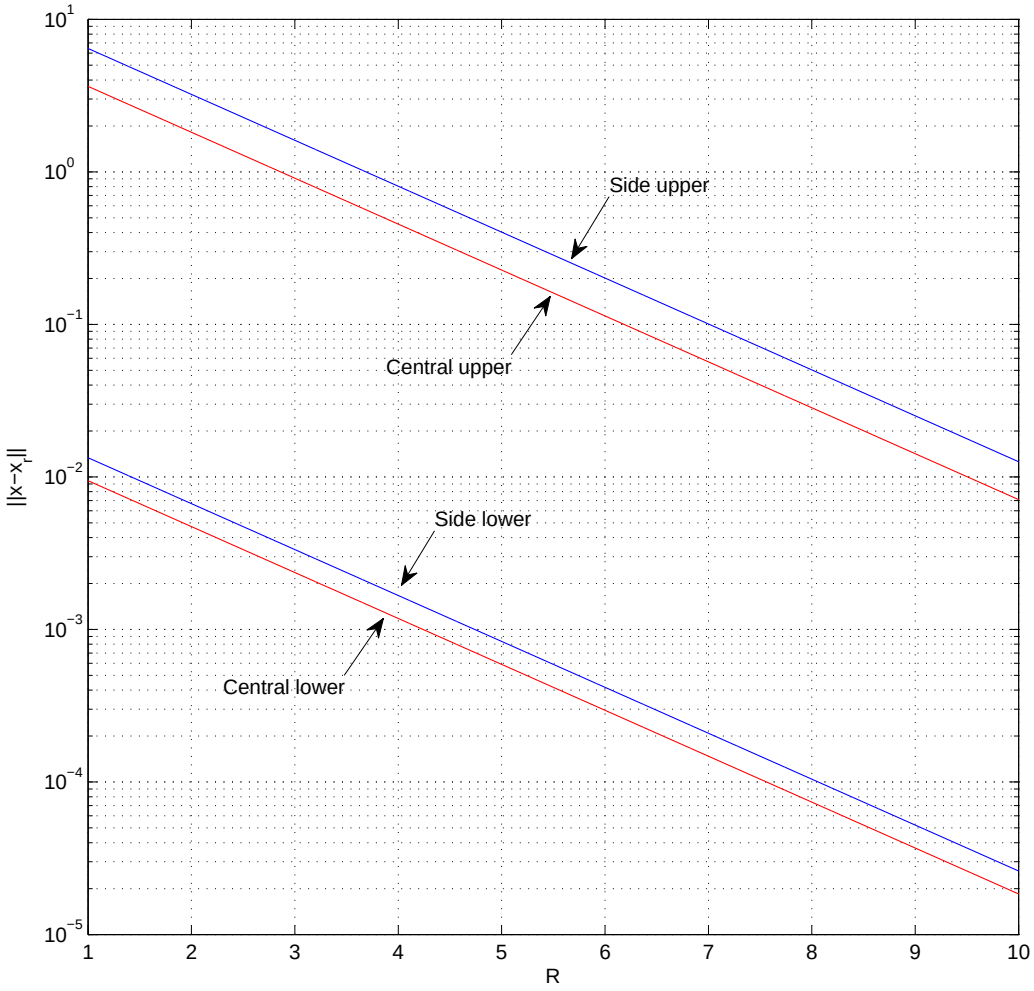

Figure 3.12. Lower and upper bound for the rate-distortion curve

Figure 3.13 reports an experimental check on the oracle lower bound. The theoretical curve is the one reported above, obtained for $\delta = 0$. The experimental curve is obtained from a large number of simulations applying a uniform scalar quantization to the measurements and then reconstruction knows the support perfectly. The figures report the norm of reconstruction error. A penalty of $\sqrt{\frac{\pi e}{6}}$ ($20 \log_{10} \sqrt{\frac{\pi e}{6}} = 1.34$dB) [20] is accounted because of the use of the uniform scalar quantizer. The simulation seems to support the theoretical result. The slope of the two curves coincides and after correcting for the quantizer penalty, the remaining offset is explained by the fact that the RIP constant is in fact greater than zero. Also note that in the side case, a lower number of measurements makes the RIP constant somewhat worse, hence the greater offset.

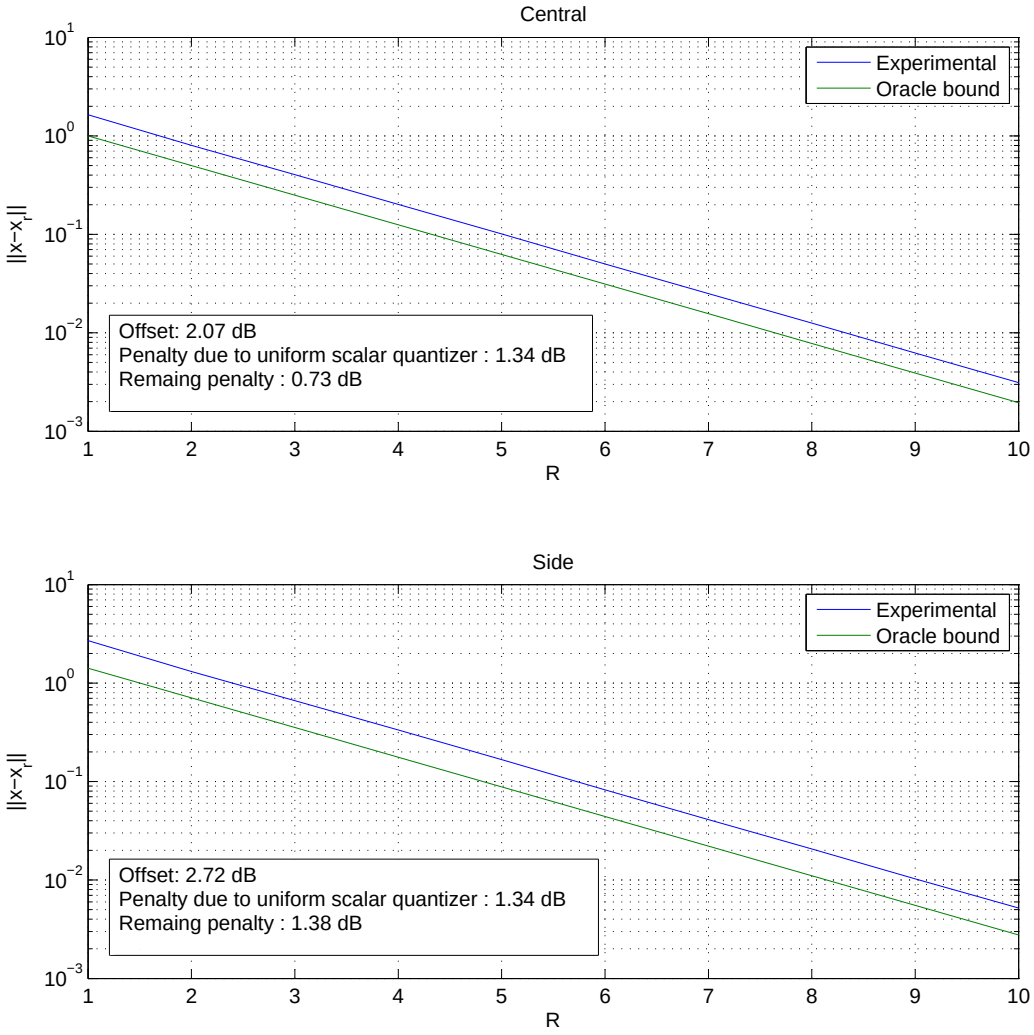

Figure 3.13. Experimental vs. theoretical lower bound. The theoretical curve is a limit case in which the RIP constant tends to zero. Simulations are performed with a Gaussian sensing matrix.

3.2.2 RD performance of CS-MDSQ

Theorem 15. *Assume that the signal $\underline{x} \in \mathbb{R}^n$ is k-sparse and that the amplitude of the nonzero entries follows a distribution with zero mean and variance σ_x^2. Assume that measurements $\underline{y} = \Phi\underline{x}$, $\underline{y} \in \mathbb{R}^m$ are obtained using a sensing matrix Φ whose entries are i.i.d. Normal random variables with $\Phi_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ and such that $m > 60 \log n$, $k < \frac{1}{4}(\frac{1}{\mu}+1)$, where μ is the coherence of the matrix. Furthermore, assume that the BPDN algorithm is used for reconstruction. If the assumptions hold, then the distortion $D = \|\underline{x} - \hat{\underline{x}}\|_2^2$ in the reconstructed signal as a function of rate R is bounded as follows, with high probability:*

$$\frac{\sigma_x^2}{m} k^2 2^{-2R} \leq D_{side}(R) \leq \frac{4\sigma_x^2 k 2^{-2R}}{1 - \sqrt{\frac{15 \log n}{m}}\,(4k-1)} \tag{3.19}$$

$$\frac{\sigma_x^2}{m} k^2 2^{-4R} \gamma_D \leq D_{central}(R) \leq \frac{4\sigma_x^2 k 2^{-4R} \gamma_D}{1 - \sqrt{\frac{15 \log n}{m}}\,(4k-1)} \tag{3.20}$$

The rate-distortion analysis of the scheme that applies a multiple-description scalar quantizer to the measurements considers the Ozarow bound [16] as starting point. We can state the Ozarow bound for a balanced MDC case in which the rate is the same for both descriptions $R_1 = R_2 = R$ and the side distortion is also the same $D_1 = D_2 = D_{\text{sm,side}}$ (we call D_{sm} the mean squared error on the single measurement):

$$D_{\text{sm,side}} \geq \frac{\sigma_x^2}{m} k 2^{-2R} \tag{3.21}$$

$$D_{\text{sm,central}} \geq \frac{\sigma_x^2}{m} k 2^{-4R} \gamma_D \tag{3.22}$$

with

$$\gamma_D = \frac{1}{1 - \left(\left(1 - \frac{D_{\text{sm,side}}}{\frac{\sigma_x^2}{m} k} \right) - \sqrt{\frac{D_{\text{sm,side}}^2}{\frac{\sigma_x^4}{m^2} k^2} - 2^{-4R}} \right)^2}$$

Following the same reasoning used in the analysis of the CS-SPLIT method, the lower bound on the performance after reconstruction is obtained for an oracle-assisted reconstruction. An oracle tells us the support $\mathcal{S}$ of the signal and recovery is performed as $\underline{\hat{x}} = \left(\Phi_S^T \Phi_S\right)^{-1} \Phi_S^T \underline{\bar{y}}$, being Φ_S the sensing matrix restricted to the columns indexed by support $\mathcal{S}$, i.e. with zeros in the other columns. It is known from [18] that the oracle-assisted recovery produces an error bounded by

$$\frac{k}{1+\delta} D_{sm} \leq \|\underline{x} - \underline{\hat{x}}\|_2^2 \leq \frac{k}{1-\delta} D_{sm} \tag{3.23}$$

where δ is the RIP constant of matrix Φ. Computing the RIP constant of a matrix can be very complex. However, the best value that could be ideally achieved is $\delta = 0$, so we will consider this limit case in the analysis. It immediately follows that the lower bound for the side and central cases are:

$$\|\underline{x} - \underline{\hat{x}}\|_{2,\text{side}}^2 \geq \frac{\sigma_x^2}{m} k^2 2^{-2R} \tag{3.24}$$

$$\|\underline{x} - \underline{\hat{x}}\|_{2,\text{central}}^2 \geq \frac{\sigma_x^2}{m} k^2 2^{-4R} \gamma_D \tag{3.25}$$

Notice that the performance of the central case is also affected by the performance of the side case through the γ_D parameter.

The upper bound is established for the BPDN reconstruction algorithm. BPDN is known to reconstruct a sparse signal from noisy measurements with an error that is proportional to the

noise power. The proportionality constant depends on the properties of the sensing matrix, in terms of RIP constant or coherence parameter. Davenport *et al.* report in [2] a reconstruction guarantee that explicitly expresses the constant as function of the coherence of the sensing matrix. Calling $\varepsilon^2 = \mathbb{E}\left[\|e\|_2^2\right]$ the power of noise, and assuming that the constraint in the BPDN optimization problem uses exactly ε, and that $k < \frac{1}{4}(\frac{1}{\mu} + 1)$, the reconstruction error is bounded by

$$\|\underline{x} - \underline{\hat{x}}\|_2 \leq \frac{2\varepsilon}{\sqrt{1 - \mu(4k-1)}} \tag{3.26}$$

where μ is the coherence of the sensing matrix.

We limit ourselves to the case in which the sensing matrix in made of Gaussian entries, which is one of the most common cases in practical applications and among the most studied from a theoretical standpoint. In fact Bajwa *et al.* have derived in [19] worst-case and average-case expressions for the coherence of a Gaussian matrix that hold with high probability. The following worst-case bound is of particular interest to our analysis. Assume that $\Phi_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ and that $m > 60 \log n$, then

$$\mu \leq \sqrt{\frac{15 \log n}{m}} \tag{3.27}$$

with probability exceeding $1 - \frac{2}{m}$. A proof of this result is given in [19]. The final bound on the reconstruction error can be obtained by considering that the noise power is in fact the distortion on the measurement vector $\varepsilon^2 = mD_{sm}$. Hence:

$$\|\underline{x} - \underline{\hat{x}}\|_{2,\text{side}}^2 \leq \frac{4\sigma_x^2 k 2^{-2R}}{1 - \sqrt{\frac{15 \log n}{m}}\,(4k-1)} \tag{3.28}$$

$$\|\underline{x} - \hat{\underline{x}}\|^2_{2,\text{central}} \leq \frac{4\sigma_x^2 k 2^{-4R}\gamma_D}{1 - \sqrt{\frac{15\log n}{m}}\,(4k-1)} \tag{3.29}$$

We are now going to check if the results predicted by the theory are consistent with the outcomes of some simulations. We will focus on the lower bound as the upper bound is somewhat cumbersome to be checked due to the assumptions needed to make the guarantees hold. Due to the dependency of the central distortion on the value of the side distortion, the theoretical curve for central distortion is actually obtained as a hybrid of theory and simulation. Our implementation of the MDSQ fixes a certain side distortion, that is not the minimal one, either because it is difficult to achieve and because minimal side distortion would imply maximum central distortion. This value of side distortion on the single measurement, before reconstruction, is evaluated experimentally, compared against the minimum achievable, and then plugged in (3.25) to determine what is the minimum central distortion that can be achieved for that side distortion. Figure 3.14 graphically shows the method. The first experiment is aimed at checking the reconstruction distortion achieved by CS-MDSQ with a practical multiple description scalar quantizer, as function of the rate, i.e. the binary logarithm of the number of rows (or columns) of the index assignment matrix. This would be the number of bits used to represent each measurement in the descriptions. Figure 3.15 reports the results. The curves labeled as '*Theory*' implement equations (3.24) and (3.25) by the procedure previously explained. It can be seen that the experimental outcome follows the theoretical curve with the same slope and a penalty that may be due by several factors such as the implementation of the MDSQ, that is not able to work at the rate-distortion limit, and the oracle-assisted recovery that is penalized

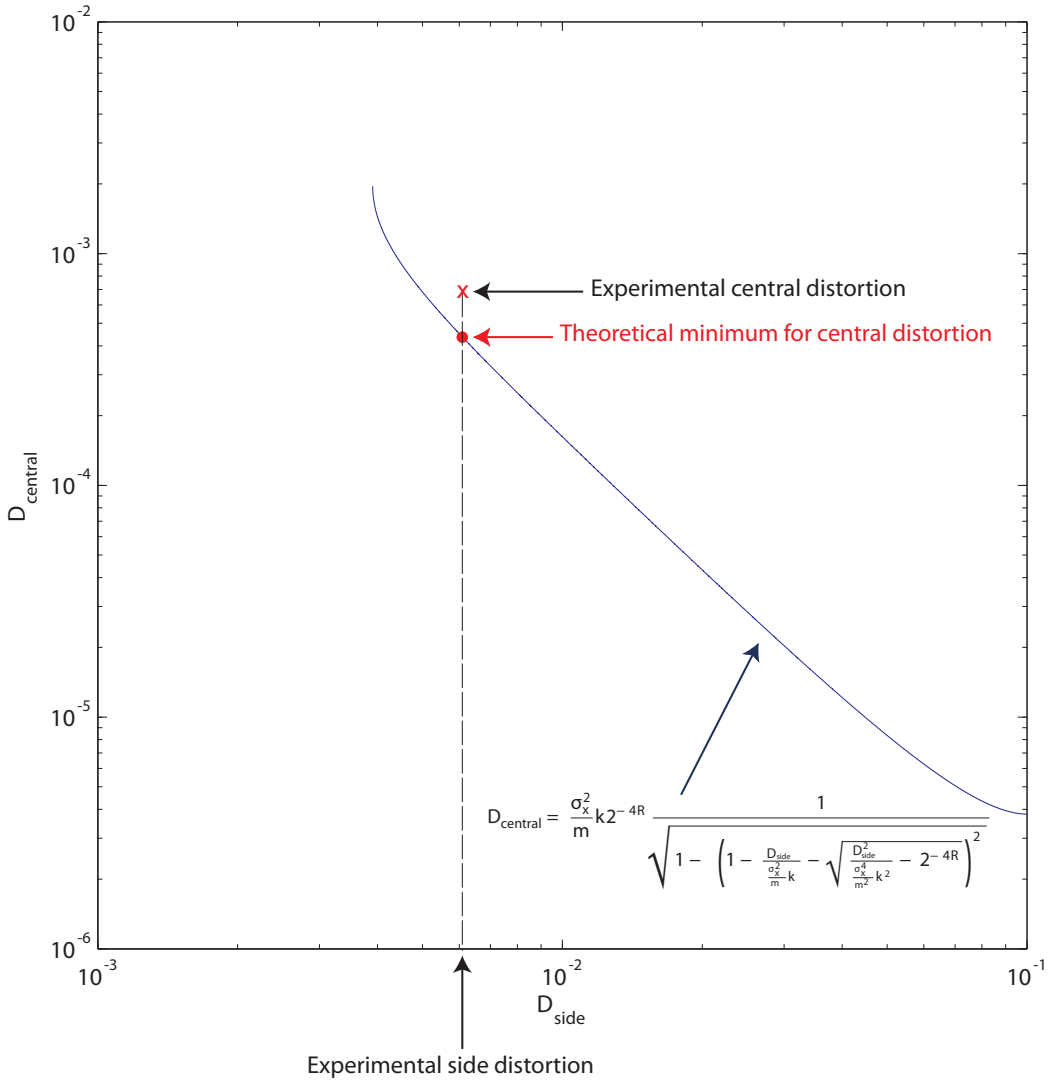

Figure 3.14. Tradeoff curve of an example MDSQ applied to the measurements. In this figure only D_{side} and D_{central} are intended as the mean square error on the measurements before any reconstruction.

by a RIP constant that is actually greater than zero. Figure 3.16 shows the dependency of the reconstruction error norm on the number of measurements. We can notice a weak reduction of the error as the number of measurements is increased.

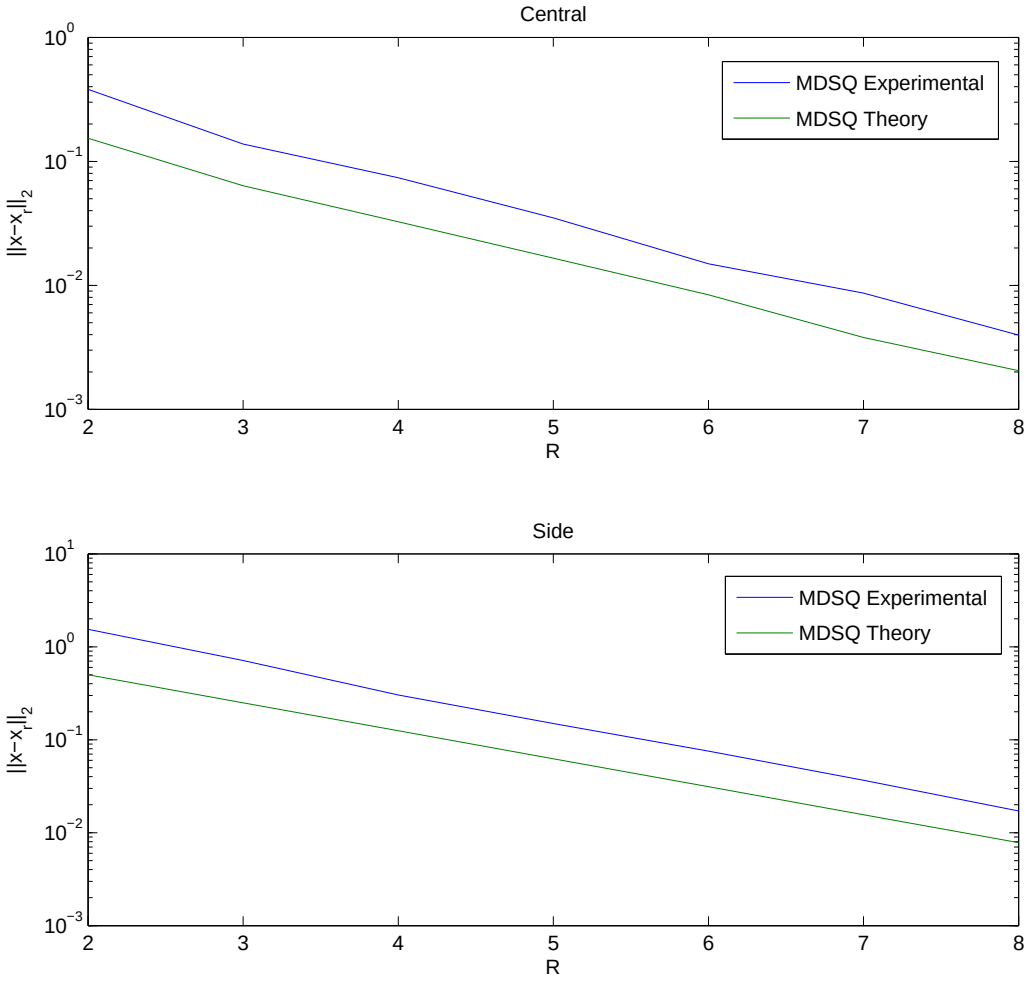

Figure 3.15. Experimental vs. theoretical lower bounds. The theoretical curve for the side case implements the minimum value of (3.24). The theoretical curve for the central case uses (3.25) using the experimental side MSE before reconstruction as $D_{\text{sm,side}}$. Simulations are performed with $k = 20$, $n = 256$, $\sigma_x = 1$ and a Gaussian sensing matrix with $m = 100$.

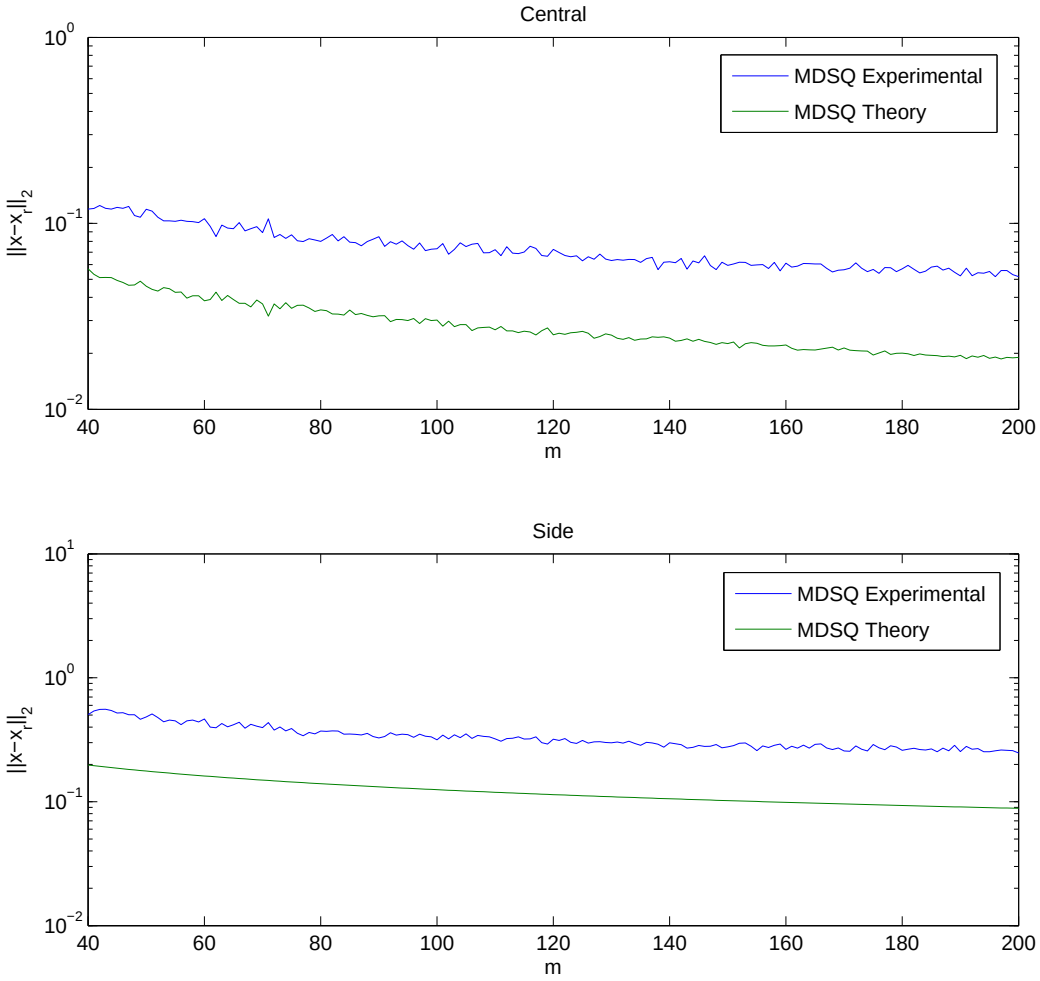

Figure 3.16. Experimental vs. theoretical lower bound. The theoretical curve for the side case implements the minimum value of (3.24). The theoretical curve for the central case uses (3.25) using the experimental side MSE before reconstruction as $D_{\text{sm,side}}$. Simulations are performed with $k = 20$, $n = 256$, $\sigma_x = 1$, $R = 5$ bps and a Gaussian sensing matrix.

3.2.3 Theoretical comparison of CS-SPLIT and CS-MDSQ

The results previously outlined show that it is difficult to compare the performance of CS-SPLIT and CS-MDSQ, using only a theoretical analysis. The bounds are, in fact, too loose and subject to a number of limitations concerning the signal parameters, especially due to the current results concerning the characterization of recovery procedures of compressed sensing. However, it is possible to give some intuitions concerning the lower bounds, obtained by means of oracle-assisted recovery. We stress that a fair comparison must not use the same parameters (e.g. same number of measurements and same rate) for both systems because, otherwise, CS-MDSQ would be transmitting twice the data with respect to CS-SPLIT. As already explained in the section considering experimental results, CS-MDSQ uses m measurement, each represented with R bits in each description, so $2mR$ bits overall. Instead, CS-SPLIT uses $\frac{m}{2}$ measurements, quantized with R bits per description, so mR bits overall. Hence, a fair comparison would use, for instance, twice as many measurements for CS-SPLIT and the same rate, or twice the rate and the same number of measurements.

The following figures report the lower bounds for CS-SPLIT and CS-MDSQ in the central distortion vs. side distortion plane. Those curves allow to define a region of the plane that is the set of all achievable pairs $(D_{\text{side}}, D_{\text{central}})$. It is interesting to notice that CS-SPLIT requires no tradeoff between central and side distortion, while, on the contrary, the use of a MDSQ requires to have worse performance in the side decoders if we want to improve central decoding. It is also interesting to notice from Figure 3.17 that, for those parameters, CS-SPLIT is, in principle, able to achieve more advantageous $(D_{\text{side}}, D_{\text{central}})$.

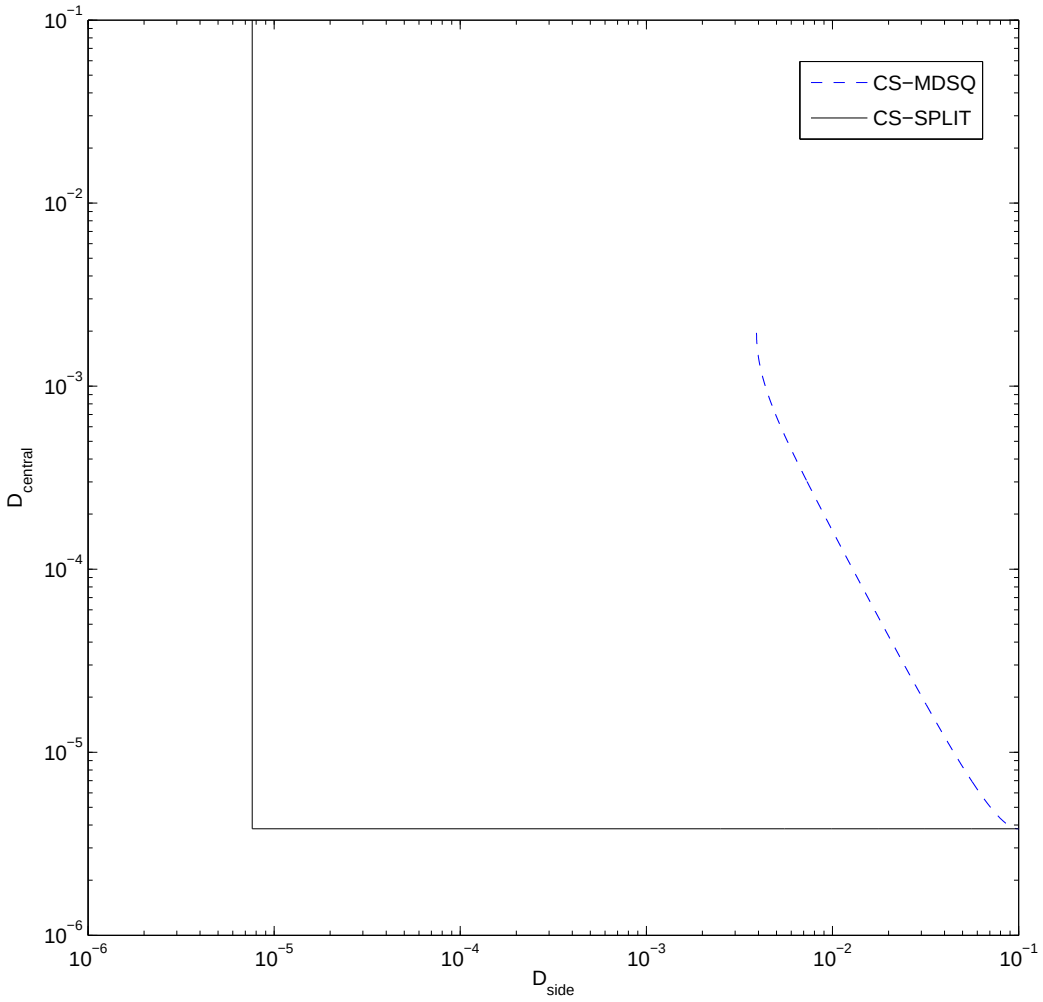

Figure 3.17. Lower bound tradeoff curves. $k = 20$, $n = 256$, $\sigma_x = 1$, $R = 5$ bps for CS-MDSQ and $R = 10$ bps for CS-SPLIT, Gaussian sensing matrix with $m = 100$.

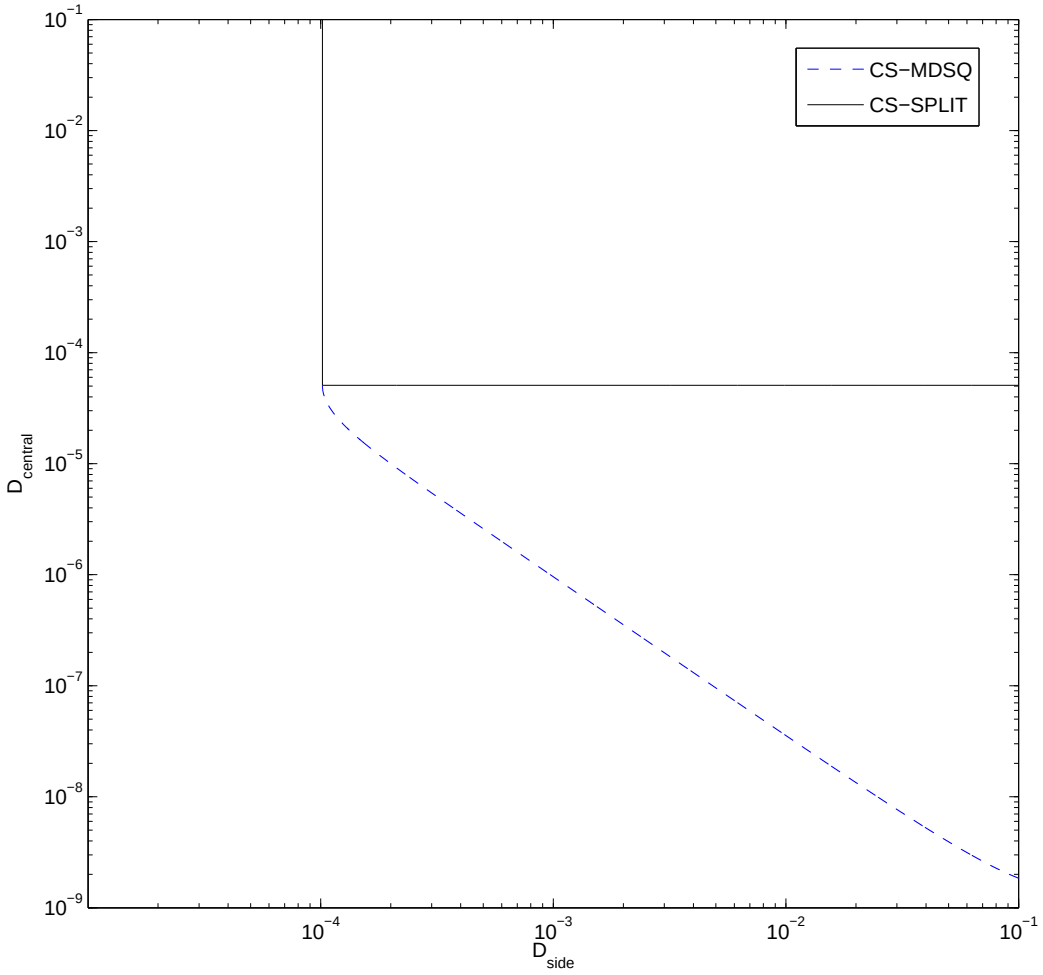

Figure 3.18. Lower bound tradeoff curves. $k = 20$, $n = 256$, $\sigma_x = 1$, $R = 8$ bps, Gaussian sensing matrix with $m = 60$ for CS-MDSQ and $m = 120$ for CS-SPLIT.

CHAPTER 4

SENSOR NETWORKS

Sensor networks may be regarded as an evolution of the traditional paradigm of sensing quantities. A sensor network is endowed with a possibly huge number of sensors that cooperate to carry out some measurements. Cooperation is key to the success of the network because it makes it fault-resilient as well as efficient. Applications for sensor networks have been proposed in a various range of fields, among which medical, military, environment monitoring, home automation and several others. Such flexibility is due to the characteristics of sensor networks. Sensors are ideally tiny objects, equipped with wireless connectivity and some processing power, that can be deployed on the site of interest without engineering their positions but rather in a random way. The network must be able to self-assemble and rewire its structure dynamically without any external intervention, as the conditions may change over time because of failures (e.g. a sensor runs out of power) or because sensors have moved. One crucial constraint that must be faced when dealing with sensor networks is the limited available power. Sensors are battery-powered and may or may not be able to recharge depending on the availability of solar cells. Energy efficiency is sought at all levels of design, from the physical layer in which binary modulations are preferred to M-ary constellations, to the link and network layers that use protocols specifically aimed at power conservation. The on-board processing power allows to perform simple preprocessing operations on the collected data, but data are routed through the network to a sink and then to the final user that has all the processing power needed to

decode and process the data. Hence, to maintain the network cost-efficiency and meet the power constraints, sensors must be kept simple and all the complex tasks must be performed at the receiver.

4.1 Distributed Compressed Sensing and Joint Sparsity Models

The framework of sensor networks significantly differs from what we analysed in the previous chapters because it involved a single signal. When dealing with networks, we have to consider an ensemble of signals, that may be generated by multiple sensors monitoring the spatial distribution of a variable (e.g. temperature). It is often the case that the observations made by the sensors are both *inter-* and *intra-correlated*. By *intra-correlation* we mean that a certain amount of redundancy within the signal itself is present and that there exists some basis Ψ that allows a compact (sparse) representation of the signal. The term *inter-correlation* is instead referring to the amount of redundancy among different signals in the ensemble. How much inter-correlation can be found depends on the specific application but it is reasonable to think that whenever we are sensing the distribution of a variable that varies smoothly over space, then the measurements collected by the sensors will be very much similar to each other. Furthermore, sensor networks are studied to be deployed in a dense fashion in the region of interest, so that it is reasonable to assume that sensors that are close to each other collect similar measurements. Sensor networks also have some strict requirements that make them a challenging topic to investigate. For example, sensors are typically small-size, battery-powered devices, so there are strong limitations to the complexity of tasks that can be performed by them. Hence, a coding scheme that has a simple signal acquisition procedure and that moves all the complexity to the receiver is desirable

for this kind of application. Compressed sensing clearly has this important feature, hence the interest in investigating applications of compressed sensing to sensor networks.

The theory of Distributed Compressed Sensing, that has a pivotal reference in [21], focuses on applying compressed sensing to a distributed scenario in which we want to leverage both inter- and intra-correlation to achieve high coding efficiency while keeping low complexity on the sensors side. It is interesting to notice that the use of Distributed Compressed Sensing achieves a reduction in the number of measurements that have to be acquired and transmitted by each sensor, thanks to the presence of many sensors collecting correlated measurements, but, in principle, it requires no cooperation among the sensors. This is important because it limits dramatically the complexity of the system if no cooperative scheme is needed to achieve the desired performance.

Baron *et al.* introduce in [21] the notion of *joint sparsity* and provide three example models of joint sparsity that have been referenced by the literature (e.g. [22] [23] [24]).

JSM-1 Sparse common component + sparse innovations. In this model each signal in the ensemble is made by a sparse component $\underline{x}_c$ that is common to all the J signals in the ensemble and by a innovation component $\underline{x}_{I,i}$ that is peculiar of sensor i.

$$\underline{x}_i = \underline{x}_c + \underline{x}_{I,i} \quad i = 1, ..., J \tag{4.1}$$

This model well represents measurements of quantities that are affected by some global factors common to all measurement nodes, thus contributing to the common component,

while the innovation signals may arise from local phenomena affecting each node in a different manner. Smoother variations in time and space tend to make the signals sparser (more intra-correlated) and each less innovative, respectively.

JSM-2 **Common sparse support** In this model all the signals share the same sparsity support but the amplitudes may be different. The JSM-2 model is appropriate for acoustic sensor arrays or MIMO communication devices where each node receives replicas of the same signal, with some modifications due to different phase shifts or attenuations. Another example is medical imaging, where recovery from a JSM-2 ensemble is often referred to as the Multiple Measurement Vector (MMV) problem.

JSM-3 **Nonsparse common component + sparse innovations** This model is similar to JSM-1 in the sense that there is a component that is common to all signals but it is not sparse. The innovation component of each signal is sparse. This model well suits the case of video sequences where each frame may not be sparse by itself, but it may have minor modifications with respect to the previous frame, hence it has sparse innovations.

4.2 Algorithms for joint reconstruction

In the following we are going to present a few algorithms that explicitly exploit models of joint sparsity to improve reconstruction performance or to decrease the number of measurements that is needed to achieve a desired quality in the reconstructed signal. It is interesting to notice that no cooperation among sensors is required by these algorithms, which keeps complexity to a minimum and does not require exchange of data inside the network. Finally, most of the algorithms require the use of side information. By side information we mean that the signal acquired by one of the sensors is fully known by the receiver. In practice, this just means that one sensor acquires more measurements than the others and quantizes them in a fine manner, so that reconstruction can be performed with very low error. If the number of sensors in the network is large, the overhead due to side information tends to be small or negligible and, in fact, we already discussed how sensor networks typically employ several sensor nodes. Since having the side information at the receiver is critical, it may be interesting to devise protocols that take care of replacing the node providing the side information in case of failure, but this is beyond the scope of this work. We introduce joint reconstruction algorithms with a short literature review. In particular, the *Intersection* and *Sort* algorithms have been proposed in [23], and the *Texas Hold 'Em* in [22]. Let's clarify some notation that is used throughout the explanation of the algorithms.

$\underline{x}_j$: j-th signal in the domain in which sensing takes place

$\underline{\theta}_j$: j-th signal in the domain in which it is sparse

Ψ : sparsifying basis $\underline{x}_j = \Psi\underline{\theta}_j$, $\underline{\theta}_j = \Psi^T\underline{x}_j$

Φ : sensing matrix

$\underline{y}_j$: measurement vector of signal j $\underline{y}_j = \Phi \underline{x}_j$

4.2.1 Intersection algorithm

This algorithm assumes the JSM-1 signal model. Its working principle can be summarised in the idea that if we are able to subtract the measurements of the common component from the measurements, then we are left just with the measurements of the innovation part, which is sparser. After recovering the innovation component, it can be added to the common component to get the full signal. Even in the general case with J sensor nodes, the algorithm works pairwise considering the measurements from sensor j and the side information (without loss of generality assume that sensor 1 provides the side information). The common component is recovered by reconstructing the signal from sensor j separately in order to estimate its support $\mathcal{S}_j$ and, since the common component is shared by 1 and j, its support $\mathcal{S}_c$ can be estimated by intersecting $\mathcal{S}_j \cap \mathcal{S}_1$. Once the support is known it is enough to solve a least squares problem to get the coefficients. Let $\underline{\theta}_j = \Psi^T \underline{x}_j$, $\underline{\theta}_1 = \Psi^T \underline{x}_1$, $\underline{y}_j = \Phi \underline{x}_j$, $\underline{y}_1 = \Phi \underline{x}_1$

Algorithm 1 *Intersection* algorithm

Require: $\mathcal{S}_1 = \{i : (\underline{\theta}_1)_i \neq 0\}$, t, $A = \Phi\Psi$

Recover $\hat{\underline{\theta}}_j$ from $\underline{y}_j$

$\hat{\mathcal{S}}_j \leftarrow \left\{ i : \left| \left(\underline{\theta}_j\right)_i \right| > t \right\}$

$\hat{\mathcal{S}}_c \leftarrow S_1 \cap \hat{\mathcal{S}}_j$

$\left(\hat{\underline{\theta}}_c\right)_{\hat{\mathcal{S}}_c} = (A^T_{\hat{\mathcal{S}}_c} A_{\hat{\mathcal{S}}_c})^{-1} A^T_{\hat{\mathcal{S}}_c} \underline{y}_j$

$\underline{y}_{\mathrm{I,j}} = y_j - A\hat{\underline{\theta}}_c$

Recover $\hat{\underline{\theta}}_{\mathrm{I,j}}$ from $\underline{y}_{\mathrm{inn,j}}$

return $\underline{\theta}_j = \hat{\underline{\theta}}_c + \hat{\underline{\theta}}_{\mathrm{I,j}}$

4.2.2 Sort algorithm

This algorithm is very similar to the *Intersection* algorithm except for the estimation of the common component that is performed in a different fashion. The coefficients of the reconstructed signal are sorted by decreasing magnitude and, proceeding in order, if the component also belongs to the support of the side information then it is added to to the support of the common component. The algorithm terminates when k_c positions have been added to the support of the common component (k_c being the sparsity of the common component).

Algorithm 2 *Sort* algorithm

Require: $\mathcal{S}_1 = \{i : (\underline{\theta}_1)_i \neq 0\}$, k_c, $A = \Phi\Psi$
  Recover $\hat{\underline{\theta}}_j$ from $\underline{y}_j$
  $\hat{\mathcal{S}}_c \leftarrow \emptyset$
  for $i = \max_i \left|\left(\hat{\underline{\theta}}_j\right)_i\right|$ to $\min_i \left|\left(\hat{\underline{\theta}}_j\right)_i\right|$ **do**
    if $i \in \mathcal{S}_1$ **then**
      $\hat{\mathcal{S}}_c \leftarrow \hat{\mathcal{S}}_c \cup i$
      if $|\hat{\underline{\theta}}_c| = k_c$ **then**
        break
      end if
    end if
  end for
  $\left(\hat{\underline{\theta}}_c\right)_{\hat{\mathcal{S}}_c} = (A^T_{\hat{\mathcal{S}}_c} A_{\hat{\mathcal{S}}_c})^{-1} A^T_{\hat{\mathcal{S}}_c} \underline{y}_j$
  $\underline{y}_{\mathrm{I,j}} = y_j - A\hat{\underline{\theta}}_c$
  Recover $\hat{\underline{\theta}}_{\mathrm{I,j}}$ from $\underline{y}_{\mathrm{inn,j}}$
  return $\underline{\theta}_j = \hat{\underline{\theta}}_c + \hat{\underline{\theta}}_{\mathrm{I,j}}$

4.2.3 Texas Hold 'Em algorithm

The previous algorithms proceed essentially pairwise, so they may not fully exploit the system properties when the system has a large number of sensors. *Texas Hold 'Em* relies on sharing all or part of the measurements of all the J sensors in the network in the reconstruction process. The idea is that the innovation components can be treated as noise, so averaging the measurements from all the sensors we get something close to the measurements of the common component. This is clearly unbiased since in the limit of a network with infinitely many sensors we have that:

$$\underline{\hat{y}}_c = \lim_{J\to\infty} \frac{1}{J}\sum_{j=1}^{J} \underline{y}_j = \lim_{J\to\infty} \left[\underline{y}_c + \frac{1}{J}\sum_{j=1}^{J} \underline{y}_{I,j}\right] = \underline{y}_c \tag{4.2}$$

Moreover, this algorithm does not require any side information. Once the common component has been estimated by recovering from the averaged measurements, it can be subtracted from the measurements of each signal to recover the innovation component.

Algorithm 3 *Texas Hold 'Em* algorithm

Require: J, k_c, $A = \Phi\Psi$

$\underline{\hat{y}}_c = \frac{1}{J}\sum_{j=1}^{J} \underline{y}_j$
Recover $\underline{\theta}_c$ from $\underline{\hat{y}}_c$
for j in $1 : J$ **do**
  $\underline{\hat{y}}_{I,j} = \underline{y}_j - A\underline{\theta}_c$
  Recover $\underline{\hat{\theta}}_{I,j}$ from $\underline{\hat{y}}_{I,j}$
  $\underline{\hat{\theta}}_j = \underline{\theta}_c + \underline{\hat{\theta}}_{I,j}$
end for

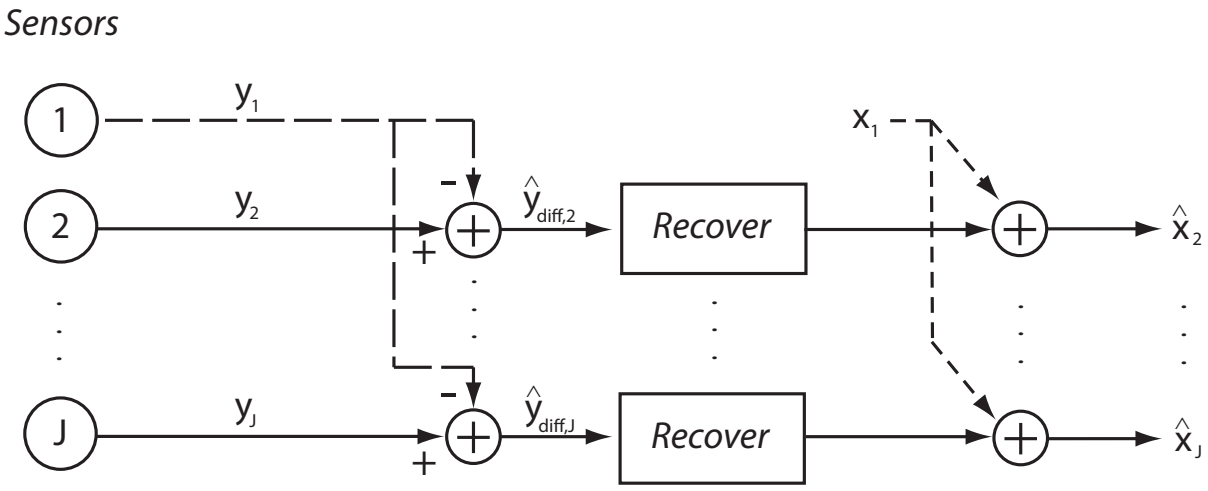

Figure 4.1. *Difference* algorithm

4.2.4 Difference algorithm

This is the first new algorithm that we propose. The underlying idea is to exploit side information to eliminate the need to recover the common component. Figure 4.1 presents a schematic representation of the algorithm. Proceeding pairwise by using the side information and one of the signals in the ensemble, it is possible to compute the difference between the measurements of the side information and of signal j. This clearly removes any component that is common to the two signals, so we are left with measurements of the difference of the innovation components. It is then possible to recover the difference signal from these measurements using any recovery procedure. Once the difference signal is recovered, it is then enough to add the side information to fully recover signal j. Before going into further details, it is interesting to notice that this algorithm does not introduce any error due to an incorrect estimation of the

common component, simply because this is removed from any recovery procedure and only side information is used to get it. A shortcoming of the algorithm is that the difference signal is less sparse than the innovation component of a single signal, so it needs more measurements to be recovered than if we were to recover just a single innovation component. However, this procedure seems advantageous in a scenario in which the sparsity of the common component is much larger than the sparsity of innovations. This is often the case when the signals in the ensemble are highly correlated. If we call k_I the sparsity of the innovation component of each signal in the ensemble, then in the worst case, the difference signal has sparsity $2k_I$. As a rule of thumb, if $2k_I < k_c$ then the difference algorithm may be favourable since it does not have to estimate the common component. This is even more relevant if the overall number of measurements is limited, and it does not allow to estimate the common component with great accuracy, as it is required by the *Intersection* and *Sort* algorithms.

Algorithm 4 *Difference* algorithm

Require: $A = \Phi\Psi$
  for j in 2:J **do**
    Compute $\underline{y}_{\text{diff,j}} = \underline{y}_j - \underline{y}_1$
    Recover $\underline{\theta}_{\text{diff,j}}$ from $\underline{y}_{\text{diff,j}}$
    $\hat{\underline{\theta}}_j = \underline{\theta}_1 + \underline{\theta}_{\text{diff,j}}$
  end for

4.2.4.1 A performance bound

$$\left\|\hat{\underline{\theta}}_j - \underline{\theta}_j\right\|_2 = \left\|\underline{\theta}_1 + \underline{\theta}_{\text{diff,j}} - \underline{\theta}_j\right\|_2 = \left\|\underline{\theta}_{I,1} - \underline{\theta}_{I,j} + \underline{\theta}_{\text{diff,j}}\right\|_2 = \tag{4.3}$$

$$= \left\|\underline{\theta}_{\text{diff,j}} - \left(\underline{\theta}_{I,j} - \underline{\theta}_{I,1}\right)\right\|_2 \leq C\epsilon \tag{4.4}$$

where $\epsilon = \left\|\underline{y}_{\text{diff,j}} - \Phi\left(\underline{x}_{I,1} - \underline{x}_{I,j}\right)\right\|$ is the norm of the noise affecting the measurements of the difference signal and C is a constant that depends on the method used for reconstruction and on the RIP constant of the sensing matrix. Ideally (no quantization, etc.) $\epsilon = 0$ and provided that there are enough measurements reconstruction is perfect. This means that the *Difference* algorithm is exact. Perfect reconstruction is possible if there is no loss of information, unlike *Texas Hold 'Em* which is limited by the residual noise in the averaging procedure.

4.2.4.2 Some distortion-rate bounds

We will follow a line of reasoning similar to the one used in the analysis of CS-SPLIT. This allows to get some bounds for the distortion-rate performance of the algorithm under some conditions that are essentially the same as the ones reported for CS-SPLIT. Let's first introduce some notation:

k_j: total sparsity of signal j

k_c: sparsity of the common component

$k_{1\cap j}$: sparsity of the intersection of the supports of signals 1 and j (note that if $J = 2$ this is k_c, but in the case with more sensors it may be greater because the common component

is common to all signals, so it is possible that the innovation component of signal j partially overlaps with the innovation component of signal 1)

$k_{\text{diff,j}}$: sparsity of the difference signal $\theta_{I,j} - \theta_{I,1}$

y_{ij}: measurement i of signal j

$y_{ij} \sim \mathcal{N}\left(0, \frac{k_j}{m}\sigma_x^2\right)$

Theorem 16. *Let an ensemble of J signals follow the JSM-1 joint sparsity model and the* Difference *algorithm be used for recovery. The nonzero entries of each signal in the sparse basis have variance σ_x^2 and zero mean. Assume that $m = O(k \log n)$ measurements are obtained with a Gaussian sensing matrix Φ as $\underline{y} = \Phi\underline{\theta}$ and that $\Phi_{ij} \overset{i.i.d.}{\sim} \mathcal{N}\left(0, \frac{1}{m}\right)$. Also assume that that $m > 60 \log n$, $k < \frac{1}{4}\left(\frac{1}{\mu} + 1\right)$ where μ is the coherence of Φ. Furthermore, assume that the BPDN method is used in the recovery steps inside the* Difference *algorithm. If the assumptions hold, then the distortion $D_j = \left\|\underline{\theta} - \hat{\underline{\theta}}\right\|_2^2$ in the j-th reconstructed signal as a function of rate R is bounded as follows, with high probability.*

$$\frac{k_1 + k_j - 2k_{1\cap j}}{m} k_{diff,j} \sigma_x^2 2^{-2R} \leq D_j(R) \leq \frac{4\left(k_1 + k_j - 2k_{1\cap j}\right)\sigma_x^2 2^{-2R}}{1 - \sqrt{\frac{15\log n}{m}}\left(4k_{diff,j} - 1\right)} \tag{4.5}$$

The proof will follow after stating a lemma considering the correlation of measurements of different signals. This lemma will be useful in the proof of the theorem and it is actually more general than what is needed, because it considers an arbitrary basis Ψ under which the signal is sparse.

Lemma 17. *Let y_{ij} and y_{i1} be the i-th measurement of signals j and 1 respectively. Let those signals be two signals in a JSM-1 ensemble. Measurements are obtained as $\underline{y} = \Phi\Psi\underline{\theta}$ with $\Phi_{ij} \overset{i.i.d.}{\sim} \mathcal{N}\left(0, \frac{1}{m}\right)$. Then:*

$$\mathbb{E}\left[y_{ij}y_{i1}\right] = \frac{k_{1\cap j}}{m}\sigma_x^2 \tag{4.6}$$

Proof. Before the math it is useful to notice that

$$\mathbb{E}\left[(\theta_c)_l\,(\theta_c)_{l'}\right] = \begin{cases} \mathbb{E}\left[(\theta_c)_l^2\right] = \sigma_x^2 & \text{if } l = l' \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbb{E}\left[(\theta_c)_l\,(\theta_{I,j})_{l'}\right] = 0$$

$$\mathbb{E}\left[(\theta_{I,j})_l\,(\theta_{I,1})_{l'}\right] = \begin{cases} \sigma_x^2 & \text{if } l = l' \text{ and } l, l' \in \mathcal{S}_{1\cap j} \\ 0 & \text{otherwise} \end{cases}$$

We used the symbol $\mathcal{S}_{1\cap j}$ to denote the support resulting from the intersection of the supports of signals 1 and j.

$$
\begin{aligned}
\mathbb{E}\left[y_{ij}y_{i1}\right] &= \mathbb{E}\left[\left(\sum_{k,l}\Phi_{ik}\Psi_{kl}\left((\theta_c)_l+(\theta_{I,j})_l\right)\right)\left(\sum_{k',l'}\Phi_{ik'}\Psi_{k'l'}\left((\theta_c)_{l'}+(\theta_{I,1})_{l'}\right)\right)\right] \\
&= \mathbb{E}\left[\sum_{k,k',l,l'}\Phi_{ik}\Psi_{kl}\left(\theta_c\right)_l\Phi_{ik'}\Psi_{k'l'}\left(\theta_c\right)_{l'}\right] \\
&+ \mathbb{E}\left[\sum_{k,k',l,l'}\Phi_{ik}\Psi_{kl}\left(\theta_{I,j}\right)_l\Phi_{ik'}\Psi_{k'l'}\left(\theta_c\right)_{l'}\right] \\
&+ \mathbb{E}\left[\sum_{k,k',l,l'}\Phi_{ik}\Psi_{kl}\left(\theta_c\right)_l\Phi_{ik'}\Psi_{k'l'}\left(\theta_{I,1}\right)_{l'}\right] \\
&+ \mathbb{E}\left[\sum_{k,k',l,l'}\Phi_{ik}\Psi_{kl}\left(\theta_{I,j}\right)_l\Phi_{ik'}\Psi_{k'l'}\left(\theta_{I,1}\right)_{l'}\right] \\
&= \sum_{k,k',l,l'}\mathbb{E}\left[\Phi_{ik}\Phi_{i'k'}\right]\mathbb{E}\left[\Psi_{kl}\left(\theta_c\right)_l\Psi_{k'l'}\left(\theta_c\right)_{l'}\right] \\
&+ \sum_{k,k',l,l'}\mathbb{E}\left[\Phi_{ik}\Phi_{i'k'}\right]\mathbb{E}\left[\Psi_{kl}\left(\theta_{I,j}\right)_l\Psi_{k'l'}\left(\theta_c\right)_{l'}\right] \\
&+ \sum_{k,k',l,l'}\mathbb{E}\left[\Phi_{ik}\Phi_{i'k'}\right]\mathbb{E}\left[\Psi_{kl}\left(\theta_c\right)_l\Psi_{k'l'}\left(\theta_{I,1}\right)_{l'}\right] \\
&+ \sum_{k,k',l,l'}\mathbb{E}\left[\Phi_{ik}\Phi_{i'k'}\right]\mathbb{E}\left[\Psi_{kl}\left(\theta_{I,j}\right)_l\Psi_{k'l'}\left(\theta_{I,1}\right)_{l'}\right] \\
&= \frac{1}{m}\sum_{k,l}\mathbb{E}\left[\Psi_{kl}\Psi_{kl}\left(\theta_c\right)_l\left(\theta_c\right)_l\right]+0+0+\frac{1}{m}\sum_{k,l}\mathbb{E}\left[\Psi_{kl}\Psi_{kl}\left(\theta_{I,j}\right)_l\left(\theta_{I,1}\right)_l\right] \\
&= \frac{k_c}{m}\sigma_x^2+\frac{k_{\{1\cap j\}/c}}{m}\sigma_x^2 \\
&= \frac{k_{1\cap j}}{m}\sigma_x^2
\end{aligned}
$$

□

We notice that the algorithm performs recovery from $\underline{y}_{\text{diff,j}}$, that can be seen as a memoryless Gaussian source. Each component has zero mean and the variance can be readily computed by exploiting the previous lemma and the result obtained in (3.7).

$$\begin{aligned}\mathbb{E}\left[y^2_{\text{diff,ij}}\right] &= \mathbb{E}\left[y^2_{\text{i1}} + y^2_{\text{ij}} - 2y_{\text{i1}}y_{\text{ij}}\right] = \frac{k_1}{m}\sigma_x^2 + \frac{k_j}{m}\sigma_x^2 - 2\,\mathbb{E}\left[y_{\text{i1}}y_{\text{ij}}\right] = \frac{k_1 + k_j}{m}\sigma_x^2 - 2\frac{k_{1\cap j}}{m}\sigma_x^2 \\ &= \frac{k_1 + k_j - 2k_{1\cap j}}{m}\sigma_x^2\end{aligned}$$

Hence, defining distortion as the squared error norm, the distortion-rate curve on the measurements is

$$D_{sm} = \frac{k_1 + k_j - 2k_{1\cap j}}{m}\sigma_x^2 2^{-2R}$$

Now, we can recall the performance bounds presented in (3.23) and (3.26) concerning CS recovery by means of an oracle and by means of the basis pursuit denoising algorithm respectively. We also recall that we presented in (3.27) a result about the coherence of a Gaussian sensing matrix. Putting all the results together, it follows that the reconstruction error is bound as stated in theorem (16), where the lower bound assumes reconstruction by solving a least squares problem when the support is told by an oracle and the upper bound is given by the reconstruction performance of the BPDN algorithm.

4.2.5 Texas Difference algorithm

This is an improvement that we propose over the *Texas Hold 'Em* algorithm. Figure 4.2 presents a schematic representation of the algorithm. Albeit it maintains the original idea of averaging the collected measurements, it is aimed at providing some performance gain by

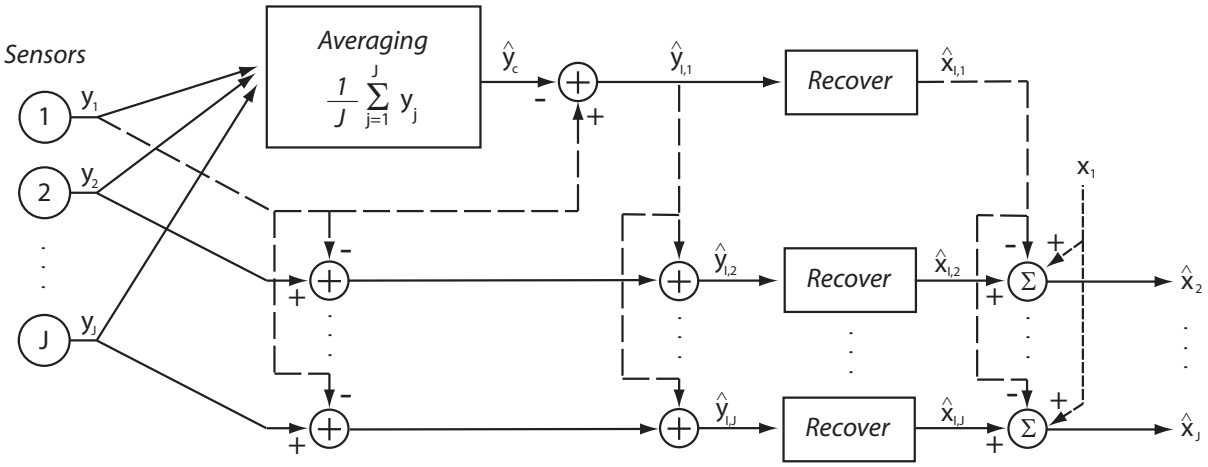

Figure 4.2. *Texas Difference* algorithm

avoiding any reconstruction of the common component and rather using it directly from the measurements, in a fashion similar to the *Difference* algorithm. As it is conceptually close to the *Difference* algorithm, it mainly targets situations in which the signals in the ensemble are highly correlated, thus having innovations that are much sparser than the common component, and few available measurements. It also differs from *Texas Hold 'Em* because it requires side information, but since both this and Texas are well suited for networks with a large number of sensors (so that the residual noise after averaging is small), the overhead due to side information may be negligible or very small anyway.

Algorithm 5 *Texas Difference* algorithm

Require: J,$A = \Phi\Psi$, k_I

$\hat{\underline{y}}_c = \frac{1}{J}\sum_{j=1}^{J}\underline{y}_j$

$\hat{\underline{y}}_{I,1} = \underline{y}_1 - \hat{\underline{y}}_c$

Recover $\hat{\underline{\theta}}_{I,1}$ from $\hat{\underline{y}}_{I,1}$

for j in 2:J **do**

$\underline{y}_{\text{diff,j}} = \underline{y}_j - \underline{y}_1$

$\hat{\underline{y}}_{I,j} = \underline{y}_{\text{diff,j}} + \hat{\underline{y}}_{I,1}$

Recover $\hat{\underline{\theta}}_{I,j}$ from $\hat{\underline{y}}_{I,j}$

$\hat{\underline{\theta}}_j = \theta_1 - \hat{\underline{\theta}}_{I,1} + \hat{\underline{\theta}}_{I,j}$

end for

4.2.5.1 A performance bound

Let's derive a simple bound on the reconstruction performance of the algorithm to gain some intuition about the conditions that make it effective, when compared to other algorithms.

$$\begin{aligned}\left\|\hat{\underline{\theta}}_j - \underline{\theta}_j\right\|_2 &= \left\|\underline{\theta}_1 - \hat{\underline{\theta}}_{I,1} + \hat{\underline{\theta}}_{I,j} - \underline{\theta}_j\right\|_2 = \left\|\underline{\theta}_{I,1} - \hat{\underline{\theta}}_{I,1} - \underline{\theta}_{I,j} + \hat{\underline{\theta}}_{I,j}\right\|_2 \\ &= \left\|\left(\underline{\theta}_{I,1} - \hat{\underline{\theta}}_{I,1}\right) + \left(\hat{\underline{\theta}}_{I,j} - \underline{\theta}_{I,j}\right)\right\|_2 \leq \left\|\underline{\theta}_{I,1} - \hat{\underline{\theta}}_{I,1}\right\|_2 + \left\|\hat{\underline{\theta}}_{I,j} - \underline{\theta}_{I,j}\right\|_2\end{aligned}$$

Let's analyse how the innovation components are recovered.

$$\hat{\underline{y}}_{I,1} = \underline{y}_1 - \hat{\underline{y}_c} = \underline{y}_c + \underline{y}_{I,1} - \underline{y}_c - \frac{1}{J}\sum_{l=1}^{J}\underline{y}_{I,l} = \underline{y}_{I,1} - \frac{1}{J}\sum_{l=1}^{J}\underline{y}_{I,l}$$

$$\underline{\hat{y}}_{I,j} = \underline{y}_j - \underline{y}_1 + \underline{\hat{y}}_{I,1} = \underline{y}_c + \underline{y}_{I,j} - \underline{y}_c - \underline{y}_{I,1} + \underline{y}_{I,1} - \frac{1}{J}\sum_{l=1}^{J}\underline{y}_{I,l} = \underline{y}_{I,j} - \frac{1}{J}\sum_{l=1}^{J}\underline{y}_{I,l}$$

Let's call $\underline{n} = \frac{1}{J}\sum_{l=1}^{J}\underline{y}_{I,l}$, i.e. the "noise" component in the measurements that is due to the averaging procedure, which does not exactly cancel out the innovation components. Hence, $\underline{\hat{y}}_{I,j} = A\underline{\theta}_{I,j} - \underline{n}$ for $j \in [1, J]$. If we use a reconstruction procedure from noisy measurements that has a performance guarantee of the type that the reconstruction error is proportional just to the noise power, and we assume that $\left\|\underline{\theta}_{I,j}\right\|_2 = \eta$ for all $j \in [1, J]$, we can write:

$$\left\|\underline{\hat{\theta}}_{I,j} - \underline{\theta}_{I,j}\right\|_2 \leq C \cdot \|\underline{n}\|_2 = C \cdot \left\|\frac{1}{J}\sum \Phi\underline{\theta}_{I,j}\right\|_2 \leq C \cdot \frac{\sqrt{1+\delta}}{\sqrt{J}} \left\|\underline{\theta}_{I,j}\right\|_2 = C \cdot \frac{\sqrt{1+\delta}}{\sqrt{J}}\eta$$

where δ is the RIP constant of matrix $A = \Phi\Psi$. Finally we can plug this result in the derivation for the recovery error and we obtain

$$\left\|\underline{\hat{\theta}}_j - \underline{\theta}_j\right\|_2 \leq \left\|\underline{\theta}_{I,1} - \underline{\hat{\theta}}_{I,1}\right\|_2 + \left\|\underline{\hat{\theta}}_{I,j} - \underline{\theta}_{I,j}\right\|_2 \leq 2C \cdot \frac{\sqrt{1+\delta}}{\sqrt{J}}\eta \tag{4.7}$$

This analysis points out a few interesting facts about the algorithm. Even if quantization or other sources of noise are not considered by the analysis, the algorithm is still bound to a certain reconstruction error. This does not happen for other algorithms (e.g. *Difference*) that we have seen to be exact. Here, the limiting factor is the averaging procedure that imposes a floor on the reconstruction error, that cannot be overcome by adjustments on the rate. However, this

error floor decreases as $\frac{1}{\sqrt{J}}$, so a network with a large number of sensors may indeed be limited by the quantization rate or other phenomena rather than the averaging procedure.

We have seen how the *Texas Difference* algorithm borrows ideas from both the *Texas Hold 'Em* strategy and the *Difference* procedure, so one may wonder whether it really improves over those methods and when it may be interesting to use it. *Texas Difference* inherits the averaging procedure from *Texas Hold 'Em*, that makes it suitable when the number of sensors is large as already discussed. However, it improves over Texas when the signals are highly correlated and few measurements are available because Texas may have difficulties in recovering the common component. The idea of computing some differences to use the common component in an implicit way is derived from the *Difference* algorithm. *Texas Difference* is able to run the compressed sensing recovery procedures only for the the innovation components themselves, whereas *Difference* has to recover a difference between two innovation signals, which is typically less sparse.

4.3 Performance comparison

We are going to test the performance of the algorithms outlined in the previous section by running some simulations, under different conditions. It can be seen that the experimental results follow our intuition, concerning the performance of the methods with respect to each other. We remark that the y-axis, labelled Mean Square Error, actually represents the mean MSE over all the J signals in the ensemble.

Figure 4.3 shows a scenario in which a very limited number of measurements is taken from each sensor but the number of sensors is relatively high ($J = 100$). This is the best case for the *Texas Difference* algorithm, which, in fact, outperforms all the others by a great margin. It is also interesting to remark how the plain *Texas* algorithm has very bad performance due to the constraint on the number of measurements. The exact algorithms perform very close to each other, and, in particular, *Difference* is not able to improve over *Intersection* because the sparsity of the innovation component is quite high compared to the sparsity of the common component. In particular, since $k_c = 15$ and $k_I = 10$, the rule of thumb $2k_I < k_c$, that we have given as an indicator of when *Difference* can be useful, does not hold. Figure 4.4 is interesting because it highlights the error floor reached by the *Texas Difference* algorithm for high values of rate R. As previously discussed this error floor is due to the averaging procedure: no matter how fine measurements are quantized, the residual noise of averaging will prevent any improvement of performance. This also happens for the *Texas* algorithm, albeit not evident in that picture. Again, the poor performance of *Difference* is due to the strong contribution of innovation components to signals. Figure 4.6 and Figure 4.5 are worth to be analysed together

because they have very similar parameters and reveal some interesting phenomena. As first comment, *Texas* performs poorly mainly because it needs more measurements to improve its performance significantly. Then, it is interesting to notice how the increase in the number of sensors from 10 to 100 significantly lowers the error floor for *Texas Difference*, which is also the best method at low bitrates. At high bitrates the *Difference* algorithm is the best when the number of measurements is limited ($m = 60$) and close to the other algorithms when more measurements are available. Finally, Figure 4.7 shows a situation that is extremely unfavourable to the *Difference* and *Texas Difference* algorithms because because of the high value of k_I compared to k_c. However, this and the high number of sensors make *Texas* perform rather well.

Due to the fact that the various algorithms seem to perform differently depending on the number of available measurements, we include a simulation of the reconstruction error as a function of how many measurements are collected by each sensor. The result is reported in Figure 4.8 and it confirms what we previously reported, i.e. *Difference* and *Texas Difference* work are the best when few measurements are available, if the correct assumptions on the signal are satisfied. However, they are outperformed by other algorithms when more and more measurements are available. This fact should be regarded as an important point: thanks to the improvements made possible by *Difference* and *Texas Difference* it is possible to reduce the number of measurements that is needed to achieve a good reconstruction error. The benefits of this fact are evident in terms of complexity of the system for measurement acquisition, as well as in terms of data transmission.

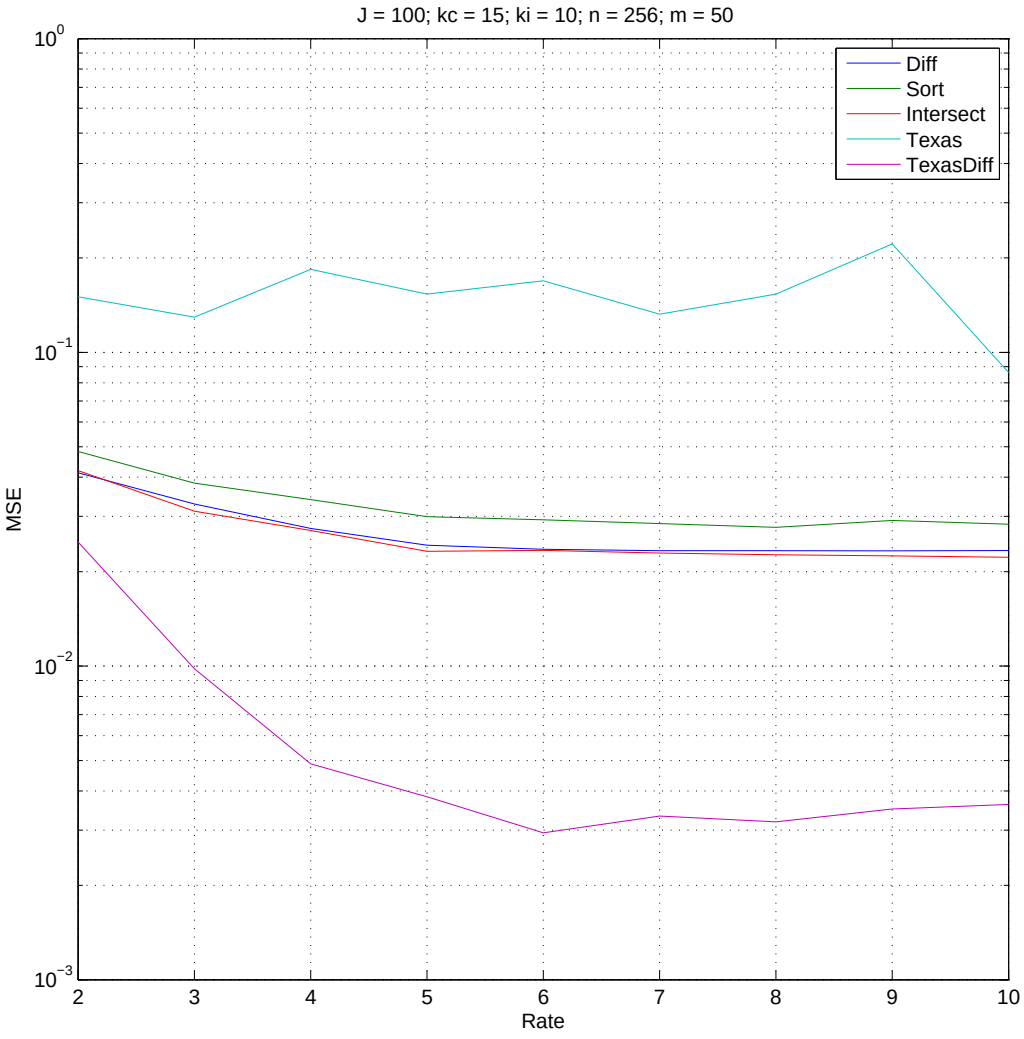

Figure 4.3. Reconstruction error. $n = 256$, $k_c = 15$, $k_I = 10$, $m = 50$, $J = 100$

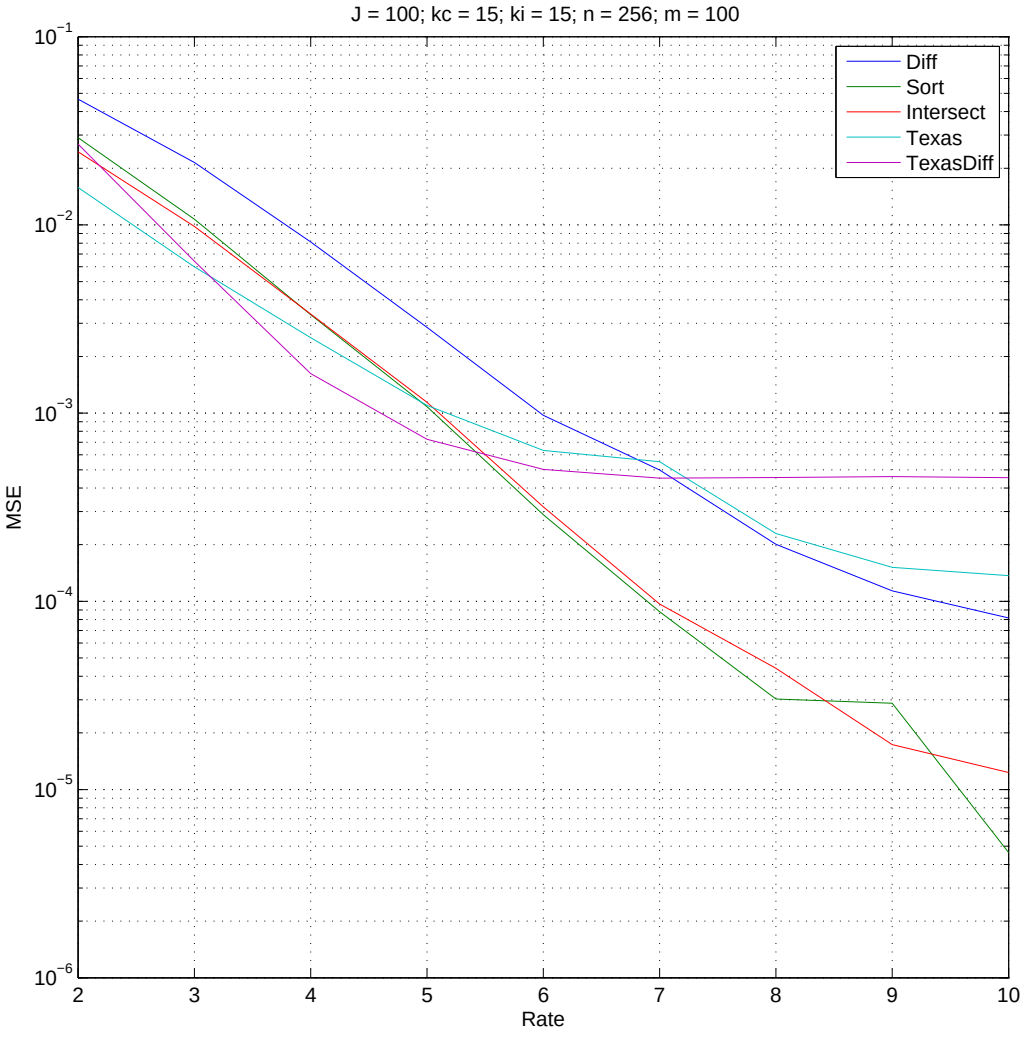

Figure 4.4. Reconstruction error. $n = 256$, $k_c = 15$, $k_I = 15$, $m = 100$, $J = 100$

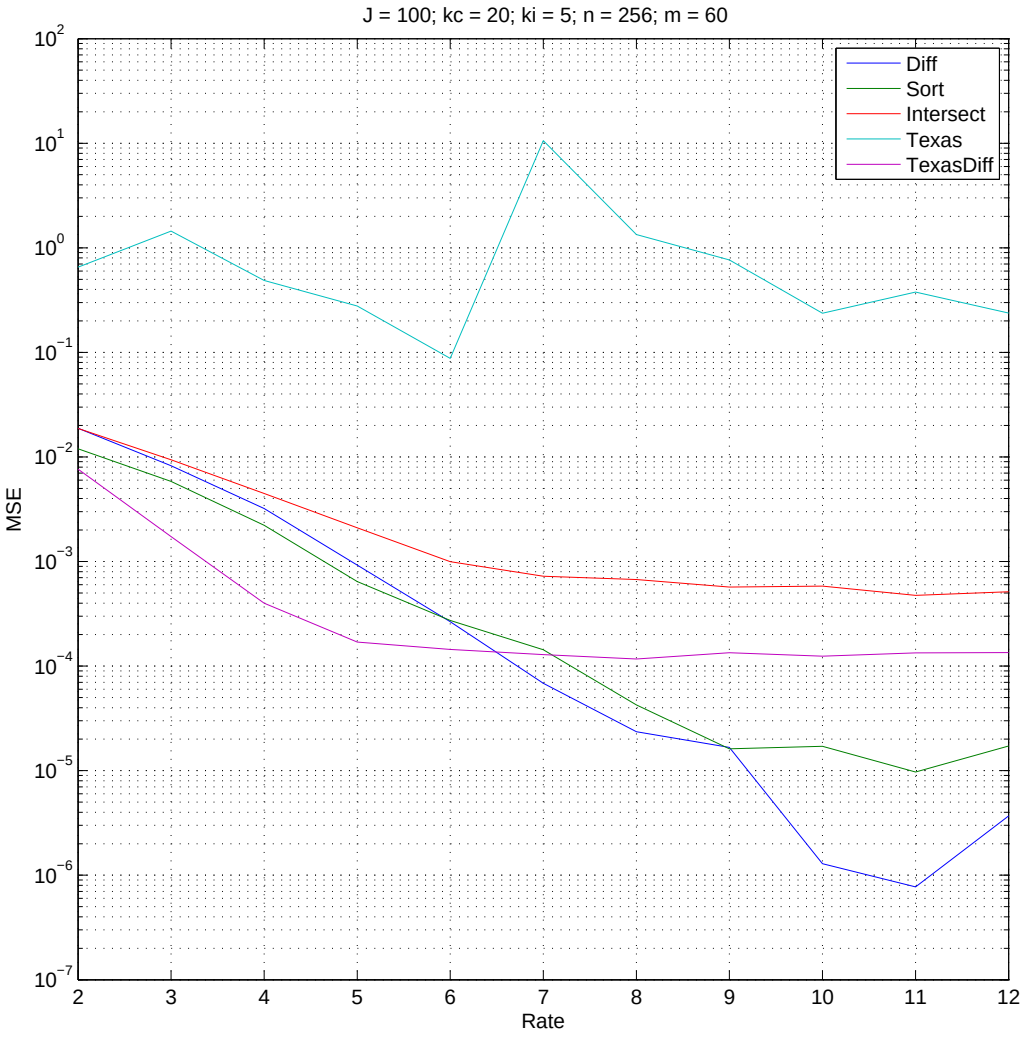

Figure 4.5. Reconstruction error. $n = 256$, $k_c = 20$, $k_I = 5$, $m = 60$, $J = 100$

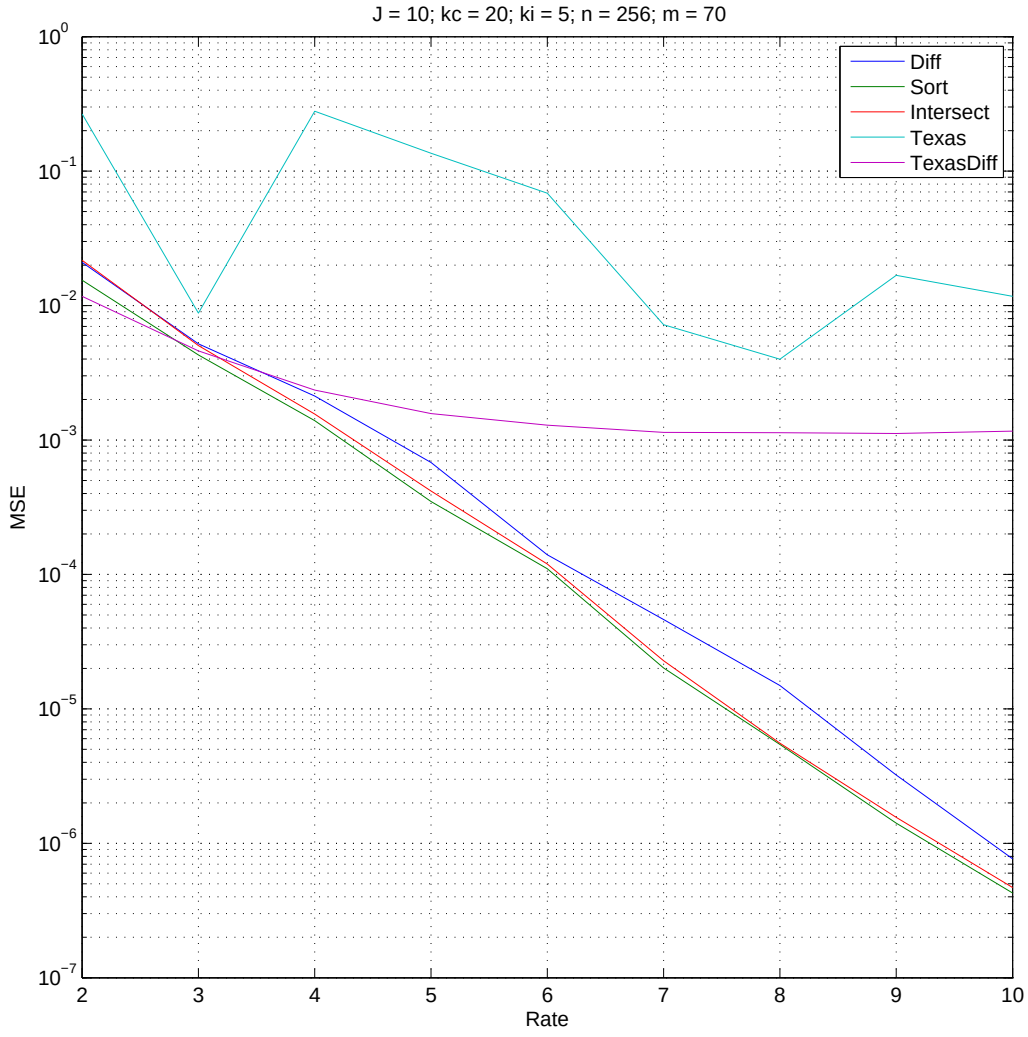

Figure 4.6. Reconstruction error. $n = 256$, $k_c = 20$, $k_I = 5$, $m = 70$, $J = 10$

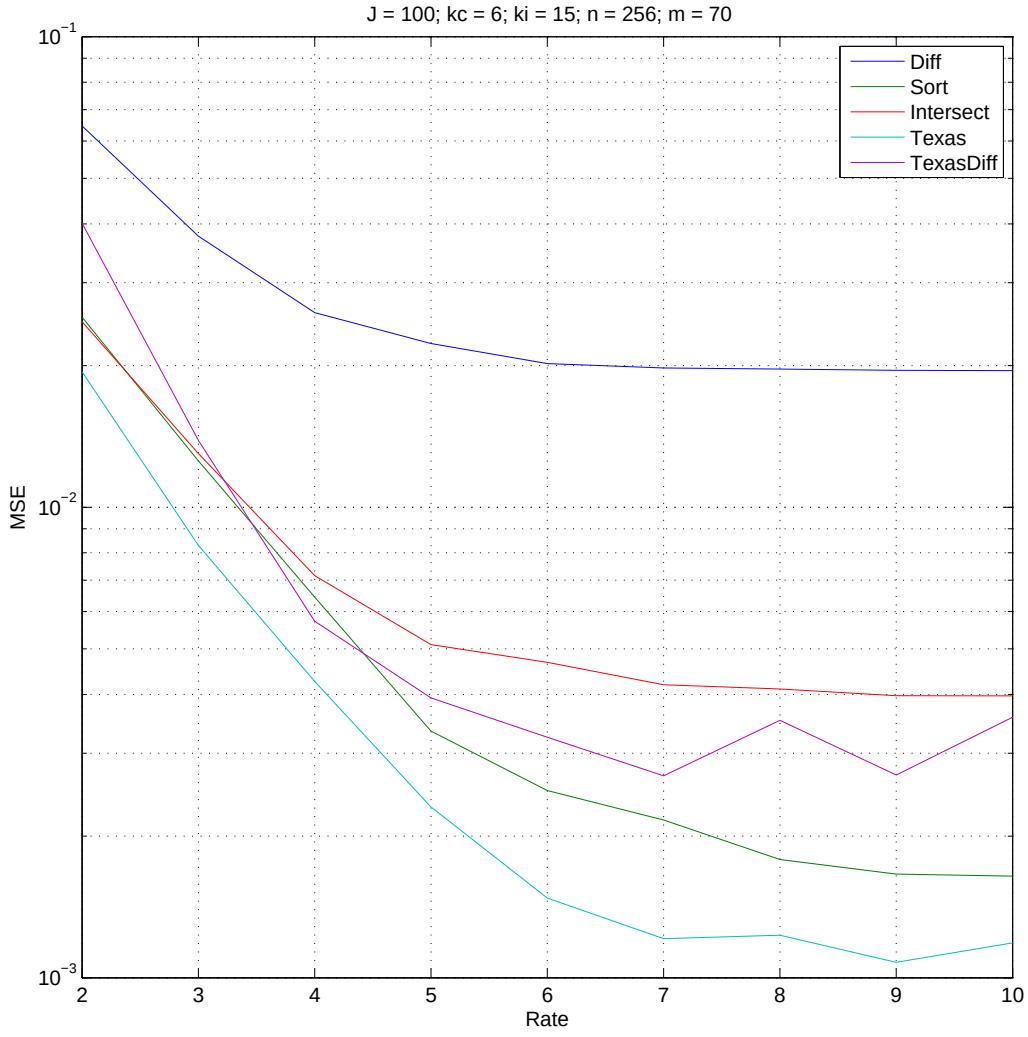

Figure 4.7. Reconstruction error. $n = 256$, $k_c = 6$, $k_I = 6$, $m = 70$, $J = 100$

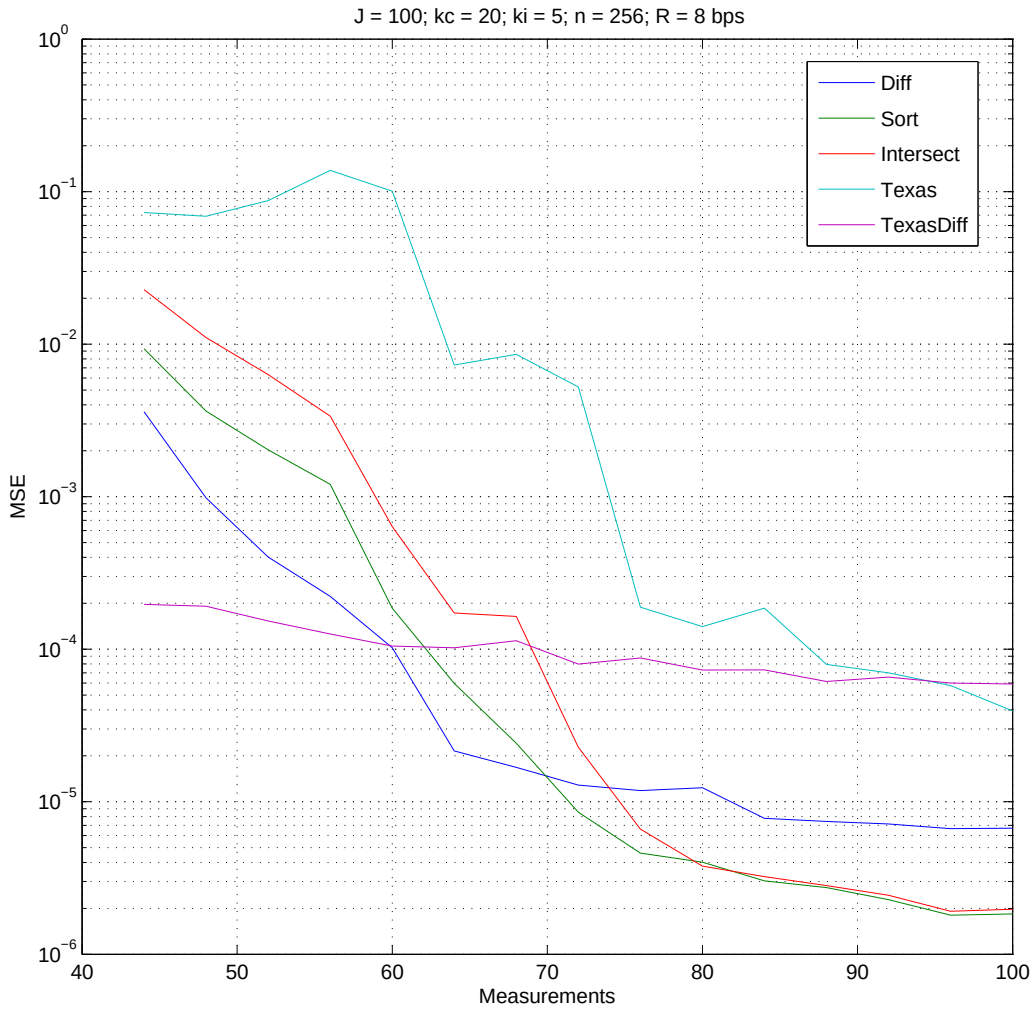

Figure 4.8. Reconstruction error as function of number of measurements. $n = 256$, $k_c = 20$, $k_I = 5$, $J = 100$, $R = 8$ bps

CHAPTER 5

A COMPRESSED-SENSING, MULTIPLE-DESCRIPTION SCHEME OVER A PACKET ERASURE CHANNEL

This chapter will draw concepts from all the previous chapters in order to analyse a system (e.g. a sensor network) that collects measurements of some quantity of interest by means of compressed sensing, bundles them in packets and transmits the packets over a packet erasure channel. The channel is unreliable, it may lose a packet with probability p_{loss} and using a connection-oriented protocol that takes care of retransmission is not applicable because of efficiency constraints (with one exception that will be noted in the following). This is an interesting scenario to apply a multiple-description coding technique, whose job is to make the system more robust to the channel unreliability. However, the technique must also be simple enough to be implemented in a sensor network, that has very tight complexity constraints. Hence, the CS-SPLIT method developed in chapter 3 seems to be a good candidate to solve the problem because it fits the compressed sensing frame and it moves all the complexity to the receiver. Moreover, an assumption is made on the joint distribution of the sensor data. We assume that they follow the JSM-1 model, that we thoroughly discussed in the previous chapter, along with some joint recovery algorithms. Most of the algorithms that we analysed require the use of side information, i.e. the perfect knowledge of one of the signals in the ensemble. From a practical standpoint, requiring side information just means that one of the signals will be sensed taking more measurements to ensure a high quality reconstruction. This method causes some overhead

to be present if compared to a scheme without the need of side information. Let's call J the number of sensors, m_1 the number of measurements for signal 1, that we suppose without loss of generality, to be our side information, and m the number of measurements for each of the other signals. Then, a relative measure of the overhead can be:

$$OH_{\text{base}} = \frac{m_1 - m}{Jm} \tag{5.1}$$

that is simply the ratio between how many extra measurements are taken for signal 1 and the total number of measurements taken from the ensemble if no side information was needed. We will refer to that quantity as base overhead because it is the minimum amount of overhead that system may have. In fact, as side information is critical we may decide that retransmission should be performed unless at least one description is received. Clearly, this scheme increases the total overhead because of retransmissions and the increase is more and more significant as p_{loss} gets higher. We can easily compute the expected total overhead as a function of p_{loss} by

considering that every time a retransmission occurs we add $\frac{m_1}{Jm}$ to the current overhead, starting from the base overhead.

$$\begin{aligned}
\mathbb{P}\left(\text{k total retransmissions}\right) &= \mathbb{P}\left(OH_{\text{tot,retr}} = k\frac{m_1}{Jm}\right) = \left(p_{\text{loss}}^2\right)^k\left(1-p_{\text{loss}}^2\right) \\
\mathbb{E}\left[OH_{\text{tot,retr}}\right] &= \sum_{k=0}^{\infty} k\frac{m_1}{Jm}\left(p_{\text{loss}}^2\right)^k\left(1-p_{\text{loss}}^2\right) \\
&= \frac{m_1}{Jm}\left(1-p_{\text{loss}}^2\right)\sum_{k=1}^{\infty} k\left(p_{\text{loss}}^2\right)^k \\
&= \frac{m_1}{Jm}\left(1-p_{\text{loss}}^2\right)p_{\text{loss}}^2\sum_{k=1}^{\infty}\frac{d}{d\left(p_{\text{loss}}^2\right)}\left[\left(p_{\text{loss}}^2\right)^k\right] \\
&= \frac{m_1}{Jm}\left(1-p_{\text{loss}}^2\right)p_{\text{loss}}^2\frac{d}{d\left(p_{\text{loss}}^2\right)}\left[\frac{p_{\text{loss}}^2}{1-p_{\text{loss}}^2}\right] \\
&= \frac{m_1}{Jm}\frac{p_{\text{loss}}^2}{1-p_{\text{loss}}^2}
\end{aligned}$$

$$\mathbb{E}\left[OH\right] = OH_{\text{base}} + \mathbb{E}\left[OH_{\text{tot,retr}}\right] = \frac{m_1-m}{Jm} + \frac{m_1}{Jm}\frac{p_{\text{loss}}^2}{1-p_{\text{loss}}^2} \tag{5.2}$$

The result agrees with the outcome of the simulations as shown in Figure 5.1. For clarity, let us state explicitly how packets are formed and how CS-SPLIT is compared against a system without any MDC strategy. When using CS-SPLIT, we collect m measurements from each sensor and two packets of $\frac{m}{2}$ measurements are formed with the first half and second half of the measurement vector. Each packet is called a description because it can be decoded separately according to the method and the performance discussed when presenting CS-SPLIT. Notice that CS-SPLIT could be easily generalized to a higher number of descriptions. If no MDC scheme is used, all the m measurements are grouped in a single packet.

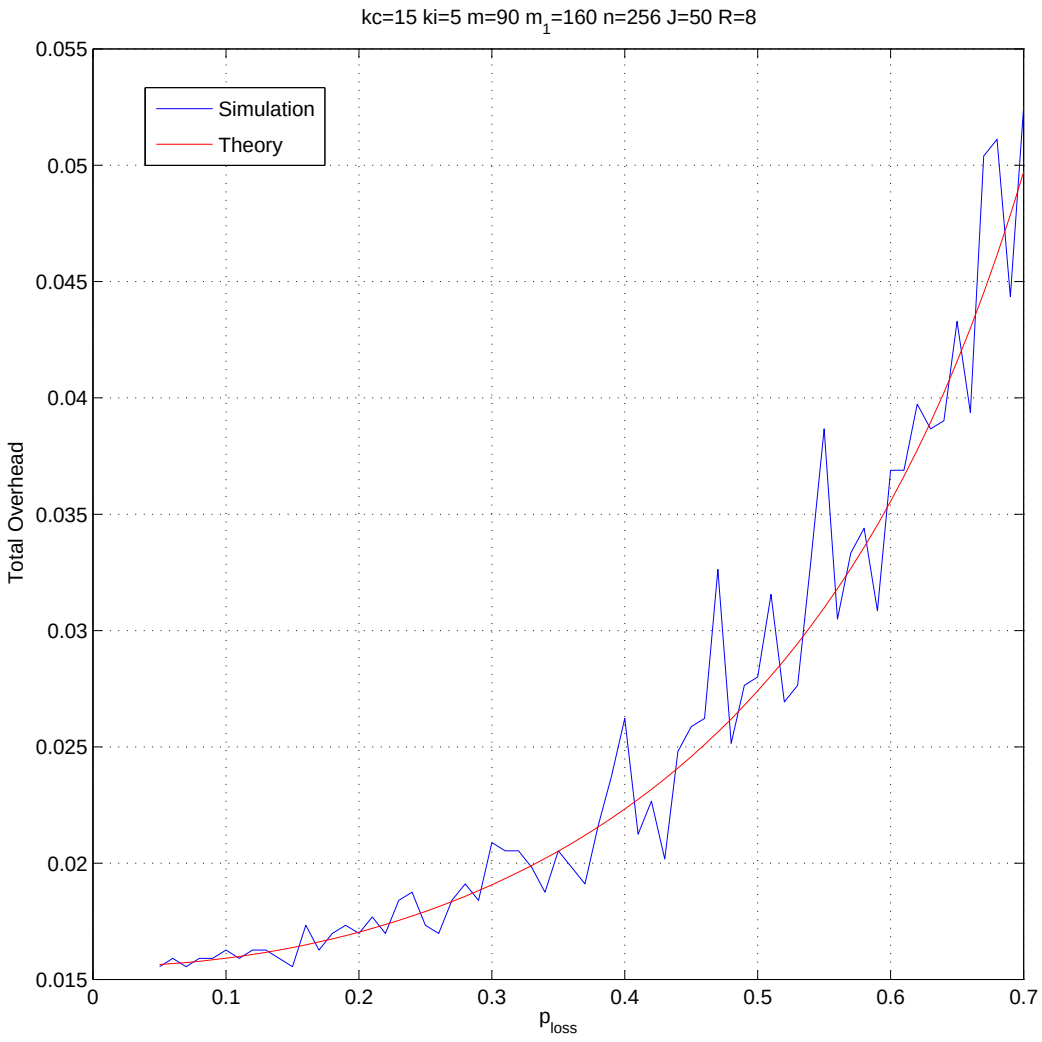

Figure 5.1. Total overhead as function of packet loss probability

It is now interesting to discuss the performance, in terms of reconstruction mean squared error, to actually verify if our multiple-description coding strategy is beneficial to the system. Also, various reconstruction algorithms are compared to evaluate their relative performance. The first interesting result from the simulations is the poor performance of the *Texas Difference* algorithm. This is essentially due to the fact that the algorithm does not fit well the packetization model used by the system. One of the key steps in the algorithm is averaging all the available measurements to produce an estimate of the measurements of the common component. However, when CS-SPLIT is used and descriptions may be lost by the channel, the receiver finds itself with some non-homogeneous data. It might have received both descriptions from a few sensors, but, for some, it only has the first description and, for some others, the second description only. This condition makes it difficult to perform averaging because it makes no sense to sum measurements coming from different rows of the sensing matrix. Hence, the best strategy that the algorithm can adopt is checking which description is the most available and average using those data only. It is clear that there is a forced waste of data, that limits the quality of the averaging procedure and, hence, compromises the performance of the whole algorithm. Albeit not present in the simulation results that will be discussed in the following, the *Texas Hold 'Em* algorithm is expected to suffer from the same problem since it relies on averaging as well. Another interesting phenomenon that is evident in all the figures concerns the performance of the algorithms in the system without MDC. All the algorithms display almost identical performance. This is mainly due to the packet losses in the channel. When a packet is lost and there is no alternative description, all the information from that sensor is lost and the MSE is

capped to the maximum for all the algorithms. The MSE displayed in the figures is an average quantity obtained from the MSEs of all the sensors, so in the average all the algorithms appear to have the same performance because their relative differences when correctly operating are much smaller than difference between the MSE of a correct reconstruction and the maximum MSE. This phenomenon is less evident in the case with multiple descriptions thanks to the fact that it less probable to all the descriptions are lost. However, when p_{loss} increases, the same tendency to compaction appears. The results clearly show that the CS-SPLIT method is able to provide a performance gain to the system in terms of average MSE. Since the method has very low complexity, it is indeed recommended to adopt it in a packet-loss scenario like the one described. Figure 5.2 also confirms that the *Difference* algorithm suits well the multiple description scenario. In fact, we have seen how it was able to improve over existing algorithms when the number of measurements is limited. This is interesting if we consider that, when a description is lost and only a single one is received, it may contain barely enough measurements to make the recovery procedure work correctly. Hence, using *Difference* is a way to improve the quality of the reconstruction in such a case. The figures use the same algorithm for both side and central decoding to have a fair comparison among the algorithms. However, it seems reasonable to suggest that, in practice, different algorithms can be used. For example, *Difference* seems naturally suited to side decoding while *Intersect* or *Sort* to central decoding.

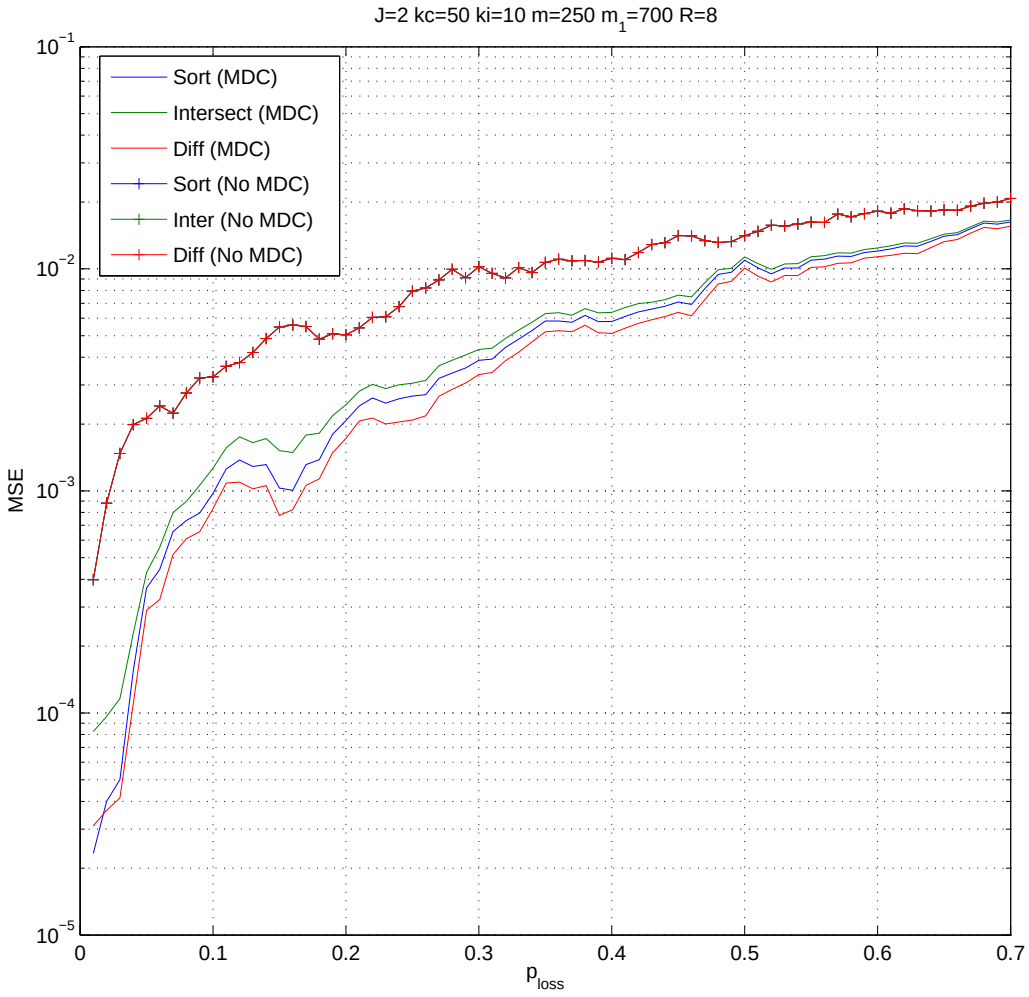

Figure 5.2. Mean MSE as function of p_{loss}. $J = 2$, $k_c = 50$, $k_I = 10$, $m = 250$, $m_1 = 700$

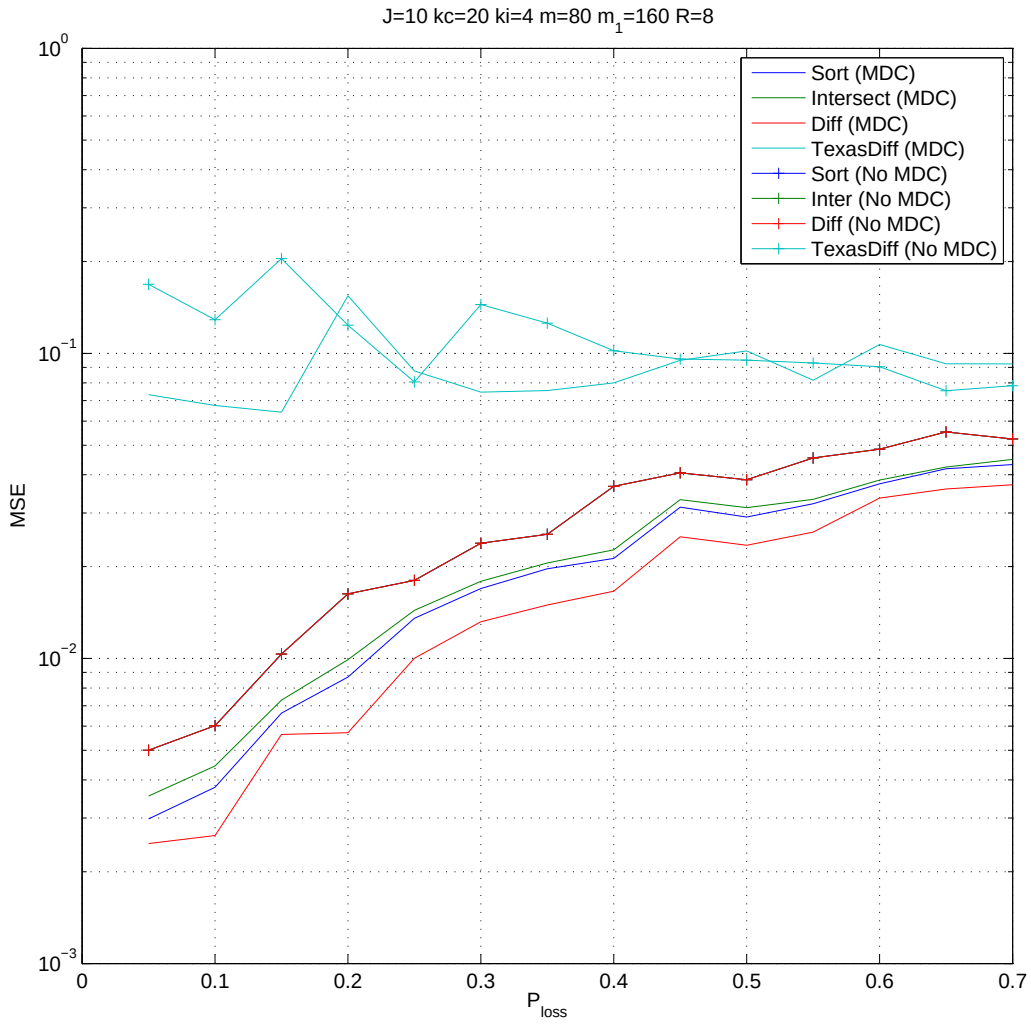

Figure 5.3. Mean MSE as function of p_{loss}. $J = 10$, $k_c = 20$, $k_I = 4$, $m = 80$, $m_1 = 160$

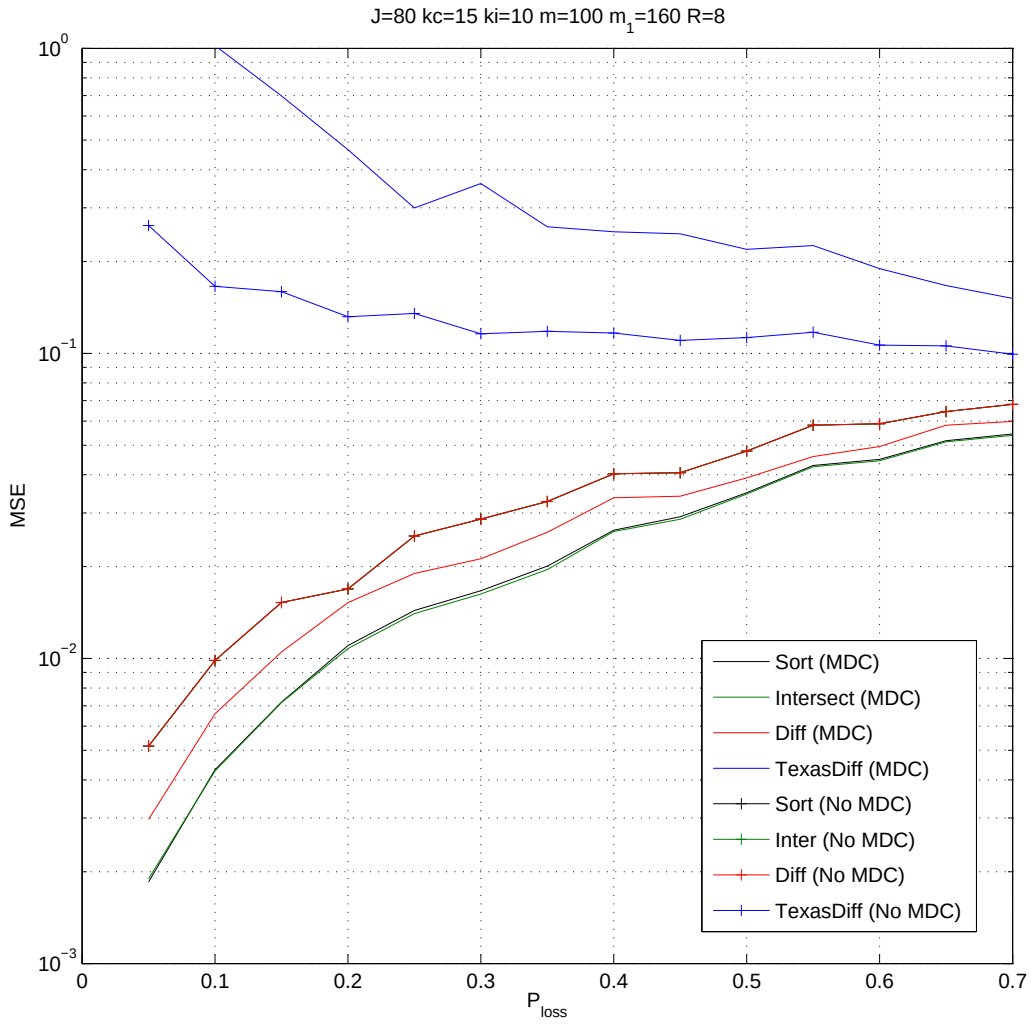

Figure 5.4. Mean MSE as function of p_{loss}. $J = 80$, $k_c = 15$, $k_I = 10$, $m = 100$, $m_1 = 160$

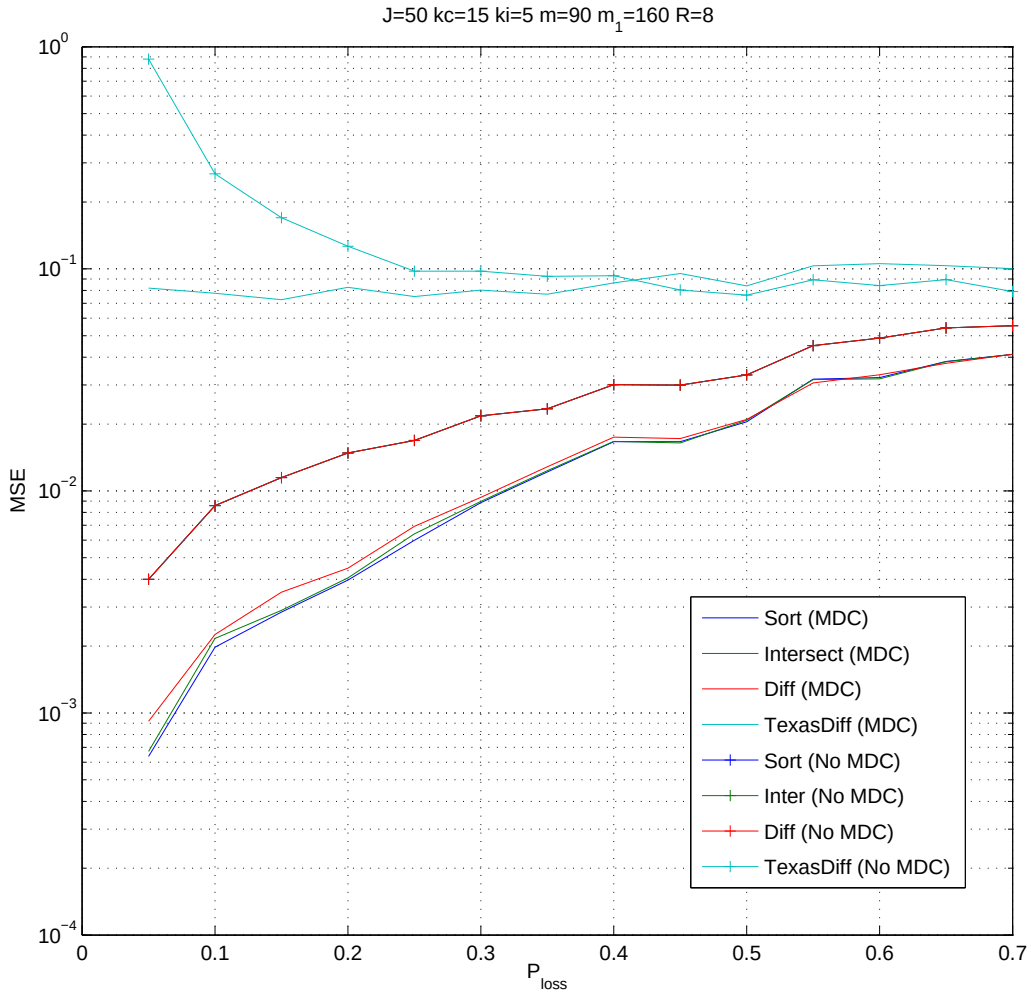

Figure 5.5. Mean MSE as function of p_{loss}. $J = 50$, $k_c = 15$, $k_I = 5$, $m = 90$, $m_1 = 160$

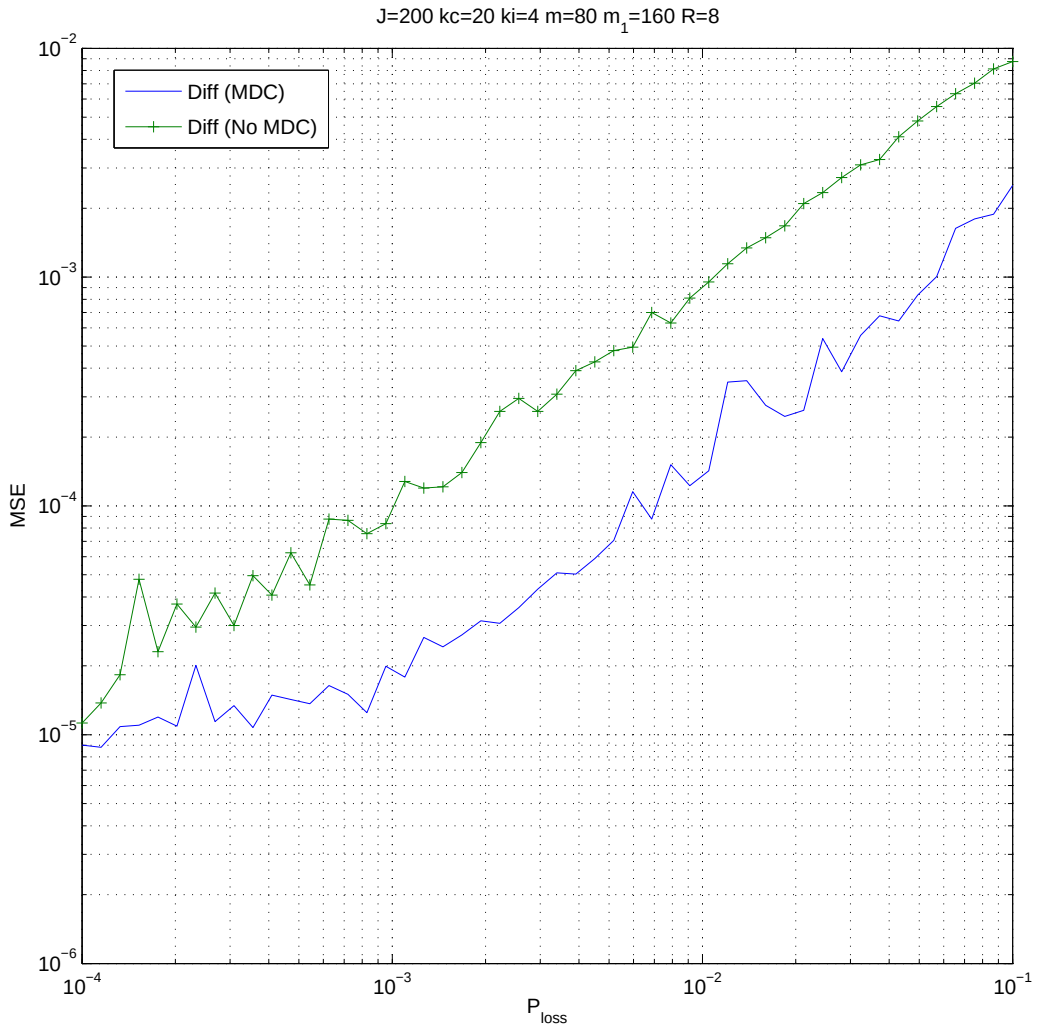

Figure 5.6. Mean MSE as function of p_{loss}. $J = 200$, $k_c = 20$, $k_I = 4$, $m = 80$, $m_1 = 160$

CHAPTER 6

CONCLUSIONS

Compressed sensing is a new technique to acquire a signal in a compressed fashion. It is able to go beyond the traditional Shannon-Nyquist paradigm and reduce the number of measurements that are needed in order to reconstruct the signal exactly, by exploiting the sparse nature of the signal. Since many signals of interest (such as natural images or sensors data, etc.) are sparse or compressible in some domain, the theory of compressed sensing has recently gathered much attention. Many interesting results concerning the validity of the theory and useful applications have already been established and intense research is focused on expanding the theory and adopting it for real systems and applications.

In the previous chapters we tried to analyse techniques to improve the robustness of systems adopting compressed sensing to errors, or channel losses, by means of multiple description coding. We also considered the case of a distributed application of compressed sensing, e.g. in a sensor network, that is able to leverage what is known as joint sparsity to improve coding efficiency and multiple description coding to improve its error resilience. Aiming at understanding if compressed sensing reconstruction could benefit from it, we also considered the topic of consistent reconstruction. Experiments showed what the theory suggested, that is enforcing the recovered signal to have measurements consistent with the quantization intervals, improves the quality of the reconstruction only in a limited number of cases, such as when having many more measurements than what is needed to perform a good-quality reconstruction. We discussed

that the benefits seem to be small and limited to few cases, so the increase in complexity that consistent reconstruction requires may not be worth it.

CS-SPLIT is the multiple-description scheme that has been proposed to increase the resilience of compressed sensing to losses. Its main idea is creating two descriptions by identifying two sets of measurements. Each description is decodable by itself, but if both descriptions are available we are able to recover the signal from a greater number of measurements, thus improving the quality of the reconstruction. The scheme is extremely simple thanks to the democracy of the measurements, which allows to simply form the descriptions by splitting the original vector of measurements into two halves. We have seen how it may be possible to packetize each description in a separate packet and ensure that some reconstruction is always possible, unless the network loses both the descriptions. It has been experimentally verified that this scheme offers a gain of performance with respect to a scheme that doesn't use multiple descriptions, when the communication channel tends to be unreliable. In principle, many techniques have been studied in the literature to create multiple descriptions, but before the advent of compressed sensing. One of the most popular is the multiple description scalar quantizer (MDSQ) that creates two descriptions during the quantization process. We experimentally verified that applying a MDSQ to the measurements (scheme that we referred to as CS-MDSQ) is less efficient than using CS-SPLIT, that is able to leverage the democracy property of measurements. We also characterized CS-SPLIT and CS-MDSQ from an analytic perspective by deriving bounds to the rate-distortion function for both the methods. Although those bounds do not allow to explicitly claim that one method is clearly better than the other, they do give some intuition on their

possible performance. In particular, we noticed how CS-SPLIT may outperform CS-MDSQ when oracle-assisted recovery from measurements is adopted. Finally, the complexity balance is so much in favour of CS-SPLIT to make it much more interesting than CS-MDSQ.

Low complexity is one the key requirements of sensor networks, that operate on small, low-cost, battery-powered devices. Compressed sensing is a promising technology in this field as it well suits the low complexity constraint (on the sensor side) and the underlying sparsity of the data to be sensed. When considering a distributed scenario with many sensors, measurements can be both intra-correlated and inter-correlated. Joint sparsity models allow to exploit inter-correlation to reduce the overall number of measurements to be taken. We proposed two new algorithms for joint reconstruction from measurements of a JSM-1 ensemble: *Difference* and *Texas Difference*. Those algorithms are able to improve the quality of the reconstruction when few measurements are available with respect to other existing algorithms. It is observed that this is of particular interest when CS-SPLIT is also used. In fact, each description must be decodable by itself and this requires that $\frac{m}{2}$ measurements should be enough to perform recovery, but we also don't want to be forced to take too many measurements overall. Hence, it may happen that $\frac{m}{2}$ measurements are barely enough to provide a decent quality for the side decoders. However, our joint reconstruction algorithms can improve the quality in exactly that situation.

CITED LITERATURE

1. Duarte, M. F., Davenport, M. A., Takhar, D., Laska, J. N., Sun, T., Kelly, K. F., and Baraniuk, R. G.: *Single-pixel imaging via compressive sampling.* IEEE Signal Processing Magazine , 25(2):83–91, March 2008.

2. Davenport, M. A., Duarte, M. F., Eldar, Y. C., and Kutyniok, G.: *Introduction to compressed sensing.* In Compressed Sensing: Theory and Applications . Cambridge University Press, 2012.

3. Fornasier, M. and Rauhut, H.: *Compressive sensing.* In Handbook of Mathematical Methods in Imaging . Springer, 2011.

4. Candes, E. J. and Tao, T.: *Decoding by linear programming.* IEEE Trans. on Information Theory , 51(12):4203–4215, December 2005.

5. Cormen, T. H., Leiserson, C. E., Riverst, R. L., and Stein, C.: *Introduction to Algorithms* . MIT Press, 2001.

6. Claerbout, J. and Muir, F.: *Robust modeling with erratic data.* Geophysics Magazine , 38(5):826–844, October 1973.

7. Santosa, F. and Symes, W.: *Linear inversion of band-limited reflection seismograms.* SIAM J. Sci. Statist. Comput. , 7(4):1307–1330, 1986.

8. Cai, T. T. and Wang, L.: *Orthogonal matching pursuit for sparse signal recovery.* IEEE Transactions on Information Theory , 57(7):4680–4688, July 2011.

9. Indyk, P.: *Explicit constructions for compressed sensing of sparse signals.* Symp. on Discrete Algorithms , 2008.

10. Jacques, L., Hammond, D., and Fadili, M. J.: *Dequantizing compressed sensing : When oversampling and non-gaussian constraints combine.* IEEE Transactions on Information Theory , 57(1):559–571, January 2011.

CITED LITERATURE (continued)

11. Dai, W., Pham, H. V., , and Milenkovic, O.: *Information theoretical and algorithmic approaches to quantized compressive sensing.* IEEE Transactions on Information Theory , 59(7):1857–1866, July 2011.

12. Boufounos, P. and Baraniuk, R.: *1-bit compressive sensing.* In Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on , pages 16 –21, march 2008.

13. van den Berg, E. and Friedlander, M. P.: *SPGL1: A solver for large-scale sparse reconstruction*, June 2007. http://www.cs.ubc.ca/labs/scl/spgl1.

14. Grant, M. and Boyd, S.: *CVX: Matlab software for disciplined convex programming*, version 1.21, April 2011. http://cvxr.com/cvx.

15. Goyal, V. K.: *Multiple description coding: Compression meets the network.* IEEE Signal Processing Magazine , 18(5):74–93, September 2001.

16. Ozarow, L.: *On a source coding problem with two channels and three receivers.* Bell Syst. Tech. J. , 59(10):1909–1921, December 1980.

17. Vaishampayan, V. A.: *Design of multiple description scalar quantizers.* IEEE Transactions on Information Theory , 39(3):821–834, May 1993.

18. Davenport, M., Laska, J., Treichler, J., and Baraniuk, R.: *The pros and cons of compressive sensing for wideband signal acquisition: Noise folding vs. dynamic range.* Arxiv preprint arXiv:1104.4842 , 2011.

19. Bajwa, W. U., Calderbank, R., and Jafarpour, S.: *Why gabor frames? two fundamental measures of coherence and their role in model selection*, 2010.

20. Proakis, J.: *Digital Communications* . McGraw-Hill Science/Engineering/Math, 2000.

21. Baron, D., Duarte, M. F., Wakin, M. B., Sarvotham, S., and Baraniuk, R. G.: *Distributed compressive sensing.* Preprint , Jan. 2009.

22. Schnelle, S., Laska, J., Hegde, C., Duarte, M., Davenport, M., and Baraniuk, R.: *Texas hold 'em algorithms for distributed compressive sensing.* In IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), 2010, pages 2886 –2889, march 2010.

CITED LITERATURE (continued)

23. Coluccia, G., Magli, E., Roumy, A., and Toto-Zarasoa, V.: *Lossy compression of distributed sparse sources: a practical scheme*. In 2011 European Signal Processing Conference (EUSIPCO-2011), Barcelona, Spain, September 2011.

24. Duarte, M., Wakin, M., Baron, D., and Baraniuk, R.: *Universal distributed sensing via random projections*. In The Fifth International Conference on Information Processing in Sensor Networks, 2006. IPSN 2006., pages 177 –185, 0-0 2006.

VITA

NAME: Diego Valsesia

EDUCATION: Laurea di I livello in Electronic and Computer Engineering, Politecnico di Torino, 2010

Laurea Magistrale in Telecommunications Engineering, Politecnico di Torino, 2012

Master of Science in Electrical and Computer Engineering, University of Illinois at Chicago, 2013