"GHEORGHE ASACHI"
TECHNICAL UNIVERSITY OF IAŞI

Faculty of Electronics, Telecommunications
and Information Technology

Contributions to signal analysis and processing using compressed sensing techniques

- DOCTORAL THESIS -

Supervisor:
Prof. univ. dr. Liviu Goraş

Candidate:
ing. Nicolae Cleju

IAŞI - 2012

UNIUNEA EUROPEANĂ
GUVERNUL ROMÂNIEI
MINISTERUL MUNCII, FAMILIEI ŞI PROTECŢIEI SOCIALE
AMPOSDRU
Fondul Social European
POSDRU 2007-2013
Instrumente Structurale
2007-2013
MINISTERUL EDUCAŢIEI CERCETĂRII TINERETULUI ŞI SPORTULUI
OIPOSDRU
UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI" DIN IAŞI

"GHEORGHE ASACHI"
TECHNICAL UNIVERSITY OF IAŞI

Faculty of Electronics, Telecommunications
and Information Technology

Contributions to signal analysis and processing using compressed sensing techniques

- Doctoral thesis -

Supervisor:
prof. univ. dr. Liviu Goraş

Candidate:
ing. Nicolae Cleju

IAŞI - 2012

UNIUNEA EUROPEANĂ
GUVERNUL ROMÂNIEI MINISTERUL MUNCII, FAMILIEI ŞI PROTECŢIEI SOCIALE AMPOSDRU
Fondul Social European POSDRU 2007-2013
Instrumente Structurale 2007-2013
MINISTERUL EDUCAŢIEI CERCETĂRII TINERETULUI ŞI SPORTULUI
OIPOSDRU
UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI" DIN IAŞI

Teza de doctorat a fost realizată cu sprijinul financiar al proiectului „Burse Doctorale pentru Performanța în Cercetare la Nivel European (EURODOC)".

Proiectul „Burse Doctorale pentru Performanța în Cercetare la Nivel European (EURODOC)", POSDRU/88/1.5/S/59410, ID 59410, este un proiect strategic care are ca obiectiv general „Dezvoltarea capitalului uman pentru cercetare prin programe doctorale pentru îmbunătățirea participării, creșterii atractivității şi motivației pentru cercetare. Dezvoltarea la nivel european a tinerilor cercetători care să adopte o abordare interdisciplinară în domeniul cercetării, dezvoltării şi inovării.".

Proiect finanțat în perioada 2009 - 2012.

Finanțare proiect: 18.943.804,97 RON

Beneficiar: Universitatea Tehnică "Gheorghe Asachi" din Iaşi

Partener: Universitatea „Babeş Bolyai" din Cluj-Napoca

Director proiect: Prof. univ. dr. ing. Mihaela-Luminița LUPU

Responsabil proiect partener: Prof. univ. dr. ing. Alexandru OZUNU

UNIUNEA EUROPEANĂ
GUVERNUL ROMÂNIEI
MINISTERUL MUNCII, FAMILIEI ŞI PROTECŢIEI SOCIALE
AMPOSDRU
Fondul Social European
POSDRU 2007-2013
Instrumente Structurale
2007-2013
MINISTERUL EDUCAŢIEI CERCETĂRII TINERETULUI ŞI SPORTULUI
OIPOSDRU
UNIVERSITATEA TEHNICĂ
"GHEORGHE ASACHI"
DIN IAŞI

Acknolwdgements

I would like to thank prof. Liviu Goraş for his help and support throughout these years. I am also grateful to all the people in the Signals, Circuits and Systems laboratory for the special and stimulating environment I worked in.

Special thanks go to prof. Mark Plumbley for accepting and overseeing my stage in the Centre for Digital Music at the Queen Mary, University of London. The time I spent in C4DM was an enriching experience.

Contents

Chapter 1

Introduction

In numerous digital applications, the typical processing chains starts with an acquisition stage, usually by sampling with a frequency as high as possible, immediately followed by compression, to reduce the dimensionality of the data and to eliminate redundancy in the acquired samples. A typical example is that of digital point-and-shoot photo cameras with internal JPEG compression: the image is spatially sampled with a very high density sensor chip, consisting nowadays of tens of millions of pixels, but immediately after this around 95% of the acquired samples are to be removed through JPEG compression, which preserves only the dominating coefficients resulting after the discrete cosine transform. This surely seems a waste of all this dense sampling power. This gives rise to the following question: if we know *a priori* that an image is compressible through some orthogonal transform, which holds for the vast majority of natural images, can we use this to reduce the number of samples that we need to acquire?

The answer to this question is given by the theory of *compressed sensing*. It shows that the number of samples needed to uniquely recover a signal is related to the "information content" of the signal, which can mean degrees of liberty, rate of innovation, number of non-zero coefficients etc. This can lead to significant dimensionality reduction when acquiring a signal and when post-processing it.

In this paper we focus on the digitized interpretation of the compressed sensing theory, where we consider signals that are already digital. If it is known *a priori* that an n-dimensional signal x has a decomposition with few significant coefficients in some known orthogonal basis (the locations and values of these coefficients are not known), compressed sensing theory shows that the signal is uniquely determined and can be recovered from a smaller m-dimensional vector of measurements. A *measurement* means the inner product of the signal with a vector known as *projection vector*. As long as the projection vectors

obey an incoherence property with the sparsity basis, the original signal can be found as the unique solution of a rigorously defined optimization problem. One of the major practical benefits is that the projection vectors can be random. Signal recovery amounts to solving the corresponding non-trivial optimization problem, using one of the many algorithms developed in the literature.

1.1 Motivation

One of the challenges of the future of signal processing is handling extremely large and high dimensional data. Up to now, compressed sensing seems to bring significant advantages in this respect, reducing the effective dimensionality by exploiting signal sparsity. As such, compressed sensing, as well as exploiting signal sparsity in general, has rapidly evolved to become a very large research area, with active research extending from analog signal acquisition to digital signal recovery and from complex theoretical analysis of the precise conditions for signal recovery to practical applications in various domains. As such, many issues have not yet been thoroughly analysed, numerous possible applications have been overlooked, and there are many practical issues that require consideration.

One of the interesting developments has been the extension of compressed sensing theory to non-orthogonal bases and overcomplete dictionaries. Although it can be seen as a natural extension of the orthogonal basis case, this introduces a new set problems: how do the intrinsic dependencies between the dictionary vectors ((known as *atoms*) affect the recovery process? how to design "good" projection vectors for overcomplete dictionaries? how to find good bases and overcomplete dictionaries for certain classes of signals in the first place? or even, can we use different a priori information instead of sparsity in some basis/dictionary?

This PhD thesis focuses on selected developments related to these promising extensions of compressed sensing and to the new challenges they bring, from a general practical-based perspective.

1.2 Thesis overview

The thesis is comprised of seven chapters, detailed bibliography and one appendix. Apart from this introduction, the rest of the thesis is structured as follows.

Chapter 2 contains a short introduction to the fundamentals of compressed sensing theory, which is the larger context of this thesis. We start with introducing the key

concepts of *sparsity* and *sparse* representations of signals. We discuss the central problem of compressed sensing, i.e. how to adequately recover sparse signals from a small number of measurements, as well as the multiple formulations of the reconstruction problem. A large part of the chapter is devoted to some of the most important conditions necessary and/or sufficient to guarantee accurate recovery. The aim is to introduce the reader to the basic results, without the burden of detailed proofs. In addition, we also present a few of the popular reconstruction and optimization algorithms that we use throughout the thesis.

Chapter 3 presents an alternative sparsity model known as *analysis* sparsity, that offers similar recovery guarantees as the classical *synthesis* sparsity model. We explore the relation between the two models, based on the idea that the analysis model can be thought of as a synthesis model with a set of additional constraints. As such, we develop a series of theorems proving that analysis-based signal recovery can be reformulated as a augmented version of synthesis-based sparse signal recovery. The practical advantage of this reformulation is that one can use existing recovery algorithms, designed solely for synthesis-based recovery, for the analysis-based recovery as well, as we show through practical simulations. The second part of the chapter consist of an experimental comparison of the two models, based on the above reformulation. The simulations reveal a fundamental difference between the two models in terms of the signals that are adequate to one or the other, as well as different sensitivity to noise and number of measurements.

Chapter 4 presents an innovative solution for finding improved acquisition matrices for signals that are sparse in non-orthogonal bases and overcomplete dictionaries. We first present three existing state-of-the-art algorithms and we propose modifications for improving each of them, based on the analysis of some unfavourable scenarios. We then propose a unified approach for finding an optimal acquisition matrix that encompasses all the three modified algorithms. We formulate the general problem as a rank-constrained nearest correlation matrix problem, i.e. finding the matrix of minimal distance from a given correlation matrix, subject to rank, semipositivity and unit-diagonal constraints. The modified versions of the three algorithms correspond to different ways of choosing a distance metric. Experimental results confirm superior performance and increased robustness, particularly for higher compression ratios and with dictionaries that are highly correlated, e.g. learned from a particular data set.

Chapter 5 presents a surprising result on using correlated projection vectors, i.e. random vectors drawn from a multivariate probability distribution with a non-unit covariance matrix. We develop a series of experiments that show that the covariance matrix

of the reconstruction errors obtained with a number of well-known recovery algorithms is heavily dependent on the covariance matrix of the random projection vectors: the two matrices have roughly the same eigenvectors, and we also find a rather precise relation between the eigenvalues of the two matrices, up to a constant factor. This means that one can effectively control the "shape" of the error distribution by properly designing the covariance of the random projection vectors. In this way, one can control the recovery accuracy along some directions in space at the expense of others. This opens the door for a number of interesting compressed sensing applications, scarcely analysed in the literature up to now. We validate our assumptions with two practical scenarios: recovering signals that are sparse in non-orthogonal bases, and recovering signals with unequal atom importance. In both cases, simulations show that using correlated projection vectors leads to significant improvements over the non-correlated case.

Chapter 6 evaluates the possibility of using the compressed sensing theory in the practical context of electrocardiographic (ECG) signal acquisition. We start with the preprocessing of ECG signals, based on segmentation into independent heart beats, and we investigate the performance of standard wavelet bases and custom learned bases in terms of sparse representations of the ECG segments. We evaluate the possibility of classifying the ECG segments into normal/pathological classes using solely the compressed random measurements. We also investigate recovery using dictionaries of different sizes and created with different algorithms. We propose a hybrid reconstruction method based on classifying the acquired measurements in the compressed space, followed by reconstruction with a dictionary dedicated to the particular signal class. The simulation results indicate superior reconstruction accuracy compared to the single dictionary scenario.

Finally, Chapter 7 summarizes the conclusions and reviews the main contributions of the thesis. An additional appendix contains further results for ECG signal classification in chapter 6.

Chapter 2

Fundamentals of sparse approximations and compressed sensing

This chapter provides a short introduction to the main theoretic results in the field of compressed sensing and sparsity in general. An alternative brief introduction can be found in [1].

2.1 Signals and sparse decompositions

In the widest sense, a signal x is called *sparse* if its ℓ_p norm has a small value, for some value of p between $0 \leq p \leq 1$. In signal processing, the ℓ_p norm of a signal $x \in \mathbb{R}^N$, denoted as $||x||_p$ or as $||x||_{\ell_p}$, is defined as:

$$||x||_p = \left(\sum_{i=1}^{N} x_i^p \right)^{\frac{1}{p}}. \tag{2.1}$$

For $p \geq 1$, the function defined in (2.1) obeys the three conditions imposed by the definition of a norm function of a vector space:

1. $||ax|| = a||x||$ (homogeneity)
2. $||x + y|| \leq ||x|| + ||y||$ (triangle inequality)
3. $||x|| = 0 \Leftrightarrow x = 0$

For $0 < p < 1$, (2.1) defines a function that does not obey the triangle inequality, and therefore is nor rigorously a norm function. However, these functions are widely used in

research literature and they are often referred to as norms, and therefore we keep the same convention and also refer to them as ℓ_p *norms*.

For $p = 0$, one uses a definition obtained as a limit case of (2.1) for $p \to 0$:

$$||x||_0 = \sum_i c_i, \text{ where } c_i = \begin{cases} 1 & x_i \neq 0 \\ 0 & x_i = 0 \end{cases} \tag{2.2}$$

Basically, (2.2) defines the ℓ_0 norm of a vector x as the number of non-zero coefficients. Similarly, this function does not obey the first two conditions in the definition of a norm, but we use the accepted convention in the literature and refer to it as the ℓ_0 *norm* nevertheless.

In general, the two most used values for p in practice are $p = 0$ and $p = 1$. In the strictest sense, the term *sparsity* implies a small value of the ℓ_0 norm of a signal. If $||x||_0 = k$, meaning that the signal has exactly k non-zero components, we say that x is k*-sparse*. In many cases, however, this definition is too strict, for example when analysing signals that are affected by noise, where it is virtually impossible for a coefficient to be rigorously zero. Therefore the definition is usually "relaxed" to ℓ_p norms with $0 < p \leq 1$, less rigorous but allowing a more robust analysis.

One observes that for $p < 1$ the ℓ_p "norms" do not satisfy the triangle inequality, which implies that any optimization problem requiring the minimization of the ℓ_p norm over a set of signals is non-convex, and as such is usually intractable. This kind of optimization problems are essential in compressed sensing theory, and therefore their non-convexity is a major disadvantage. On the other hand, the ℓ_1 norm satisfies this condition, and therefore minimizing the ℓ_1 norm is a convex optimization problem, with efficient solving algorithms available. As such, the ℓ_1 norm is usually regarded as an optimal tradeoff between the rigorous measure of sparsity (requiring p as close to 0 as possible) and the efficiency of solving algorithms (convex optimization for $p \geq 1$), and therefore is widely used both in theory and in practice.

In a larger context, sparsity can be defined with respect to a certain basis or overcomplete dictionary. Given a signal x from a vector space and a basis B for this space, we say that x is *sparse in the basis* B if the decomposition γ of x in the basis B is sparse. This definition is a generalization of the previous definitions, which constitute a special case with B being the canonical basis $B = I_n$. An overcomplete dictionary is a generalization of the concept of basis, an overcomplete dictionary D being composed of a set of N vectors which span the vector space S, but their number N is larger than the dimension of the space, $N > n$ (in literature, the elements of a basis or dictionary are usually called *atoms*). Any signal x therefore has an infinite number of possible decompositions with

respect to a dictionary D. We say that x is *sparse in the dictionary* D if at least one of the decompositions in D is sparse.

2.2 Compressed sensing

Let us consider the vector $x \in \mathbb{R}^n$ being a sparse vector in the basis defined by the matrix $B \in \mathbb{R}^{n \times n}$. The vector x can therefore be written as

$$x = B\gamma \tag{2.3}$$

where γ is the sparse decomposition vector of x in B.

Let us assume that we acquire a set of m measurements of x obtained by projecting the vector on a set of m measurement vectors. Considering these projection vectors arranged as the rows of an acquisition matrix $P \in \mathbb{R}^{m \times n}$, measuring (i.e. acquiring) the signal x is described by its multiplication with the matrix P:

$$y = Px = \underbrace{PB}_{A}\gamma = A\gamma \tag{2.4}$$

If we define the product of P and B as the *effective dictionary* $A = PB$, we reach the standard form of the compressed sensing problem:

$$y = A\gamma \tag{2.5}$$

Equation (2.5) states that the sparse vector γ is *acquired* with the matrix A.

The question that is central to the compressed sensing theory is the following: under which conditions is it possible to recover the n-dimensional sparse vector γ from the $m << n$ measurements y? If one can recover γ, one can immediately recover the original signal x through (2.3).

A more robust analysis must take into account the possibility of additive noise z overimposed on the measurements, and therefore the acquisition system becomes:

$$y = A\gamma + z \tag{2.6}$$

The above equations remain valid in the general case when instead of the basis B one has an overcomplete dictionary D composed of N atoms, $N > n$, the only difference being that in this case the dimensions of A are $m \times N$ and γ is N-dimensional vector.

2.3 Solutions and regularization

The equation system (2.5) is an undetermined system, as the matrix A is of size $m \times N$. As such, recovering the sparse decomposition vector γ is not possible without an additional regularizing term than ensures the uniqueness of the solution. This regularization term must reflect an *a priori* information known about the vector γ or about x.

Compressed sensing theory exploits the additional information that the signal is sparse in a known basis/dictionary. A fundamental result [2, 3] states that, if the decomposition γ is sufficiently sparse, then, under some conditions on the matrix A that are analysed in Section 2.4, the vector γ is *the sparsest solution in the set of all solutions of* (2.5). As such, one can recover γ by solving the following constrained optimization problem:

$$\hat{\gamma} = \arg\min_{\gamma} ||\gamma||_p, \text{ s.t. } y = A\gamma \qquad (P_p)$$

The problem (P_p) consists in finding the sparsest solution of $y = A\gamma$. For the two most widely used values of p, $p = 0$ and $p = 1$, equation (P_p) becomes (P_0) and (P_1), respectively:

$$\hat{\gamma} = \arg\min_{\gamma} ||\gamma||_0, \text{ s.t. } y = A\gamma \qquad (P_0)$$

$$\hat{\gamma} = \arg\min_{\gamma} ||\gamma||_1, \text{ s.t. } y = A\gamma \qquad (P_1)$$

The problem (P_0), that uses the ℓ_0 norm, is NP-hard [4], which means it has a non-polynomial solving complexity; NP-hard problems are considered virtually impossible to solve for moderate sizes. The second optimization problem (P_1), that uses the ℓ_1 norm for enforcing sparsity, is known in the literature by the name *Basis Pursuit* [5]. This is a convex optimization problem, that can be converted to a linear programming problem, which is a well studied optimization problem with many efficient solving algorithms available; we present it in more details in Section 2.5.1.

Both (P_0) and (P_1) require the constraint $y = A\gamma$, i.e. they require an exact decomposition of y. In practice, however, signals and measurements are always affected by noise, and a certain degree of error can always be tolerated. As such, a robust analysis must take into account the possibility of additive noise added to the measurement vector y, as in (2.6). In this case, the exact condition $y = A\gamma$ cannot be rigorously satisfied. A relaxation of this constraint that allows for an approximation error less than a certain tolerance level given by the estimated noise energy leads to a more robust form of the optimization problems:

$$\hat{\gamma} = \arg\min_{\gamma} ||\gamma||_0, \text{ s.t. } ||y - A\gamma|| \leq \epsilon \qquad (\hat{P^{\epsilon}}_0)$$

$$\hat{\gamma} = \arg\min_{\gamma} ||\gamma||_1, \text{ s.t. } ||y - A\gamma|| \leq \epsilon \qquad (\hat{P^{\epsilon}}_1)$$

The condition $||y - A\gamma|| \leq \epsilon$ can be relaxed even further by making it part of the minimizing goal function:

$$\hat{\gamma} = \arg\min_{\gamma} ||y - A\gamma|| + \lambda \cdot \|\gamma\|_0 \qquad (\hat{P^{\lambda}}_0)$$

$$\hat{\gamma} = \arg\min_{\gamma} ||y - A\gamma|| + \lambda \cdot \|\gamma\|_1 \qquad (\hat{P^{\lambda}}_1)$$

Equations $(\hat{P^{\lambda}}_0)$ and $(\hat{P^{\lambda}}_1)$ are unconstrained formulations of the optimization problems. The parameter λ controls the tradeoff between the approximation error, expressed by the first term of the goal function, and the sparsity of the desired solution, expressed by the second term.

Yet another alternative formulation, known as the *LASSO* algorithm [6], is given as:

$$\hat{\gamma} = \arg\min_{\gamma} ||y - A\gamma|| \; s.t. \; ||\gamma||_1 \leq \tau \qquad (P_1^{lasso})$$

The goal is to minimize a quadratic functional under the constraint that the ℓ_1 norm of the solution is bounded by a parameter τ. This problem can be reformulated a a double-sized quadratic programming problem, for which efficient solving algorithms are available [7].

All the above formulations represent different ways of expressing the same fundamental problem of finding the sparsest solution to an undetermined equation system, with various degrees of robustness against noise and inexact sparsity. Rigorous guarantees that the solution to some of these problems is close to the original sparse signal are given in Section 2.4.

2.4 Conditions for sparse signal recovery

In this section we present fundamental results regarding the conditions on the matrix A of (2.5) and (2.6) that guarantee the uniqueness of the sparse solution γ, and thus guarantee successful recovery by solving the optimization problems presented above.

2.4.1 The spark of a matrix

One of the first recovery conditions relies on the concept of the *spark* of a matrix, introduced in [2]:

Definition 1 (**[2]**). *Consider the matrix $A \in \mathbb{R}^{m \times n}$. The* spark σ *of A is the minimum number of columns of A that are linearly dependent*

The spark of a matrix is important because of the following: if some signal has two sparse decompositions of sparsity k_1 and k_2 in the dictionary A, then their difference is a vector of sparsity at most $k1 + k2$ that lives in the nullspace of A, meaning that at most $k_1 + k_2$ columns of A form a linearly dependent set. Therefore, a high enough value of $\sigma > k_1 + k_2$ prevents this, effectively guaranteeing a uniquely sparse solution.

The following theorem rigorously states a uniqueness guarantee for (P_0):

Theorem 2 ([2]). *Let γ be a sparse vector with $||\gamma||_0 = k$, acquired with a matrix A as in (2.5). Let σ be the spark of A. If $k < \sigma/2$, then γ is the unique solution to the optimization problem (P_0).*

Proof. The proof is obvious: if (P_0) had a different (even sparser) solution with sparsity $k' \leq k$, than the difference of the two solutions would yield a vector of sparsity $(k'+k) < \sigma$ living in the nullspace of A. This, however, means having a linear dependent set of columns of A less than the spark of A, which contradicts the definition of the spark. □

Unfortunately, computing the spark of a matrix is known to be combinatorial and thus NP-hard [8], which limits the practical utility of the spark.

2.4.2 Restricted Isometry Property (RIP)

One of the main research directions is based on the study of some properties of restricted isometry of the matrix A [5, 9, 10].

Definition 3 ([5]). *Consider the matrix $A \in \mathbb{R}^{m \times n}$ having as columns the vectors $a_i \in \mathbb{R}^m$. For every integer number k with $1 \leq k \leq n$ let us define the Restricted Isometry Constant (RIC) of order k, with notation δ_k, as the smallest real value satisfying:*

$$(1 - \delta_k) \, ||\gamma||_2^2 \leq ||A\gamma||_2^2 \leq (1 + \delta_k) \, ||\gamma||_2^2 \tag{2.7}$$

for all vectors γ having at most k non-zero coefficients (k-sparse).

Let T be the indices of the non-zero coefficients of γ, and A_T the matrix composed only of the columns of A from the set T. The condition (2.7) expresses that all the sub-matrices A_T composed of at most $|T|$ columns of A behave close to orthonormal matrices. One is reminded that satisfying the isometry condition $||x||_2 = ||Ax||_2$, $\forall x$, is a necessary and sufficient condition for a matrix A to be orthonormal. Instead of rigorous equality, (2.7) allows a tolerance of $\pm\delta_k$, hence the name "*restricted* isometry". A smaller value of δ_k implies that the matrices A_T behave closer to orthonormal matrices.

One property that follows naturally from the definition is that the sequence of RIC constants δ_k is monotonously increasing, $\delta_k \leq \delta_{k+1}$. If the values of the δ_k constants are smaller, it is easier to recover sparse signals, as shown in the following theorems.

Theorem 4 ([**5, 10**])**.** *Let γ be a sparse vector with $||\gamma||_0 = k$, acquired with a matrix A as in* (2.5)*. If the constant δ_{2k} satisfies $\delta_{2k} < 1$, then the optimization problem (P_0) has a unique solution and that solution is γ*

Proof. Let $\delta_{2k} < 1$, and let us suppose that the solution $\hat{\gamma}$ of the problem (P_0) is actually different from γ. As a consequence, the sparsity of the solution $\hat{\gamma}$ is at most equal to the sparsity of γ, $||\hat{\gamma}||_0 \leq k$, because (P_0) searches for the minimum ℓ_0 norm solution. Since both $\hat{\gamma}$ and γ are solutions of the system in (2.5), it follows that their difference is in the nullspace of the matrix A:

$$0 = A\left(\hat{\gamma} - \gamma\right). \tag{2.8}$$

But $\hat{\gamma} - \gamma$ is a vector of sparsity at most equal to $2k$, since it is the difference of two at most k-sparse vectors. The definition of the RIC constant (2.7) implies that $\delta_{2k} \geq 1$, which contradicts our initial assumption. As such, the solution of the problem is unique and it is $\hat{\gamma} = \gamma$ □

Theorem 4 shows that one can recover a k-sparse signal γ compressively sensed through an acquisition matrix A if A satisfies $\delta_{2k} < 1$. In this case we say that the matrix A satisfies the *Restricted Isometry Property* (RIP). Intuitively, the RIP property necessitates that the columns of A are rather dissimilar to each other: indeed, if columns would be similar, the sub-matrices A_T would be far from orthogonal and thus would have large values of the RIC constants.

However, the optimization problem (P_0) is NP-hard; in addition, real signals are affected by noise and thus the sparsity condition $||\gamma||_0 = k$ is never rigorously fulfilled in practice. The following theorems establish sufficient recovery conditions when using the ℓ_1 norm.

Theorem 5 ([**10**])**.** *Let γ be a signal, and let γ_k be a signal containing only the largest k of its coefficients (in absolute value), the rest of the coefficients being zero. Let γ be acquired through a matrix A as in* (2.5)*. If δ_{2k} satisfies $\delta_{2k} < \sqrt{2} - 1$, then the solution $\hat{\gamma}$ of the optimization problem (P_1) satisfies the following:*

$$||\hat{\gamma} - \gamma||_1 \leq C_0||\gamma - \gamma_k||_1 \tag{2.9}$$

and

$$||\hat{\gamma} - \gamma||_2 \leq C_0 k^{-1/2}||\gamma - \gamma_k||_1 \tag{2.10}$$

Proof. The inequality (2.10) is a special case of the following theorem 6, for the noiseless case. The inequality (2.9) derives from the same proof. □

The following theorem extends the analysis to noise measurements as well.

Theorem 6 ([10]). *Let γ be a signal, and let γ_k be a signal containing only the largest k of its coefficients (in absolute value), the rest of the coefficients being zero. Let γ be acquired with a matrix A followed by additive noise z over the measurements, as in* (2.6). *If δ_{2k} satisfies $\delta_{2k} < \sqrt{2}-1$ then, for $||z||_2 \leq \epsilon$, the solution $\hat{\gamma}$ of the optimization problem $(\hat{P^{\epsilon}}_1)$ satisfies:*

$$||\hat{\gamma} - \gamma||_2 \leq C_0 k^{-1/2} ||\gamma - \gamma_k||_1 + C_1 \epsilon \tag{2.11}$$

Proof. The proof is found in [10] and we do not include it here in the interest of brevity. □

If γ is exactly k-sparse, then $\gamma = \gamma_k$ and Theorems 6 shows that one can recover γ up to a precision depending linearly on the energy of the noise. In noiseless conditions, the recovery is perfect.

Theorems 4, 5 and 6 prove the fact that a small value of the RIC constant guarantees accurate recovery of a sufficiently sparse vector by solving the optimization problems (P_0), (P_1) and $(\hat{P^{\epsilon}}_1)$, respectively. Thus, these theorems constitute a central part of the foundations of compressed sensing theory.

Unfortunately, finding the precise value of the RIP constant δ is also proven to be a NP-hard problem [8]. This prevents a direct application in practice of the RIP guarantees for general deterministic matrices.

2.4.3 The Null Space Property (NSP)

Another interesting property, related to the RIP property, is the *Null Space Property* (NSP) introduced in [11].

Definition 7 ([11]). *Consider the matrix $A \in \mathbb{R}^{m \times n}$ and let $\mathcal{N}(A)$ be the nullspace A. We say that A satisfies the* Null Space Property *(NSP) of order k with the constant C in ℓ_1 if $\forall h \in \mathcal{N}(A)$ and for all sets of coefficients $T \subseteq 1, 2...n$ with $card(T) \leq k$, the following equation holds:*

$$||h_T||_1 \leq C||h_{T^c}||_1 \tag{2.12}$$

where h_T contains only the coefficients in the set T (the other being zero) and h_{T^c} contains only the coefficients not belonging to the set T.

We typically consider the minimum value of C that satisfies (2.12). The definition (7) says that all vectors $h \in \mathcal{N}(A)$ in the nullspace of A have low sparsity, their energy being spread rather uniformly over the coefficients instead of being concentrated in a few coefficients. This is the opposite of sparsity: for a $k-$sparse vector h all the energy is concentrated in k coefficients, and thus if we include the locations of all non-zero coefficients in T we obtain $||h_T||_1 > 0$ and $||h_{T^c}||_1 = 0$, implying a constant C going to infinity. On the contrary, a vector h that has equal coefficients h_i has a minimal constant C, irrespective of how the set T is chosen. Thus, a low value of C in (2.12) implies we are closer to the latter case.

Having the NSP with a constant $C < 1$ is a necessary and sufficient condition for the success of P_1:

Theorem 8 ([11, 1]). *Consider the matrix $A \in \mathbb{R}^{m \times n}$. Let γ be a k-sparse signal acquired with A as in (2.5). Then γ is the solution of (P_1) for all k-sparse signals γ if and only if A satisfies the NSP with a constant $C < 1$.*

We point out that the NSP is a necessary and sufficient condition for sparse recovery via ℓ_1 minimization, whereas the RIP is only a sufficient condition, but not necessary. As such, having the RIP implies having the NSP, as shown in Section 2.4.5.

Unfortunately, finding the precise value of the NSP constant for a given matrix A is also NP-hard, as it was recently proven in [8].

2.4.4 Mutual coherence

An alternative direction in studying the conditions guaranteeing successful recovery of compressively sensed k-sparse signals is based on the *mutual coherence* of the matrix A from (2.5), [12, 3].

Definition 9. *Consider a matrix $A \in \mathbb{R}^{n \times N}$ with columns a_i. The* mutual coherence *μ of A is the maximum absolute value of the correlation between any two distinct normalized columns a_i and a_j of the matrix.*

$$\mu = \underset{i \neq j}{\operatorname{argmax}} \frac{\langle a_i, a_j \rangle}{||a_i||||a_j||} \tag{2.13}$$

The smaller the mutual coherence is, the more orthogonal are the columns of A (for an orthogonal matrix $\mu = 0$), and the more useful is the matrix in recovering sparse signals, as shown in the following theorems.

Theorem 10. *[[2, 13, 12]] Consider a matrix $A \in \mathbb{R}^{n \times N}$ with mutual coherence μ and a vector γ with $||\gamma||_0 = k$, in a noiseless compressed sensing setup as in (2.5). If condition (2.14) is true:*

$$||\gamma||_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(D)}\right) \tag{2.14}$$

then the following hold:

1. *γ is the sparsest decomposition of x in D, i.e. it is the solution of (P_0).*
2. *γ is recoverable using ℓ_1 minimization, i.e. it is also the solution of (P_1).*
3. *γ is recoverable using Orthogonal Matching Pursuit [14, 15].*

For the noisy case, the following theorem establishes a similar result:

Theorem 11 ([3]). *Consider a matrix $A \in \mathbb{R}^{n \times N}$ with mutual coherence μ and a vector γ with $||\gamma||_0 = k$, in a noisy compressed sensing setup as in (2.6), with $||z||_2 \leq e$. Then:*

1. *If $k \leq \frac{1}{2}\left(1 + \frac{1}{\mu}\right)$, then the vector $\hat{\gamma}$ obtained as the solution of the optimization problem $(\hat{P^{\epsilon}}_0)$ satisfies:*

$$||\hat{\gamma} - \gamma||_2^2 \leq \frac{(\epsilon + e)^2}{1 - \mu(2k-1)}, \quad \forall \epsilon \geq e > 0 \tag{2.15}$$

2. *If $k \leq \frac{1}{4}\left(1 + \frac{1}{\mu}\right)$, then the vector $\hat{\gamma}$ obtained as the solution of the optimization problem $(\hat{P^{\epsilon}}_1)$ satisfies:*

$$||\hat{\gamma} - \gamma||_2^2 \leq \frac{(\epsilon + e)^2}{1 - \mu(4k-1)}, \quad \forall \epsilon \geq e > 0 \tag{2.16}$$

3. *If $k \leq \frac{1}{2}\left(1 + \frac{1}{\mu}\right) - \frac{1}{\mu} \cdot \frac{e}{|\gamma_k|}$, where γ_k is the smallest of the k non-zero coefficients of γ, then the vector $\hat{\gamma}$ obtained by the Orthogonal Matching Pursuit (OMP) [14, 15] algorithm satisfies:*

$$||\hat{\gamma} - \gamma||_2^2 \leq \frac{e^2}{1 - \mu(k-1)} \tag{2.17}$$

In the above inequalities, ϵ is the accepted tolerance in the optimization problems $(\hat{P^{\epsilon}}_0)$ and $(\hat{P^{\epsilon}}_1)$, and e is the noise energy.

Theorems 10 and 11 imply that small values of the mutual coherence are beneficial, as they relax the upper bound of the sparsity of the vectors that are guaranteed to be perfectly recovered.

We point out that the mutual coherence of a dictionary is a sufficient condition that is very easy to compute, contrary to the spark, RIP and NSP. As such, it provides significant benefit in practice.

2.4.5 Relation between spark, RIP, NSP and mutual coherence

Some known relations between the above conditions and constants are summarized in Theorem 12 from [1].

Theorem 12 ([1]). *Consider a matrix $A \in \mathbb{R}^{n \times N}$ with spark σ, RIP constants δ_k, NSP constant C and mutual coherence μ. Then the following relations hold:*

1. $$\sigma \geq 1 + \frac{1}{\mu}$$

2. *A satisfies the RIP of order k with*
$$\delta_k = k\mu, \forall k < \frac{1}{\mu}$$

3. *Suppose A satisfies the RIP of order $2k$ with $\delta_{2k} < \sqrt{2} - 1$. Then A satisfies the NSP of order $2k$ with constant*
$$C = \frac{\sqrt{2}\delta_{2k}}{1 - (1 + \sqrt{2})\delta_{2k}}.$$

2.4.6 Random matrices

Considering a given general matrix A, computing the spark or the RIP or NSP constants are NP-hard problems [8], and thus directly using these properties in practice to check whether an acquisition matrix is "good" for compressed sensing is not possible. However, a series of fundamental results have shown that a random matrix in which each element is an i.i.d random variable with normal, Bernoulli or other probability distributions, has a very high chance of satisfying the RIP if the number of rows is large enough.

The foundation of these results is the *concentration of measure* phenomenon, consisting in the fact that, given a fixed vector $x \in \mathbb{R}^n$, the ℓ_2 norm of its projections on random subspaces is a random variable with the values strongly concentrated around the value $||x||_2$. The concentration of measure phenomenon is well known in mathematics. For rigorousness, we use the definition 13 used in [16].

Definition 13. *Let r be a random variable in the space Ω distributed with the probability distribution ρ. We construct a random matrix Φ of size $m \times n$ composed of independent realisations of the random variable r. We say that the probability density ρ satisfies the concentration of measure inequality if, for any $x \in \mathbb{R}^n$ we have:*

$$P\left(|\frac{n}{m}||\Phi x||_2^2 - ||x||_2^2| \geq \epsilon||x||_2^2\right) \leq 2e^{-mc_0(\epsilon)},\ 0 < \epsilon < 1, \tag{2.18}$$

where the probability is taken for the set of all possible matrices Φ *of size* $m \times n$*, and* $c_0(\epsilon)$ *is a constant depending only on* ϵ *such that* $c_0(\epsilon) > 0,\ \forall \epsilon \in (0,1)$*.*

Definition 13 implies that, for any random matrix Φ composed of independent realisations of a random variable distributed with the probability law ρ, the projection of any fixed vector x on the subspace defined by Φ, Φx, has the norm concentrated around the initial value of the norm of x, i.e. the probability that the resulting norm varies by a factor larger than $\pm\epsilon$ is upper bounded by an exponential depending on ϵ. The factor $\frac{n}{m}$ is only needed to normalize the value of the norm, as the norm ℓ_2 of Φx is defined on the vector space $\mathbb{R}^m$ whereas the norm ℓ_2 for x is defined in $\mathbb{R}^n$.

Among the probability distributions that satisfy concentration of measure inequalities we have the normal distribution and the Bernoulli distribution, as shown in the Theorem 14 from [17].

Theorem 14 ([**17**])**.** *Let* R *be a matrix of size* $m \times n$ *composed of independent random variables* r_{ij}*, distributed with one of the following distributions:*

$$r_{ij} = \mathcal{N}(0,1)$$

$$r_{ij} = \begin{cases} +1 & \textit{with probability } 1/2 \\ -1 & \textit{with probability } 1/2 \end{cases}$$

$$r_{ij} = \begin{cases} +1 & \textit{with probability } 1/6 \\ 0 & \textit{with probability } 2/3 \\ -1 & \textit{with probability } 1/6 \end{cases}$$

For $\forall \epsilon > 0$ *and for any vector* $v \in \mathbb{R}^n$ *the following inequalities hold:*

$$P\left(||Rv||_2^2 \leq (1+\epsilon)k/n\right) < e^{-\frac{m}{2}(\epsilon^2/2 - \epsilon^3/3)} \tag{2.19}$$

$$P\left(||Rv||_2^2 \leq (1-\epsilon)k/n\right) < e^{-\frac{m}{2}(\epsilon^2/2 - \epsilon^3/3)} \tag{2.20}$$

For the three distributions above, concentration of measure is satisfied with $c_0(\epsilon) = \frac{\epsilon^2/2 - \epsilon^3/3}{2}$.

For random matrices that have as elements independent realisations of a random variable with a probability distribution that satisfies concentration of measure, it can be shown that they obey the RIP with high probability. This is proven in Theorem 15 from [16].

Theorem 15 ([16]). *Given m, n and δ with $0 < \delta < 1$, if the probability distribution that generates the matrices Φ satisfies the concentration of measure inequality* (2.18)*, then there exist the constants c_1 and c_2 depending only on δ such that Φ satisfies the RIP with the chosen constant δ for all $k < c_1 m / \log(n/k)$ with probability $\geq 1 - e^{-c_2 m}$.*

Theorem 15 shows that a random matrix created from a distribution satisfying concentration of measure allows, with high probability, recovering sufficiently sparse signals. We can rewrite $k < c_1 m / \log(n/k)$ in the form $m > 1/c_1 \cdot k \log(n/k)$, which means that the number of necessary projections (i.e. rows of the acquisition matrix) for recovering k-sparse signals is of order $\mathcal{O}(k \log(N/k))$. Thus, we can recover a k-sparse signal from a measurement vector $\mathcal{O}(k \log(N/k))$ long. This is essentially a tight bound, since $k \log(N/k)$ is the minimum memory requirement if the locations of the non-zero coefficients would be known in advance.

In practice, the probability of successful recovery of k-sparse signals as a function of m, n and k via ℓ_1 minimization has been analysed in [18] using extended numerical experiments. Results show a surprisingly sharp transition between the parameter domain in which successful recovery is very likely and the domain where it is virtually impossible, at least in the asymptotic case when the signal dimensions are very large (in finite dimensional case the transition is less sharp). As the borders of the domain of successful recovery are rather sharp, one can estimate with a certain degree of accuracy the required number of measurements, which is very useful in practical applications.

Finally, let us note that random projections have two appealing characteristics in practical applications: they are (i) *universal* are (ii) *democratic*. Universality refers to the fact that a random projection matrix is, with high probability, incoherent with *any* fixed orthogonality basis (alternatively, any fixed orthogonal transformation of a random projection matrix will likely retain its characteristics); as such, it can be used with signals that are sparse in any fixed basis, even if this basis is unknown at acquisition time. Democracy [19] refers to the fact that each measurement carries roughly the same amount of information, and the measurements are interchangeable and replaceable with any other random measurements. This is similar in spirit with LT erasure codes [20] (randomly combining symbols for protection against packet losses on the transmission channel) and random network coding [21, 22, 23] (randomly combining data packets for increased throughput in multicast networks).

2.5 Reconstruction algorithms

In this section we briefly present some of the most used reconstruction algorithms in the literature for finding sparse solutions of undetermined equation systems. These algorithms solve the various formulations of the optimization problems (P_0), (P_1), $(\hat{P^{\epsilon}}_0)$, $(\hat{P^{\epsilon}}_1)$, $(\hat{P^{\lambda}}_0)$ and $(\hat{P^{\lambda}}_1)$.

The problems that imply minimizing the ℓ_0 norm are NP-hard, meaning that there is no known algorithm to solve them in polynomial time. Basically, they are considered intractable problems. impossible to solve for moderate sizes of interest. As such, finding the optimal solution of the minimization problems (P_0), $(\hat{P^{\epsilon}}_0)$ and $(\hat{P^{\lambda}}_0)$ is virtually impossible in general. Alternatively, there are a number of approaches that find a local optima. Besides the ℓ_0 norm, minimization problems involving ℓ_p norms with $0 < p < 1$ are also non-convex optimization problem, and as such are also very difficult to solve. On the contrary, minimizing the ℓ_p norm with $p \geq 1$ is a convex optimization problem. with only one local optimum, i.e. the global optimum. As such, these problems are much easier to solve.

2.5.1 Linear programming (Basis Pursuit)

Linear programming problems are a much studied class of optimization problems that require minimization of a linear function subject to linear equality and inequality constraints. The standard form is [24, page 146]:

$$d^* = \arg\min_{d} \sum_{p=0}^{N-1} w_p \cdot d_p \text{ with } c = P \cdot d \text{ and } d_i \geq 0, \tag{2.21}$$

but other equivalent formulations are possible. A linear programming problem in standard form requires finding the vector d, with non-negative elements, that minimizes the scalar product with a vector of weights w, among all the vectors that satisfy the linear constraint $c = P \cdot d$.

In mathematical literature there are well-known algorithms for solving these kind of problems (*simplex*, *primal-dual* or *interior point* algorithms) and, as such, we will not enter the details of their functioning.

In the case of compressed sensing, the optimization problem (P_1) (known by the name of *Basis Pursuit*) can be reformulated as an equivalent linear programming problem (2.21), of double size, as follows [25, chapter 12]: we introduce the variables $u_p \geq 0$ and $v_p \geq 0$, such that $y = u - v$, and we use the definitions (2.22) (square parentheses denote

concatenation).

$$\begin{aligned} P &= [A, -A] \\ w_p &= 1, \forall p \\ d &= \begin{bmatrix} u \\ p \end{bmatrix} \\ c &= y \end{aligned} \tag{2.22}$$

With the parameters defined this way, one solves the linear programming problem (2.21) with one of the available algorithms in the literature. The solution $\hat{\gamma}$ of the original problem is obtained as $\hat{\gamma} = u^* - v^*$, where u^* is the first half of the vector d^*, and v^* is the second half. The vector $\hat{\gamma}$ satisfies the condition $y = A \cdot \hat{\gamma}$ since $y = c = P \cdot d^* = A \cdot u^* - A \cdot v^* = A\,(u^* - v^*) = A \cdot \hat{\gamma}$

2.5.2 The Matching Pursuit family

One important family of algorithms widely used in practice for the problems $(\hat{P^{\epsilon}}_0)$ and $(\hat{P^{\epsilon}}_1)$ is the Matching Pursuit family of algorithms, consisting of Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), Batch OMP, Compressive Sampling Matching Pursuit (CoSaMP) et.al. These algorithms share the same heuristic approach for the problem of finding a sparse decomposition of a vector in a dictionary. The basic idea is to have a greedy search, at every iteration, for the best atom from the dictionary. Without guaranteeing in general that the rigorous optimal solution is found, these algorithms typically succeed in finding a sufficiently sparse solution with a reduced algorithm complexity, compared with other methods.

Every algorithm keeps a list of the coefficients $\gamma_i, i = 1, ...N$ corresponding to the atoms a_i of the dictionary A. The atoms that are not used in the decomposition have a corresponding coefficient equal to $\gamma_i = 0$. The approximation error at every iteration k is called the *residual* at iteration k:

$$r^{(k)} = y - A\gamma^{(k)}. \tag{2.23}$$

At every iteration the residual is reduced by adding one or more atoms of A in the current decomposition. Various algorithms differ in the rule used for selecting atoms and in setting their corresponding coefficients.

The simplest, and also the least effective method, is known as Matching Pursuit (MP) [26]. At every iteration, it chooses an atom which minimizes the current residual: it computes the correlation between the current residual $r^{(k)}$ and all the atoms a_i of the

dictionary A, and it selects the atom a_m with maximum correlation with the residual. The coefficient of atom a_m is set to the correlation coefficient, and the residual is updated by subtracting the atom a_m with the corresponding coefficient; the rest of the coefficients are unchanged. Convergence is guaranteed for every vector y in the span of the dictionary, although in practice convergence can be very slow.

The main advantage of MP is its simplicity, as it has linear complexity with respect to the size of y, m, as well as to the number of dictionary atoms N. As such, it can be used with extremely large signals (e.g. decomposing musical signals of considerable size, sampled at typical frequencies of tens of kHz, with millions of atoms [27], where no other method can be used). Compared to other methods, however, the sparsity of the solution found is poorer.

Orthogonal Matching Pursuit (OMP) [14] is similar, but adds a projection step after every iteration. In this way, when adding a new atom all the coefficients of the previous atoms are modified, such that the current approximation is precisely the projection of the signal on the subspace spanned by the selected atoms. OMP is described in more detail in the following Section 2.5.2.

Compressive Sampling Matching Pursuit (CoSaMP) [28] is a generalization of OMP that adds at every iteration the most significant $2s$ atoms instead of only one. The signal is then projected on the set spanned by all the selected atoms, and only the most significant s atoms are retained in the decomposition. The CoSaMP algorithm produces a decomposition with s non-zero components, the approximation error being bounded within a linear factor by the ℓ_1 error of the optimal decomposition with $s/2$ non-zero coefficients and by the energy of the additive noise [28]. Such a bound on the error is often the most that can be guaranteed for sub-optimal solutions of NP-hard problems. The drawback is the requirement that the dictionary A satisfies the RIP for good performance, making CoSaMP an algorithm dedicated to compressed sensing and less usable for other applications of sparse signal approximations.

Orthogonal Matching Pursuit

Orthogonal Matching Pursuit (OMP) [14, 15], [25, cap. 12] leads to improved decompositions by introducing a Gram-Schmidt orthogonalization after every iteration. Since the atoms are in general non-orthogonal to each other, adding a new atom in the MP algorithm introduces new components along the directions of the other atoms previously selected. Let us denote with T the list of the atoms already selected for the decomposition. Similarly to MP, the OMP algorithm looks for the atom a_m that has maximum

Algorithm 1 Orthogonal Matching Pursuit (OMP)

1: $r^{(0)} \leftarrow y$
2: $\gamma_i \leftarrow 0, \forall i$
3: **repeat**
4: Find $a_m \in A$ with maximum correlation coefficient $\left\langle r^{(k)}, a_m \right\rangle$
5: Add m to the set of indices of already selected atoms, $T \leftarrow T \cup \{m\}$
6: Project x on the set of atoms $a_{\{T\}}$, obtaining the coefficient vector at step k: $\gamma_{\{T\}}^{(k+1)} = a_{\{T\}}^{\dagger} x$
7: Update the residual $r^{(k+1)} \leftarrow x - A \cdot \gamma^{(k+1)}$
8: **until** stopping criterion, e.g. $\|r^{(k)}\|_2 \leq \epsilon$ or fixed number of iterations

correlation with the residual, $\left\langle r^{(k)}, a_m \right\rangle$, and then adds a_m to the set $a_{\{T\}}$ of selected atoms, but then all the coefficients $\gamma_i, i \in T$ are recomputed by projecting the signal on the set of selected atoms $a_{\{T\}}$. The difference from MP consists, therefore, in that once atom a_m is selected at step k, all the coefficients of the current decomposition are updated, including the coefficients of the previously selected atoms, in order for the new approximation to be the projection of the original signal on the set of selected atoms. This is described conceptually in step (6) of Algorithm 1, but in practice it can be implemented with computationally simpler iterative methods .

The particularity of OMP is that after each iteration the signal approximation with the currently selected atoms is precisely the projection of the signal on the subspace spanned by these atoms. In other words, the current signal approximation is orthogonal on the residual, $A\gamma^{(k)} \perp r^{(k)}$, or $\left(y - r^{(k)}\right) \perp r^{(k)}, \forall k$. This guarantees that the algorithm stops in a number of iterations at most equal to the original signal sparsity, and the solutions found are typically much better than with MP. Basically, if one chooses a fixed number of iterations, one is able to impose the desired sparsity of the decomposition, since each iteration adds one and only one atom to the decomposition.

The disadvantage of OMP is that, even with an efficient implementation that smartly reuses the coefficients γ_i from the previous iteration to reduce the computing effort required for the orthogonal projection at the current iteration, the complexity is quadratic in N, $\mathcal{O}(N^2)$, and thus much larger than for MP. As a consequence, even if the number of iterations is small, in general the computing effort for the OMP algorithm is significantly larger than for MP.

2.5.3 Smoothed ℓ_0

Algorithm 2 SL0

1: Initialization: $\gamma^{(0)} = \text{argmin}||\gamma||_2$ s.t. $y = A\gamma$
2: Initialization: fix a suitable decreasing sequence $\{\sigma_i\}$, $1 \leq i \leq N$
3: **for** $i = 1 \to N$ **do**
4: Initialization with previous solution: $\gamma^{(i)} \leftarrow \gamma^{(i-1)}$
5: Perform L steepest ascent steps:
6: **for** $k = 1, ...L$ **do**
7: Compute negative gradient $\delta = [s_1 e^{\frac{s_1^2}{2\sigma^2}}, ...s_n e^{\frac{s_n^2}{2\sigma^2}}]^T$
8: Step towards gradient $\gamma^{(i)} \to \gamma^{(i)} - \mu\delta$
9: Enforce the constraint $y = A\gamma$ by projecting on the feasible set:

$$\gamma^{(i)} \leftarrow \gamma^{(i)} - A^T(AA^T)^{-1}(A\gamma^{(i)} - y)$$

10: **end for**
11: **end for**
12: **return** $\gamma^{(N)}$

The Smoothed ℓ_0 algorithm (SL0) [29] is an algorithm for solving (P_0), based on the idea of replacing the ℓ_0 minimization with a sequence of non-convex functions that become progressively closer to the ℓ_0 norm. The ℓ_0 minimization is replaced with maximization of the following proxy function:

$$f(\gamma) = \sum_{i=1}^{n} e^{-\frac{\gamma_i^2}{2\sigma^2}}. \tag{2.24}$$

As the variance $\sigma \to 0$, every exponential goes to 0 if the corresponding γ_i is non-zero, but becomes 1 if $\gamma_i = 0$. As such, the proxy function $f(\gamma)$ tends to the *complement* of the ℓ_0 norm:

$$\lim_{\sigma \to 0} \sum_{i=1}^{n} e^{-\frac{\gamma_i^2}{2\sigma^2}} = n - ||\gamma||_0. \tag{2.25}$$

and therefore maximizing the left hand of the equation is equivalent to minimizing the ℓ_0 norm. The advantage is that, contrary to the ℓ_0 norm, $f(\gamma)$ is a smooth function and thus one can standard optimization methods. The authors propose using steepest ascent. However, for small values of σ the function $f(\gamma)$ is not convex, and thus steepest ascent only finds a local optimum of the problem. The solution is to start with a σ large enough that the minimization problem is convex, and then repeatedly solve a sequence of problems with progressively smaller value of σ, using the previous solution as the starting point for the next one. The idea is that, for a small decrease of σ, the solution will remain

in the vicinity of the previous solution.

The complete algorithm is presented as Algorithm 2. It consists of repeatedly maximizing $f(\gamma)$ for a decreasing sequence of σ's, via steepest ascent followed by projecting on the feasible set to enforce the constraint $y = A\gamma$. Each single maximization need not be very accurate, since it is only used as a starting point for the next one. The goal is to get in the vicinity of the solution, not to accurately find it. As such, a few iterations of steepest ascent are enough. As a result, the complete algorithm is very fast, at the same time being very effective.

2.6 Dictionary optimization algorithms

Numerous classes of signals have a "natural" basis or dictionary adequate for sparse or compressible representations. For example, natural or medical images are in general compressible with the cosine or wavelet transforms, i.e. they are approximately sparse in the orthogonal bases associated with these transforms. However, other classes of signals do not have these kind of natural bases, or they have certain characteristics which are not sufficiently exploited in standard bases. For these signals it is desirable to find bases or overcomplete dictionaries that are adapted to their characteristics. For this purpose, there have been developed dictionary optimization algorithms, which use a representative training set to produce dictionaries adapted to the particularities of various classes of signals.

2.6.1 The K-SVD algorithm

One of the most used dictionary optimization algorithms is the K-SVD algorithm [30]. Having a set of training vectors $\{d_n\}$, for a desired sparsity level L and a desired dimension N of the dictionary, K-SVD produces a dictionary $\mathcal{D}$ composed of N atoms ϕ_i optimized for the training set, i.e. the atoms ϕ_i minimize the residual $\sum_n \|d_n - \mathcal{D} \cdot y_L\|_2$ obtained after decomposing the training signals in the dictionary $\mathcal{D}$ with a sparsity level equal to L ($\|y_L\|_0 = L$). The dictionary obtained with K-SVD is a local optimum, and the algorithm can be viewed as a generalization of the classic K-Means algorithm for vector quantization, the difference being that each training vector is approximated not by a single centroid, but as a linear combination of L vectors from a dictionary.

As K-Means, K-SVD executes two steps for each iteration: (i) sparse decomposition of the training vectors in the current dictionary and (ii) updating each dictionary atom to better approximate the training vectors that used that atom in the decomposition.

Algorithm 3 K-SVD

1: Dictionary initialization $\mathcal{D}^{(0)} \in \mathbb{R}^{nxN}$
2: **repeat**
3: Step I. Sparse decomposition of every training vector d_n in the current dictionary $\mathcal{D}^{(k)}$, with L atoms, using OMP. We obtain the matrix of coefficients $\Gamma^{(k)}$.
4: Stage II. Dictionary update:
5: compute the matrix of residuals of every vector $R^{(k)} = D - \mathcal{D}^{(k)} \cdot \Gamma^{(k)}$
6: **for all** atom $\phi_n \in \mathcal{D}^{(k)}$ **do**
7: find the set ω_n of all training vectors that use atom ϕ_n in their decomposition
8: restrict $R^{(k)}$ only to the columns ω_n, $R_n^{(k)}$, and add the component of the current atom to the residuals $R_n^{*(k)} = R_n^{(k)} + \phi_n \cdot row\ n\ of\ \Gamma^{(k)}$
9: compute the SVD decomposition of the matrix $R_n^{*(k)}$, $R_n^{*(k)} = USV^T$ and update:
10: $\phi_n^{(k+1)} \leftarrow$ first column of U
11: row n of $\Gamma \leftarrow$ first column of $V \cdot S(1,1)$
12: **end for**
13: **until** stopping criterion, e.g. fixed number of iterations

The first step requires using a sparse decomposition algorithm; the original version of the algorithm uses OMP, but any other option is possible. As a consequence, the quality of the produced dictionary depends on the particularities of the sparse decomposition algorithm used, and the dictionary is better suited for using with that particular algorithm. In the second step, each atom is updated by replacing it with the most significant singular vector from the SVD decomposition of the matrix formed by the residuals of every training vectors that use that particular atom in their decomposition. At the first iteration, the dictionary can be initialized either with random vectors, a standard basis/dictionary or with random vectors from the training set.

In the description of Algorithm 3, D is the matrix of the training vectors (the vectors are aligned as the columns of D), $\mathcal{D}^{(k)}$ is the dictionary at iteration k, $\Gamma^{(k)}$ is the matrix of the decomposition coefficients of all training vectors at iteration k, and as such the row n from $\Gamma^{(k)}$ contains the coefficients of atom ϕ_n in all decompositions. The matrix $R_n^{*(k)}$ contains the residuals of every vector that uses atom ϕ_n, plus the component of the atom ϕ_n (this is added to the residuals). In other words, $R_n^{*(k)}$ is obtained by subtracting from the training vectors all selected atoms except ϕ_n itself.

Updating atom ϕ_n aims to minimize the Frobenius norm ($\| \cdot \|_F$, extension of the

ℓ_2 norm to matrices) of the matrix $R_n^{*(k)}$: the atom ϕ_n is chosen such that it globally minimizes the residuals of all vectors that use it:

$$\phi_n = \arg\min_{\phi_n} ||R_n^{*(k)} - \phi_n \cdot \gamma_n^T||_F \tag{2.26}$$

Since $\phi_n \cdot \gamma_n^T$ is a matrix of rank 1 (a column vector multiplied with a row vector), we may say that we aim to find the best rank 1 approximation of the matrix $R_n^{*(k)}$, in the sense of minimizing the Frobenius norm of the difference. A classic mathematical result, the Eckart - Young theorem, states that the best low rank r approximation of a matrix, in the sense of minimizing the Frobenius norm of the difference, is the matrix obtained from the most significant singular vectors of the SVD decomposition. In our case $r = 1$, meaning that we choose atom ϕ_n to be the most significant left singular vector (first column of U from the SVD decomposition $R_n^{*(k)} = USV^T$), while the coefficients of this atom, y_n^T, are the first row of V^T multiplied with the first singular value, $S(1,1)$.

In the ideal case that all training vectors have an exact decomposition with L atoms of the dictionary, the decomposition residuals would be zero, so the matrix $R_n^{*(k)}$ from step (8) of the algorithm contains only the components of the current atom, $R_n^{*(k)} = 0 + \phi_n \cdot row\ n\ of\ \Gamma^{(k)}$. Therefore the matrix $R_n^{*(k)}$ itself would be of rank 1, and thus the minimization of (2.26) would keep the vector ϕ_n unchanged.

Increased performance can be obtained in practice by adding various heuristic modifications (e.g. see the implementation of [31]). After each iteration, if an atom is used by too few vectors in their decomposition, it can be reinitialized with a vector from the training set to avoid over-training it on too few particular atoms instead of capturing the more general features of the signal class. Moreover, if after the update step any two atoms become too similar (very correlated), one of them is reinitialized, in order to preserves the generalization capability of the dictionary.

2.7 Algorithms for optimizing the acquisition matrix

The algorithms for optimizing the acquisition matrices aim to obtain an acquisition matrix P from (2.4) that improves sparse signal recovery, by adapting the acquisition matrix to the sparsity basis/dictionary of the signals. One fundamental result presented in Section 2.4.6 shows that random matrices can successfully be used for signal acquisition when the signals are sparse in orthonormal bases. However, in the case of overcomplete dictionaries, the existence of correlations between the dictionary atoms can negatively affect the performance if random matrices with independent elements are being used.

This effect is worse for dictionaries that are produced with optimization algorithms, because in this case the atoms can be significantly correlated with each other if the training signals themselves are grouped around a rather narrow subspace.

Chapter 4 of this thesis is dedicated to improving some existing acquisition matrix optimization algorithms. As such, we find appropriate to defer the description of existing state-of-the-art algorithms to that chapter instead of adding them to this survey, since our contributions in that chapter will make heavy use of the these algorithms and as such it is more adequate to have their description close at hand.

2.8 Additional topics

We mention here other interesting research directions in the compressed sensing literature, which we cannot present in more detail due to space constraints.

One important field is exploiting the additional structure of sparse signals. The basic compressed sensing theory makes no assumption on how the non-zero components are distributed in a sparse signal. Additional benefits can be gained if some additional sparsity *model* is known [32], which is often the case in practice. The *block-sparsity* model relies on the fact that the non-zero components of a sparse signals are often clustered together in blocks [33, 34]. The *tree-sparsity* model assumes that the non-zero coefficients live on a connected tree structure where some sub-trees are all zero [32]. A typical example is a signal that has a sparse representation in a multi-level wavelet decomposition. The multi-level decomposition coefficients can be naturally aligned in a tree structure with the coarser coefficients on top and the details at bottom. In such a decomposition, it is often the case that if one coarse coefficient is equal to zero, all the coefficients in the corresponding sub-tree below it are also zero. This is a structure that can be exploited to achieve better recovery performance [32]. Yet a third model is to assume a common sparsity support of multiple sensed vectors, i.e. acquiring a set of vectors that all share the same sparsity profile, known as the *Multiple Measurement Vectors* (MMV) problem [35, 36]. These models can be exploited in reconstruction algorithms, leading to improved recovery performance.

Another important field is construction of deterministic matrices guaranteeing the RIP or equivalent properties [37, 38, 39]. In comparison with random acquisition matrices, explicitly constructed matrices provide tighter recovery guarantees and often lend themselves to fast implementations. The main drawback is that in many applications the acquisition matrix cannot be freely chosen due to constructive issues, and thus explicit

acquisition matrices are sometimes difficult to implement.

A third extensive research area is the development of recovery algorithms. There is a huge number of algorithms and techniques for sparse signal recovery. A partial list is available online at [40]. Throughout this thesis we use only a few of the most well known algorithms.

We conclude by mentioning two of the most important online resources for the interested reader: the compressed sensing resources page at [41] and the Nuit-Blanche blog [42] featuring the latest research papers and discussions.

Chapter 3

Synthesis and analysis sparsity

In this chapter we introduce a different sparsity model used as *a priori* information for recovering signals from few random projections, that offers similar guarantees to the standard model in the compressive sensing literature. We explore the relation between the two models, exposing key similarities and differences from a practical point of view.

3.1 The analysis sparsity model

The de-facto sparsity model used in compressed sensing and sparse approximations literature is the generative model of (3.1), which asserts that a signal $x \in \mathbb{R}^n$ can be decomposed as a weighted sum of a few k atoms from a known set D

$$x = D\gamma_S, \text{ with } ||\gamma_S||_0 = k \tag{3.1}$$

According with the established terminology, we call this model *synthesis sparsity*

Recently, a different sparsity model has been proposed [43], asserting that the signal x produces a sparse result when analysed with an operator Ω of size $N \times n$:

$$\gamma_A = \Omega x, \text{ with } ||\gamma_A||_0 = N - l \tag{3.2}$$

The signal x is thus orthogonal to l rows of Ω. According to [44], we call this model the *analysis sparsity* model, or simply the *cosparse* model, Ω the *analysis operator*, and the quantity l the *cosparsity* of the signal x with respect to the operator Ω.

Both of these models can successfully be used as regularizing terms in various ill-posed problems. We focus on the problem of recovering a signal $x \in \mathbb{R}^n$ that is observed only through a set of $m < n$ random linear measurements, by exploiting the prior information on the its sparsity. The measurement vectors are aligned as the rows of a $m \times n$ acquisition

matrix P, and the measurements are possibly contaminated by noise e, i.e.:

$$y = Px + z. \tag{3.3}$$

This is the classic problem (2.6) of compressed sensing, and it is well known [9] that a sufficiently synthesis-sparse signal can be accurately recovered by solving the synthesis-based optimization problem ($\hat{P}^{\epsilon}{}_0$):

$$\hat{x} = D \arg\min_{\gamma_S} \|\gamma_S\|_0 \text{ with } \|y - PD\gamma_S\|_2^2 < \epsilon \tag{3.4}$$

where ϵ is the estimated noise energy, as long as the measurement vectors satisfy an incoherence property with the atoms of the dictionary.

A similar result has been proven for analysis-sparse signals [45]. In this case the analysis-based optimization problem that needs to be solved is:

$$\hat{x} = \arg\min_{x} \|\Omega x\|_0 \text{ with } \|y - Px\|_2^2 < \epsilon. \tag{3.5}$$

The ℓ_0 minimization in (3.4) makes the problem combinatorial, and thus NP-hard, and we hypothesize that (3.5) is similarly difficult. In addition, using the ℓ_0 norm makes both problems very sensitive to noise. At the expense of stronger sparsity requirements, the ℓ_0 norm can be replaced with a more relaxed ℓ_p norm. A value $0 < p < 1$ increases the robustness to noise but still leads to non-convex optimization problems, whereas the ℓ_1 norm makes the optimization problems convex, and as such has been studied for both problems [5, 46] and is widely used in practice. Other approaches and related problems have also been studied, e.g. unconstrained versions [47] or exchanging the goal function and the constraint in (3.4) [6].

This chapter presents new results on the relation between the analysis and synthesis sparsity models. The first part of this work deals with an innovative way of performing analysis-based recovery using synthesis-based solvers, whereas the second part of this work investigates under what conditions is analysis recovery preferable to synthesis recovery.

The synthesis sparsity model (3.1) describes a signal as the weighted sum of k atoms of a dictionary D, while the complementary analysis model (3.2) specifies instead what the signal is orthogonal to. Both of them are instances of a general Union-of-Subspaces (UoS) model [45]. The synthesis model with sparsity k asserts that the signal lives in one of the $\binom{N}{k}$ k-dimensional subspaces defined by combinations of any k atoms of D, whereas the analysis model with cosparsity l asserts that the signal lives in one of the $\binom{N}{l}$ $(d-l)$-dimensional subspaces that are orthogonal to any combination of l rows of Ω. It is usually assumed that Ω is in *general position* [44], i.e. any set of n or fewer rows are

linearly independent. We follow this assumption in this work as well. In this case the cosparsity l is upper bounded by $n-1$, since for a non-zero n-dimensional signal one can find at most $n-1$ linearly independent orthogonal signals. No such restriction exists for the sparsity k, which can as small as 1.

The relationship between the analysis and synthesis sparsity models is investigated in [43, 45]. One important issue of interest is the relation between the analysis model with an operator Ω and the corresponding synthesis model with the dictionary $D = \Omega^{\dagger}$, where $\dagger$ defines the Moore-Penrose pseudoinverse. In this case it is known [43] that for $N \leq n$ the analysis and synthesis reconstruction problems are equivalent. However, for the general overcomplete case $N > n$ the equivalence no longer holds, with (3.4) and (3.5) leading to different solutions. This is caused by the different geometries of the defining polytopes, which become more different as the overcompleteness factor of the dictionary/operator increases. It has also been noted that the vector $\gamma_A = \Omega x$ can be viewed as a *poor man's* sparse decomposition of x in $\Omega^{\dagger}$ [45], since it implies $x = \Omega^{\dagger}\gamma_A$, but in general this decomposition is much less sparse than the sparsest decomposition implied in the synthesis model, hence the differences between the two models.

Another important issue is the representation power of the two sparsity models. It is observed in [45] that the two models do not vary similarly with k and l. A synthesis-sparse signal (small k in the synthesis model) can live in a low number of low-dimensional subspaces, but, contrarily, a co-sparse signal in the analysis model (large value of l) generally implies a larger number of low dimensional subspaces. Based on this behaviour it is thus argued that, in general, a synthesis model might make it simpler to recover a signal known to be living in a low-dimensional subspace, since the number of subspaces in which the signal is sought is also low, contrary to the analysis model.

3.2 Analysis-based Sparse Reconstruction via Least-Squares Constrained Synthesis Sparsity

3.2.1 Analysis as least-squares synthesis sparsity

Following our recent work in [48], in this work we propose an innovative approach to the analysis sparsity model and its relation with the synthesis model. One observes than (3.2) can be considered as a way of finding the least-squares solution γ_A to the undetermined equation system

$$x = \Omega^{\dagger}\gamma \tag{3.6}$$

Here, $\Omega^\dagger$ is regarded as an overcomplete dictionary of size $n \times N$. The analysis-based vector γ_A is therefore the least-squares decomposition of the signal x in the overcomplete dictionary $\Omega^\dagger$. Since a least-squares solution is the one solution that is orthogonal to the system's nullspace (alternatively, that lives in the row space of $\Omega^\dagger$, i.e. in the column space of Ω), we may write (3.2) equivalently as:

$$\begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} \Omega^\dagger \\ P_{\Omega^\dagger}^T \end{bmatrix} \gamma_A \tag{3.7}$$

where the rows of $P_{\Omega^\dagger}^T$ form any basis for the nullspace of $\Omega^\dagger$, and thus $P_{\Omega^\dagger}^T$ can easily be found, e.g. with the SVD decomposition. One observes that the upper part of (3.7) is similar to the synthesis sparsity model of x with the overcomplete dictionary $D = \Omega^\dagger$. The additional lower constraint enforces that γ_A is the least-squares decomposition, and thus ensures that it lies in the row span of $\Omega^\dagger$ (column span of Ω), which is implied in the definition of the analysis model (3.2).

We obtained in (3.7) a convenient reformulation of the analysis sparsity model as a *least-squares constrained synthesis* sparsity model with $D = \Omega^\dagger$, characterised by the additional orthogonality constraint on the decomposition vector γ. This shows that analysis sparsity can be viewed as synthesis sparsity *for the least-squares decomposition only*, i.e. a particular case of synthesis sparsity which requires not just any signal decomposition to be sparse, but specifically the least-squares decomposition.

From (3.7) one can directly deduce many facts already known on the relation between the analysis and synthesis sparsity models. Synthesis-based recovery is known to be more general than analysis recovery [43, 45], which is obvious here since analysis sparsity is regarded as synthesis plus an extra constraint. Whenever Ω is square or undercomplete ($N < d$), the additional lower constraint from (3.7) vanishes since there exists only one solution γ, leaving analysis and synthesis recovery equivalent, which is known from [43]. It is also known that analysis and synthesis recovery are equivalent in the case of ℓ_2 minimization [43]. In our approach this follows straightforwardly, since it is precisely the least-squares minimization that characterizes analysis recovery from the more general synthesis recovery problem.

From a practical point of view, the above considerations suggest that, for recovering a signal x from a few linear measurements y, the analysis recovery problem (3.5) can be

replaced with a modified version of (3.4) in the form:

$$\begin{aligned} \hat{x} = D \arg\min_{\gamma} \|\gamma\|_0 \text{ with:} \\ ||y - PD\gamma||_2^2 \leq \epsilon \text{ and} \\ 0 = P_{\Omega^\dagger}^T \gamma \end{aligned} \tag{3.8}$$

where $D = \Omega^\dagger$. Instead of directly finding x as in (3.5), (3.8) first requires finding the sparsest decomposition vector γ_A in the row space of the dictionary $D = \Omega^\dagger$. This is merely a synthesis recovery problem with a reduced search space, i.e. with an additional orthogonality constraint on the solution. This equivalence of (3.8) with (3.5) is formally stated in Theorems 16 and 17 in the following sections, for recovery with both equality and quadratic constraints (noiseless / noisy measurements).

The additional constraint $0 = P_{\Omega^\dagger}^T \gamma$ in (3.8) can easily be integrated in many existing algorithms for synthesis-based recovery, since it is merely a linear equality constraint no different from the other equality or quadratic constraints that these algorithms are required to consider. This opens the way for using existing synthesis-based recovery algorithms for analysis-based recovery (3.8) in a practical scenario.

Interestingly, in a very recent paper [49] that appeared independently of this work, the authors investigate the link between Total Variation (TV) denoising and wavelet shrinkage denoising and reach similar conclusions. TV denoising is based on enforcing a small value of the ℓ_1 norm of the vector containing the discrete gradients of a signal. In the 1-D case, for example, this amount to minimizing a regularizing term of the form $||\Omega x||_1$, where Ω is an analysis operator composed of Haar wavelets. It is therefore in the framework of the analysis sparsity model. On the other hand, wavelet shrinkage with cycle spinning is a denoising method relying on synthesis sparsity, which sparsifies the decomposition of the signal in an overcomplete wavelet dictionary. Along similar lines to our work, the authors show that TV denoising (analysis-based) can be achieved by performing wavelet shrinkage (synthesis-based) with an additional least-squares constraint identical to the one in (3.8). The difference between our work and theirs is that we use constrained optimization, appropriate for the task of recovering a signal from a few measurements, whereas for their denoising task they use unconstrained formulations of the analysis / synthesis regularizing priors. In addition, we consider a more general problem involving a general analysis operator Ω, while their work deals mainly with Haar-wavelet shrinkage, stemming from the particular task of TV denoising. Nevertheless, it is an excellent practical example of performing analysis-based tasks using synthesis-based algorithms together with an additional least-squares type constraint.

3.2.2 Exact signal recovery from noiseless measurements

In this section we consider the case of reconstruction with equality constraints, i.e. $\epsilon = 0$ in (3.5), which is appropriate in the case of noiseless measurements. The following theorem, first introduced in our previous work [48], establishes the equivalence between analysis recovery with exact constraints and a least-squares constrained synthesis recovery problem.

Theorem 16. *The solution of the analysis recovery problem with equality constraints*

$$\hat{x} = \arg\min_{x} \|\Omega x\|_p \text{ with } y = Px \tag{3.9}$$

is identical to the solution of the least-squares constrained synthesis recovery problem

$$\hat{x} = D \arg\min_{\gamma} \|\gamma\|_p \text{ with } \tilde{y} = \tilde{A}\gamma \tag{3.10}$$

where $D = \Omega^\dagger$, $\tilde{y} = \begin{bmatrix} y \\ 0 \end{bmatrix}$, $\tilde{A} = \begin{bmatrix} PD \\ P_D^T \end{bmatrix}$ *with* P_D *being any basis of the nullspace of* D.

Proof. We show the equivalence of (3.9) with (3.10), starting from the approach in [43]. Making the notation $\Omega x = \gamma$, it follows from $\Omega^\dagger \Omega = I_d$ that $x = \Omega^\dagger \gamma$. We proceed to substitute the unknown variable x in (3.9) introducing γ instead, but in doing that we must keep in mind that γ is allowed to live only in the column span of Ω, which we can express as the extra constraint $\gamma = \Omega\Omega^\dagger\gamma$. Therefore we arrive to

$$\hat{x} = \Omega^\dagger \arg\min_{\gamma:\gamma=\Omega\Omega^\dagger\gamma} \|\gamma\|_p \text{ with } y = P\Omega^\dagger\gamma. \tag{3.11}$$

We rewrite the constraint $\gamma = \Omega\Omega^\dagger\gamma$ as $0 = (I_N - \Omega\Omega^\dagger)\gamma$. We can join this with the constraint $y = P\Omega^\dagger\gamma$ and construct a single augmented constraint system

$$\underbrace{\begin{bmatrix} y \\ 0 \end{bmatrix}}_{\tilde{y}} = \underbrace{\begin{bmatrix} P\Omega^\dagger \\ I_N - \Omega\Omega^\dagger \end{bmatrix}}_{\tilde{A}} \gamma. \tag{3.12}$$

Let us define $D = \Omega^\dagger$. The lower constraint $0 = (I_N - \Omega\Omega^\dagger)\gamma$ is equivalent to γ living in the column space of Ω, i.e. being orthogonal to the nullspace of $D = \Omega^\dagger$; therefore this constraint can be expressed as $0 = P_D^T\gamma$ with P_D being any basis of the nullspace of D. Replacing $\Omega^\dagger$ with D and rewriting (3.11) with the augmented constraint (3.12) yields

$$\begin{aligned} \hat{x} &= D \arg\min_{\gamma} \|\gamma\|_p \\ &\text{with } \begin{bmatrix} y \\ 0 \end{bmatrix} = \begin{bmatrix} PD \\ P_D \end{bmatrix} \gamma \end{aligned} \tag{3.13}$$

which is what we wanted to prove. □

Algorithm 4 Proposed Analysis-By-Synthesis approach for exact reconstruction (ABS-*exact*)

Require: Analysis operator Ω, measurements vector y, measurement matrix P

Ensure: Recovered signal

$$\hat{x} = \arg\min_x \|\Omega x\|_p \text{ with } y = Px$$

1: Define $D = \Omega^\dagger$ and compute a basis for the null space of D using the SVD decomposition, arranging the vectors as the rows of a $(N - n) \times N$ matrix denoted as P_D^T

2: Create augmented constraint matrix $\tilde{A}$ and measurement vector $\tilde{y}$

$$\tilde{A} = \begin{bmatrix} PD \\ P_D^T \end{bmatrix} \qquad \tilde{y} = \begin{bmatrix} y \\ 0 \end{bmatrix}$$

3: Solve

$$\hat{x} = D \arg\min_\gamma \|\gamma\|_p \text{ with } \tilde{y} = \tilde{A}\gamma$$

using a synthesis-based solver.

Theorem 16 is the straighforward consequence of our reformulation of analysis sparsity from the previous section, applied in the context of signal recovery with equality constraints from a set of fewer exact measurements. It proves that one can perform analysis-based reconstruction by solving an augmented synthesis recovery problem. Even though the more general character of synthesis over analysis recovery, as well as the additional constraints implied by the latter were already known from [43, 45], to our knowledge this theorem is the first explicit statement of the equivalence of analysis-based exact reconstruction with an augmented synthesis problem.

Based on Theorem 16, our practical approach for analysis-based recovery with equality constraints is summarized in Algorithm 4, which we denote as *Analysis-By-Synthesis exact* (ABS-*exact*). It consists of building the augmented constraint matrix $\tilde{A}$ and measurement vector $\tilde{y}$ and then solving with a synthesis-based algorithm.

3.2.3 Signal recovery from noisy measurements

In this section we apply the reformulation of analysis sparsity from Section 3.1 to the context of signal recovery with quadratic constraints, i.e. in the case of noisy measure-

ments. We first present a theorem similar to Theorem 16 followed by a second theorem as a way of avoiding having mixed equality and quadratic constraints.

Theorem 17. *The solution of the analysis recovery problem with quadratic constraints*

$$\hat{x} = \arg\min_x \|\Omega x\|_p \text{ with } ||y - Px||_2^2 \leq \epsilon \tag{3.14}$$

is identical to the solution of the synthesis recovery problem with quadratic and equality constraints:

$$\begin{aligned} \hat{x} = & D \arg\min_\gamma \|\gamma\|_p \text{ with} \\ & ||y - PD\gamma||_2^2 \leq \epsilon \text{ and} \\ & 0 = P_D^T \gamma \end{aligned} \tag{3.15}$$

where $D = \Omega^\dagger$ and P_D is any basis of the nullspace of D.

Proof. The proof is similar to the proof of Theorem 16, with the difference that in this case we cannot join the two constraints to form a single equation system, since now one of the constraints is quadratic and the other is equality. □

Theorem 17 parallels the exact reconstruction Theorem 16 in showing that analysis-based recovery can be conveniently expressed as a least-squares constrained synthesis recovery problem. It suggests that synthesis-based recovery algorithms can be easily adapted for analysis recovery by enforcing the additional exact constraint $0 = P_D^T \gamma$ on the solution γ. This requires the algorithms to work simultaneously with both quadratic and equality constraints, thus possibly requiring some minor modifications of the existing implementations. These are particularly easy for algorithms which explicitly enforce the constraints, e.g. by repeatedly projecting on the feasible set. Details about the required modifications of the algorithms we used in our experiments can be found in Section 3.2.4. Unfortunately, pursuit and thresholding algorithms do not in general explicitly use their constraints. Instead, they are tailored for the quadratic constraints they are typically used with, and do not lend themselves to naturally incorporate additional equality constraints. As such, they require a problem with quadratic constraints only.

The following Theorem 18 replaces the equality constraint from Theorem 17 with an arbitrarily precise quadratic constraint, thus obtaining a system with quadratic constraints only. However, the equivalence holds only as a limit case.

Theorem 18. *The solution of the analysis recovery problem with quadratic constraints*

$$\hat{x} = \arg\min_x \|\Omega x\|_p \text{ with } ||y - Px||_2^2 \leq \epsilon \tag{3.16}$$

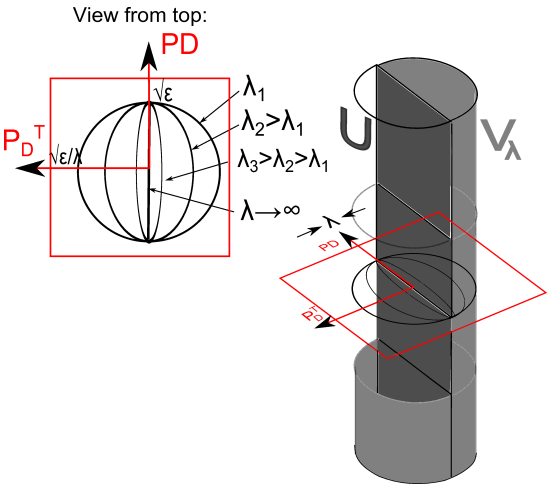


Figure 3.1: Geometrical interpretation of Theorem 18. U is the feasible set of analysis recovery (3.16), V_λ is the feasible set of (3.17) for some $\lambda \in \mathbb{R}$. As $\lambda \to \infty$, the set V_λ flattens and converges to U, implying that the global $\ell_{p>0}$ minimizers over the two sets become identical.

for $p > 0$ is identical to the solution obtained as the limit of

$$\lim_{\lambda\to\infty} \hat{x}_\lambda = D \arg\min_{\gamma} \|\gamma\|_p \text{ with } \left|\left| \begin{bmatrix} y \\ 0 \end{bmatrix} - \begin{bmatrix} PD \\ \lambda\ P_D^T \end{bmatrix} \gamma \right|\right|_2^2 \leq \epsilon. \tag{3.17}$$

where $D = \Omega^\dagger$ and P_D is any basis of the nullspace of D.

Proof. We first use Theorem 17 to rewrite (3.16) as

$$\begin{aligned} \hat{x} =& D \arg\min_{\gamma} \|\gamma\|_p \text{ with} \\ & ||y - PD\gamma||_2^2 \leq \epsilon \text{ and} \\ & 0 = P_D^T \gamma \end{aligned} \tag{3.18}$$

We denote the feasible set of (3.18) as U and the feasible set of (3.17) for some λ as V_λ. We show that $V_{\lambda\to\infty}$ converges to U, and, for $p > 0$, the continuity of the ℓ_p norm implies that the sequence of minimizers over V_λ converges to the minimizer over U.

The intuition behind the theorem is that as λ becomes arbitrarily large, the lower constraint becomes arbitrarily close to an exact equality constraint as in Theorem 17. The geometrical interpretation of the theorem in shown in Fig.3.1. The feasible set of (3.17), V_λ, is a cylindrical body of elliptical section, such that the projection of any feasible solution falls inside an ellipse extending by $\sqrt{\epsilon}$ along the MD axis and by $\sqrt{\epsilon}/\lambda$

along the P_D^T axis. We would like the solution to be orthogonal to P_D^T. The feasible set of (3.18), U, is orthogonal to P_D^T. As the parameter λ increases to infinity, the ellipse degenerates into a segment, i.e. the set V_λ degenerates into U that is orthogonal to P_D^T as required. As long as the minimization function is continuous (ℓ_p, $p > 0$), the convergence of the feasible sets means that the optimal solutions of the two problems become identical as well.

Inspired by the geometrical interpretation, the rigorous proof is by showing that any potential solution lying just outside the set U will eventually remain outside of V_λ for a large enough value of λ: given any potential solution $\gamma^\star \notin U$, there exists $\lambda^\star$ such that, for any $\lambda > \lambda^\star$, $\gamma^\star \notin V_\lambda$.

First, if the potential solution is such that $||y - PD\gamma^\star||_2^2 > \epsilon$, it is automatically outside V_λ for any $\lambda > 0$, since it does not agree well enough with the measurements.

If $||y - PD\gamma^\star||_2^2 \leq \epsilon$, since $\gamma^\star \notin U$ it follows that $P_D^T\gamma^\star \neq 0$. Let us define the two errors $\epsilon_1^\star$ (the measurement error) and $\epsilon_2^\star$ (the orthogonality error):

$$\epsilon_1^\star = ||y - PD\gamma^\star||_2^2 \tag{3.19}$$

$$\epsilon_2^\star = ||0 - P_D^T\gamma^\star||_2^2 \tag{3.20}$$

$$= ||P_D^T\gamma^\star||_2^2 \tag{3.21}$$

If we define $\lambda^\star = \frac{\sqrt{\epsilon - \epsilon_1^\star}}{\sqrt{\epsilon_2^\star}}$, then we observe that $\gamma^\star$ lives on the boundary of $V_{\lambda^\star}$, since it satisfies the constraint of (3.17) with equality for $\lambda = \lambda^\star$:

$$\epsilon_1^\star + (\lambda^\star)^2\epsilon_2^\star = \epsilon_1^\star + \frac{\epsilon - \epsilon_1^\star}{\epsilon_2^\star}\epsilon_2^\star = \epsilon$$

For any $\lambda > \lambda^\star$, it follows that $\epsilon_1^\star + \lambda^2\epsilon_2^\star > \epsilon$, which means that $\gamma^\star$ does not anymore satisfy the constraint of (3.17), i.e. $\gamma^\star \notin V_{\lambda > \lambda^\star}$.

Since any $\gamma^\star$ just outside if U will eventually remain outside of V_λ for some λ large enough, we conclude that $V_{\lambda \to \infty} = U$. As any ℓ_p norm with $p > 0$ is continuous, it follows that the sequence of minimizers over V_λ

$$\gamma_\lambda = \arg\min_\gamma \|\gamma\|_p \text{ with}$$
$$\left|\left|\begin{bmatrix} y \\ 0 \end{bmatrix} - \begin{bmatrix} PD \\ \lambda P_D^T \end{bmatrix}\gamma\right|\right|_2^2 < \epsilon.$$

is converging towards the minimizer over U. Multiplication with D leads us to $\lim_{\lambda\to\infty} \hat{x}_\lambda = \hat{x}$, which is what we wanted to prove. □

Although Theorem 18 is rigorously valid only for $p > 0$ due to the required continuity of the ℓ_p norm, the geometric interpretation of the proof suggests that ℓ_0 minimization

would fail to comply only in very particular cases, e.g. when the feasible set U of (3.16) is tangent to the hyperplanes of the ℓ_0 "ball", such that a sparser solution might exist just outside of, but infinitely close to the set U. In other cases the theorem is also valid for ℓ_0 minimization, albeit it might require impractically large values of λ. This motivates using the theorem in practice with ℓ_0 minimization algorithms as well.

Following the theorems presented above, we propose the two approaches presented as Algorithm 5 (ABS-*mixed*) and Algorithm 6 (ABS-λ) for approximate reconstruction with synthesis-based solvers. The difference between the two is that ABS-*mixed* relies on Theorem 17 and assumes that the solving algorithms are capable of handling the equality constraint $0 = P_D^T \gamma$ alongside the quadratic constraints, whereas ABS-λ enforces the equality constraint as a degenerate case of a quadratic constraint, as in Theorem 18.

Within ABS-*mixed* the solver must handle both equality and quadratic constraints. Within ABS-λ the solvers are given only quadratic constraints, but the equivalence with analysis recovery holds only as a limit case, thus reducing its practical usefulness. As a consequence, if a synthesis-based solver is capable of handling both types of constraints, the preferred method for analysis recovery is the ABS-*mixed* approach. ABS-λ is useful mostly for pursuit and thresholding algorithms, which cannot easily accommodate equality constraints.

3.2.4 Adapting existing synthesis-based algorithms for analysis recovery

In this section we provide details about the required modifications (if any) of synthesis-based solvers for analysis recovery. We consider the three approaches separately: ABS-*exact* for recovery with exact constraints from noiseless measurements, ABS-*mixed* for recovery from noisy measurements with mixed equality and quadratic constraints, and ABS-λ for recovery from noisy measurements with quadratic constraints only.

ABS-*exact*

In the case of signal recovery with exact constraints, Theorem 16 shows that analysis recovery requires only the addition of the extra equality constraint alongside the measurements. Existing synthesis algorithms need not be modified at all, one only has to provide them the extra constraint, as in Algorithm 4. Therefore any existing algorithm for ℓ_p minimization, $\forall p$, can be used for analysis recovery, albeit in practice the recovery performance depends on the algorithm. The algorithms that we use in our experiments

Algorithm 5 Proposed Analysis-By-Synthesis (ABS-*mixed*) approach for approximate reconstruction with both equality and quadratic constraints

Require: Analysis operator Ω, measurements vector y, measurement matrix P

Ensure: Recovered signal

$$\hat{x} = \arg\min_x \|\Omega x\|_0 \text{ with } ||y - Px|| \leq \epsilon$$

1: Define $D = \Omega^\dagger$ and compute a basis for the null space of D using the SVD decomposition, arranging the vectors as the rows of a $(N - n) \times N$ matrix denoted as P_D^T

2: Add the constraint

$$0 = P_D^T \gamma$$

to the list of constraints enforced by the solver

3: Solve

$$\begin{aligned} \hat{x} = & D \arg\min_\gamma \|\gamma\|_0 \\ & \text{with } ||y - Px|| \leq \epsilon \\ & \text{and with } 0 = P_D^T \gamma \end{aligned}$$

using a synthesis-based solver.

are presented in section 3.2.5.

ABS-λ

The ABS-λ approach is a formulation using only quadratic constraints for recovering signals from noisy measurements. Therefore we can use with no modifications, in principle, any synthesis-based algorithm for noisy measurements . However, Theorem 18 guarantees equivalence with analysis recovery only in the limit case of a very ill-conditioned optimization problem ($\lambda \to \infty$). In practice, ill conditioned problems mean at best convergence problems, and at worst severe numerical imprecisions compromising the result. This approach is therefore usable only for reasonable values of λ, depending on the particular algorithm, and does not guarantee accuracy. As such, ABS-λ is a solution of last-resort, useful mainly for pursuit and thresholding algorithms which cannot accommodate both equality and quadratic constraint at the same time.

One of the algorithms we use for ℓ_0 minimization is *Smoothed* ℓ_0 (SL0) [29]. The

Algorithm 6 Proposed Analysis-By-Synthesis (ABS-λ) approach for approximate reconstruction with implicit extra equality constraints

Require: Analysis operator Ω, measurements vector y, measurement matrix P

Ensure: Recovered signal

$$\hat{x} = \arg\min_{x} \|\Omega x\|_0 \text{ with } ||y - Px|| \leq \epsilon$$

1: Define $D = \Omega^{\dagger}$ and compute a basis for the null space of D using the SVD decomposition, arranging the vectors as the rows of a $(N - n) \times N$ matrix denoted as P_D^T

2: Create augmented constraint matrix $\tilde{A}_\lambda$ and measurement vector $\tilde{y}$

$$\tilde{A}_\lambda = \begin{bmatrix} PD \\ \lambda \ P_D^T \end{bmatrix} \qquad \tilde{y} = \begin{bmatrix} y \\ 0 \end{bmatrix}$$

3: Solve

$$\hat{x} = D \arg\min_{\gamma} \|\gamma\|_0 \text{ with } ||\tilde{y} - \tilde{A}_\lambda x|| \leq \epsilon$$

for a sufficiently large λ using a synthesis-based solver.

original SL0 algorithm [1] is a steepest descent algorithm designed to work with equality constraints (i.e. in the noiseless case), consisting of optimizing a sequence of smoother and progressively more accurate approximations of the ℓ_0 norm. Each descent step is followed by projection on the feasible set given by the constraints. In our experiments we modify the SL0 algorithm to adapt it to noisy measurements. We modify the projection step in the following manner: instead of projecting on a subspace defined by the exact constraints, we project onto the ellipsoid feasible set defined by the quadratic constraints, while keeping everything else in the SL0 algorithm unchanged. Projection on an ellipsoid is a non-trivial optimization problem in itself, but fast iterative algorithms have been developed [50], and we use the general algorithm from [50]. While steepest descent followed by projection on an ellipsoid might not be the most efficient implementation, we take this approach to maintain maximum similarity with the original SL0 implementation. In particular, this implementation offers better results than Robust-SL0 [51], at a higher computational cost.

[1] we use the implementation from http://http://ee.sharif.edu/ SLzero/

ABS-*mixed*

For recovering signals from noisy measurements, Theorem 17 and Algorithm 5 require solving an optimization problem with both equality and quadratic constraints. However, existing synthesis-based algorithms for approximate recovery typically use quadratic constraints only. Enforcing additional exact constraints is not difficult in principle, but it still requires modifications inside the algorithms.

For ℓ_0 minimization, we use Smoothed ℓ_0 (SL0). The original algorithm implementation is modified in two ways. First, as explained above, we enable recovery with quadratic constraints (noisy measurements) by replacing the projection step with projection on the ellipsoidal feasible set given by the quadratic constraints with the algorithm from [50]. Enforcing the additional least-squares equality constraint requires an additional exact projection stage. Therefore the full projection step of our modified SL0 algorithm for the ABS-*mixed* approach consists first of an exact projection on the subspace given by the analysis-based least-squares constraint, followed by projection on an ellipsoid given by the quadratic measurement constraints.

In principle, this approach is valid for all synthesis-based algorithms that enforce the constraints by projecting on the feasible set. One only needs to change this projection step by adding the exact projection required by the additional least-squares exact constraint.

For ℓ_1 minimization, we use the synthesis-based Basis Pursuit algorithm with quadratic constraints as implemented in the ℓ_1-*magic* toolbox [52]. This is an implementation of a generic log-barrier solver for a second-order cone program, which, in its general form, can handle both quadratic and exact constraints [2]. Our modification amounts to using the general form of the algorithm that allows equality constraints as well.

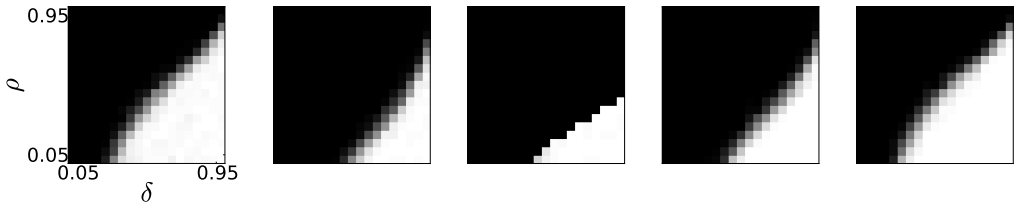
3.2.5 Experimental Results

In this section we present experimental results for analysis-based signal reconstruction using our proposed Analysis-By-Synthesis approaches, for both equality and quadratic constraints.

Reconstruction with equality constraints

In the exact reconstruction case, we investigate the performance obtained using ABS-*exact* with four existing synthesis-based algorithms and compare it with the results of Greedy-Analysis-Pursuit (GAP) [44], a solver designed specifically for solving analysis

[2] see eq.(8) from [52]



(a) GAP (b) OMP-ϵ (c) TST (d) BP (e) SL0

Figure 3.2: Percentage of perfectly reconstructed signals for analysis-based recovery with different algorithms: GAP [44] (a) and our proposed ABS-*exact* approach with four different synthesis solvers ((b), (c), (d) and (e)). White indicates 100% recoverability and black 0%. SL0 is virtually identical to GAP, but all other algorithms also perform well.

Table 3.1: Total running times for exact reconstruction ($\times 10^3$ seconds)

	ABS-*exact*:			
GAP	OMP-ϵ	TST	BP	SL0
36.742	3.844	55.649	90.106	6.044

recovery (3.5) directly. With our ABS-*exact* approach we use Orthogonal Matching Pursuit [14] with stopping criterion being residual error below 10^{-9} (OMP-ϵ), Two Stage Thresholding (TST) [53] (a generalization of CoSaMP [28] and Subspace Pursuit [54]), ℓ_1 minimization using the *cvxopt* Python convex optimization package [3] (Basis Pursuit, BP), and Smoothed ℓ_0 (SL0) [29] (ℓ_0 minimization via optimization of smooth but progressively more accurate approximations of the ℓ_0 norm).

Following [44], we investigate the perfect recoverability domain of the above mentioned algorithms. The dimension of the signals is set to $n = 200$. The analysis operator is created as the transposition of a random tight frame, having $N = 240$ rows, thus being 1.2 times overcomplete. We define the parameters $\delta = \frac{m}{n}$ and $\rho = \frac{n-l}{m}$ that define the compression ratio and the relative cosparsity. For every pair (δ, ρ) we generate 100 signals x_i such that $\|\Omega x_i\|_0 = N - l$ and we project them using a random measurement matrix P of size $m \times n$, with zero-mean unit-norm normal i.i.d. random elements. We then attempt reconstruction with the above mentioned algorithms. We consider a signal as perfectly recovered if the reconstruction error is below 10^{-6}.

[3] available at http://abel.ee.ucla.edu/cvxopt/

For reconstruction with equality constraints, Fig.3.2 displays the percentage of perfectly recovered signals, with white indicating 100% recoverability and black 0%.

Fig.3.2 shows that our ABS-*exact* approach is a viable solution to analysis-based recovery with equality constraints, with SL0 performing as good as GAP but with better running time. OMP-ϵ and BP also provide good results. TST also performs well but requires more favourable parameters. Interestingly, the transition from success to failure for TST is much steeper than for other algorithms (very sharp transition from white to black), something that was also observed in [55] in the synthesis case. This suggests that the synthesis-based algorithms generally exhibit the same characteristics and performance as in the usual case of synthesis recovery.

For completeness, we also present the total running times of the algorithms in Table 3.1. The overall experiment consisted in recovering a total of 36100 signals (19×19 pairs (δ, ρ) $\times 100$ signals for each) on a 2.83GHz Intel Core 2 Quad Q9550 machine running MATLAB 7.7.0. We find that for our experiments OMP and SL0 are the fastest whereas BP is the slowest, with TST and GAP yielding intermediate times.

Reconstruction with quadratic constraints

In the case of reconstruction with quadratic constraints we also use various synthesis-based solvers within our two approaches ABS-*mixed* and ABS-λ. We compare with the results of the GAP and NESTA [56] algorithms that solve the analysis recovery problem directly by ℓ_0 and ℓ_1 minimization, respectively.

With the ABS-*mixed* approach we use the SL0 algorithm [29] for ℓ_0 minimization and the ℓ_1 minimization algorithm from the ℓ_1-MAGIC toolbox [52] (denoted as BP). As detailed in section 3.2.4, both algorithms are modified to enable them to consider the extra equality constraint as well.

In the ABS-λ approach we use OMP-ϵ and TST, the stopping criterion for both of them being residual error below ϵ, and also SL0 and BP with quadratic constraints. To investigate the convergence implied by Theorem 18, we take a progressive sequence of values for λ: $1, 10^2, 10^4$.

The analysis operator is created as the transposition of a random tight frame, with varying overcompleteness factor. The dimension of the signals is $n = 50$. The parameters δ and ρ are defined in the same way as for exact reconstruction. For every pair (δ, ρ) we generate 100 signals x_i and we project them using a random measurement matrix P of size $m \times n$, with zero-mean unit-norm normal i.i.d. random elements. We add random white noise to the measurements and then attempt reconstruction with the above algorithms.

We are interested in the distortion of the reconstructed signals. We define the percentage RMS error of a reconstructed signal $\hat{x}$ as

$$R = \sqrt{\frac{\sum (x_i - \hat{x}_i)^2}{\sum x_i^2}} \tag{3.22}$$

A smaller value of R indicates a better reconstruction, with $R = 0$ meaning perfect reconstruction. We compute the average R for every pair (δ, ρ) and we display them in a suggestive manner, with white indicating $R = 0$ (i.e. perfect reconstruction) and black $R \geq 1$ (i.e. distortion energy higher than signal energy).

Fig.3.3 presents the results obtained with the dimension of the signals is set to 50 and the analysis operator Ω is 60×50, therefore being 1.2 times overcomplete. The measurements are contaminated with $SNR = 40$dB white noise. For easier comparison, the contours delimit the areas where R is below $R \leq 0.05$, $R \leq 0.2$ and $R \leq 0.5$.

In Fig.3.4 we increase the overcompleteness factor to 2, with Ω being 100×50, and keep the same SNR value of 40dB for the measurements.

In Fig.3.5 and 3.6 we decrease the measurements' SNR to 20dB, with the analysis operator being again 1.2 times and then 2 times overcomplete, respectively.

We observe that in the ABS-*mixed* approach with mixed equality and quadratic constraints, the performance of synthesis-based SL0 is very similar to the performance of GAP for ℓ_0 minimization, for smaller or larger noise as well as with an analysis operator less or more overcomplete. This is also true for ℓ_1 minimization, the synthesis-based BP algorithm providing similar performance to the analysis-based NESTA algorithm. This shows that the ABS-*mixed* approach is a a viable alternative for analysis-based signal recovery under a variety of circumstances.

The ABS-λ approach is also successful under some conditions. For small noise and small overcompleteness factor, Fig.3.3, all algorithms perform well under ABS-λ approach. As the the operator becomes more overcomplete, Fig.3.4, the performance of OMP-ϵ and TST worsens, especially as λ increases, indicating that the algorithms encounter problems when the constraint system becomes more ill-conditioned. Higher noise also affects the results, Fig.3.5 and Fig.3.6, for all algorithms.

We conclude therefore that the ABS-*mixed* approach of solving an augmented synthesis problem with mixed quadratic and equality constraints enables performance almost similar to dedicated analysis-based signal reconstruction, whereas the ABS-λ approach, relying only on quadratic but ill-conditioned constraints, is useful mostly with small noise and for analysis operators with small overcompleteness factor.

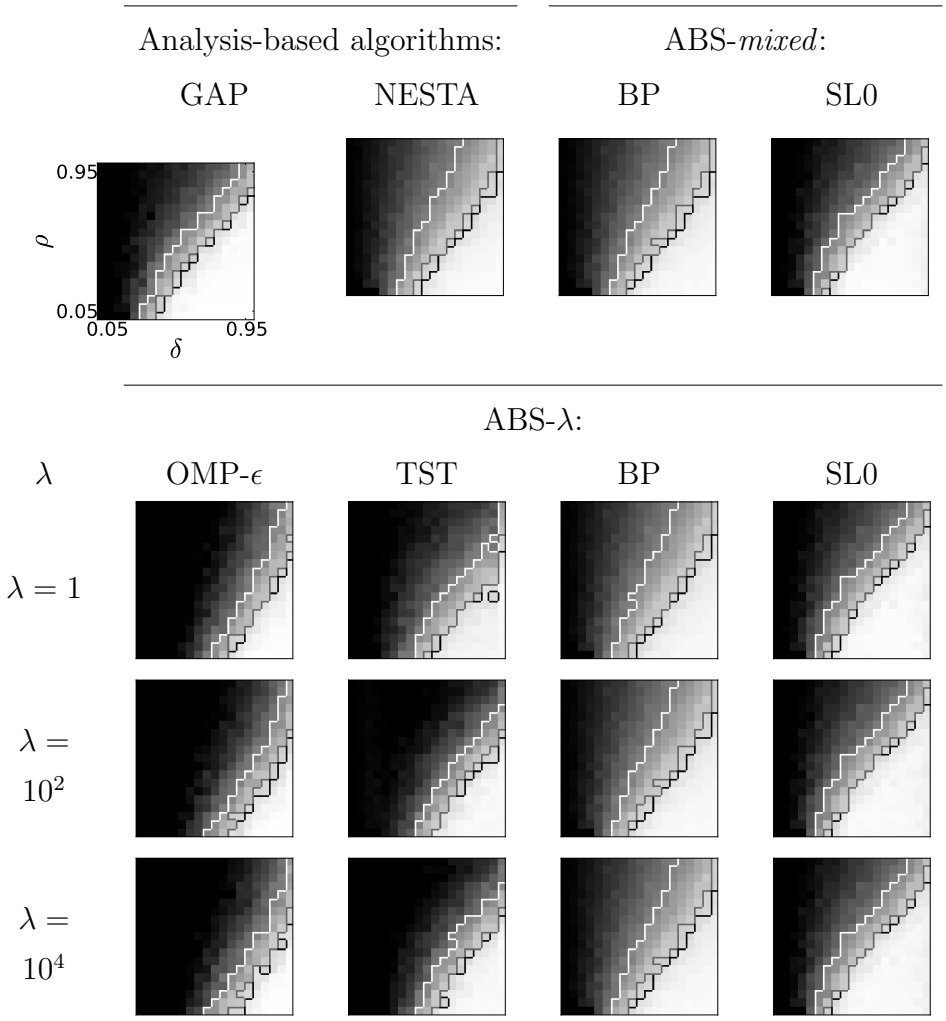


Figure 3.3: **SNR = 40dB, 1.2 times overcomplete.** Average reconstruction error for analysis-based recovery with quadratic constraints. White indicates $R = 0$ (perfect reconstruction) and black $R > 1$. The analysis operator Ω is 60×50, the noise of the measurements is 40dB. The contours delimit the areas where $R \leq 0.05$, $R \leq 0.2$ and $R \leq 0.5$.

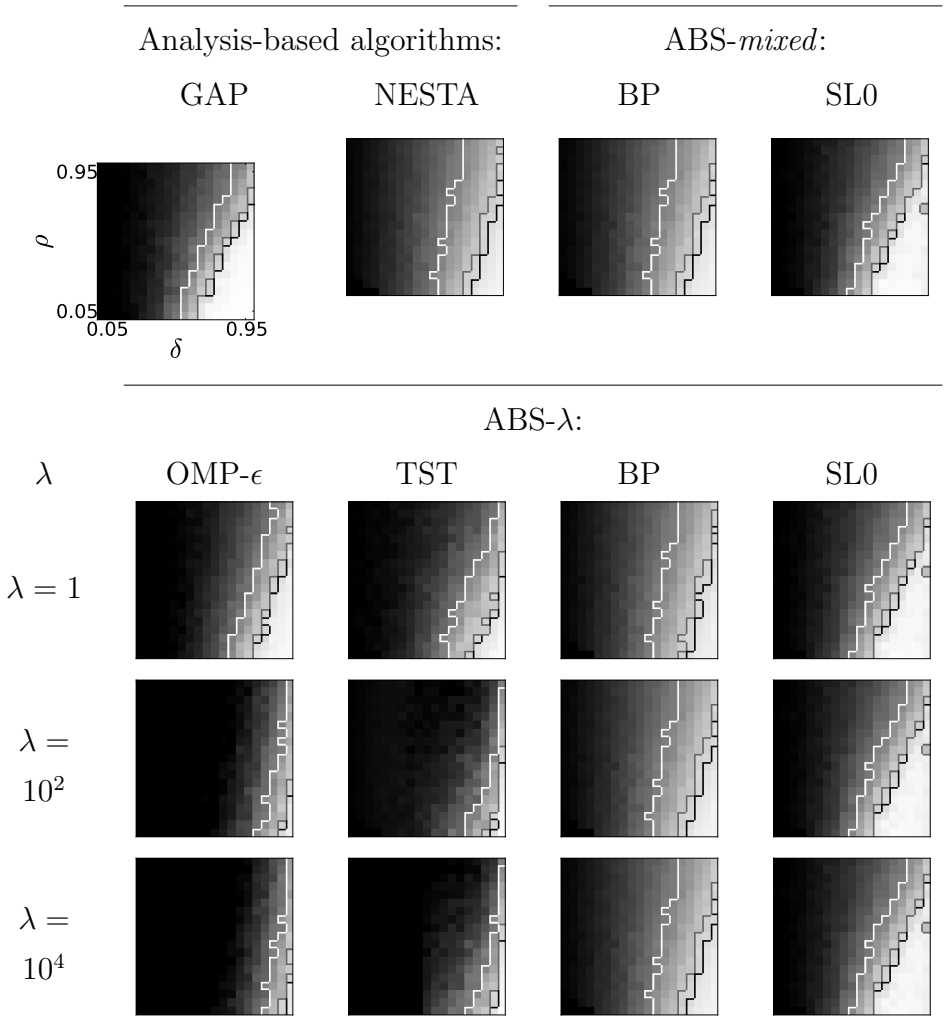


Figure 3.4: **SNR = 40dB, 2 times overcomplete.** Average reconstruction error for analysis-based recovery with quadratic constraints, white indicating $R = 0$ (perfect reconstruction) and black $R > 1$. The analysis operator Ω is 100×50, the noise of the measurements is 40dB. The contours delimit the areas where $R \leq 0.05$, $R \leq 0.2$ and $R \leq 0.5$..

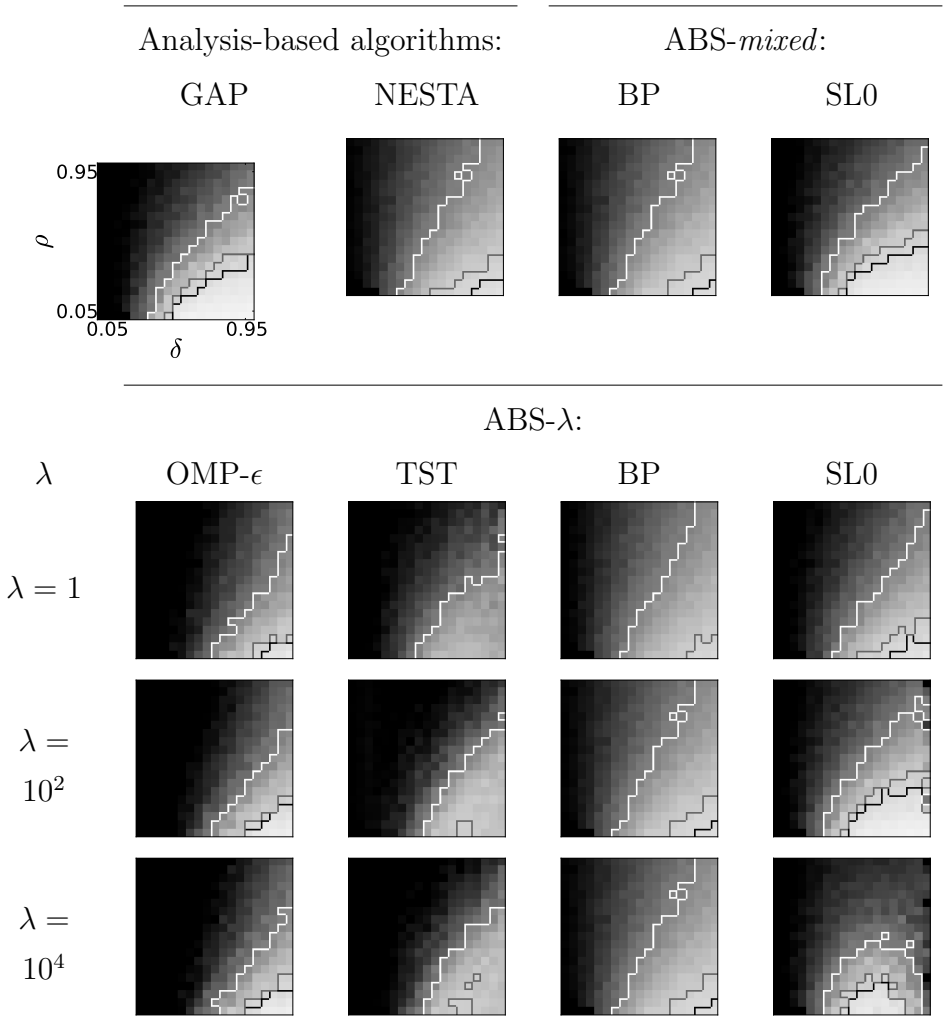


Figure 3.5: **SNR = 20dB, 1.2 times overcomplete.** Average reconstruction error for analysis-based recovery with quadratic constraints, white indicating $R = 0$ (perfect reconstruction) and black $R > 1$. The analysis operator Ω is 60×50, the noise of the measurements is 20dB. The contours delimit the areas where $R \leq 0.05$, $R \leq 0.2$ and $R \leq 0.5$.

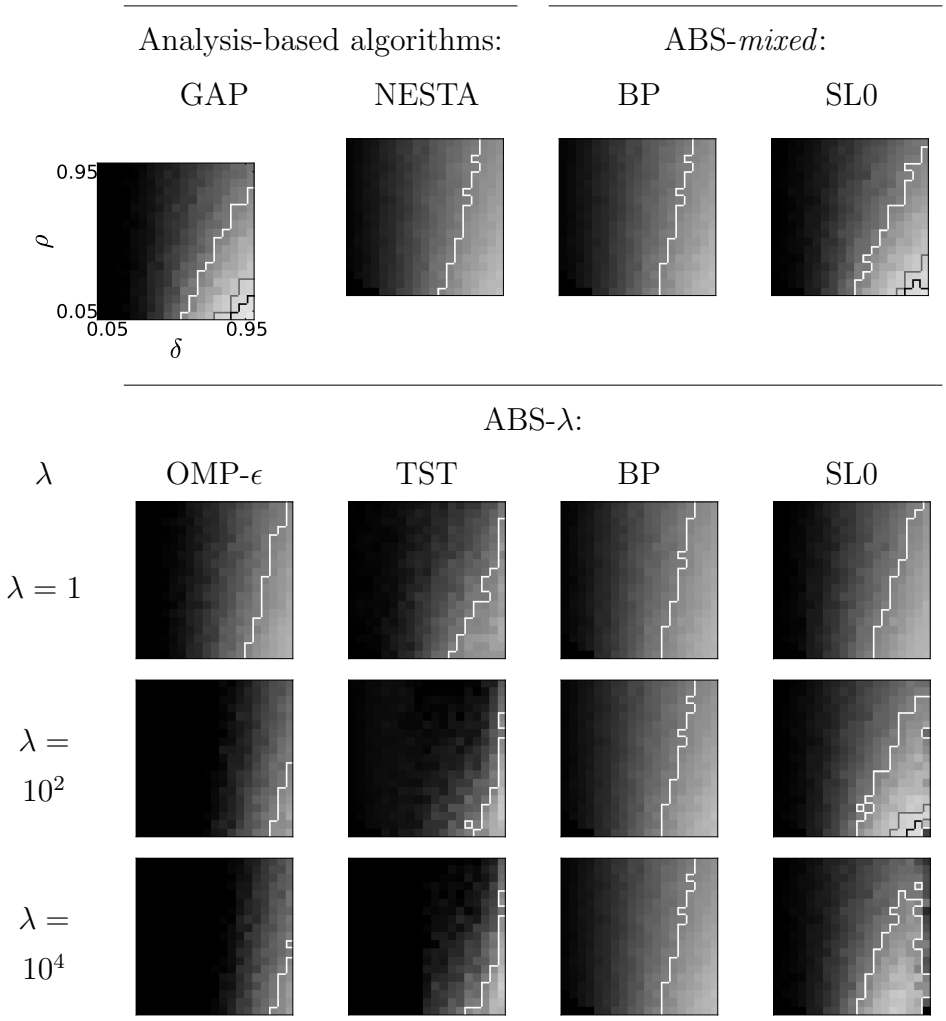


Figure 3.6: **SNR = 20dB, 2 times overcomplete.** Average reconstruction error for analysis-based recovery with quadratic constraints, white indicating $R = 0$ (perfect reconstruction) and black $R > 1$. The analysis operator Ω is 100×50, the noise of the measurements is 20dB. The contours delimit the areas where $R \leq 0.05$, $R \leq 0.2$ and $R \leq 0.5$.

3.2.6 Conclusions

In this section we introduced a new approach to analysis-based signal recovery, reformulating the analysis recovery problem as least-squares constrained synthesis-based recovery. The foundation of our approach resides in the fact that the analysis sparsity model can be viewed as a synthesis sparsity model with the additional constraint that the decomposition vector must lie in the row space of the dictionary, or, equivalently, that it is also of minimum ℓ_2 norm. This interpretation enables us to adapt for analysis recovery algorithms that were originally designed solely with synthesis-based sparsity in mind.

Building on this relation between analysis and synthesis sparsity, we developed three theorems that allow to reformulate the analysis recovery problem as augmented synthesis problems. In the case of exact constraints, analysis recovery is shown to be equivalent to synthesis recovery with a set of extra equality constraints. As a result, synthesis solvers can be directly applied to analysis problems. In the case of analysis recovery with quadratic constraints, this is shown to be equivalent to synthesis recovery with mixed quadratic and equality constraints. Some existing synthesis solvers can be easily modified to include both types of constraints simultaneously, so they can be adapted for analysis recovery. The extra equality constraint can also be expressed as a limit case of a quadratic constraint, enabling synthesis solvers that accept only quadratic constraints to function as well.

Experiments show that this approach is a viable alternative to analysis recovery with both equality or quadratic constraints, well-known synthesis solvers, most notably SL0, successfully matching under a variety of conditions the performance of the GAP algorithm which is specifically designed for analysis recovery.

3.3 Choosing Analysis or Synthesis Recovery for Sparse Reconstruction

In this section we compare synthesis recovery (3.4) with $\epsilon = 0$ with the reformulated analysis recovery (3.8), aiming to establish when one is better than the other, for a general dictionary D. One observes that following our reformulation of analysis sparsity in Section 3.2.1, the two recovery problems have a very similar form. The difference is that synthesis recovery (3.4) seeks the *sparsest* decomposition γ_S of x in D, whereas analysis recovery (3.8) requires finding the *least-squares* decomposition γ_A of x in D. As such, analysis recovery (3.8) benefits from having available an additional orthogonality

constraint, at the expense of seeking a solution which is less sparse, since the co-sparsity of γ_A is upper bounded by $l \leq n-1$ while no such restriction exists for γ_S.

Choosing between synthesis and analysis-based recovery implies, therefore, evaluating the benefits of having an extra orthogonality constraint versus higher sparsity of the solution. In this section we evaluate this trade-off from a practical point of view, using numerical experiments to establish when one recovery problem is more successful than the other, depending on the sparsity and cosparsity of a signal. This kind of empirical numerical experiments have been presented in compressed sensing literature before [57], providing insight into the theory as well as significant help for practical applications.

A meaningful comparison requires the synthesis dictionary and the analysis operator to be pseudoinverses of each other, i.e. using the same D in (3.4) and (3.8). This also follows naturally from the equivalence of the two models in the complete case. In a practical application, however, if different dictionary and analysis operator are available, one must keep in mind that the difference of their inherent quality will correspondingly make one recovery method preferable.

3.3.1 Bisparse signals

In this work we consider signals that are jointly synthesis and analysis sparse for some random dictionary, and we perform reconstructions with both synthesis and analysis algorithms separately, in order to establish which recovery is better, depending on the sparsity and cosparsity of the signals. We focus mainly on exact-sparse signals, but we optionally add small non-sparse random components to the signals to also investigate robustness against non-exact sparsity.

Let us consider a signal $x \in \mathbb{R}^n$ and a dictionary $D \in \mathbb{R}^{n \times N}$. Denote with γ_S the sparsest decomposition of x in D, and with γ_A the least-squares decomposition of x in D, with $k = ||\gamma_S||_0$ (the number of non-zero coefficients in γ_S) and $l = N - ||\gamma_A||_0$ (the number of zero coefficients in γ_A). We refer to such a signal as a (k,l)-*bisparse* signal, or simply a *bisparse* signal, throughout the rest of this chapter.

We seek to investigate the following question: given k and l, is it better to recover the signal using synthesis-based recovery (3.4) or using analysis-based recovery (3.5) reformulated as (3.8)? As $k \in 1, 2, ...n$ and $l \in 0, 1, ...(n-1)$, there are n^2 possible pairs (k,l) in all. We test every combination and determine in which region of the (k,l) space is synthesis recovery performing better than its analysis counterpart, and vice-versa.

To generate (k,l)-bisparse signals for some dictionary D, we must first see when is it possible for such signals to exist. We rely on Theorem 19 that formulates the existence

conditions.

Theorem 19. *Consider an overcomplete dictionary $D \in \mathbb{R}^{n \times N}$ and denote $M = D^\dagger D$. Given any subsets $I, J \subset \{1, 2, ...N\}$ with $card(I) = l$ and $card(J) = k$, there exists a non-zero vector $x \in \mathbb{R}^n$ having simultaneously a decomposition γ_S with the non-zero coefficients on locations in J and a least-squares decomposition vector γ_A with zero elements on locations I if and only if the rank of the $l \times k$ minor matrix M_{IJ} obtained by keeping only the rows with indices I and columns J from M is strictly smaller than k.*

Proof. The signal x satisfies both (3.1) and (3.2) with $\Omega = D^\dagger$. Replacing x from (3.2) with (3.1) yields:

$$\gamma_A = \underbrace{D^\dagger D}_{M} \gamma_S, \text{ with } ||\gamma_A||_0 = N - l \text{ and } ||\gamma_S||_0 = k. \tag{3.23}$$

If there exist γ_S and γ_A obeying (3.23), then, if we keep only the rows in I and the columns in J from M in (3.23), we have

$$0 = M_{IJ} \gamma_{S_J} \tag{3.24}$$

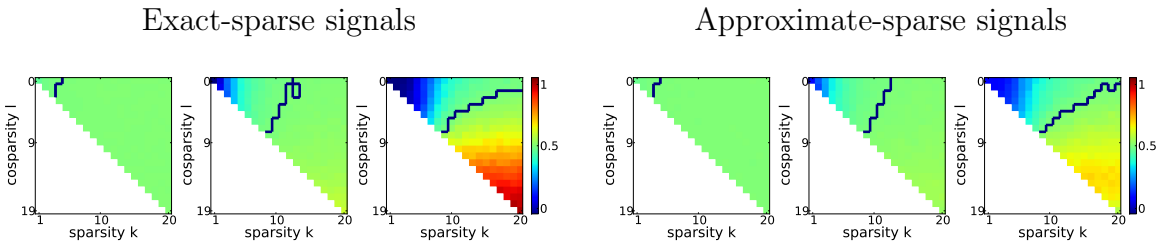
where M_{IJ} is a minor matrix obtained by keeping only the rows with indices I and the column with indices J from M, and γ_{S_J} is the restriction of γ_S to the indices J. Since γ_{S_J} is nonzero, this means that the k columns of the M_{IJ} are linear dependent, i.e. $rank(M_{IJ}) < k$.

Conversely, if an $l \times k$ minor matrix M_{IJ} of M has rank smaller than k, it means that its k columns are linear dependent, and therefore there exists a set of non-zero coefficients γ_{S_J} such as (3.24) is true. One has only to find such a solution of (3.24) and then place the coefficients on the locations J of γ_S. The signal x can be then generated as $x = D\gamma_S$. The least-squares solution $\gamma_A = D^\dagger x$ will have zeros at the locations in I.

When generating bisparse signals this way, it is possible for γ_A to accidentally have additional zero coefficients besides the locations in I. Thus, an $l \times k$ matrix M_{IJ} with rank smaller than k only guarantees that cosparsity l' of γ_A is at least l, but not necessarily equal to it, $l' \geq l$. □

Theorem 19 shows that it is always possible to find k-sparse signals that are also l-cosparse up to $l \leq k - 1$, irrespective of where the non-zero coefficients of γ_S and the zeros of γ_A are located, since $l < k$ implies that $rank(M_{IJ}) < k, \forall I, J$.

For $l \geq k$, however, there exist (k, l)-bisparse signals only if the sparsity and cosparsity patterns I and J happen to correspond to a rank deficient minor matrix M_{IJ} of $D^\dagger D$. Thus, such signals are not guaranteed to exist. Their existence is strictly



(a) $m = 3$ (b) $m = 11$ (c) $m = 17$ (d) $m = 3$ (e) $m = 11$ (f) $m = 17$

Figure 3.7: Ratio of average reconstruction errors obtained with synthesis and analysis recovery, respectively, for signals simultaneously k-sparse and l-cosparse. On the left side, the signals are exact-sparse, whereas on the right side a small non-sparse component of 1% energy is added. A small value indicates smaller errors with synthesis recovery, large values indicate smaller errors for analysis recovery. The dark separation line indicates the 0.5 frontier (similar performance)

determined by the distribution of linear dependent minors in the $D^\dagger D$ matrix. Our experiments show that the second case is negligible for a reasonable high overcompleteness factor (about $N/d > 1.5$), i.e. the vast majority of bisparse signals have $l < k$. An in-depth characterization of this distribution for a general dictionary D would be interesting, but is outside the scope of the current work.

3.3.2 Results: comparing synthesis and analysis recovery

For each of the n^2 pairs (k, l) with $k = 1, 2, ...n$ and $l = 0, 1, 2, ...n - 1$ we attempt to generate 1000 (k, l)-bisparse test signals with a dictionary D. We generate D as a random tight frame of size 20×50. The signals are generated according to the following procedure: (i) choose random subsets $I, J \subset \{1, 2, ...N\}$ with $card(I) = l$ and $card(J) = k$; (ii) check whether the minor matrix M_{IJ} has rank smaller than k (Theorem 19); (iii) if yes, find a random solution to (3.24) and place the coefficients on the locations J of γ_S; (iv) compute the actual signal $x = D\gamma_S$, and (v) compute $\gamma_A = D^\dagger x$, count the number l' of zeros and assign x to the set of (k, l')-bisparse signals.

To investigate robustness against approximate sparsity, when generating signals we optionally add a small non-sparse random component to the exact-sparse decomposition γ_S, with energy equal to 1% of γ_S. Thus the resulting signal x will be only approximately sparse and cosparse.

We take m zero-mean, unit-norm random linear measurements of each signal and reconstruct independently with synthesis-based recovery (3.4) and analysis-based recovery

(3.8), respectively. For synthesis recovery we use the Smooth L0 algorithm (SL0) [29]. For analysis recovery we also use a version of SL0 adapted to analysis recovery via the ABS-*mixed* approach, as detailed in Section 3.2.4. We define the percentage RMS error of the reconstructed signal $\hat{x}$ as

$$R(x) = \sqrt{\frac{\sum (x_i - \hat{x}_i)^2}{\sum x_i^2}}. \tag{3.25}$$

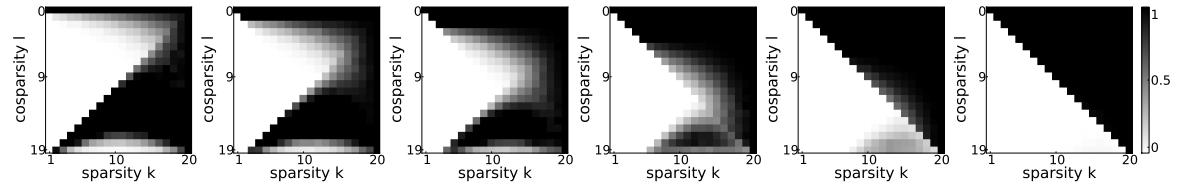
A smaller value of R indicates a better reconstruction, with $R = 0$ meaning perfect reconstruction. For every pair (k, l) we define the averaged error $R_{kl}^{S,A}$ as the average $R(x)$ for all the signals of the (k, l) pair, with indices S and A indicating synthesis or analysis recovery, respectively. The ratio R_{kl}^S / R_{kl}^A indicates which recovery is better: a value smaller than 1 indicates that synthesis reconstruction achieves lower average errors, otherwise analysis recovery is the better option.

In Fig.3.7, we plot the quantity

$$R_{kl} = \left(1 + \frac{R_{kl}^A}{R_{kl}^S}\right)^{-1} \tag{3.26}$$

for exact-sparse signals as well as approximate sparse signals. A value of $R_{kl} = 0.5$ indicates similar performance of the two reconstruction algorithms. If synthesis performance is better, the value of R_{kl} tends to 0, whereas if analysis is better, it approaches 1. For each (k, l) pair we attempt to generate 1000 signals. However, for $l \geq k$ we cannot guarantee to find as many, as explained in Section 3.3.1. We use the intensity of the color to indicate the number of (k, l)-bisparse signals actually found in 1000 attempts, with pure white indicating that no such signal could be found. We show the results for three different values of the m (3, 11, and 17) to illustrate the influence of the number of measurements.

One notices first that we could not find any (k, l)-bisparse signals for $l \geq k$, and thus we have no data for the lower triangular part of every plot. Further experiments confirm this finding for all other random dictionaries, as long as the overcompleteness factor N/d remains reasonably large (e.g. N/d larger than about 1.5). Under these assumptions, the results suggest that the $(k, k-1)$ main diagonal is in general a maximum limit for joint sparsity and cosparsity, i.e. a k-sparse signal cannot be in general more than $(k-1)$-cosparse at the same time. In other words, a very synthesis-sparse signal x cannot simultaneously be very analysis-cosparse for the same D, and vice-versa. However, this does not hold for certain dictionaries which enforce particular relations between the atoms. For example, if D is an equiangular tight frame, all the off-diagonal elements of $D^\dagger D$ have the same absolute value, and it is therefore much easier to find linear dependent minor



(a) $d \times N = 20 \times 21$ (b) $d \times N = 20 \times 22$ (c) $d \times N = 20 \times 23$ (d) $d \times N = 20 \times 25$ (e) $d \times N = 20 \times 27$ (f) $d \times N = 20 \times 30$

Figure 3.8: Percentage of successfully generated signals that are jointly k-sparse and l-cosparse, for a random dictionary of size $d \times N$ with small overcompleteness factor. White indicates that no (k, l)-bisparse signals could be generated in 1000 attempts, and black indicates the all 1000 (k, l)-bisparse signals have been generated.

matrices of $D^\dagger D$ of larger size than usual, and thus (k, l)-bisparse signals that are more sparse than usual. This suggests than the $(k, k-1)$ joint sparsity limit is valid only in a probabilistic sense. Even though, it can still be useful when working with learned dictionaries, which generally do not have any particular relation enforced between the atoms.

When approaching the complete case (N/d approaching 1), the pattern evolves dramatically, as depicted in Fig.3.8. Theorem 19 shows that finding a (k, l)-bisparse signal depends on the rank of the corresponding M_{IJ} minor matrix being smaller than k. The patterns in Fig.3.8 are, therefore, an illustration of the probability of finding linear dependent minor matrices of size $l \times k$ inside the matrix $D^\dagger D$, for random dictionaries D of various sizes. When D is a basis, $N/d = 1$, the plot concentrates strictly on the first diagonal. This is because in this case the two solutions γ_S and γ_A are identical, and therefore a k-sparse signal is automatically $(d-k)$-cosparse. Further analysis of these distribution patterns, although interesting, is however outside the scope of this work.

A second interesting observation is that analysis recovery performs well with sufficiently cosparse signals when the number of measurements is large, Fig.3.7(c), but is less reliable than its synthesis counterpart for fewer measurements and for approximately-sparse signals. In Fig.3.7(b), analysis recovery is not accurate even for the most cosparse signals (lower-right corner), while synthesis recovery works better for very sparse signals (upper-left corner). In Fig.3.7(d),(e),(f), a small non-sparse component of 1% energy is added to the test signals. One observes that analysis recovery is significantly more affected by this non-sparse component than synthesis recovery. We conclude therefore that analysis recovery is less robust to approximate sparsity and insufficient measurements than synthesis recovery.

3.3.3 Conclusions

In this section we conducted an experimental investigation into determining when is one of the analysis and synthesis sparsity models better than the other in terms of recovering a signal from a few random measurements. Our approach is based on reformulating analysis sparsity as least-squares constrained synthesis sparsity. We consider (k, l)-bisparse signals, i.e. signals that are simultaneously k-sparse in a random dictionary and l-cosparse with the pseudoinverse of that dictionary. We generate bisparse test signals for every possible (k, l) pair, reconstruct them separately using both synthesis and analysis recovery, and compare the average recovery errors obtained with the two methods.

The results indicate than the two recovery options perform similarly when recovering signals that are sparse according to the corresponding sparsity model, when the number of measurements is sufficient. However, analysis recovery is significantly more affected than synthesis recovery by a reduction in the number of measurements, and is also less robust with signals that are only approximately sparse. In addition, we find that for random dictionaries with reasonably large overcompleteness factor there is a limit of how much joint sparse and cosparse a signal can be, with a k-sparse signal being in general at most $(k-1)$-cosparse.

As future work, it will be interesting to conduct similar experiments with ℓ_1-sparse signals instead of exact-sparse, increasing the practical usefulness of the results. We also aim to fully investigate the influence of the dictionary size for small overcompleteness factors.

3.4 Chapter conclusions

Analysis sparsity is a new alternative to the well known synthesis sparsity model, introducing new opportunities for recovering a signal from a few random measurements with *a priori* information. Fundamental theory results indicate similar recovery guarantees as its synthesis counterpart.

In this chapter we analysed the relation between the two models, starting with the basic observation that analysis sparsity can be viewed as synthesis sparsity for the least-squares decomposition of a signal in an overcomplete dictionary. Based on this, the first part of the chapter introduces an innovative approach for signal recovery in the analysis model based on algorithms up to now designed solely for the synthesis formulation. The second part of this chapter constitutes of an experimental investigation that seeks to answer the question when is the analysis model preferable over its synthesis counterpart.

Chapter 4

Optimizing projections for compressed sensing

4.1 Introduction

The choice of the acquisition matrix P is governed by the principle of incoherence with the sparsity basis/dictionary D: a "good" acquisition matrix has its rows (i.e. the projection vectors) incoherent with the columns of D [58]. This fundamental result has lead to two major approaches for choosing acquisition matrices, especially in the well-studied case when D is an orthonormal basis. The first approach is to use a random matrix that has as elements i.i.d. random variables from certain distributions e.g. normal or Bernoulli. Such a random matrix is with high probability incoherent with any fixed basis or dictionary D. The second approach is to use orthogonal projection vectors taken from another orthonormal basis, known beforehand to be incoherent with D. This requires that the sparsity basis / dictionary D is known in advance, but allows more efficient recovery algorithms if the projection vectors are selected from a basis which has fast transform algorithms available (e.g. Fourier, wavelet).

In the overcomplete case, however, it is not uncommon that dictionaries exhibit significant atom correlation. This is especially true with dictionaries that are learned, i.e. optimized for a particular class of signals. In this case the acquisition matrix must be adapted to the dictionary, with the goal of ensuring that the sparse decomposition can be recovered from the measurements. A number of algorithms for finding optimized projections for signals that are sparse in overcomplete dictionaries have been developed [59, 60, 61].

This chapter reviews three well-known algorithms for finding optimized projections,

proposing modifications for improving the performance of all of them. We show that our improvements can be unified in a single formulation based on solving a *rank-constrained nearest correlation matrix* problem [62].

Throughout this chapter we shall use the following notations. The signal that we want to acquire is $x \in R^n$, the sparsity dictionary is D of size $n \times N$. A decomposition of x in D is typically denoted as γ. The acquisition matrix is P of size $m \times n$. The effective dictionary is $D_e = PD$, of size $m \times N$. The Gram matrix of D is denoted $G = D^T D$, while the Gram matrix of the effective dictionary D_e is denoted G_e and referred to as *effective Gram matrix*.

4.2 Acquisition matrices and mutual coherence

A widely used approach to ensure the uniqueness of the solution $\hat{\gamma}$ in (P_0) or (P_1) is by way of the mutual coherence of the effective dictionary $D_e = PD$ [2, 13, 12]. We introduced in Section 2.4.4 the definition of the mutual coherence as well as the sparse signal recovery guarantees. Theorems 10 and 11 presented in Section 2.4.4 shows that having a smaller mutual coherence of the dictionary is a desirable property, as it increases the set of recoverable signals. As such, most optimization algorithms are designed with the goal of minimizing the mutual coherence of the effective dictionary $D_e = PD$.

4.3 Existing optimization algorithms

4.3.1 The Elad algorithm

The algorithm in [59], which we henceforth call *Elad algorithm*, aims to reduce the t-averaged mutual coherence μ_t of the effective dictionary D_e, defined as the average of the largest off-diagonal values of the Gram matrix:

$$\mu_t(D_e) = \frac{\sum_{1 \le i,j \le k, i \ne j} (|g_{ij}| > t) \cdot |g_{ij}|}{\sum_{1 \le i,j \le k, i \ne j} (|g_{ij}| > t)} \tag{4.1}$$

The parameter t is either a fixed threshold or a percentage indicating the top fraction of the matrix elements that are to be considered. The reason for minimizing the t-averaged mutual coherence μ_t instead of the coherence μ is that the latter is a worst-case scenario bound. Even under more relaxed conditions than the ones in Theorem 10, in practice almost all signals can still be adequately recovered, at the expense of a small fraction of unrecoverable signals. For this reason, [59] argues that the t-averaged mutual coherence is a better measure for the average behaviour of the effective dictionary.

Algorithm 7 Elad algorithm

1: **repeat**
2: Compute the effective dictionary $D_e = P_k \cdot D$, normalize its columns, and compute its Gram matrix $G_e^{(k)} = D_e^T \cdot D_e$
3: Apply shrinking function to $G_e^{(k)}$, $\hat{G}_e^{(k)} = f\left(G_e^{(k)}\right)$
4: Find the best rank m approximation of $\hat{G}_e^{(k)}$ using SVD decomposition.
5: Extract square root D_k, where $\hat{G}_e^{(k)} = D_k^T \cdot D_k$
6: Find P_k as to minimize $||D_k - P_k D||_F$, i.e. $P_k = D_k D^\dagger$
7: **until** Until stop criterion

The algorithm's idea is to iteratively shrink the off-diagonal elements of the effective Gram matrix G_e, while keeping the rank of the matrix equal to $m < N$. At every iteration k, a shrinking function $f_t(u)$ is applied to the off-diagonal elements of the current Gram matrix $G_e^{(k)}$, that will reduce the largest values while keeping the lower values untouched. However, shrinking leads in general to a full rank matrix that is not semipositive definite (SPD), whereas the Gram matrix of any $m \times N$ dictionary is required to be SPD and of rank at most m. Therefore the algorithm finds the best rank-m approximation of the shrinked Gram matrix, then extracts the square root D_k, which presumably is a better effective dictionary than the initial one. The acquisition matrix at step k, denoted as P_k, is then found as the matrix which brings D as close to the desired D_k as possible, i.e. $P = D_k D^\dagger$, (where † denotes the Moore-Penrose pseudoinverse).

The shrinking function is empirically chosen as in (4.2), where the parameter α is the shrinkage factor.

$$f_t(u) = \begin{cases} u & \|u\| \leq \alpha t \\ \alpha t \cdot sgn(u) & \alpha t \leq \|u\| \leq t \\ \alpha u & t \leq \|u\| \end{cases} \tag{4.2}$$

The complete algorithm is summarized in Algorithm 7. The algorithm requires as input the dictionary D, an initial acquisition matrix P_0 and the shrinking function $f_t(u)$. The output is the optimized acquisition matrix P.

The algorithm is typically run for a predefined number of iterations. However, convergence is not guaranteed, therefore the recommended stopping criterion is by checking the resulting t-averaged mutual coherence after each iteration and stopping when a significant increase is observed.

Algorithm 8 Xu algorithm

1: **repeat**

2: Compute the effective dictionary $D_e = P \cdot D$, normalize its columns and compute the Gram matrix $G_e^{(k)} = D_e^T \cdot D_e$

3: Project G_e on Λ^k by enforcing:

$$g_{ij} = \begin{cases} 1 & i = j \\ g_{ij} & |g_{ij}| < \mu_G \\ sgn(u) \cdot \mu_G & |g_{ij}| \geq \mu_G \end{cases} \tag{4.5}$$

4: New solution is between the projection G_P and the previous solution:

$$G_k = \alpha G_P + (1 - \alpha) G_{k-1}, 0 < \alpha < 1 \tag{4.6}$$

5: Update the acquisition matrix P using QR factorization with eigenvalue decomposition

6: **until** Until stop criterion

4.3.2 The Xu algorithm

The algorithm presented in [60], which we refer to as the *Xu algorithm*, aims to make the effective dictionary D_e as close as possible to an equiangular tight frame (ETF), because an ETF has minimal coherence among all matrices of the same dimensions. Thus, it aims to solve the optimization problem

$$\hat{G_e} = \min_{G_G \in A_m^n} ||G_e - G|| \tag{4.3}$$

where A_m^n is the set of the Gram matrices of all $m \times n$ ETFs. Since this set is not convex, they replace it with the convex set Λ^n:

$$\Lambda^n = \{G \in \mathbb{R}^{n \times n} : G = G^T, diag(G) = 1, \max_{i \neq j} \|g_{ij}\| < \mu_G\} \tag{4.4}$$

where $\mu_G = \sqrt{\frac{n-m}{m(n-1)}}$ is a lower bound for the coherence of an $m \times n$ ETF. The algorithm itself, presented as Algorithm 8, is based on the method of alternating projections, and is very similar to Elad algorithm.

One observes that the Xu algorithm is very similar to the Elad algorithm. Projection on Λ^k is very similar to the shrinking step in the Elad algorithm, only with a modified

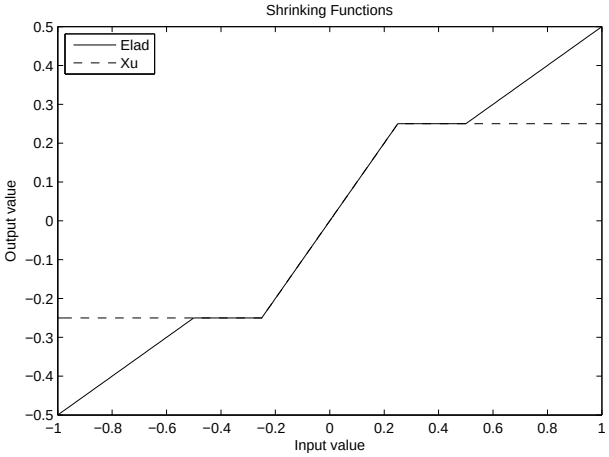


Figure 4.1: Comparison of shrinking functions in Elad and Xu algorithms

shrinking function that hard-limits all the off-diagonal elements larger than μ_G. The two shrinking functions are shown in Fig.4.1 for comparison. The only other difference is that the new solution is selected as an intermediate point between the previous solution and the result of the projection, in an effort to improve the algorithm stability. Apart from these two differences, the two algorithms are essentially the same.

4.3.3 The Duarte algorithm

Duarte-Carvajalino & Sapiro introduce in [61] a different algorithm for finding optimized projections for a given fixed dictionary, as well a a method of jointly optimizing the acquisition matrix and the dictionary. Since we consider fixed (i.e. given) dictionaries, we will focus only on the former, which we refer to as the *Duarte algorithm* for brevity. The authors seek the acquisition matrix $P^\star$ that minimizes:

$$P^\star = \underset{P}{\operatorname{argmin}} \, ||DD^T - DD^T P^T P DD^T||_F \tag{4.7}$$

The problem (4.7) has a closed-form solution in the form

$$P^\star = \Lambda_{1:m}^{-1/2} \cdot U_{1:m}^T \tag{4.8}$$

where Λ and U come from the eigenvalue decomposition of $DD^T = U\Lambda U^T$ and the notation $_{1:m}$ indicates a restriction to the top m rows. In other words, considering an SVD decomposition of $D = USV^T$, the optimal acquisition matrix is given by the top m principal components of D scaled with the inverse of the corresponding singular values, $P^\star = S_{1:m}^{-1} U_{1:m}^T$. As a consequence, the effective dictionary is a tight frame defined by the restriction of V^T to the top m rows:

$$D_e = P^\star \cdot D = S_{1:m}^{-1} U_{1:m}^T \cdot USV^T = V_{1:m}^T \tag{4.9}$$

4.4 Improving existing optimization algorithms

4.4.1 Reformulating as constrained optimization

As a first step towards improving the algorithms, we try to reformulate the algorithms as constrained optimization problems.

We note that the Elad algorithm can be thought of as a way of robustly solving

$$\begin{aligned} &\min ||G_e - I_N||_\infty \\ &\text{s.t. } G_e \succeq 0 \\ &\quad \text{rank}(G_e) = m \\ &\quad \text{diag}(G_e) = 1 \end{aligned} \tag{4.10}$$

by iteratively shrinking the top largest off-diagonal elements of G_e with a custom shrinking function. Indeed, at every iteration the largest off-diagonal values are shrinked, followed by enforcing the rank and unit-diagonal constraints. Reducing the largest off-diagonal values effectively means reducing the ℓ_∞ norm of the distance $||G_e - I_N||_\infty$. Thus, Elad's algorithm can be thought as an iterative method for constrained ℓ_∞ minimization.

A similar reasoning holds for the Xu algorithm. It is well known that the maximum correlation of two atoms is minimal for an ETF among all other matrices of same size [63, 64]. It follows that projecting on the set of ETFs is similar to minimizing the largest absolute off-diagonal value of G_e. As such, the Xu algorithm can also be thought of as a way of solving the same constrained optimization problem as the Elad algorithm

$$\begin{aligned} &\min ||G_e - I_N||_\infty \\ &\text{s.t. } G_e \succeq 0 \\ &\quad \text{rank}(G_e) = m \\ &\quad \text{diag}(G_e) = 1 \end{aligned} \tag{4.11}$$

by iteratively shrinking the large off-diagonal elements of G_e and enforcing the constraints, only with a custom shrinking function that is slightly different from the Elad algorithm.

The case of the Duarte algorithm is different. The algorithm starts from the problem of minimizing the distance between the same two matrices, but with the ℓ_2 norm instead of ℓ_∞:

$$||I_N - G_e||_2,$$

but then both sides of the two terms are multiplied with D and D^T, leading to a relaxed optimization problem of minimizing

$$||G - DG_eD^T||_2.$$

The latter problem is a relaxed version of the former since the "target" matrix G is now of smaller rank (n) than the original I_N. The reason for this modification, however, is not explained in the original paper.

As such, we propose an alternative view on the Duarte algorithm. We observe that the Duarte algorithm is very similar to solving the following optimization problem:

$$\begin{aligned} &\min||G_e - G||_2 \\ &\text{s.t. } G_e \succeq 0 \\ &\quad \text{rank}(G_e) = m \end{aligned} \tag{4.12}$$

i.e. trying to make G_e as close as possible to G under rank and semipositivity constraints. The justification is as follows. Considering an SVD decomposition of $D = USV^T$, the solution to (4.12) is obtained by keeping only the most significant m left singular vectors and values of D

$$P^\star D = S_{1:m} * V_{1:m}^T$$

which leads to the optimal acquisition matrix being

$$P^\star = U_{1:m}^T.$$

The only difference from the Duarte solution (4.8) is that the projection vectors resulting from (4.12) have unit norm, whereas the vectors from the Duarte solution of (4.8) are scaled with the inverted singular values. However, the scaling of the acquisition vectors is not very important, and in a practical scenario it may actually prove more robust to have unit-norm acquisition vectors. Thus, one can think of Duarte's algorithm as essentially a way of solving (4.12) followed by scaling the projection vectors.

4.4.2 Improving the Elad and Xu algorithms

We analyse the Elad and Xu algorithms in two corner cases that suggest that the shrinkage of the Gram matrix may be inadequate.

In the first situation, let us consider that the dictionary D contains two identical atoms d_i and d_j. While this can be regarded as an extreme scenario, it may happen that an atom is a linear combination of some very few other atoms, since D is overcomplete. In any case, since the dictionary D is given as an input to the optimization algorithm and is therefore out of its control, we would naturally require that the optimization algorithm is robust to this scenario.

If columns d_i and d_j are identical, then the effective dictionary $D_e = PD$ will also have two identical columns d_{e_i} and d_{e_j}, for any P. The effective Gram matrix G_e will therefore

have a pair of 1's outside the main diagonal. The mutual coherence of the effective dictionary is thus maximal, and both algorithms work to shrink these off-diagonal 1's. The reason for which values of 1 outside the main diagonal are bad is that two identical columns in the effective dictionary leads to ambiguity and non-uniqueness of the resulting solution vector: if atoms i and j are identical, the i^{th} and the j^{th} coefficients of a solution vector can be swapped and combined in any way and still yield a valid solution as long as their sum remains the same.

We observe, however, that the effort of reducing the off-diagonal 1's in this case is unnecessary since this ambiguity in the effective dictionary is actually *inherited* from the dictionary D. No matter how a recovery algorithm places the i^{th} and j^{th} coefficients in the solution vector γ, when we reconstruct the original signal x as $x = D\gamma$ we obtain the same result because the same ambiguity of the i^{th} and j^{th} atoms is also present in D. The correct signal x is generated irrespective of whether the i^{th} or the j^{th} are swapped. Therefore, in this case the optimization algorithm should not worry about the off-diagonal 1's of the Gram matrix. We may think of this as a guideline that *correlations in the effective dictionary D_e inherited from the original dictionary D are not bad*. Any optimization algorithm shouldn't try to decrease atom correlations in the effective dictionary beyond the correlations present in the original dictionary D itself.

In the second scenario, consider that the initial acquisition matrix is simply the full-size identity matrix I_n. Of course, in this case there is no compression, the measurement vector y is identical with x, the effective dictionary D_e is simply D, and any decomposition of y in D_e leads to a perfect reconstruction of x. In this case there is nothing to be optimized, as the acquisition matrix is already perfect, irrespective of the coherence of D. However, both Elad and Xu algorithms fail to notice this fact. They compute the Gram matrix $G_e = D_e^T D_e = D^T D$ and proceed to shrink its large off-diagonal values as usual, aiming to make the dictionary more incoherent or closer to an ETF, respectively. But since the acquisition matrix is already optimal, the algorithms are clearly pursuing an overly ambitious goal.

In the view of the two scenarios above we conclude that the goals of the Elad and Xu algorithms are set too high. Instead of improving the *absolute* coherence of the effective dictionary D_e by indiscriminately shrinking the large off-diagonal elements of its Gram matrix G_e, we propose to take into account the Gram matrix G of the original dictionary D. While the Elad algorithm tries to make the Gram matrix G_e closer to I_N and the Xu algorithm tries to make G_e closer to the Gram matrix of a ETF, our approach is instead to try and make G_e as close as possible to the Gram matrix G of the original dictionary.

This means reducing the correlations of the atoms of D_e, but not beyond the original correlations of the atoms in D. This is appropriate in the two scenarios above: if there are two identical columns in D, the algorithm does not try to orthogonalize them in D_e, and if $G_e = G$, the algorithm simply stops, as nothing can be further improved.

We therefore propose the modification of the two algorithms in the following manner: instead of shrinking the matrix G_e directly, we propose to shrink the difference matrix $G - G_e$, thus aiming to make G_e progressively closer to G. This amounts to replacing I_N with G in (4.10) and (4.11), similar to (4.12). We refer to these algorithms as Elad-G and Xu-G, and their complete description is given in Section 4.5.

4.4.3 Improving the Duarte algorithm

The Duarte algorithm presented in Section 4.3.3 is choosing the projection vectors to be the principal components of the dictionary, scaled with the inverse of its singular values. As mentioned in Section 4.4.1, this can be thought of as minimizing (4.12) followed by a rescaling of the resulting projection vectors with the inverse of the dictionary's singular values. Unlike Elad and Xu algorithms, therefore, it aims to make G_e as close as possible to G, not to the identity matrix. In this respect it is similar to the modifications we proposed above for the Elad and Xu algorithms.

There is however an essential condition missing from (4.12) as well as the original Duarte formulation: there is no guarantee that the atoms of effective dictionary D_e are normalized, i.e. $diag\left(\hat{G}_e\right) = 1$. The resulting effective dictionary, as mentioned in Section 4.3.3, is a tight frame obtained as the top m rows of the unitary matrix V^T, where $D = USV^T$ is an SVD decomposition of D. This is not, however, a unit-norm tight frame.

The mutual coherence of a dictionary is equal to the maximum off-diagonal element of the Gram matrix only if the atoms are normalized. Minimizing atoms' correlations without ensuring that they are normalized is not so effective. For example, consider two atoms whose inner product is 0.5; if their norm is 1, the angle between them equals 60 deg, whereas if their norm is 0.9, their angle is only 52 deg. Thus, minimizing $G - G_e$ without imposing atom normalization will result in the atoms being more coherent than initially thought. Note that Elad and Xu algorithm explicitly performed a normalization of the effective dictionary at every iteration.

A worst case scenario for the Duarte algorithm is when $V^T = I_N$, and restriction to the top m rows results in an effective dictionary which contains some all-zero atoms. An an example, consider the case when D is as simple as an orthogonal matrix, but whose

atoms are not normalized. This may arise, for example, in a practical application with limited precision. An SVD decomposition of D is therefore $D = D_n S I_N$, where Dn is the normalized D and S is a diagonal matrix containing the norms of the atoms. The optimal acquisition matrix found by the Duarte algorithm is

$$P^\star = S_{1:m}^{-1} D_{n_{1:m}}^T.$$

However since D is orthogonal, this acquisition matrix is orthogonal to the last $(N-m)$ atoms of the dictionary, resulting in an effective dictionary with $(N-m)$ all-zero columns

$$D_e = [I_m; 0].$$

Thus, the measurements do not capture anything about the remaining $(N-m)$ atoms, and thus one can never hope to recover any of these components.

We propose therefore a more accurate optimization problem that constrains the effective atoms to have unit norm:

$$\begin{aligned} &\min ||G_e - G||_2 \\ &\text{s.t. } G_e \succeq 0 \\ &\quad \text{rank}(G_e) = m \\ &\quad \textbf{and } \text{diag}(G_e) = 1 \end{aligned} \tag{4.13}$$

As explained in the next section, this problem is a rank-constrained nearest correlation matrix problem (RCNCM), and can be solved with algorithms developed for robustly estimating correlation matrices [65, 62]. We refer to this problem as *RCNCM*.

4.5 Rank-constrained nearest correlation matrix for optimized projections

The considerations in Section 4.4 lead us to proposing the following class of optimization problems from choosing the best acquisition matrix:

$$\begin{aligned} &\min ||G_e - G||_p \\ &\text{s.t. } G_e \succeq 0 \\ &\quad \text{rank}(G_e) = m \\ &\quad \text{diag}(G_e) = 1 \end{aligned} \tag{4.14}$$

This formulation is a natural generalization of all the three algorithms presented above. For $p = 2$, the problem reduces to the Duarte optimization problem with the additional unit-norm constraint, as introduced in 4.4.3. For $p = \infty$, (4.14) can be solved with the modified versions of the Elad and Xu algorithms introduced in Section 4.4.2.

The optimization problem (4.14) is a rank-constrained nearest correlation matrix problem [62]. This family of problems has received much attention in the recent years, with applications in finance as well as engineering. A matrix X is called a *correlation matrix* if $X \succeq 0$ (i.e. it is semipositive definite) and $X_{ii} = 1$. In many practical scenarios, the correlation matrix estimated from noisy, unreliable or possibly incomplete data can turn out to violate the rank and positivity constraints required of a correlation matrix. In these cases, one needs to find a matrix that fulfils the constraints and is close as possible to the input matrix using a distance metric. This leads to an optimization problem formulated as in (4.14).

4.5.1 Solving for $p = 2$

When considering $p = 2$, the optimization problem (4.14) becomes similar to the Duarte problem with additional atom normalization constraint. This constraint prevents having a simple closed-form solution as in the original Duarte problem.

For $p = 2$ several approaches have been developed for solving (4.14) [65, 62]. In this paper we use the algorithm presented in [62], based on eigenvalue penalization and majorization, which we summarize here. First, let us note that in absence of the rank constraint, the problem would be a convex optimization problem and thus tractable [66]. To enforce the additional rank-constraint, the authors of [62] propose to minimize a function that penalizes the last $(N - m)$ eigenvalues

$$\begin{aligned} \textbf{minimize } & f(G_e) = ||G_e - G||_2 + c \sum_{m+1}^{N} \lambda_i \\ \text{s.t. } & G_e \succeq 0 \\ & \text{diag}(G_e) = 1 \end{aligned} \tag{4.15}$$

where c is a trade-off constant. This is not equivalent to the original problem (4.14), but for a large enough value of c the solution of (4.15) is arbitrarily close to the solution of (4.14). Solving (4.15) is achieved iteratively using the majorization technique [65], which consists in solving a sequence of simpler convex optimization problems: given the estimate $G_e^{(k)}$ at iteration k, one constructs a simpler convex function g_k that majorizes f, $g_k(X) \geq f(X)$, $\forall X$, and minimizes g instead, obtaining the new estimate $G_e^{(k+1)}$. The

Algorithm 9 Elad-G algorithm

1: **repeat**
2: Compute the effective dictionary $D_e = P_k \cdot D$, normalize its columns, and compute its Gram matrix $G_e^{(k)} = D_e^T \cdot D_e$
3: Compute the difference $\Theta^{(k)} = G_e^{(k)} - G$, where $G = D^T D$
4: Apply shrinking function to $\Theta^{(k)}$, $\hat{\Theta}^{(k)} = f\left(\Theta^{(k)}\right)$
5: Compute the estimate matrix by adding back G: $\hat{G_e}^{(k)} = \hat{\Theta}^{(k)} + G$
6: Find the best rank m approximation of $\hat{G_e}^{(k)}$ using SVD decomposition.
7: Extract square root D_k, where $\hat{G_e}^{(k)} = D_k^T \cdot D_k$
8: Find P_k as to minimize $||D_k - P_k \cdot D||_F$, i.e. $P_k = D_k D^\dagger$
9: **until** Until stop criterion

sequence of estimates $G_e^{(k)}$ converges to the solution of (4.15). Further details can be found in [62].

Once the optimal G_e is found, the optimal acquisition matrix $P^\star$ is obtained, as in the other algorithms, by first factorizing $G_e = (D_e^\star)^T D_e^\star$ and then $P^\star = D_e^\star D^\dagger$.

We refer to this algorithm for finding optimized projections as *RCNCM*.

4.5.2 Solving for $p = \infty$

For $p = \infty$, the problem (4.14) can be solved with the modified versions Elad-G and Xu-G introduced in Section 4.4.2, by iteratively shrinking the top largest elements of the difference matrix $G_e - G$. The essential difference to the original algorithms is that we make G_e as close as possible to G rather than to I_N or the Gram matrix of an ETF. As such, the modified algorithms behave appropriately in the two corner cases analysed in Section 4.4.2: if the original dictionary D happens to have identical atoms, the algorithms do not inadvertently try to orthogonalize them in D_e, and if the initial acquisition matrix is the identity matrix, they stop because there is nothing to optimize.

The complete description of the two proposed algorithms Elad-G and Xu-G, is given as Algorithm 9 and Algorithm 10. We point our that although Xu-G keeps the same projection step in the original Xu algorithm, this has little relation with the original purpose of projecting on the set of ETFs. Instead, we consider projecting on Λ^k as just a different way of reducing large off-diagonal values from the one in Elad-G, by hard-limiting large values instead of reducing them by a linear factor.

Algorithm 10 Xu-G algorithm

1: **repeat**

2: Compute the effective dictionary $D_e = P \cdot D$, normalize its columns and compute the Gram matrix $G_e^{(k)} = D_e^T \cdot D_e$

3: Compute the difference $\Theta^{(k)} = G_e^{(k)} - G$, where $G = D^T D$

4: Project $\Theta^{(k)}$ on Λ^k by enforcing:

$$g_{ij} = \begin{cases} 1 & i = j \\ g_{ij} & |g_{ij}| < \mu_G \\ sgn(u) \cdot \mu_G & |g_{ij}| \geq \mu_G \end{cases} \tag{4.16}$$

5: Add back G: $G_P = \hat{\Theta}^{(k)} + G$

6: New solution is between the projection G_P and the previous solution:

$$G_k = \alpha G_P + (1 - \alpha) G_{k-1}, 0 < \alpha < 1 \tag{4.17}$$

7: Update the acquisition matrix P using QR factorization with eigenvalue decomposition

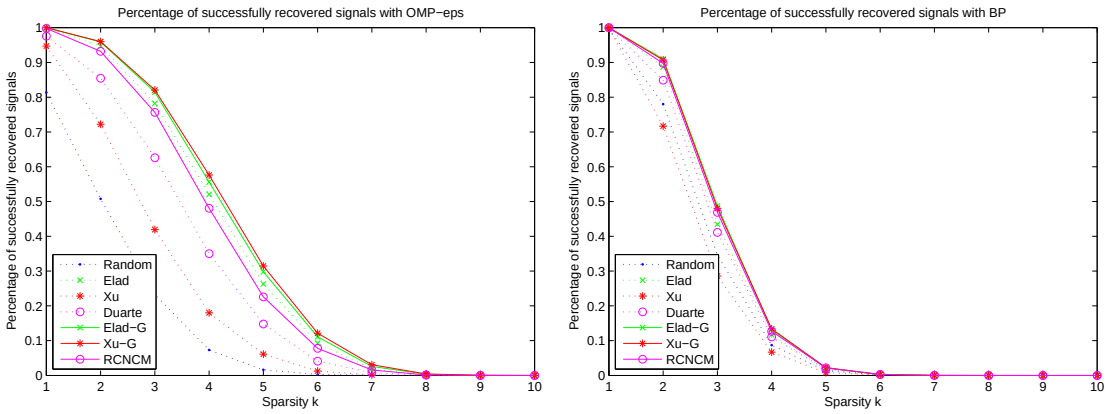
8: **until** Until stop criterion

4.6 Simulation results

We test the performance of the three proposed algorithms Elad-G, Xu-G and RCNCM in four different setups:

1. Random dictionary, exact sparse data
2. Dictionary is a random orthogonal matrix with slightly different atom norms, exact-sparse data
3. Real dictionary obtained with K-SVD from image patches, exact-sparse data
4. Real dictionary obtained with K-SVD from image patches, real image patches

The algorithms under test are: (i) random acquisition matrix with i.i.d. normal elements, (ii) Elad algorithm, (iii) Xu algorithm, (iv) Duarte algorithm, (v) proposed Elad-G algorithm, (vi) proposed Xu-G algorithm and (vii) proposed RCNCM algorithm.



(a) OMP (b) ℓ_1-minimization

Figure 4.2: **Random dictionary, exact-sparse data: successful recovery.** Percentage of successfully recovered exact-sparse signals using various projection optimizing algorithms. The number of measurements is $m = 16$, the dictionary is a random matrix of size 64×256.

For reconstruction we use (i) Orthogonal Matching Pursuit [14] with stopping criterion being the residual error below a certain threshold ϵ, denoted as *OMP-*ϵ, and (ii) ℓ_1 minimization with linear programming (denoted as *BP*). The OMP residual ϵ is set to 10^{-9} when reconstructing exact-sparse data, and 10^{-3} for data being real image patches (test 4).

The signal dimension is $n = 64$ and the dictionary size is $N = 256$. All the results are averaged over 10 different random initializations, i.e. we generate 10 optimized acquisition matrices using 10 initial random matrices for every test.

4.6.1 Test 1: Random dictionary, exact-sparse data

In the first case, we randomly generate a dictionary as a $\mathbb{R}^{n\times N}$ matrix with i.i.d. random normal entries, and we generate exact-sparse data as random linear combinations of atoms.

Fig.4.2 shows the probability of exact reconstruction for test 1, for different sparsity levels k, when using $m = 16$ optimized measurements with the algorithms under test, i.e. a compression ratio of $1/4$. One can see that in this case all optimizing algorithms perform similarly. This is because the dictionary is generated randomly with i.i.d elements, and thus the atoms have little correlations. As such, there are no significant correlations in the dictionary that the algorithms could exploit, and therefore the optimization algorithms perform close to each other.

Table 4.1: Percentage of successful recovery with orthogonal but not perfectly normalized dictionary and exact-sparse data

	Percentage of successfully recovered signals					
	OMP-ϵ			BP		
Projections	$k=2$	$k=4$	$k=6$	$k=2$	$k=4$	$k=6$
Random	87.1%	47.2%	12.7%	95.1%	46.7%	8.2%
Elad	99.3%	92%	60.8%	98.4%	58%	11.2%
Xu	100%	94.5%	63.7%	100%	60.7%	11.7%
Duarte	**5.9%**	**0.4%**	**0.0%**	**5.9%**	**0.4%**	**0.0%**
EladG	99.3%	92%	60.8%	98.4%	58%	11.2%
XuG	100%	94.7%	64.2%	100%	60.2%	11.9%
RCNCM	99.2%	86.8%	49.5%	98.7%	59%	12.2%

4.6.2 Test 2: Random orthogonal dictionary with inaccurate normalization, exact sparse data

The second scenario illustrates the problem of the Duarte algorithm, simulating a dictionary with imprecise normalization. The dictionary is created as a random orthogonal square matrix, with the atoms norms randomly selected in the interval $(1-10^{-6}, 1+10^{-6})$. The differences between atom norms are therefore very small. We generate exact-sparse data for this dictionary as random linear combinations of its atoms.

Table 4.1 shows the probability of exact reconstruction of exact-sparse data, when using $m = 16$ optimized measurements with the algorithms under test. One observes that the Duarte algorithm severely compromises the acquisition. All other algorithms have close performance, since there are no correlations between the atoms that can be exploited.

We conclude therefore that the Duarte algorithm is not very robust in some common scenarios such as this one.

4.6.3 Test 3: Learned dictionary, exact-sparse data

In the third scenario, we generate the dictionary from a set of image patches obtained by randomly selecting 150 patches of size 8×8 from each of 37 test images from the

miscellaneous section of the USC-SIPI image database, resulting in a total of 5550 patches (there are 47 images in the database, but we removed 10 of them that were too uniform and affected the dictionary learning algorithm). The patches are reshaped columnwise as 64×1 vectors. The dictionary consists of $N = 256$ atoms. The K-SVD algorithm is used to train the dictionary, using all the patches as training set. We then create exact-sparse test data by finding the best k-term approximation of the patches in the learned dictionary with OMP. As a consequence, the dictionary atoms are not used uniformly in the decompositions, thus better modelling a real-life scenario.

Fig.4.3 shows the probability of exact reconstruction for test 3, when using $m = 16$ optimized measurements with the algorithms under test, for different values of the sparsity k. In this case, one sees significant differences between the algorithms.

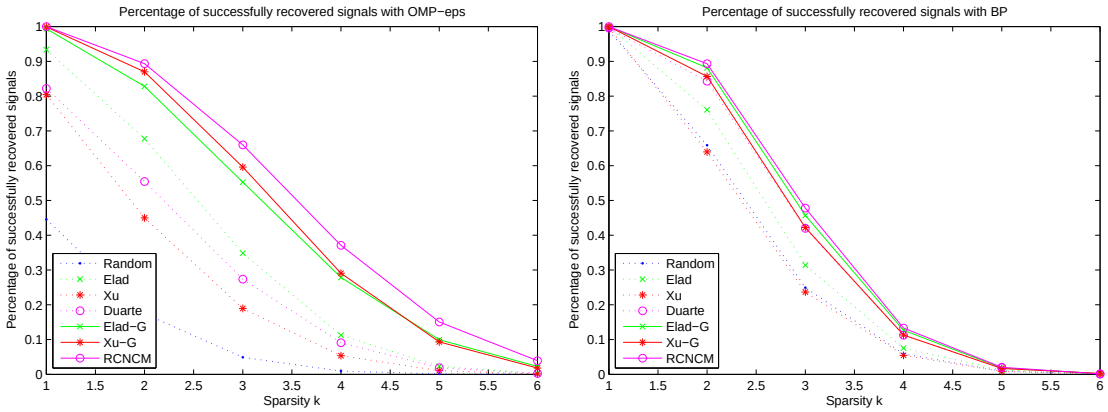
When recovering with OMP, our proposed algorithms significantly outperform all the existing ones, with RCNCM being the winner, followed by Elad-G and Xu-G. For ℓ_1 minimization the improvements over Duarte are smaller, whereas the improvement over Elad and Xu algorithms remain similarly large. Similar improvements are obtained when using Smoothed-ℓ_0 (SL0) [29] for recovery.

Fig.4.4 shows the histogram of the off-diagonal elements of the effective Gram matrix G_e. It shows that the Elad, Xu and Duarte algorithms are indeed reducing the effective dictionary coherence, however this does not translate to improved recovery in Fig.4.3. This constitutes therefore a proof that indiscriminately shrinking off-diagonal elements of G_e, as Elad and Xu algorithms achieve, does not constitute alone a guarantee for improved recovery.

4.6.4 Test 4: Learned dictionary, patches data

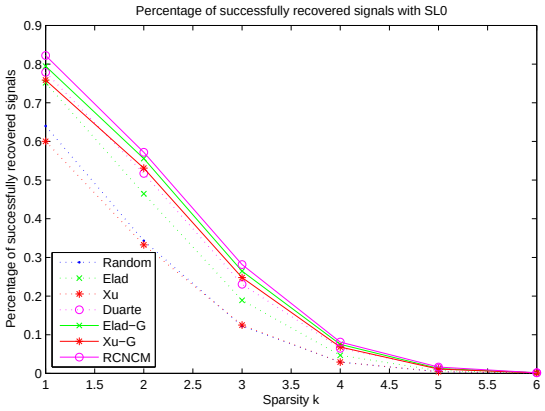
In the final scenario the dictionary is created as in Test 3, i.e. learned with K-SVD from a set of 5550 randomly selected patches from 37 test images, but we use the actual set of patches as testing data. As such, the data is not perfectly sparse and we cannot consider successful reconstruction as a performance criterion. Fig.4.5 shows instead the Mean Squared Error (MSE) of the recovered signals, when varying the number of measurements.

We see that Duarte's algorithm slightly outperforms our proposed RCNCM for reduced number of measurements, Elad-G and Xu-G performing worse than them but still better than the original Elad and Xu algorithms. However, in this scenario PCA (i.e. projection on principal components followed by linear reconstruction) actually performs better than any of the compressed sensing acquisition schemes. This suggests that the data is not sufficiently sparse in the considered dictionary in order for a compressed sens-



(a) OMP

(b) ℓ_1-minimization



(c) SL0

Figure 4.3: **Learned dictionary, exact-sparse data: successful recovery.** Percentage of successfully recovered signals using projection optimizing algorithms. The number of measurements is $m = 16$, the dictionary learned from real patches and has significant correlations, the data is exact-sparse.

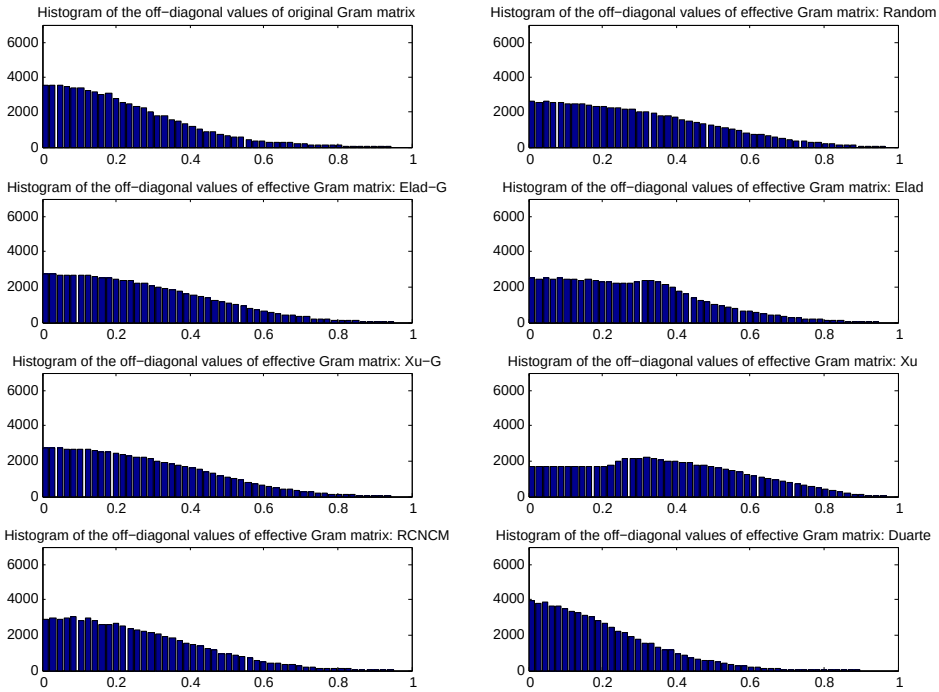
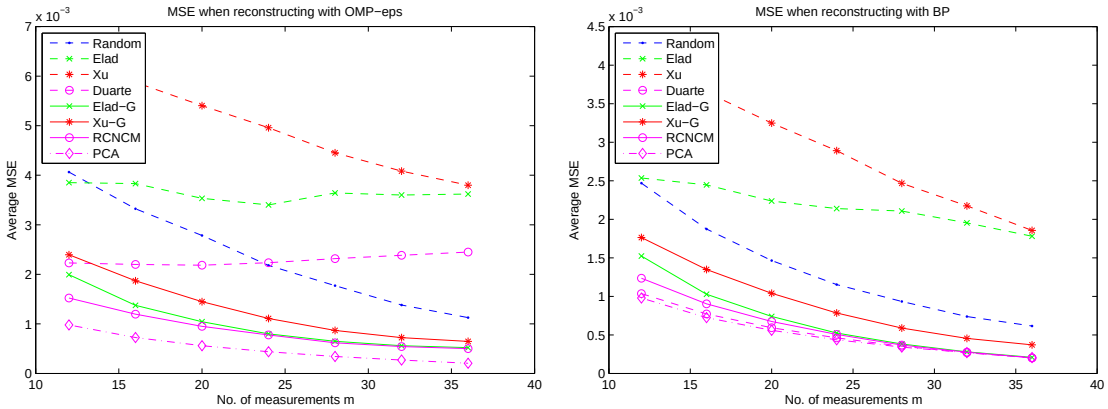


Figure 4.4: **Learned dictionary, exact-sparse data.** Histogram of the effective Gram matrix off-diagonal elements



(a) OMP (b) ℓ_1-minimization

Figure 4.5: **MSE: Learned dictionary, patches data.** Average MSE of recovered signals using various projection optimizing algorithms. The data are random image patches from the database, the dictionary is learned from random image patches and has significant correlations.

ing scheme to be useful, and thus the relevance of this test is limited. The performance of the Duarte projections can be explained in this case through their similarity with the principal components, since they only differ in normalization.

4.7 Conclusions

In this section we focus on optimizing the acquisition matrix for compressively sensing signals that are sparse in overcomplete dictionaries, possibly with high atom intercorrelations. We start by reviewing and improving three existing state-of-the art algorithms, and we identify corner cases when the algorithms perform sub-optimally. We argue that the Elad and Xu optimization algorithms can be improved by bringing the Gram matrix of the effective dictionary progressively closer to the Gram matrix of the dictionary, rather than to the identity matrix. The Duarte algorithm is based on a similar idea; however, it does not guarantee that the resulting effective dictionary is normalized, thus estimating the coherence inaccurately. We propose improving the algorithm by adding an additional unit-norm constraint to the optimization problem.

All the three proposed algorithms can be viewed as special instances of a single unified formulation: find the matrix that is of minimal distance from the dictionary Gram matrix, subject to rank, unit-norm and semipositivity constraints. This is known as the rank-constrained nearest correlation matrix problem. When the distance metric is the usual

ℓ_2 (Frobenius) norm, the problem is similar to the Duarte optimization problem with the proposed additional unit-norm constraint. Several algorithms have already been developed for this optimization problem, and we use an existing algorithm based on a majorized penalty approach. When the distance metric is ℓ_∞, one can use the modified Elad and Xu algorithms, iteratively minimizing the largest entries of the difference matrix.

We test the proposed algorithms considering random and learned dictionaries. Results show significant improvements for learned dictionaries, our proposed approach being capable of better exploiting the correlations of the dictionary atoms than the Elad and Xu algorithms. The Duarte algorithm is also surpassed when recovering with OMP, but performs as good as the proposed algorithms for ℓ_1 minimization. However, it completely fails with poorly normalized dictionaries, a problem which does not affect any other algorithm. We conclude that our formulation the optimization problem as a rank-constrained nearest correlation matrix problem is a more accurate and at the same time a more robust approach to optimizing the acquisition matrix.

Chapter 5

Compressed sensing with correlated projection vectors

5.1 Introduction

Recently, the successful use of random projection matrices in compressed sensing theory was proven without the RIP mechanism [67]. The authors of [67] treat the rows of the projection matrix as random vectors from some multivariate probability distribution F, and impose conditions on F rather than on each element of the matrix. They derive conditions analogous to the Restricted Isometry Property (RIP) conditions by implying, among others, that the probability distribution is isotropic.

In this chapter, we also treat the rows of the projection matrix as random vectors from a multivariate distribution, but we consider correlated and distorted probability distributions, obtained after some linear transformation of the initial space of a set of uncorrelated vectors. While the vector distribution in [67] is spherical and uncorrelated (i.e. isotropic, the covariance matrix is the canonical matrix), we consider distribution that can any real symmetric covariance matrix. In other words, while their random projection vectors are uniformly pointed in all directions in space, our projection vectors can point with higher probabilty into some directions in the space (the significant eigenvectors of the distribution's covariance matrix) and with a lower probability towards other directions (least significant eigenvectors).

We show that choosing projection vectors from a non-isotropic distribution affects the covariance matrix of the resulting recovery errors. Surprisingly, we experimentally establish a precise relationship between the covariance matrix of the projection vectors and that of the errors, up to a scaling factor. Even more, we find that this relation holds

unchanged for a number of very different solving algorithms (ℓ_1 minimization (Basis Pursuit, BP) [5], Orthogonal Matching Pursuit (OMP) [14] and Smoothed L_0 (SL0) [29]). Practically, this allows a sort of *error shaping*, i.e. the possibility of controlling the covariance matrix of the errors by appropriately choosing the covariance matrix of the projection vectors. This provides significant benefit in applications like compressed sensing with non-orthogonal bases or with unequal atom importance.

5.2 Problem statement

We describe below in Section 5.2.1 two scenarios that require some sort of *error shaping* mechanism, i.e. a way of controlling the energy of the errors along some critical directions of the space. A first approximation of the shape of a distribution is given by its covariance matrix. The covariance matrix of a distribution specifies the estimated variance of the random vectors along a set of mutually orthogonal directions. For the two examples below, controlling the error along orthogonal directions is enough; therefore one only needs to find a way of imposing a desired covariance matrix on the errors. We point out that the controlling the covariance matrix may not be enough in more complex scenarions, e.g. non-orthogonal bases with unequal atom importance, which need controlling the variances along a set of non-orthogonal directions.

In this work, we focus on shaping the covariance matrix of the reconstruction errors. In addition, we deal here only with compressed sensing in the absence of noise. Under these assumptions, the problem we investigate can be stated as follows: given a CS acquisition system as

$$y = A\gamma, \tag{5.1}$$

where A is the acquisition matrix and γ is a sparse vector (in the canonical basis), and defining the recovery error

$$e_\gamma = \gamma - \hat{\gamma}, \tag{5.2}$$

where $\hat{\gamma}$ is the recovered vector, how can we control the covariance matrix of the errors e_γ? If one has a way of controlling the covariance matrix, one can impose smaller or larger variances along an arbitrary set of orthogonal directions by properly choosing the eigenvectors and eigenvalues.

5.2.1 Motivation and applications

We provide here some practical scenarios where error shaping is highly beneficial.

Atoms with unequal importance

Consider the problem of recovering a sparse vector γ when the coefficients γ_i have unequal importance. As an example, consider the problem of recovering an image that is sparse in the Discrete Cosine Transform (DCT) basis. Because of the characteristics of human visual perception, some spatial frequencies can tolerate less noise and errors than others. Therefore, it is desirable that some components are reconstructed more accurately (typically the lower frequencies), whereas larger errors can be acceptable for other components. As another example, γ can be an estimated sparse error vector that affects an underlying signal of interest, and recovering γ is similar to denoising (low-rank matrix estimation [68, 69]). In this case, there often exists a region-of-interest (ROI) consisting of a critical part of the undelying matrix that one would like to estimate more precisely.

We can encompass these scenarios by associating a cost c_i associated with coefficient γ_i, that describes the cost of misestimating that coefficient. Assuming that γ is the original sparse signal and $\hat{\gamma}$ is the recovered estimation, the goal is to minimize the total estimation cost:

$$\textbf{minimize } C = c^T(\gamma - \hat{\gamma}) = c^T e_\gamma = \sum_i c_i e_\gamma(i) \tag{5.3}$$

In order to minimize the total cost, we need a way to make the recovery process more accurate for coefficients with higher error cost c_i at the expense of the coefficients with lower costs. This is the same as saying that we desire a covariance matrix of the errors C_e^γ as

$$C_e^\gamma = c \text{ diag}(c_i^{-2}), \tag{5.4}$$

where c is a scaling factor, i.e. C_e^γ is a custom diagonal covariance matrix depending on the values of the misestimation costs.

Non-orthogonal bases

A second scenario is with signals that are sparse in non-orthogonal bases. Suppose the signal x is sparse in the non-orthogonal basis B

$$x = B\gamma, ||\gamma||_0 = k \tag{5.5}$$

and we acquire it with an acquisition matrix P

$$y = Px = \underbrace{PB}_{A}\gamma. \tag{5.6}$$

The matrix $A = PB$ can be viewed as an equivalent sparsity dictionary for y, as y has the sparse decomposition γ in A, or it can be regarded as an equivalent projection matrix for the sparse vector γ. In this chapter, we favour the second approach.

Let us assume that the matrix A has strong RIP characteristics, i.e it is adequate for compressed sensing. For simplicity, let us consider that A it is a random matrix with i.i.d. normal elements (meaning that the original acquisition matrix P was defined as $P = AB^{-1}$), which has the RIP with high probability. This guarantees that the recovered sparse vector $\hat{\gamma}$ is as accurate as possible.

The recovered signal in the original signal space, is found as

$$\hat{x} = B\hat{\gamma}, \tag{5.7}$$

and the signal reconstruction error is

$$e_x = x - \hat{x} = B(\gamma - \hat{\gamma}) = Be_\gamma, \tag{5.8}$$

where e_γ denotes the error $e_\gamma = \gamma - \hat{\gamma}$.

As B is a (possibly highly) non-orthogonal matrix, the non-linear transformation from the decomposition space to the signal space $\hat{x} = B\hat{\gamma}$ enlarges the direction corresponding to the most significant right singular vectors of B, and compresses the directions of the least significant vectors. In other words, if $B = USV$ is a singular decomposition of B, whenever e_γ happens to lie along the significant vectors of V (corresponding to the larger singular values in S), it will result in higher error e_x in the signal domain, whereas if e_γ lies along the least singular directions, the error will be reduced. As a result, when considering a large number of reconstructions, the average error in the signal space can actually be higher that in the decomposition space.

This effect of the non-linear transformation induced by the non-orthogonal basis B is illustrated in Fig.5.1 and Fig.5.2. In the decomposition space (figure on the right), the vectors e_γ are drawn from a multivariate random distribution with unit covariance matrix. Since B is non-orthogonal, some of the errors are increased in the signal space whereas others are decreased. This affects the optimality of the average error, since, in the average sense, an n-fold increase of a signal error weights more than a n-fold decrease of another one.

Intuitively, a better scenario would be the one depicted in Fig.5.2. The distribution of the error vectors in the decomposition space is such that the resulting errors in the signal space are isotropic. Thus, when recovering a signal, we require smaller errors (better recovery precision) along the critical principal components of B, since they will be enlarged after the transformation. Conversely, it is acceptable to have larger errors along the least significant components of B, since they will be compressed by the transformation to the signal space.

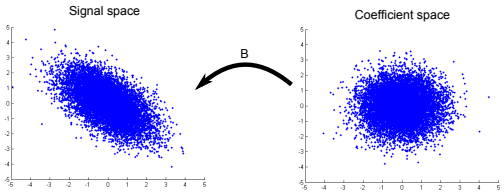


Figure 5.1: Illustration of the non-orthogonal transformation from the decomposition space (right) to the signal space(left) induced by the non-orthogonal basis B. The average error of the resulting signals is increased.

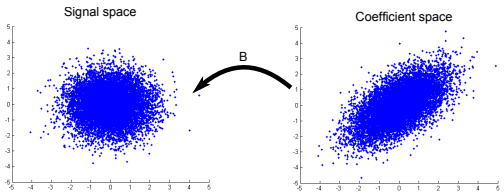


Figure 5.2: To achieve better average signal error in the signal space after the transformation induced by the non-orthogonal basis B, the error in the coefficient space must be non-isotropic.

Considering the error vector e_γ as a random variable from a multivariate distribution with covariance matrix

$$C_e^\gamma = \mathbb{E}\{e_\gamma e_\gamma^T\}, \tag{5.9}$$

the covariance matrix of the errors in the signal domain is

$$C_e^x = \mathbb{E}\{Be_\gamma (Be_\gamma)^T\} = B\mathbb{E}\{e_\gamma e_\gamma^T\}B^T = BC_e^\gamma B^T. \tag{5.10}$$

The above arguments suggest that we would like the error distribution in the signal space to be isotropic, i.e. have unit covariance matrix (up to a scaling factor c):

$$C_e^x = c \cdot I_n. \tag{5.11}$$

Therefore the errors in the decomposition space should have a desired covariance matrix, up to some scaling factor c, of

$$C_e^\gamma = c \cdot B^{-1}(B^T)^{-1} = c \cdot (B^T B)^{-1} = c \cdot VS^{-2}V^T, \tag{5.12}$$

where V and S are from the SVD decomposition $B = USV^T$. Equation (5.12) guarantees that after the transformation with B the distribution will be isotropic.

Extension to overcomplete dictionaries. We can think of sparsity in overcomplete dictionaries as an extension of the case with non-orthogonal bases. If x is a signal

sparse in some dictionary D of size $n \times N$

$$x = D\gamma, \tag{5.13}$$

we are typically not interested in the decomposition error along any direction in the nullspace of D, since any component living in the nullspace of D will be annihilated when reconstructing $\hat{x} = D\hat{\gamma}$. As such, similarly to (5.12), the desired decomposition error covariance matrix in the overcomplete case is

$$C_e^{\gamma} = V\tilde{S}^{-2}V^T, \tag{5.14}$$

where $\tilde{S}$ is a diagonal $N \times N$ matrix containing the n real values of the SVD decomposition $D = USV^T$ as well as $(N - n)$ unspecified values for the nullspace

$$\tilde{S}_{ii} = \begin{cases} S_{ii} & 1 \leq i \leq n \\ \forall & n < i. \end{cases} \tag{5.15}$$

5.3 The general approach

Our approach relies on the fact that the error vectors are always orthogonal to the projection vectors, since both the original decomposition γ and the reconstructed $\hat{\gamma}$ satisfy the same acquisition system:

$$0 = PB(\gamma - \gamma_e) = Ae_{\gamma} \tag{5.16}$$

Let us consider that the projection vectors are drawn from a multivariate normal distribution with covariance matrix C_P. Intuitively, if the projection vectors are agglomerated along some direction u_i, the resulting e_{γ}, being always orthogonal to the projection vectors, will be less inclined along this direction. On the contrary, if u_N is the direction where the vectors are least likely to be drawn from, the e_{γ} will agglomerate along it. Thus, it seems natural that the principal directions of the projection vectors are the same as those of the resulting e_{γ}, with the eigenvalues being in reversed order. We can formally state this as the following proposition:

Proposition 20. *Assume that the projection vectors p_i are multidimensional random variables drawn from a multivariate distribution F with covariance matrix C_P. Assume that the error vector e_{γ} is a multidimensional random variable distributed in the nullspace of the acquisition matrix P with a probability density function that is isotropic in that nullspace, i.e. it is uniform along all directions (rotationally invariant in that subspace).*

Denote the distribution of the e_γ with Γ and its covariance matrix with C_e^γ. In this case, the eigenvectors of C_e^γ are identical to the eigenvectors of C_P. The corresponding eigenvalues are different, but in reversed order, i.e. $\sigma_i^P \leq \sigma_j^P \rightarrow \sigma_i^\gamma \geq \sigma_j^\gamma$.

Proof. Let u_n be the least significant eigenvector of C_P. This means that the direction of u_n is the least likely direction of the projection vectors p_i, i.e. the direction which the p_i are least aligned with. Since the vectors e_γ are always orthogonal to p_i and uniformly spread along the directions in the nullspace of P, it follows that u_i is the most likely direction of the e_γ's. Thus, u_n is the leading eigenvector of C_e^γ, and it corresponding eigenvalue is the highest. Naturally, the second most significant eigenvector of C_e^γ is the second least significant eigenvector of C_P, u_{n-1}, and so on, with the corresponding eigenvalues decreasing in reversed order than those of C_P. □

Proposition 20 asserts that, under some mild conditions, in order for the decomposition errors to have the desired covariance matrix C_e^γ one must draw the projection vectors from a distribution with the same principal directions and inversely ordered variances (their precise values are unknown yet). We experimentally check the condition that e_γ is distributed in the nullspace of P with a rotationally invariant probability density function in Section 5.4, and find that is holds in practice.

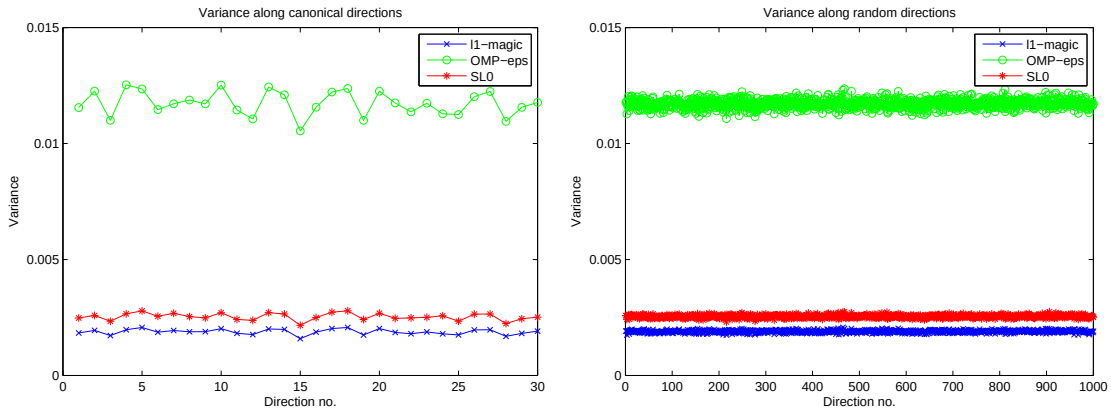
5.4 Establishing isotropy

In this section, we investigate the assumption that the distribution of the error vectors is isotropic when the projection vectors are themselves isotropic.

We generate a set S of 1000 sparse signals with a predefined sparsity k, and project each onto 1000 random projections matrices with i.i.d. normal elements. We then reconstruct with each of the BP, OMP, and SL0 algorithms, thus obtaining the set of error vectors $E = \{e \mid e = x - \hat{x}, x \in S\}$ for each algorithm.

We project the error vectors of E onto the n vectors of the canonical basis, as well as on a set of 1000 unit-norm vectors uniformly selected on the ℓ_2 ball and compute the projection variances along each vector. The results are presented in Fig.5.3.

The results in Fig. 5.3 indicate that for all of these algorithms (BP, OMP, SL0) the expected variance of the error is almost the same along any direction in space, i.e the distribution is almost isotropic. This is to be expected, since there is no reason any direction in space should get a different treatment. The projection matrices are randomly selected from an isotropic distribution, the error vectors lie uniformly in their nullspace, and thus considering the whole set of possible projection matrices we can choose, their



(a) Canonical directions (b) Random directions

Figure 5.3: Variance of error vectors projected onto randomly oriented unit-norm vectors.

nullspaces are uniformly oriented in the space. Therefore, it is not a surprise that, in average over a large number of randomly selected projection matrices, the error vectors are uniformly oriented in all directions.

5.5 Using correlated normal projection vectors

In this section we experimentally investigate the distribution of the recovery errors when the projection vectors are selected from a non-isotropic multivariate random distribution.

5.5.1 A revealing experiment

We design an experiment to reveal the relation between the variances of the distribution of projection vectors and the variances of the resulting error vectors.

Consider a set X of 100 sparse signals of size $n = 20$ obtained by randomly placing $k = 4$ non-zero coefficients. We project them on 5000 projection matrices P_{ijk} with the projection vectors drawn from multivariate random distributions with 50 covariance matrix C_{ij} obtained by combining 5 random vectors of eigenvalues Λ_i with 10 random orthogonal matrices of eigenvectors U_j:

$$C_{ij} = U_j \cdot \ \mathrm{diag}(\Lambda_i) \cdot U_j^{-1}. \tag{5.17}$$

The vectors Λ_i of eigenvalues consist of $n = 20$ random variables selected from the interval $(0.1, 1.1)$, normalized such that their product equals 1. The restriction to the interval $(0.1, 1.1)$ ensures that the ratio of any two eigenvalues is between $(1/11, 11)$, therefore avoiding extremely large or small values.

For each covariance matrix C_{ij} we create 100 projection matrices P_{ijk} by randomly selecting multivariate normal vectors with this covariance matrix. We acquire the signals with every projection matrix and then we reconstruct with various reconstruction algorithms. Then we compute the average variances of the reconstruction errors along the directions of the eigenvectors U_j.

We are interested in how the ratio of two variances in the projection vectors' distribution influences the ratio of the corresponding variances of the error vectors along the same directions. We are interested only in the ratios of variances, not their absolute values, since the absolute values are dependent on multiple factors (signal sparsity, number of measurements).

Fig.5.4, Fig.5.5 and Fig.5.6 show the ratio of the variances of error vectors depending on the ratios of the variances of the projection vectors. The three graphs reveal two surprising conclusions:

1. There is an accurate relation between the variances of the projection vectors' distribution and the variances of the resulting error vectors. If the projection vectors have a covariance matrix C_P with eigenvalues λ_i^P, the resulting error vectors have a covariance matrix C_e^γ with eigenvalues λ_i^γ such that:

$$\frac{\lambda_i^\gamma}{\lambda_j^\gamma} \approx \left(\frac{\lambda_i^P}{\lambda_j^P}\right)^{-0.424}, \tag{5.18}$$

   i.e. the ratio of any two eigenvalues of C_e^γ is approximately a power function of the ratio of the corresponding eigenvalues of C_P.

2. Different algorithms (ℓ_1-magic, OMP, SL0) exhibit virtually identical behaviour in this respect.

5.5.2 Proposed solution

In view of the above considerations, we propose the following solution to the problem stated in Section 5.2:

Objective: find a projection matrix for sparse vectors such that the covariance of the reconstruction errors is C_e^γ (up to a scaling factor c).

Solution: Consider an eigenvalue decomposition

$$C_e^\gamma = U\Lambda^{(e)}U^{-1}. \tag{5.19}$$

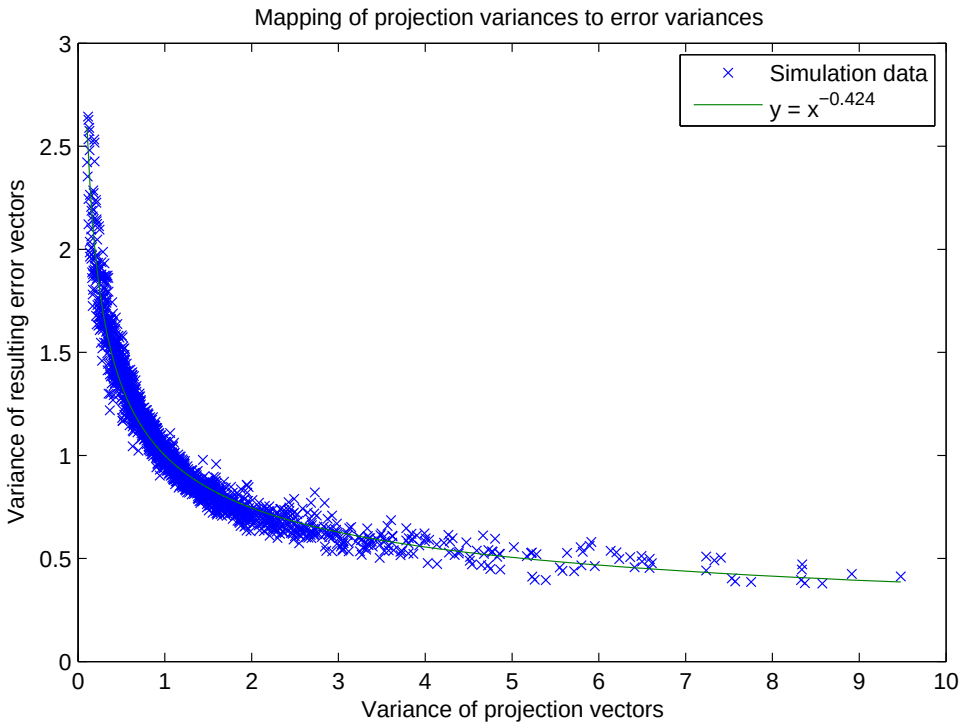


Figure 5.4: Variances of the error vectors as a function of variances of projection vectors - l1-magic

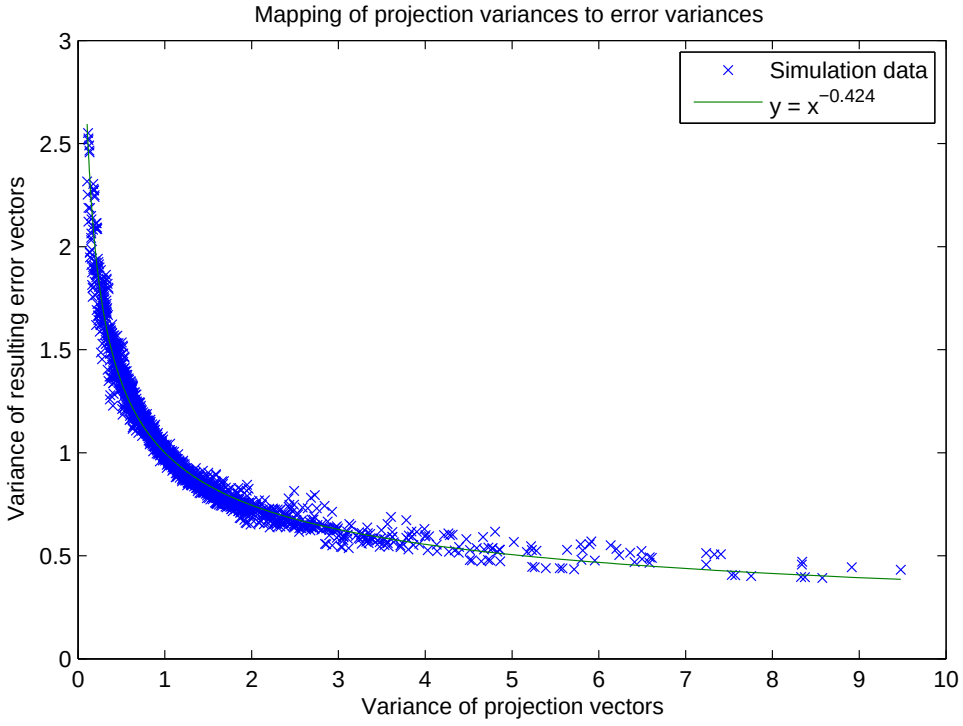


Figure 5.5: Variances of the error vectors as a function of variances of projection vectors - OMP

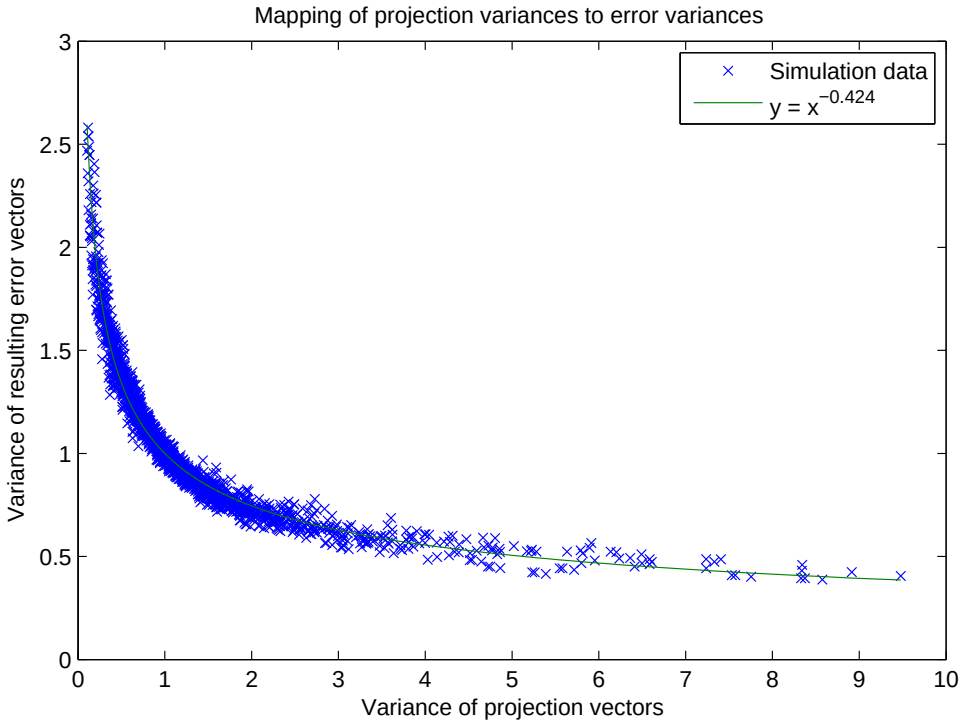


Figure 5.6: Variances of the error vectors as a function of variances of projection vectors - l1-magic

Choose multivariate random projection vectors from a distribution with covariance matrix

$$C_P = U\Lambda^{(P)}U^{-1}, \tag{5.20}$$

where

$$\lambda_i^{(P)} = \left(\lambda_i^{(e)}\right)^{-1/0.42}. \tag{5.21}$$

Practically, we draw projection vectors from a correlated distribution whose shape "extends" proportionally in the directions for which we want a higher accuracy.

Extension to sparsity in bases

When working with signals that are sparse in some basis B, on must keep in mind that (5.20) refers to the equivalent projection vectors, i.e. the rows of the effective dictionary $A = PB$, since they are the equivalent projection vectors that "measure" the sparse decomposition directly. In this case, the covariance matrix of the real projection vectors P must satisfy

$$E\{(PB)^T PB\} = B^T C_P B = U\Lambda^{(P)}U^{-1}, \tag{5.22}$$

therefore the covariance matrix of the projection vectors must be

$$C_P = (B^T)^{-1}U\Lambda^{(P)}U^{-1}B^{-1}. \tag{5.23}$$

Extension to overcomplete dictionaries

As argued in Section 5.2.1, the case of overcomplete dictionaries can be considered as an extension of the case of non-orthogonal bases, and the desired decomposition error covariance matrix has the form in (5.14). In this case, there is a natural extension of (5.23) that accounts for overcomplete dictionaries as well.

For non-orthogonal bases, the desired covariance matrix is (5.12)

$$C_e^{\gamma} = VS^{-2}V^T, \tag{5.24}$$

and thus (5.23) particularizes to

$$C_P = (B^T)^{-1}VS^{2/0.42}V^TB^{-1} = \left(U(S^T)^{-1}V^T\right)VS^{2/0.42}V^T\left(VS^{-1}U^T\right) = US^{2/0.42-2}U^T. \tag{5.25}$$

This can naturally be extended to the case of overcomplete dictionaries of size $n \times N$, as the quantities involved in (5.25) only imply the dimension n of the dictionary (if we consider the "economy size" SVD decomposition of D, in which case S is of size $n \times n$; otherwise S should be the restriction to the square matrix). Therefore, for sparsity in an overcomplete dictionary $D = USV^T$, the projection vectors should be selected from a multivariate distribution with covariance matrix equal to

$$C_P = US^{2/0.42-2}U^T, \tag{5.26}$$

where U and S come from the SVD decomposition $D = USV^T$.

Alternative interpretation. Equations (5.23) and (5.26) lend themselves to an intuitive interpretation, if the value 0.42 would be rounded to 0.5. In this case, the desired projection covariance matrix becomes

$$C_P = US^{2/0.5-2}U^T = US^2U^T, \tag{5.27}$$

which is exactly the covariance matrix obtained if we define the projection matrix as

$$P = P_0D^T, \tag{5.28}$$

where P_0 is a random matrix with i.i.d elements of size $m \times N$. Indeed in this case

$$\mathbb{E}\{P^TP\} = D\mathbb{E}\{P_0^TP_0\}D^T = DD^T = US^2U^T, \tag{5.29}$$

i.e. identical to (5.27).

Defining the acquisition matrix as in (5.28) means *selecting projection vectors as random combinations of the atoms*, instead of selecting random projection vectors directly.

In this case, the covariance matrix of the projection vectors is the same as the covariance matrix of the dictionary atoms, which is also the covariance matrix of the sparse signals, if we assume that the decomposition vectors are themselves isotropic (all atoms being used uniformly).

We find therefore the intuitive conclusion that the projection vectors should have the same covariance matrix as the signals they are applied to. This agrees with the intuition that if the signals/atoms are agglomerated along particular directions in space, the projection vectors should lie mostly along the same directions, in order to capture the most of the signals. The more accurate exponent we find in our experiments of 0.42 instead of 0.5, indicates that we should actually draw projection vectors from a distribution slightly more *pointy* along the interesting directions.

5.5.3 Particular cases

When we are not interested in reducing the error along some particular direction, and all directions in space have equal importance, the desired error covariance is $C_e^\gamma = c \cdot I_n$, and (5.21) yields a unit covariance matrix for the projection vectors. Our approach reduces therefore to drawing projection vectors from a distribution with unit covariance matrix, which is the same as creating a random matrix with i.i.d. random scalar elements. This is the usual scenario in compressed sensing, and the fact that it leads to isotropic error distribution is validated experimentally in Section 5.4.

When we are not interested in any control of the error along some particular direction or directions, i.e. the error can extend arbitrarily along those directions, the corresponding eigenvalues of the desired error covariance matrix may have very large values (virtually ∞). Accordingly, the projection vectors covariance matrix has 0 covariance along those directions, meaning that all the projection vectors will be orthogonal to the uninteresting subspace. This agrees with the intuition: if some subspace is of no interest, the projection vectors should be orthogonal to it, capturing no information regarding this subspace.

On the contrarily, if we would like absolute precision (zero error) along some direction, (5.21) suggests a infinite eigenvalue in the projection covariance matrix, i.e. a degenerate distribution extending infinitely and narrowly along the specified direction. This is clearly not appropriate, as it would suggest that all the projection vectors should be collinear, along the same specific direction of interest, making them redundant. The problem resides, in fact, in the probabilistic model that assumes that all vectors are drawn at random from a multivariate distribution. The correct solution in this case would be to fix one of the projection vectors along the specified direction, and use a probabilistic

model only for the remaining projections, in accordance with the desired error variances along the remaining subspace. The resulting errors will always be orthogonal to the fixed direction of the first vector, i.e. having zero error along that direction.

5.5.4 Relation with signal foveation

From a practical point of view, choosing random vectors from a multivariate distribution with covariance matrix

$$C_P = U\Lambda U^{-1} \tag{5.30}$$

can be achieved by multiplying a random matrix P_0 with i.i.d normal elements with $M = \Lambda^{1/2}U^{-1}$:

$$P = P_0 M. \tag{5.31}$$

Indeed, in this case the covariance matrix of the projection vectors is

$$\mathbb{E}\{P^T P\} = M^T \mathbb{E}\{P_0^T P_0\} M = M^T M = C_P, \tag{5.32}$$

since $E\{P_0^T P_0\} = I_N$ (vectors with i.i.d. normal elements have unit covariance).

Signal foveation [70] is an interesting technique that bears resemblance with this approach. Signal foveation is a way of enhancing a region of interest of a signal in a transform domain. For example, consider a signal x sparse in some basis B

$$x = B\gamma \tag{5.33}$$

that is acquired using some acquisition matrix P_0:

$$y = P_0 x \tag{5.34}$$

A foveating approach would be to use instead an acquisition matrix defined as $P = P_0 B M B^{-1}$, resulting in an acquisition system defined as:

$$\tilde{y} = Px = P_0 B M B^{-1} x \tag{5.35}$$

where M is a diagonal matrix. The idea is that x is transformed in the sparsity domain with the operator B^{-1}, a mask M is then applied in the transform domain that enhances specific desired components known a priori (M is a diagonal matrix with high values for the interesting components and small values for the uninteresting ones), then brought back into the original domain and projected on P_0. Thus, comparing with the non-foveated approach, the interesting components are artificially increased compared to the uninteresting ones. After recovering the signal, the effect of mask is undone and the

components are scaled back to their original level. The overall effect is that the enhanced components will have better reconstruction accuracy, at the expense of worse accuracy for the rest.

The covariance matrix of the foveated projection vectors in (5.35) is

$$\mathbb{E}\{P^T P\} = (B^{-1})^T M B^T \mathbb{E}\{P_0^T P_0\} B M B^{-1}. \tag{5.36}$$

Assuming that $\mathbb{E}\{P_0^T P_0\} = I_N$ and B is an orthogonal matrix, this reduces to

$$\mathbb{E}\{P^T P\} = B M^2 B^{-1}, \tag{5.37}$$

which is similar to (5.20). This means that using correlated projection vectors can essentially be thought of as signal foveating with an orthogonal transformation.

However, what is essential in our work is that we determine an accurate relationship between the amount of foveation (the degree of enhancement with the masking matrix M) and the variance of resulting errors in the context of compressed sensing, for some very different algorithms. Previously, signal foveation was typically considered with classical linear processing applications (e.g. transform signal compression in wavelet domain [70]).

5.6 Applications

5.6.1 Unequal atom importance

Consider employing compressed sensing for recovering 8×8 grayscale image patches that are approximately sparse in the Discrete Cosine Transform (DCT) basis:

$$x = B\gamma \tag{5.38}$$

where we consider x and γ as column vectors (by concatenating columns) and correspondingly B is the 64×64 2D-DCT matrix. We would like to recover the image patches from a number m of random projections that form an acquisition matrix P of size $m \times 64$.

It is well known that the human eye has different sensitivity to spatial frequencies [71], which is used in image compression (e.g. JPEG) to set different quantization steps on the DCT coefficients. In the JPEG standard, quantization steps are defined by a quantization table Q. Each element of the 8×8 DCT coefficient matrix is divided by the corresponding element in Q and then rounded. A general typical Q matrix for the luminance component is defined as in Fig.5.7 [72], however in practice different matrices can be used according to the desired compression ratio. A higher value indicates coarser

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 5.7: Typical JPEG quantization table for the luminance component.

quantization. As can be seen in Fig.5.7, lower frequencies (upper left corner) tend to have finer quantization and thus smaller quantization noise.

It is desirable, therefore, to have less noise in the lower frequencies and allow for larger noise in the higher frequencies. We can enforce such a behaviour when reconstructing image patches from random projections by using correlated projection vectors, drawn from a multivariate normal distribution with an appropriate covariance matrix.

The elements of the quantization matrix Q can be interpreted as the cost of distortion for a certain spatial frequency, e.g. a quantization error of 99 for the (8,8) spatial frequency component is as acceptable as a quantization error of 11 for the (1,2) component. Therefore the square of the elements of Q indicate the acceptable relative noise variance for every component. As such, when reconstructing from compressed measurements, we would like the error variance of the coefficient of the (i, j) component to be proportional to q_{ij}^2, i.e. the desired error covariance matrix of the sparse decomposition is

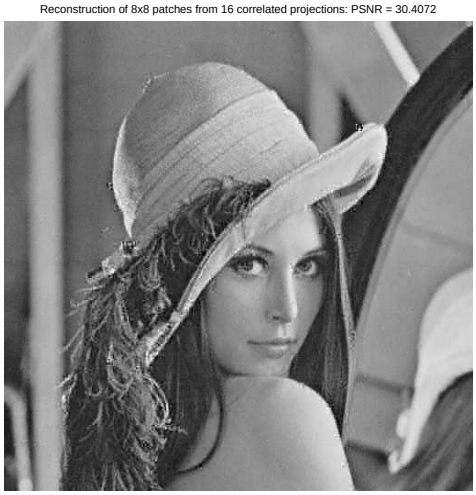
$$C_\gamma = \text{ diag}(q_{ij}^2). \tag{5.39}$$

where B is the orthonormal basis matrix of the 2D-DCT transform.

According to (5.23), we propose drawing the projection vectors from a multivariate normal distribution with covariance matrix C_P defined as

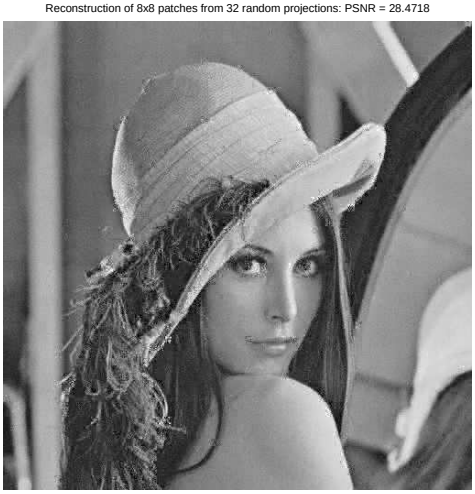
$$C_P = (B^T)^{-1} \text{ diag}(q_{ij}^{\frac{2}{-0.42}}) B^{-1} = B \text{ diag}(q_{ij}^{\frac{2}{-0.42}}) B. \tag{5.40}$$

Fig.5.8 shows the reconstruction of the *Lena* image, each 8×8 block being reconstructed independently from $m = 16$ random projections. Fig.5.9 is similar but with $m = 32$. The random projection vectors are drawn from multivariate normal distributions with a) unit covariance matrix (i.i.d. random elements) and b) the covariance matrix C_P defined in (5.40). Different projection vectors are drawn from the distribution for every 8×8 image patch. Reconstruction is done with Orthogonal Matching Pursuit (OMP) [14].



(a) Isotropic projection vectors: PSNR=16.7dB (b) Correlated projection vectors: PSNR=30.4dB

Figure 5.8: **Sparsity in orthogonal basis with unequal atom importance.** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 16$ random projections.



(a) Isotropic projection vectors: PSNR=28.5dB (b) Correlated projection vectors: PSNR=37.1dB

Figure 5.9: **Sparsity in orthogonal basis with unequal atom importance.** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 32$ random projections.

The improvements are significant: with isotropic projection vectors the PSNR is 16.7dB ($m = 16$) and 28.5dB ($m = 32$), whereas with correlated vectors it is 30.4dB and 37.1dB, respectively.

We point out that that other factors may have contributed as well to the improvements seen in Fig.5.8b and Fig.5.9b. In natural images, it is expected that the significant coefficients are concentrated in lower spatial frequencies, and thus the sparsity pattern is not uniform, whereas in our fundamental experiment in Section 5.5.1 we assumed uniform sparsity pattern (all atoms are used uniformly). As such, the lower spatial frequencies are not only the components for which we desire less noise, but they are also more likely to appear in the sparse decompositions than the high frequency components. As a consequence, shaping the projection vectors distribution along the directions of the lower frequencies will likely provide an additional benefit.

5.6.2 Sparsity in non-orthogonal bases / overcomplete dictionaries

Let us consider a similar scenario as before, but now consider the 8×8 image patches to be sparse in a learned basis/ dictionary instead of the 2D-DCT basis. We use all the overlapping 8×8 image patches from the image as a training set (a total of 255025 for a 512×512 image), and we use the dictionary learning algorithm from [73, 74] to learn a better sparsifying basis B or overcomplete dictionary D. We then reconstruct all the distinct non-overlapping image patches of the image (a total of 4096) from random projections, assuming sparsity in the basis/dictionary learned before.

For the basis case, as detailed in Section 5.2.1, we would like an error covariance matrix in the decomposition equal to (up to a scaling factor)

$$C_e^{\gamma} = B^{-1}(B^T)^{-1} = (B^T B)^{-1} = VS^{-2}V^T, \tag{5.41}$$

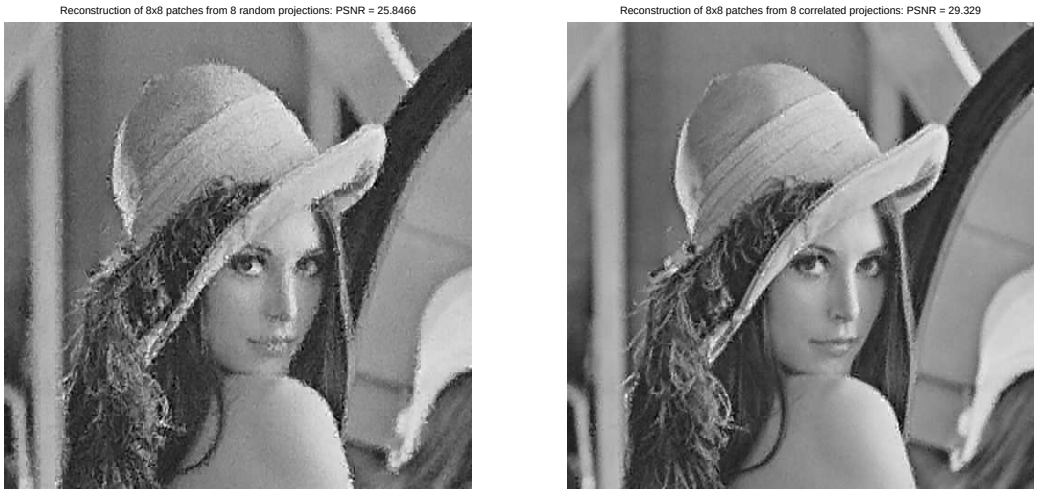
where V and S come from the SVD decomposition $B = USV^T$.

According to (5.23), we draw projection vectors from a distribution with covariance matrix equal to

$$C_P = (B^T)^{-1}VS^{2/0.42}V^T B^{-1} = US^{2/0.42-2}U^T. \tag{5.42}$$

This equation holds for the overcomplete case as well. As such, the two cases can be treated identically, the overcomplete case being a natural straightforward extension of the basis case.

Fig.5.10 and 5.11 show the reconstructions of the *Lena* image from 8×8 reconstructed patches, using a learned basis, when the projection vectors are drawn from multivariate



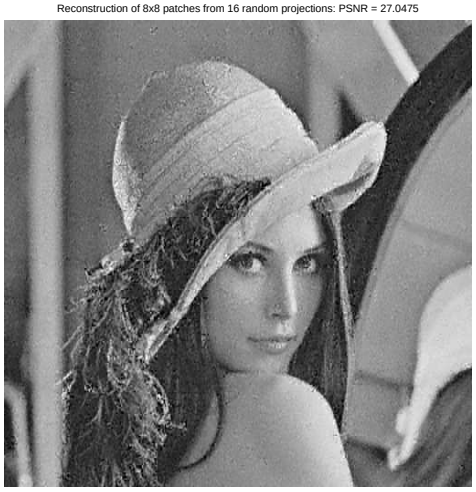
(a) Isotropic projection vectors: PSNR=25.8dB (b) Correlated projection vectors: PSNR=29.3dB

Figure 5.10: **Sparsity in learned basis (64×64).** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 8$ random projections.

normal distributions with a) unit covariance matrix (i.i.d. random elements) and b) the covariance matrix C_P defined in (5.42). Fig.5.12 and 5.13 are for the overcomplete dictionary of size 64×128. Different projection vectors are drawn from the distribution for every 8×8 image patch. Reconstruction is done with OMP.

Again, our approach leads to significant improvement over the isotropic case. With a basis, with isotropic projection vectors the PSNR is 25.8dB ($m = 8$) and 27.4dB ($m = 16$), whereas with correlated vectors it is 29.3dB and 33.3dB, respectively. For the overcomplete case we have similar values: with isotropic projection vectors the PSNR is 25.8dB ($m = 8$) and 28.2dB ($m = 16$), whereas with correlated vectors it is 29.4dB and 33.2dB, respectively. Interestingly, using a learned dictionary instead of a learned basis does not improve the results, suggesting that the number of significant components is actually far lower then 64 and additional atoms don't bring notable advantages.

The learned basis and dictionary are presented in Fig.5.14. In this test, we work with zero-mean patches, because the high DC component in natural image patches is susceptible of misleading the dictionary training algorithm. The DC component is typically the single largest component present in all the patches, and for this thing it is sometimes treated differently (e.g. the JPEG coding standard [72]). For dictionary learning, if the training data have DC component, it is distributed among all the resulting atoms,



(a) Isotropic projection vectors: PSNR=27.4dB

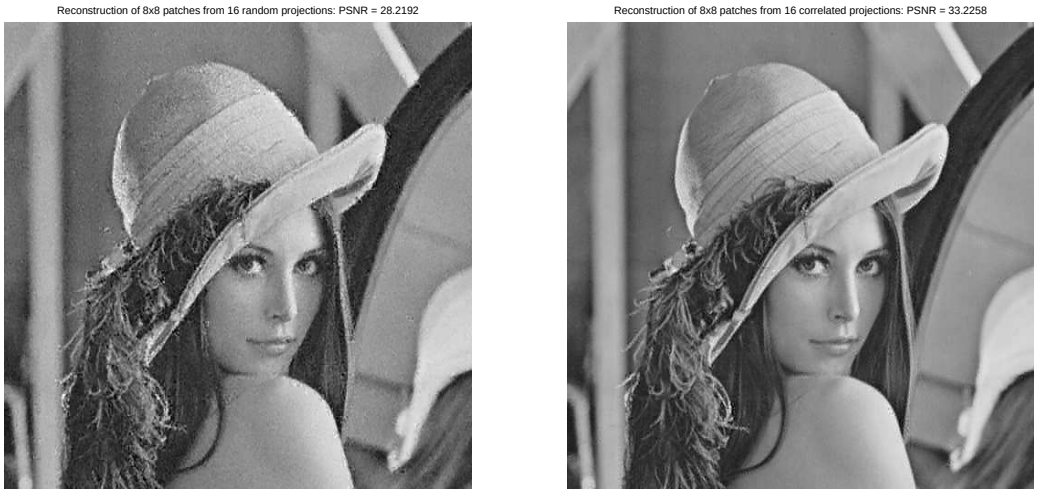
(b) Correlated projection vectors: PSNR=33.3dB

Figure 5.11: **Sparsity in learned basis (64×64).** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 16$ random projections.

(a) Isotropic projection vectors: PSNR=25.8dB

(b) Correlated projection vectors: PSNR=29.4dB

Figure 5.12: **Sparsity in learned dictionary (64×128).** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 8$ random projections.



(a) Isotropic projection vectors: PSNR=28.2dB (b) Correlated projection vectors: PSNR=33.2dB

Figure 5.13: **Sparsity in learned dictionary (**64×128**).** Reconstruction from correlated projection vectors is significantly better than from isotropic vectors. Every 8×8 image patch is reconstructed independently from $m = 16$ random projections.

preventing them from converging to other particular features and interfering with the reconstruction. As such, for this test we remove the mean from all the patches and re-add it at the end with no errors, i.e. assuming perfect acquisition and reconstruction of the DC component.

5.6.3 ECG signals

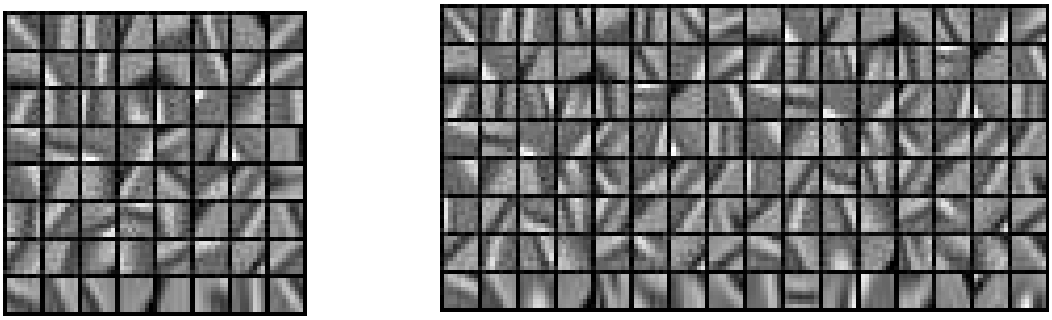
Section 6.7.1 presents another example of using correlated projections for acquiring electrocardiogram (ECG) signals that are sparse in a highly non-orthogonal basis. In that case, the projection matrix is defined as

$$P = P_0 B^T \tag{5.43}$$

where P_0 is a random matrix with i.i.d. normal elements, i.e. vectors with isotropic distribution. Therefore the covariance matrix is

$$E\{P^T P\} = BE\{P_0^T P_0\}B^T = US^2U^T, \tag{5.44}$$

which is similar to (5.42), only with the exponent 0.42 replaced by 0.5. Nevertheless, as shown, the results still outperform the results obtained with isotropic projection vectors.



(a) Learned 64×64 basis

(b) Learned 64×128 dictionary

Figure 5.14: The learned sparsity basis/dictionary of 8×8 image patches.

5.7 Conclusions

In this chapter we experimentally show that a number of popular CS reconstruction algorithms (BP, OMP, SL0) exhibit the property that, when using with correlated random projection vectors from a distribution with covariance matrix C_P, yield reconstruction errors with a covariance matrix having the same eigenvectors and a power law of the original eigenvalues. This allows for accurate control of the covariance matrix of the decomposition errors by properly choosing the covariance matrix of the projection vectors, thus permitting more accurate signal recovery along some desirable directions. This enables us to optimize the projection vectors in interesting non-standard CS scenarios involving bases with unequal atom importance and non-orthogonal bases/dictionaries.

Further work should expand the investigation to other reconstruction algorithms, as well as develop a probabilistic model that explains the precise relation between the variances of the projection vectors and of the resulting errors.

Chapter 6

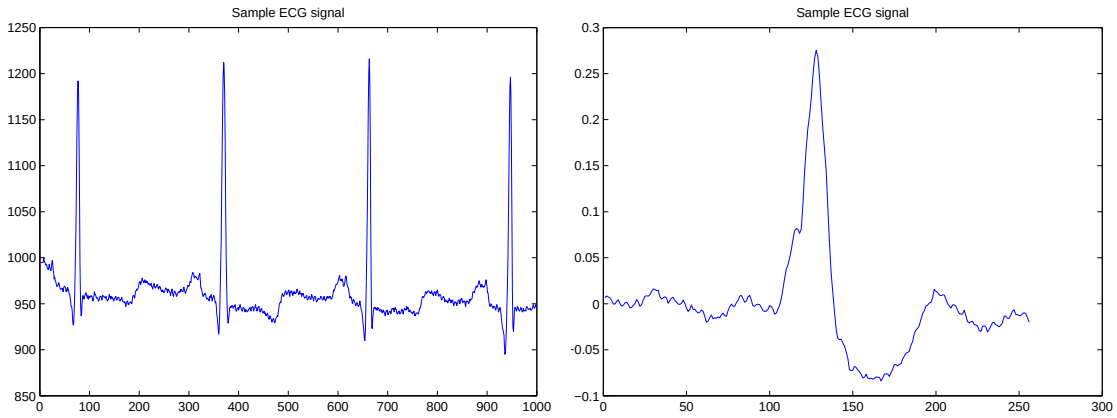
Compressed sensing of ECG signals

6.1 Introduction

ECG signals are electric signals related to the activity of the heart, obtained by measuring the electrical variations of the skin due to depolarization of the heart muscle during a heart beat, and used in medicine to diagnostic various abnormalities and heart-related conditions. The ECG acquisition procedure typically requires multiple sensors in various locations on the body. For the purposes of this work, we refer to single-channel ECG signals only, i.e. acquired through only one sensor.

We investigate two main applications of ECG signal processing: compression and classification. One of the typical applications of signal processing for ECG signals concerns their compression. ECG recordings can span tens of hours, especially when considering portable wearable devices attached to the body of a patient. In addition, a portable device imposes specific constraints on power usage and size of the device. As such, efficient compression techniques have been researched in the literature, e.g. [75, 76]. Our goal is to evaluate a compression technique based on random projections, according to the compressed sensing theory. This has the benefit that the acquisition stage is extremely simple, and thus it is potentially adequate for ECG acquisition with battery-powered, low complexity portable devices.

The second application that we investigate is that of ECG heart beat classification. Automatic classification of heart beats from an ECG signal provides an important aid for diagnosis of heart-related abnormalities, and as such has also been an active research domain [77, 78]. In this work we focus on estimating the possibility of classifying random measurements of the signals rather than the original signals themselves, with the goal of reducing dimensionality and thus the power requirements of the application.



(a) Sample portion of a ECG recording (b) Sample ECG segment after preprocessing

Figure 6.1: Sample ECG signals

6.2 ECG signals and pre-processing

The ECG data used throughout this work come from 24 recordings of the MIT-BIH Arrhythmia database [79, 80], widely used in the research literature. The signals were acquired at a sampling frequency of 360Hz, with 11 bits / sample, and include annotations containing the position of the R wave, as well as normal/pathological heart beat classification done by trained cardiologists [79, 80].

We apply an initial preprocessing stage for the ECG recordings, consisting of the following: (i) remove glitches in the signal (samples of extremely low value, possibly due to equipment malfunction), (ii) segment the ECG recording at every midpoint between two consecutive R waves, i.e. isolating heartbeats, (iii) extract mean and normalize each heartbeat segment, (iv) resampling the first and the second half of every segment independently up to a length of 128 samples each. The final ECG segments are therefore 256 samples long and with the R wave aligned on the middle of the segment. The reason is that the QRS complex accounts for large part of the signal's energy and is of essential importance for visual diagnosis as well; having a known precise alignment of it will be a helpful clue for signal compression/classification/reconstruction. The mean of each segment as well as its energy is of little importance for diagnosis, therefore we have chosen to bring the segments to zero mean and unit norm to avoid unnecessary complications.

An illustration of an ECG segment as resulting after the pro-processing step in given in Fig.6.1.

The heart beats of the signals from the MIT-BIH Arrhythmia database that we work with are annotated into 13 different classes, one normal class and 12 abnormal classes

corresponding to various heart related diseases and abnormalities. Out of these, in classification applications we typically exclude 5 classes because they are too under-represented ("Unclassifiable beat", "Ventricular escape beat", "Aberrated atrial premature beat", "Nodal (junctional) escape beat", "Supraventricular premature beat"), keeping only 8 classes of beats with enough representation: "Normal beat", "Left bundle branch block beat","Right bundle branch block beat","Premature ventricular contraction","Fusion of ventricular and normal beat","Paced beat","Fusion of paced and normal beat" and "Atrial premature beat". For compression applications, we keep all the data available.

The preprocessing stage results in a number of over 50.000 ECG segments, each grouped according to the corresponding heart beat class. We split the data of each class into a 80% training set (to be used in training compression/classification schemes) and a 20% test set used to validate the proposed techniques.

6.3 Investigating sparsity bases and dictionaries

The key condition for successful signal reconstruction from few random measurements is undoubtedly the sparsity of the ECG segments in the considered dictionary. The efficiency of various dictionaries for compressively sensing of aligned and centered ECG segments has previously been studied in [81, 82]. It has been determined that better performance is achieved when using custom dictionaries composed of other ECG segments, rather than using standard wavelet bases typically used in literature [83, 70]. This is due to the similarity of ECG segments, especially since they are all aligned with the R wave in the middle.

In this section we recreate partially the experiments in [81, 82], with the aim of determining the best sparsifying basis.

We investigate sparsity in most of the orthogonal wavelet bases available in the Wavelab 850 Matlab package [84]: Haar, Beylkin, Coiflet 3, Coiflet 4, Coiflet 5, Daubechies 6, Daubechies 8, Daubechies 10, Daubechies 12, Daubechies 14, Symmlet 5, Symmlet 6, Symmlet 7, Symmlet 8, Symmlet 9, Vaidyanathan, Battle 1, Battle 3, Battle 5.

We evaluate sparsity by computing the average error of the best k approximation of the signals in each of the above bases, with k varying from 1 to 25. As the wavelet bases are orthogonal, we compute the best k-term approximation by simply applying the transform and restricting the result to the first k significant coefficients.

Besides the standard wavelet bases, we use three methods of generating custom bases from the ECG segments training set:

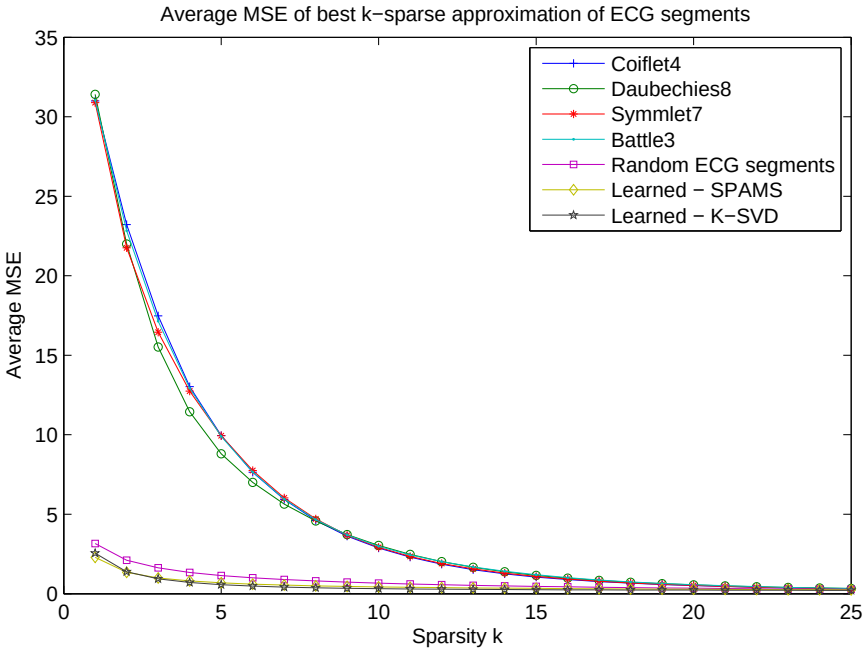


Figure 6.2: Average best k-term approximation error for different bases

1. Randomly selecting n ECG segments from the training set (RNDDICT).

2. Learning a basis using the ODT dictionary learning algorithm from [73, 74].

3. Learning a basis using the K-SVD dictionary learning algorithm [30].

These custom bases are not orthogonal and as such we use Orthogonal Matching Pursuit (OMP) [14, 15] to select the best k atoms of the signal decomposition.

The average error of the best k-term approximation for the most relevant wavelet bases and for the custom bases is displayed in Fig.6.2. More detailed results for all bases are in Table 6.1.

The results clearly show that custom bases significantly outperform the standard wavelet bases for approximating the ECG segments with few atoms. The explanation lies in the preprocessing of the ECG segments. The preprocessing makes the segments have the same length and aligns the peak (the QRS complex) at the middle of the segment, and thus significantly increases the similarity of the segments. As such, custom bases can exploit this similarity much better. Even a custom basis created by randomly selected ECG segments provides much better approximation error than the standard bases, due to the similarity of the segments. As such, we focus on using custom bases rather than standard orthogonal wavelet bases.

Table 6.1: Average best k-term approximation error for different bases

Basis	**Sparsity**		
Wavelet	$k = 5$	$k = 10$	$k = 15$
Haar	13.80	4.83	2.21
Beylkin	11.96	4.36	1.80
Coiflet 3	11.89	3.08	1.11
Coiflet 4	9.95	2.88	1.05
Coiflet 5	11.27	3.26	1.19
Daubechies 6	11.91	3.02	1.17
Daubechies 8	8.81	3.03	1.16
Daubechies 10	12.85	4.33	1.46
Daubechies 12	12.67	3.87	1.46
Daubechies 14	10.66	3.41	1.33
Symmlet 5	9.39	3.13	1.14
Symmlet 6	11.96	3.12	1.10
Symmlet 7	9.94	2.92	1.09
Symmlet 8	9.99	2.90	1.06
Symmlet 9	10.96	3.22	1.14
Vaidyanathan	11.81	4.67	1.95
Battle 1	11.28	3.03	1.02
Battle 3	9.90	2.99	1.19
Battle 5	9.93	3.45	1.42
Custom			
Random ECG segments - RNDDICT	1.15	0.67	0.46
Learned - ODT	0.69	0.43	0.34
Learned - K-SVD	0.57	0.32	0.26

6.4 Compressed sensing with a single custom dictionary

Compressed sensing can be advantageous for portable ECG devices due to its capability of fast and efficient compression with simple linear projections, the burden of complex processing being shifted at the decoding side. As such, compressed sensing for ECG acquisition has been proposed in the literature [83], typically in conjunction with orthogonal wavelets. In our setup, having custom bases that are significantly better than wavelet bases means that we should obtain better recovery performance for compressed sensing of ECG segments.

A straightforward approach is to use a single custom dictionary. The optimal number of atoms is difficult to estimate: a larger number of atoms means better representation power and thus reduced signal sparsity, however it also implies a larger decomposition vector. The number of measurements required for successfully recovering an N-dimensional sparse vector is of order $\mathcal{O}(k \log(N/k))$ (see Theorem 15 from section 2.4.6), therefore increasing logarithmically with N. Therefore the optimal value of N derives from a trade-off between good representation power (larger N) and good recovery possibilities (smaller N).

In the following, we evaluate the recovery accuracy of preprocessed ECG segments from m random measurements, considering custom dictionaries of size $256 \times N$ where N is 256, 320, 384, 448, 512, 640 or 768. The dictionaries are created with the three methods from the previous section that provide the best sparse representation: RNDDICT, ODT, K-SVD. We are interested in the average PRDN of the ECG segments, defined as

$$PRDN(x, \hat{x}) = \sqrt{\frac{||x - \hat{x}||_2^2}{||x - \mu_x||_2^2}} \times 100 \tag{6.1}$$

where μ_x is the DC component of x.

We also test three methods of creating the acquisition matrix:

1. using i.i.d. random normal elements (RNDPROJ)
2. using correlated projection vectors as explained in section 6.7.1 (CORRPROJ-0.5)
3. using correlated projection vectors as in section 5.5.2 (CORRPROJ-0.42).

As explained in the alternative interpretation in section 5.5.2, the difference between CORRPROJ-0.5 and CORRPROJ-0.42 lies in the fact that the former uses a coefficient of 0.5 instead of the more accurate value of 0.42 determined from the experiment in 5.5.1.

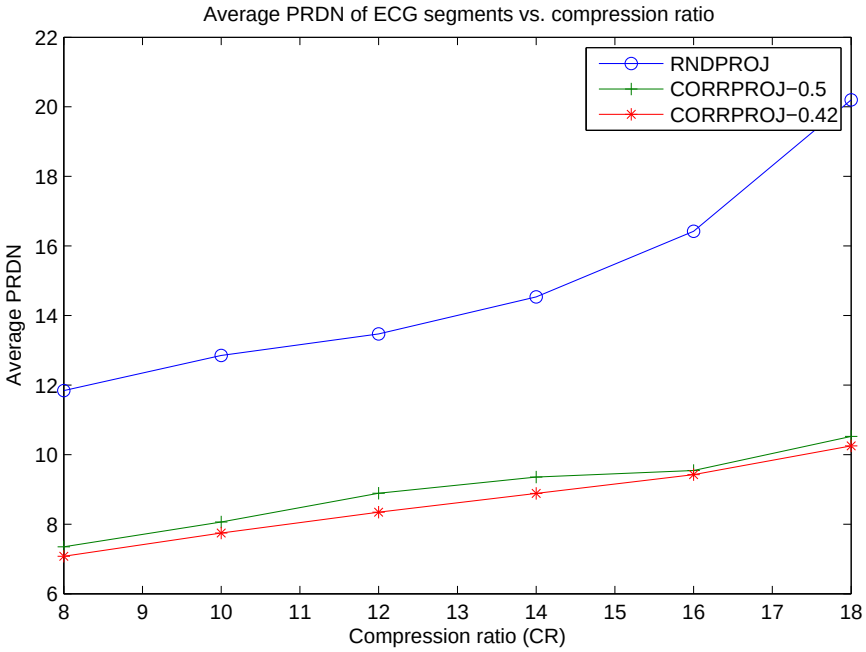


Figure 6.3: Comparison of different types of projection matrices. The dictionary is 256×768 and created with the ODT learning algorithm. The CORRPROJ-0.42 method, using correlated projection vectors as in section 5.5.2, provides the smallest average PRDN errors.

Reconstruction is done with ℓ_1 minimization from the ℓ_1-magic Matlab package [52].

We first analyse which method for creating the projection matrix leads to the best recovery results. For a dictionary of size 256×768 created with the ODT algorithm, the three types of projection matrices lead to the results presented in Fig.6.3. Similar results are obtained for the other dictionary algorithms and sizes. The best results are obtained using CORRPROJ-0.42, i.e. correlated projection vectors as proposed in section 5.5.2. Using the slightly modified value of 0.5 leads to a small performance decrease.

Second, we turn our attention on the best dictionary learning algorithm and dictionary size. In Fig.6.4 we present the average PRDN obtained with dictionaries of different sizes, created with all three dictionary learning algorithms. The projection matrix is created with CORRPROJ-0.42, which we determined above to be the best. In these conditions, the ODT algorithm provides the best dictionaries, i.e. leading to smallest reconstruction errors. For ODT, the size of the dictionaries seems not very importance in this case: larger dictionaries bring only a marginally improvement.

We conclude, therefore, that the best results are obtained using preferably large dictionaries learned with the ODT training algorithm, with significant improvements brought by using correlated projection vectors to exploit their non-orthogonality.

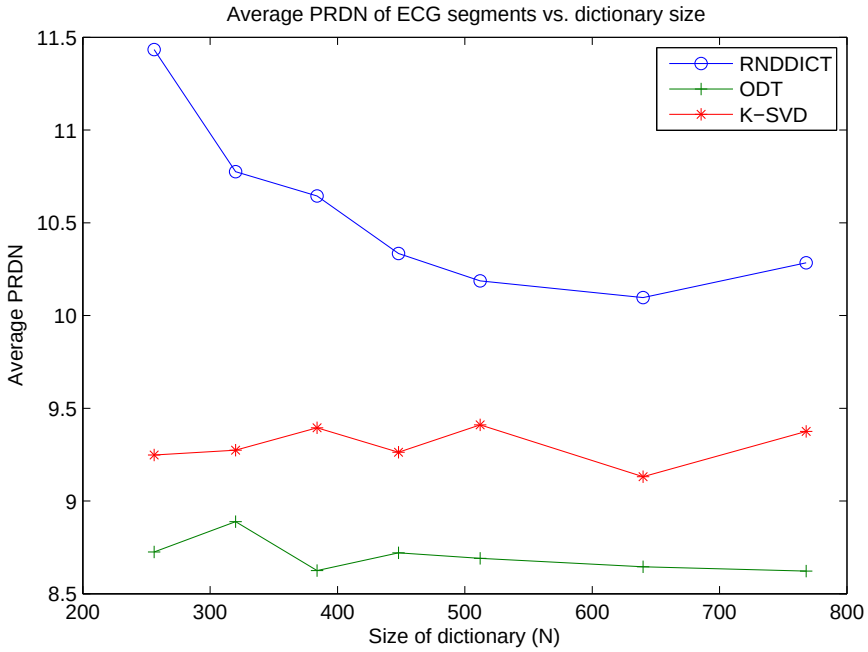


Figure 6.4: Comparison of dictionary learning algorithms for different dictionary sizes. The ODT learning algorithm leads to the smallest average PRDN errors. Larger dictionaries lead to slightly better results.

6.5 Classification in compressed space

Besides signal compression, another interesting application of signal processing for ECG signals is that of classification of heart beats into normal / pathological classes, with the aim of aiding the medical staff in detecting and diagnosing heart-related anomalies.

In [85] it is shown that smaller measurement vectors obtained with random projections can be used instead of the original signals for signal classification with linear kernel SVM, leading to little loss of performance with high probability. The reason is that random projections, due to the concentration of measure phenomenon, have the property of almost preserving linear products of signals. Similarly, in [86] it is argued that random projections can be considered a kind of *universal* features, almost as good as particular features (e.g. principal components) except in the case of very few measurements.

We investigate the possibility of classifying the ECG segments into normal and pathological heart beat classes directly in the compressed space, i.e. working only with random projections of the segments. Similar results were first published in our paper [87].

We investigate classification of compressed ECG segments in two scenarios: (i) with only two signal classes, normal vs. abnormal heart beat and (ii) with all 8 classes, one normal and the seven distinct abnormal classes. The signal are projected on random

vectors with i.i.d normal elements. We use two classification methods:

1. *k-Nearest-Neighbour (kNN)* classification

2. *Multilayer perceptron (MLP)* neural network with one input, one output and one hidden layer. The output layer is composed of 2 or 8 neurons, respectively, each one firing for one corresponding output class.

For 2 classes classification, we use equal training sets of 5000 randomly chosen ECG segments from each class and 3000 segments from each class for testing. The confusion matrices for kNN classification are presented in Table 6.2, whereas with MLP classification are in Table 6.3. Class 1 designates "normal" and class 2 designates "abnormal". As we can see, using random projections for classification reduces the classification accuracy by merely a few percent.

For classification with 8 distinct classes, we use equal training sets of 599 segments per class and 150 segments from each class for testing (the number of segments is given by the minimum class size among the 8 classes). The confusion matrices are presented in Table 6.4 and Table 6.5. In this case class 0 designates "normal" and classes 1 to 7 designate the abnormal classes: "Left bundle branch block beat","Right bundle branch block beat","Premature ventricular contraction","Fusion of ventricular and normal beat","Paced beat","Fusion of paced and normal beat" and "Atrial premature beat".

The results for 8-class kNN classification are in Table 6.4, whereas with MLP classification are in Table 6.5. One can see that MLP classification is significantly poorer, especially with the original signals; this is caused by the small number of training data available, leading to poor training of the neural network. kNN classification, on the other hand, provides good results. Similarly to the 2 classes scenario, using random projections for classification leads to only a few percent loss in classification performance.

Using random Bernoulli projection matrices (with either 0/1 or +1/-1 i.i.d elements) yields results similar to random normal projections. The corresponding confusion matrices are provided in Appendix A.

6.6 Reconstructing with specific dictionaries

In this section we propose an enhanced reconstruction technique for ECG segments, based on having specific dictionaries for each class of normal/abnormal signals, and reconstructing each segment using the corresponding dictionary of its class. The decision of the appropriate dictionary is taken by classifying in the compressed space using the

	1	2
1	98.83%	1.17%
2	1.10%	98.90%

(a) Original signals

	1	2
1	98.23%	1.77%
2	1.87%	98.13%

(b) $m = 17$ normal projections

Table 6.2: Confusion matrix for normal/abnormal classification with kNN

	1	2
1	96.80%	3.20%
2	4.70%	95.30%

(a) Original signals

	0	1
1	95.70%	4.30%
2	5.43%	94.57%

(b) $m = 17$ normal projections

Table 6.3: Confusion matrix for normal/abnormal classification with MLP

kNN method analysed above. Having a dedicated dictionary for each class means that the characteristics of each class are better captured than with a single general dictionary, and as such we can expect better reconstruction if the signal class can be correctly determined from the compressed measurements.

Our proposed approach can be summarized as follows:

1. Create dictionaries for each class, fix an acquisition matrix and set-up the training set
2. Acquire a test signal
3. Determine the signal class by classifying the compressed measurement vector with kNN
4. Reconstruct the signal using the dictionary of that class

The performance of this scheme depends on the accuracy of the classification, and we have seen in Section 6.5 that we can expect a classification accuracy almost as good as with the original signals. We create the dictionaries by simply randomly picking signals from the specified class. The reason is that some classes do not have enough signals to advocate using a dictionary training algorithm, and in addition we saw in Section 6.3 that custom dictionaries created this way are close to the performance of learned dictionaries.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	89.33%	0.00%	0.00%	1.33%	5.33%	0.67%	1.33%	2.00%
2	0.00%	98.67%	0.00%	0.00%	0.67%	0.00%	0.67%	0.00%
3	0.67%	0.67%	98.67%	0.00%	0.00%	0.00%	0.00%	0.00%
4	0.67%	0.00%	1.33%	94.00%	2.67%	1.33%	0.00%	0.00%
5	0.00%	1.33%	0.00%	2.00%	95.33%	0.00%	1.33%	0.00%
6	0.00%	0.00%	0.00%	0.00%	0.00%	99.33%	0.67%	0.00%
7	0.00%	0.00%	0.00%	0.00%	0.00%	2.00%	98.00%	0.00%
8	4.00%	0.00%	0.67%	1.33%	0.67%	0.00%	0.00%	93.33%

(a) Original signals

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	83.33%	0.67%	0.00%	1.33%	6.00%	0.67%	4.00%	4.00%
2	0.67%	98.67%	0.00%	0.00%	0.00%	0.00%	0.67%	0.00%
3	0.00%	0.67%	98.67%	0.00%	0.00%	0.00%	0.67%	0.00%
4	2.00%	1.33%	1.33%	91.33%	2.67%	1.33%	0.00%	0.00%
5	0.00%	1.33%	0.00%	4.00%	92.67%	0.00%	1.33%	0.67%
6	0.00%	0.00%	0.00%	0.00%	0.00%	99.33%	0.67%	0.00%
7	0.67%	0.67%	0.00%	0.00%	0.00%	3.33%	95.33%	0.00%
8	4.67%	0.00%	1.33%	0.00%	2.00%	0.00%	0.00%	92.00%

(b) $m = 17$ normal projections

Table 6.4: Confusion matrix for 8-class classification with kNN

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	82.00%	0.00%	3.33%	3.33%	6.00%	0.67%	1.33%	3.33%
2	2.67%	44.67%	0.00%	0.67%	49.33%	0.67%	0.00%	2.00%
3	2.00%	0.00%	96.67%	0.00%	1.33%	0.00%	0.00%	0.00%
4	1.33%	0.67%	1.33%	80.67%	3.33%	0.67%	0.67%	11.33%
5	4.00%	0.00%	0.00%	5.33%	89.33%	0.00%	0.00%	1.33%
6	0.00%	0.67%	0.67%	0.67%	2.00%	96.00%	0.00%	0.00%
7	3.33%	0.00%	0.00%	0.00%	82.00%	5.33%	8.00%	1.33%
8	4.67%	0.67%	4.00%	0.67%	2.67%	0.00%	1.33%	86.00%

(a) Original signals

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	80.00%	0.00%	2.00%	0.67%	6.00%	0.00%	6.67%	4.67%
2	0.00%	97.33%	0.00%	0.67%	0.00%	0.00%	2.00%	0.00%
3	3.33%	0.00%	95.33%	0.67%	0.00%	0.00%	0.67%	0.00%
4	2.00%	0.00%	2.00%	80.67%	2.67%	0.00%	3.33%	9.33%
5	6.67%	0.67%	0.00%	2.00%	87.33%	2.00%	1.33%	0.00%
6	0.67%	0.00%	0.00%	2.00%	0.00%	97.33%	0.00%	0.00%
7	6.67%	2.00%	2.00%	2.00%	0.00%	3.33%	81.33%	2.67%
8	9.33%	1.33%	1.33%	0.67%	0.00%	0.00%	1.33%	86.00%

(b) $m = 17$ normal projections

Table 6.5: Confusion matrix for 8-class classification with MLP

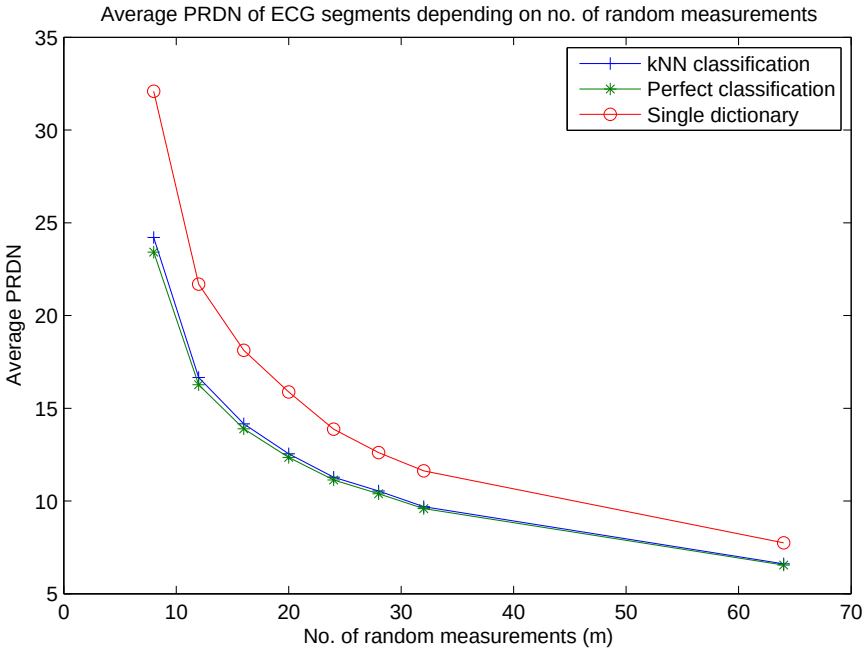


Figure 6.5: Average PRDN of ECG segments with class-specific dictionaries

The dictionary size is 599 for each class (599 is the size of the training set of the smallest class).

Fig.6.5 presents the average PRDN of reconstructed ECG segments with (i) class-specific dictionaries. For comparison, we also present the results with (ii) perfect classification, using a priori class knowledge and (iii) using a single general dictionary of same size (600 atoms), consisting of an equal number of atoms from each of the 8 classes. We observe notable improvements with class-specific dictionaries over the scenario with one single general dictionary. In addition, having perfect signal classification in the compressed space does not lead to any further improvements, which means that signal classification in the compressed domain with our kNN approach provides sufficient accuracy.

6.7 Robust reconstruction

6.7.1 Using correlated projection vectors

Two further improvements have been proposed in our paper [88]. First, given that the bases or dictionaries created by choosing ECG segments from a training set are highly non-orthogonal (due to the similarity of the atoms), using correlated projection vectors is more adequate than using isotropic projection vectors. This provides increased robustness

against the fact that the ECG segments are only approximate sparse in the considered dictionaries.

We propose choosing the projection vectors from a multivariate normal distribution with covariance matrix equal to $B^T B$. In [88] we compare three methods for choosing the acquisition matrix for compressed sensing of a vector that is sparse in a highly non-orthogonal basis or overcomplete dictionary.

The first approach is to simply select a matrix with random i.i.d. normal elements R. We denote this kind of acquisition matrix as P_1.

A second approach is to choose an acquisition matrix as $P_2 = RB^{-1}$, i.e. as a random matrix$R$ multiplied with the inverse of B. In this case B^{-1} cancels out the effect of the non-orthogonal matrix B, leaving the sparse decomposition vector being sensed directly with the random matrix. In this case the reconstruction is affected by the non-orthogonal transform as explained in Section 5.2.1. Even though the sparse vector may be accurately reconstructed, moving into the signal domain may dramatically increase the error. In addition, in case that B is badly conditioned, it is not possible to accurately compute its inverse; to avoid numerical errors, in our experiments we do not actually compute B^{-1}, but rather we compute the decompositions γ using OMP and project γ directly on the random matrix R, thus avoiding the intermediate multiplication with $B^{-1}B$. This would not be possible in a real scenario, but we only use it in order to be able to safely attribute any reconstruction error to the effect of the transform $\hat{x} = B\hat{\gamma}$ and not to numerical imprecisions.

A third alternative is to use a projection matrix defined as $P_3 = RB^T$, i.e. the projection vectors (rows of P_3) are random linear combinations of the atoms, resulting in an acquisition equation system $y = RB^T B\gamma$. In our experiments, this results in the best reconstruction errors among all three alternatives.

Reconstructing with nonorthogonal bases

In the first experiment we create a basis B for ECG representation by randomly picking 300 atoms from the training data. We compare the performance of the three projection matrices P_1, P_2, P_3 created as above; in all three cases we start from the same 20×300 random matrix R with i.i.d elements drawn from a zero-mean, unit norm normal distribution. We perform reconstruction under identical circumstances using ℓ_1 minimization with linear programming. The average error in all three cases is reported in Table 6.6. All the signals and atoms are normalized to 1 in the preprocessing stage, so the error is defined as $||x - \hat{x}||_2/||x||_2$. One observes a large reconstruction error for the second

Table 6.6: Average ECG reconstruction error

Projection matrix	$\mathbf{P_1 = R}$	$\mathbf{P_2 = RB^{-1}}$	$\mathbf{P_3 = RB^T}$
Avg. error (basis)	15.488	108.143	9.506
Avg. error (overcomplete dict.)	12.83	106.52	8.5

matrix; the third matrix is providing the best results.

For further investigation of the three types of projection matrices we create artificial testing datasets of 1000 signals of exact sparsity k in the considered basis B, by randomly combining k atoms with random weights. We set the number of measurements to $m = 40$ and we test the average reconstruction error for increasing values of k. The results are the continuous-line graphs presented in Fig.6.6. For the second matrix, as expected, we obtain perfect recovery up to a certain sparsity level k_0 ($k_0 = 4$ here); further on, the reconstruction error of the coefficient vector $\hat{\gamma}$ leads to very fast increasing error of the reconstructed signal $B\hat{\gamma}$. On the contrary, when using the other two projection matrices, perfect reconstruction stops at a higher sparsity, but later on the signal reconstruction is much more robust to the non-sparsity of the signal. It is also clear that the third projection matrix is better than the first at every sparsity degree. We conclude that the third projection matrix trades inaccurate reconstruction at high sparsity for improved robustness when the signal is not sparse enough.

Reconstructing with overcomplete dictionaries

The same considerations hold when considering overcomplete dictionaries instead of bases for sparse signal representation. In this experiment we use all the 1000 atoms of the training set to form the overcomplete dictionary D. We create the projection matrices starting from a random 20×1000 matrix R with normal distributed elements. We define in the same way $P_1 = R$ and $P_3 = RD^T$; however we cannot define P_2 using the inverse of D since D is not square anymore. For the sake of comparison we define $P_2 = RD^\dagger$ using the pseudoinverse of D instead, even though in this case we have no guarantees about perfect recovery as in the complete case. The average error for all three cases is reported in Table 6.6. We observe a reduction of the average error due to the better representation capability of the overcomplete dictionary compared to the basis.

Investigating the reconstruction error for the same exact-sparse datasets in the overcomplete case yields the dashed graphs in Fig.6.6. As in the previous case the second matrix leads to fast-increasing errors, while the third projection matrix outperforms the

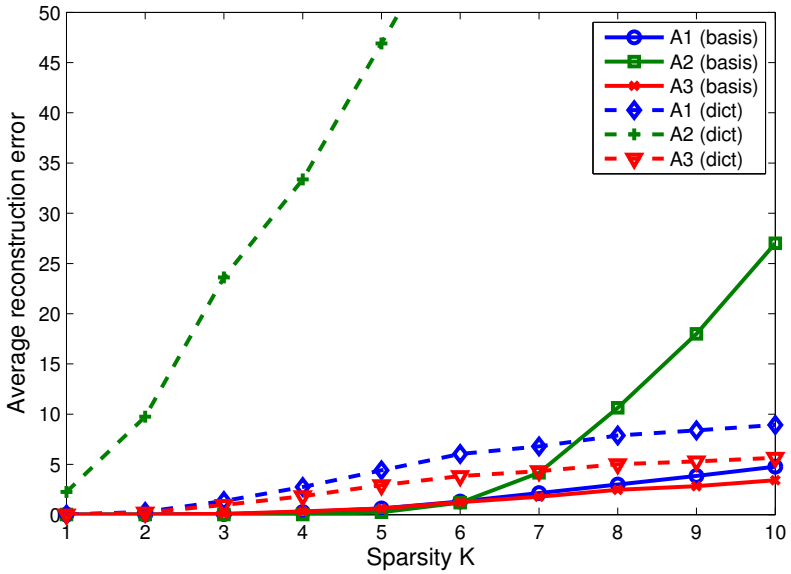


Figure 6.6: Average reconstruction error for decreasing signal sparsity in basis and over-complete dictionary for the three matrices

other two.

6.7.2 Averaging multiple reconstructions

A second improvement is based on the observation that since we create the dictionaries by randomly choosing atoms from a training set, a second dictionary created in the same way has approximately the same representation power as the first. Therefore we propose to perform multiple independent reconstructions of a signal with multiple dictionaries created in the same manner, followed by averaging the reconstructions. The idea is that large, characteristic features of the class are present in most of the ECG segments of the dictionaries and as such they will appear in every independent reconstruction, whereas the errors due to the artefacts and noise of the individual atoms of the dictionaries will be smoothed out by the averaging process.

Fig.6.7 from [88] presents the average ECG segment error after averaging multiple reconstructions. The improvements are more visible for the first 2-3 independent reconstructions.

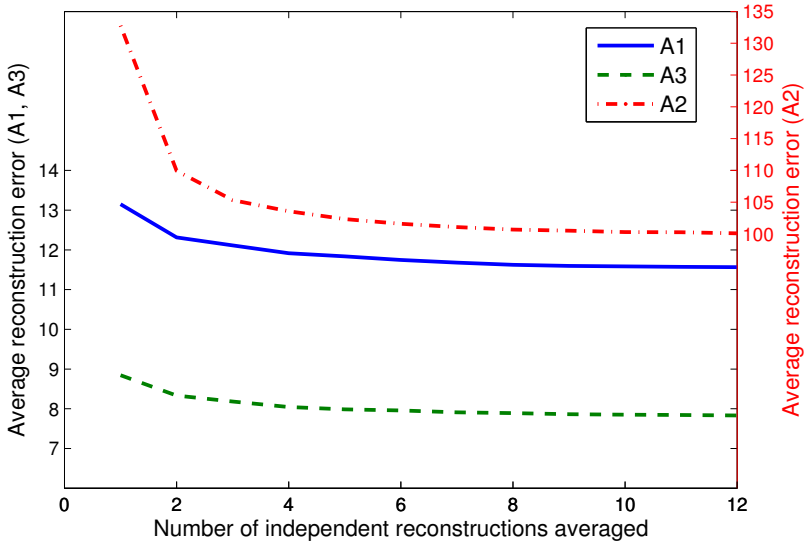


Figure 6.7: Average PRDN of ECG segments with class-specific dictionaries

6.8 Conclusions

In this chapter we investigate the application of compressive sensing to ECG signals. We consider preprocessed ECG signals that undergo segmentation and reshaping to a fixed size. We investigate the best bases for sparse representations of the ECG segments and find that custom and learned bases are much better than standard orthogonal wavelet bases for sparse approximation with few coefficients, as they are capable of much better exploiting the similarity of the ECG segments. This leads to better accuracy when recovery from compressed measurements with a high compression ratio.

We establish that classification into normal and pathological classes in the compressed domain, using significantly fewer random projections of ECG segments, is almost as successful as using the original signals themselves. This is highly beneficial in a diagnosis application.

We propose an improved system for ECG segment recovery based on compressed classification followed by reconstruction with class specific dictionaries. Each of the heart beat classes ("normal" and seven "abnormal" classes) has a dedicated dictionary. When an unknown ECG segment is acquired, we use a kNN classifier to determine its likely class, and then reconstruct the signal using the dictionary associated with its class. Simulations show notable lower errors compared with the case of using a single general dictionary.

Chapter 7

Conclusions

Compressed sensing is an exciting new research area in signal processing, focusing on recovering sparse signals from a number of measurements far fewer than suggested by the classical Nyquist theorem. The fundamental theoretical issue is that a sparse signal can be uniquely represented by a set of incoherent projections, their number being proportional to the "information content" of the signal (number of non-zero coefficients, degrees of liberty, rate of innovation). This can lead to significant dimensionality reduction when acquiring a signal and when post-processing it.

In the digital world, when considering already digitized signals, compressed sensing revolves around the idea that sparse solutions to undetermined equation systems are unique and can be found. Therefore, a sparse vector is uniquely determined by a measurement vector of smaller dimension, albeit the recovery process (finding the sparse signal associated to the measurement vector) is non-linear and in some cases even NP-hard.

7.1 Thesis review and future work

7.1.1 Analysis and synthesis sparsity

One of the interesting recent extensions of compressed sensing theory is the development of the analysis sparsity model as an alternative to the synthesis sparsity model, which we focus on in Chapter 3. The analysis sparsity model specifies that a signal produces a sparse output when analysed with a frame operator, whereas the synthesis model asserts that a signal is generated as a sparse combination of the atoms of a dictionary. Theoretical research in the literature established similar recovery guarantees for analysis sparse signals as in the synthesis case, virtually putting analysis sparsity on an equal foot with synthesis sparsity.

Our contribution focuses on the relation between the two models. The fundamental idea of our approach is that the analysis sparsity can be viewed as synthesis sparsity applied only for the least-squares decomposition of a signal in an overcomplete dictionary. We show that the analysis model can be reformulated as a synthesis model plus a least-squares constraint. The advantage is that now we can think of analysis sparsity as a more constrained version of synthesis sparsity.

The practical benefit that we derive from this interpretation is that we can adapt a variety of algorithms designed for synthesis-based recovery to analysis-based sparse signal recovery as well. We introduce three theorems asserting that the analysis-based signal recovery problem is equivalent to an augmented synthesis-based recovery problem. When considering exact constraints (noiseless case), we show that analysis recovery is equivalent to synthesis recovery with a set of extra equality constraints. As a result, synthesis solvers can be directly applied to analysis problems. The case of analysis recovery with quadratic constraints (i.e. with noisy measurements) is shown to be equivalent to synthesis recovery with mixed quadratic and equality constraints. A number of existing synthesis solvers can be easily modified to include both types of constraints simultaneously, and thus can be adapted for analysis recovery. Synthesis solvers that accept only quadratic constraints can be adapted as well, by expressing the extra equality constraint as a limit case of a quadratic constraint.

Experiments show that this approach is a viable alternative to analysis recovery with both equality or quadratic constraints. Some well-known synthesis solvers, most notably Smoothed ℓ_0, successfully match the performance of the Greedy Analysis Pursuit algorithm specifically designed for analysis recovery under a variety of conditions. Good performance is also obtained with Orthogonal Matching Pursuit and ℓ_1 minimization algorithms.

A second benefit of reformulating analysis sparsity as constrained synthesis sparsity is that we can easier compare the two models, since both of them are instances of dictionary-based sparsity, only with different conditions on the sparse decomposition. We conduct an experimental investigation for determining when is one of the models better than the other in terms of recovering a signal from a few random measurements. Our approach is based on generating bisparse signals, i.e. signals that are simultaneously k-sparse in the synthesis model and l-cosparse in the analysis model with the same dictionary. We generate bisparse test signals for all (k, l) pairs, reconstruct them separately using only synthesis or analysis recovery, and compare the average recovery errors obtained with the two methods.

One of the interesting conclusions of this experiment is that, for sufficiently overcomplete dictionaries, in general there are no bisparse signals for which both sparsity models are very adequate simultaneously. This suggests a fundamental difference of the two models: a signal is either sparse or cosparse (or neither) in some dictionary, but the two models cannot simultaneously be very good. Another conclusion is that the two recovery options perform similarly when recovering signals that are sparse according to the corresponding sparsity model, when the number of measurements is sufficient. However, analysis recovery is significantly more affected than synthesis recovery by a reduction in the number of measurements, and is also less robust with signals that are only approximately sparse.

As future work, there are interesting new research avenues for the analysis model. Dictionary learning algorithms for the analysis model, for example, are not as advanced as in the synthesis case, and this impacts negatively its practical applications. In addition, rigorous guarantees for analysis-bases recovery algorithms adapted through our proposed approach are yet to be established. Moreover, regarding the comparison between the two models, it will be interesting to conduct similar experiments with ℓ_1-sparse signals instead of exact-sparse, as a way of increasing the practical usefulness of the results.

7.1.2 Optimized projections for compressed sensing

Another contribution of this thesis is in the problem of finding optimized acquisition matrices for signals that are sparse in non-orthogonal bases and overcomplete dictionaries, which we explore in Chapter 4. We propose a new framework for finding optimized acquisition matrices by solving a rank-constrained nearest correlation matrix problem. This formulation is provides increased accuracy and robustness by better exploiting the correlations between the dictionary atoms.

We start from three existing state-of-the-art algorithms (proposed by *Elad et al*, *Xu et al*, *Duarte-Carvajalino and Sapiro*) based on the idea of minimizing the mutual coherence of the effective dictionary, and we propose modifications for improving each of them after analysing particular corner cases where they perform sub-optimally. We argue that the first two optimization algorithms can be improved by making the Gram matrix of the effective dictionary progressively closer to the Gram matrix of the dictionary, rather than to the identity matrix, whereas for the third one we propose adding an additional unit-norm constraint to the optimization problem. All the three modified algorithms can be viewed as special instances of a general unified problem: find the matrix that is at minimal distance from the Gram matrix of the dictionary, subject to rank, unit-

norm and semipositivity constraints. Moreover, for the case when the distance metric is the Frobenius norm (ℓ_2), we can use an existing efficient algorithm for solving the optimization problem.

We test the proposed algorithms with random and learned dictionaries. The results indicate significant improvements especially for learned dictionaries, where the inherent correlations of the atoms are much better exploited by our proposed algorithms. Moreover, we have better results than the state-of-the-art algorithms when the number of measurements is smaller.

7.1.3 Compressed sensing with correlated projection vectors

In Chapter 5 we investigate in more detail the effect of choosing correlated projection vectors for compressive sensing of sparse vectors. Our experiments reveal a very interesting phenomenon: when recovering sparse signals acquired with multivariate random normal projection vectors, with a number of popular compressed sensing reconstruction algorithms (Basis Pursuit, Orthogonal Matching Pursuit, Smoothed ℓ_0), the covariance matrix of the resulting reconstruction errors depends very strictly on the covariance matrix of the projection vectors. If the projection vectors are drawn from a multivariate normal distribution with covariance matrix C_P, the covariance matrix of the resulting reconstruction errors has the same eigenvectors, while the eigenvalues are a power law of the eigenvalues of C_P. Specifically, every eigenvalue of the error covariance matrix is proportional to the eigenvalue of the projection vectors covariance matrix raised to the power of approximately -0.42 (experimentally determined).

This property allows to accurately control the covariance matrix of the decomposition errors by properly choosing the covariance matrix of the projection vectors. Thus, it permits better signal recovery along some desirable directions in space, at the expense of larger errors along others. This opens the door to exciting non-standard compressed sensing applications.

One of the applications is compressive sensing of signals that are sparse in non-orthogonal bases. When considering non-orthogonal bases, we would like the recovery of the sparse decomposition vector to be more accurate along the significant components of the basis. Using correlated projection vectors allows exactly this, by properly choosing the projection vectors. This is particularly useful when considering bases that are learned with dictionary learning algorithms, as the typically are highly non-orthogonal since atoms are often rather similar. Our experiments show significant improvements in recovery of image patches with learned bases and dictionaries by considering correlated

projection vectors, resulting in PSNR increases of several dB.

Another possible application is with orthogonal bases where the atoms have unequal importance. As an example, consider image patches that are sparse in the 2D-DCT basis, where the human visual system has different tolerance to noise and errors for different spatial frequencies. Our experiments show much better visual quality when image patches are acquired with correlated projection vectors, such that the errors are smaller along the sensitive low frequency components at the expense of higher frequencies, much more tolerant to noise and errors.

There is much further work possible for exploring this phenomenon in greater depth. The investigation should be expanded to include other reconstruction algorithms that exhibit the same phenomenon. Moreover, a probabilistic model that explains the precise relation between the variances of the projection vectors and of the resulting errors has yet to be developed and explained.

7.1.4 Compressed sensing of ECG signals

In Chapter 6 we put to use many of the compressed sensing techniques with the aim of compression and classification of electrocardiographic (ECG) signals. The ECG signals first undergo a preprocessing step involving segmentation into heart beats, resizing and aligning the peak of the beat at the center of the segments. This allows for much better characterisation of the segments using learned bases/dictionaries rather than standard wavelet bases typically used in literature. Using learned bases allow reaching high compression ratios (e.g. 15:1) with random projections, with acceptable errors. For classification of pathological ECG beats, we propose using bases and dictionaries specific to each pathological class, with the aim of capturing the characteristics of each pathology. Classification in the compressed space indicates which class the signal belongs to, then the signal is reconstructed using the dictionary of the specified class.

7.2 Contributions

The main contributions of this thesis are listed below.

- Proving the equivalence between analysis-based recovery and augmented synthesis-based recovery, with practical validation by adapting synthesis-based recovery algorithms for analysis-based recovery.

- Investigating the relation between the two models based on the above relation, revealing key differences.
- Proposing improved versions for three state-of-the-art algorithms for finding optimized projection vectors, with practical validation.
- Proposing a novel unified approach for optimizing the projection vectors, based on the rank constrained nearest correlation matrix problem.
- Proposing the use of non-isotropic multivariate distributions of the projection vectors as a way of accurately controlling the covariance matrix of the errors, with relevant practical applications validated through simulations.
- Proposing an efficient technique for ECG signal acquisition based on preprocessing, acquisition with random projections and recovery with overcomplete dictionaries.
- Proposing an improved method for reconstruction of compressively sensed ECG signals, based on initial classification in the compressed space followed by reconstruction with class-specific dictionaries.

Chapter 8

Bibliography

[1] G. Kutyniok, "Compressed sensing : Theory and applications," *preprint*, pp. 1–22, 2012.

[2] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[3] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.

[4] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.

[5] E. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, pp. 4203–4215, 2005.

[6] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society. Series B. Methodological*, vol. 58, no. 1, pp. pp. 267–288, 1996.

[7] M. a. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec. 2007.

[8] A. M. Tillmann and M. E. Pfetsch, "The computational complexity of RIP, NSP, and related concepts in compressed sensing," *CoRR*, vol. abs/1205.2081, 2012.

[9] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, 2006.

[10] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, pp. 589 – 592, 2008.

[11] A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation," *Journal of the American Mathematical Society*, vol. 22, no. 1, pp. 211–231, Jul. 2008.

[12] J. Tropp, "Greed is good: algorithmic results for sparse approximation," *Information Theory, IEEE Transactions on*, vol. 50, no. 10, pp. 2231 – 2242, oct. 2004.

[13] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *Information Theory, IEEE Transactions on*, vol. 49, no. 12, pp. 3320 – 3325, dec. 2003.

[14] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Annual Asilomar Conf. on Signals, Systems, and Computers*, 1993, pp. 40–44.

[15] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive approximation*, pp. 57–98, 1997.

[16] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, Jan. 2008.

[17] D. Achlioptas, "Database-friendly random projections," in *Symposium on Principles of Database Systems*, vol. 66, no. 4. ACM, 2001, pp. 274–281.

[18] D. L. Donoho and J. Tanner, "Precise undersampling theorems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 913–924, Jun. 2010.

[19] M. A. Davenport, J. N. Laska, P. Boufounos, and R. G. Baraniuk, "A simple proof that random matrices are democratic," *CoRR*, vol. abs/0911.0736, 2009.

[20] M. Luby, "LT codes," in *Proc. of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, 2002, pp. 271 – 280.

[21] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204 –1216, jul 2000.

[22] T. Ho, M. Medard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proc. of 41st Annual Allerton Conf. on Communication, Control, and Computing*, 2003.

[23] N. Cleju, N. Thomos, and P. Frossard, "Selection of network coding nodes for minimal playback delay in streaming overlays," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 1103 –1115, oct. 2011.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[25] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 2008.

[26] ——, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[27] S. Krstulovic and R. Gribonval, "MPTK: Matching Pursuit made tractable," in *Proc. Int. Conf. Acoust. Speech Signal Process. (ICASSP'06)*, vol. 3, Toulouse, France, May 2006, pp. III–496 – III–499.

[28] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, p. 30, 2008.

[29] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289 –301, jan. 2009.

[30] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, p. 4311, 2006.

[31] "Matlab implementation of the K-SVD algorithm." [Online]. Available: http://www.cs.technion.ac.il/~elad/Various/KSVD_Matlab_ToolBox.zip

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

[33] Y. C. Eldar and M. Mishali, "Block sparsity and sampling over a union of subspaces," in *Proceedings of the 16th international conference on Digital Signal Processing*, ser. DSP'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 1–8.

[34] Y. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *Signal Processing, IEEE Transactions on*, vol. 58, no. 6, pp. 3042 –3054, june 2010.

[35] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[36] E. van den Berg and M. Friedlander, "Theoretical and empirical results for recovery from multiple measurements," *Information Theory, IEEE Transactions on*, vol. 56, no. 5, pp. 2516 –2527, may 2010.

[37] P. Indyk, "Explicit constructions for compressed sensing of sparse signals," in *SODA*, 2008, pp. 30–33.

[38] R. A. DeVore, "Deterministic constructions of compressed sensing matrices," *J. Complexity*, vol. 23, no. 4-6, pp. 918–925, 2007.

[39] A. R. Calderbank, S. D. Howard, and S. Jafarpour, "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property," *J. Sel. Topics Signal Processing*, vol. 4, no. 2, pp. 358–374, 2010.

[40] I. Carron, "Compressive sensing: The big picture." [Online]. Available: https://sites.google.com/site/igorcarron2/cs

[41] "Compressive sensing resources." [Online]. Available: http://dsp.rice.edu/cs

[42] "The Nuit-Blanche blog." [Online]. Available: http://nuit-blanche.blogspot.com

[43] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, pp. 947–968, 2007.

[44] S. Nam, M. Davies, M. Elad, and R. Gribonval, "Cosparse analysis modeling - uniqueness and algorithms," in *Proc. ICASSP 2011*, 2011, pp. 5804–5807.

[45] S. Nam, M. E. Davies, M. Elad, and R. Gribonval, "The cosparse analysis model and algorithms," INRIA, Research Report, 2011.

[46] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Computational Harmonic Analysis*, vol. 31, pp. 59–73, 2011.

[47] M. a. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec. 2007.

[48] N. Cleju, M. Jafari, and M. D. Plumbley, "Analysis-based sparse reconstruction with synthesis-based solvers," in *Proc. ICASSP 2012*, 2012, pp. 5401–5404.

[49] U. Kamilov, E. Bostan, and M. Unser, "Wavelet shrinkage with consistent cycle spinning generalizes total variation denoising," *IEEE Signal Processing Letters*, vol. 19, no. 4, pp. 187 –190, april 2012.

[50] Y.-H. Dai, "Fast algorithms for projection on an ellipsoid," *SIAM Journal on Optimization*, vol. 16, no. 4, pp. 986–1006, Apr. 2006.

[51] A. Eftekhari, M. Babaie-Zadeh, C. Jutten, and H. Moghaddam, "Robust-SL0 for stable sparse representation in noisy settings," in *Proc. ICASSP 2009*, april 2009, pp. 3433 –3436.

[52] E. Candès and J. Romberg, *ℓ_1-MAGIC: Recovery of Sparse Signals via Convex Programming*, http://users.ece.gatech.edu/ justin/l1magic/.

[53] A. Maleki and D. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 330–341, 2010.

[54] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *Information Theory, IEEE Transactions on*, vol. 55, no. 5, pp. 2230 –2249, may 2009.

[55] B. L. Sturm, "A study on sparse vector distributions and recovery from compressed sensing," *CoRR*, vol. abs/1103.6246, 2011.

[56] S. Becker, J. Bobin, and E. J. Candes, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.

[57] D. L. Donoho and J. Tanner, "Precise undersampling theorems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 913–924, 2010.

[58] E. J. Candes and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, Jun. 2007.

[59] M. Elad, "Optimized projections for compressed sensing," *Signal Processing, IEEE Transactions on*, vol. 55, no. 12, pp. 5695 –5702, dec. 2007.

[60] J. Xu, Y. Pi, and Z. Cao, "Optimized projection matrix for compressive sensing," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, p. 560349, 2010.

[61] J. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *Image Processing, IEEE Transactions on*, vol. 18, no. 7, pp. 1395 –1408, july 2009.

[62] Y. Gao and D. Sun, "A majorized penalty approach for calibrating rank constrained correlation matrix problems," National University of Singapore, Tech. Rep., 2012.

[63] L. Welch, "Lower bounds on the maximum cross correlation of signals (corresp.)," *Information Theory, IEEE Transactions on*, vol. 20, no. 3, pp. 397 – 399, may 1974.

[64] V. Malozemov and A. Pevnyi, "Equiangular tight frames," *Journal of Mathematical Sciences*, vol. 157, pp. 789–815, 2009, 10.1007/s10958-009-9366-6.

[65] R. Pietersz and P. J. F. Groenen, "Rank reduction of correlation matrices by majorization," *Quantitative Finance*, vol. 4, no. 6, pp. 649–662, 2004.

[66] Y. Gao and D. Sun, "Calibrating least squares semidefinite programming with equality and inequality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 1432–1457, Dec. 2009.

[67] E. Candes and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *Information Theory, IEEE Transactions on*, vol. 57, no. 11, pp. 7235 –7254, nov. 2011.

[68] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, p. 11, 2011.

[69] E. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925 –936, june 2010.

[70] I. B. Ciocoiu, "ECG signal compression using 2D wavelet foveation," in *ICHIT*, 2009, pp. 576–580.

[71] H. Peterson, J. A. J. Ahumada, and A. Watson, "An improved detection model for DCT coefficient quantization," in *Proc. SPIE*, 1993, pp. 191–201.

[72] ITU, "ISO/IEC 10918-1 : 1993(E) CCIT Recommendation T.81," 1993. [Online]. Available: http://www.w3.org/Graphics/JPEG/itu-t81.pdf

[73] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *ICML*, 2009, p. 87.

[74] ——, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.

[75] A. Djohan, T. Nguyen, and W. Tompkins, "ECG compression using discrete symmetric wavelet transform," in *Engineering in Medicine and Biology Society, 1995., IEEE 17th Annual Conference*, vol. 1, sep 1995, pp. 167 –168 vol.1.

[76] S.-G. Miaou, H.-L. Yen, and C.-L. Lin, "Wavelet-based ECG compression using dynamic vector quantization with tree codevectors in single codebook," *Biomedical Engineering, IEEE Transactions on*, vol. 49, no. 7, pp. 671 –680, july 2002.

[77] P. de Chazal, M. O'Dwyer, and R. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 7, pp. 1196 –1206, july 2004.

[78] S. Yu and K. Chou, "Integration of independent component analysis and neural networks for ECG beat classification," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2841–2846, 2008.

[79] G. Moody and R. Mark, "The impact of the MIT-BIH Arrhythmia database," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 20, no. 3, pp. 45 –50, may-june 2001.

[80] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[81] M. Fira and L. Goras, "Biomedical signal compression based on basis pursuit," in *Proceedings of the 2009 International Conference on Hybrid Information Technology*, ser. ICHIT '09. New York, NY, USA: ACM, 2009, pp. 541–545.

[82] M. Fira, L. Goraș, C. Barabasa, and N. Cleju, "On ecg compressed sensing using specific overcomplete dictionaries," *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 23–28, 2010.

[83] K. Kanoun, H. Mamaghanian, N. Khaled, and D. Atienza, "A real-time compressed sensing-based personal electrocardiogram monitoring system," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2011*, march 2011, pp. 1 –6.

[84] "The WAVELAB 850 Matlab package." [Online]. Available: http://www-stat.stanford.edu/~wavelab/Wavelab_850/index_wavelab850.html

[85] R. Calderbank and S. Jafarpour, "Finding needles in compressed haystacks," in *Proc. ICASSP 2012*, 2012, pp. 3441–3444.

[86] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 210–227, February 2009.

[87] M. Fira, L. Goras, N. Cleju, and C. Barabasa, "On the projection matrices influence in the classification of compressed sensed ECG signals," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 8, pp. 141–145, 2012.

[88] N. Cleju, C. M. Fira, C. Barabasa, and L. Goras, "Robust reconstruction of compressively sensed ECG signals," in *Proc. ISSCS 2011*, 2011, pp. 507–510.

Appendix A

Further classification results

This appendix contains the classification results (confusion matrices) for classification of compressively sensed ECG segments, when the acquisition matrix is a random matrix with Bernoulli i.i.d. entries, as explained in Section 6.5.

When using random matrices with Bernoulli +1/-1 entries, the results for classification with two classes are presented in Table A.1 and Table A.2, and with eight classes are in Table A.3 and Table A.4.

For random matrices with Bernoulli +1/0 entries, the corresponding results for classification with two classes are presented in Table A.5 and Table A.6, and with eight classes are in Table A.7 and Table A.8.

	1	**2**
1	98.83%	1.17%
2	1.10%	98.90%

(a) Original signals

	1	**2**
1	98.23%	1.77%
2	1.97%	98.03%

(b) $m = 17$ normal projections

Table A.1: Confusion matrix for normal/abnormal classification with kNN

	1	2
1	96.80%	3.20%
2	4.70%	95.30%

(a) Original signals

	0	1
1	95.17%	4.83%
2	6.40%	93.60%

(b) $m = 17$ normal projections

Table A.2: Confusion matrix for normal/abnormal classification with MLP

	1	2	3	4	5	6	7	8
1	89.33%	0.00%	0.00%	1.33%	5.33%	0.67%	1.33%	2.00%
2	0.00%	98.67%	0.00%	0.00%	0.67%	0.00%	0.67%	0.00%
3	0.67%	0.67%	98.67%	0.00%	0.00%	0.00%	0.00%	0.00%
4	0.67%	0.00%	1.33%	94.00%	2.67%	1.33%	0.00%	0.00%
5	0.00%	1.33%	0.00%	2.00%	95.33%	0.00%	1.33%	0.00%
6	0.00%	0.00%	0.00%	0.00%	0.00%	99.33%	0.67%	0.00%
7	0.00%	0.00%	0.00%	0.00%	0.00%	2.00%	98.00%	0.00%
8	4.00%	0.00%	0.67%	1.33%	0.67%	0.00%	0.00%	93.33%

(a) Original signals

	1	2	3	4	5	6	7	8
1	82.00%	0.67%	0.67%	1.33%	8.00%	0.00%	2.67%	4.67%
2	0.67%	98.00%	0.00%	0.00%	0.67%	0.00%	0.67%	0.00%
3	0.67%	0.67%	98.67%	0.00%	0.00%	0.00%	0.00%	0.00%
4	2.67%	0.00%	0.00%	92.67%	2.67%	0.00%	1.33%	0.67%
5	0.00%	0.00%	0.00%	0.67%	96.67%	0.00%	1.33%	1.33%
6	0.00%	0.00%	0.00%	0.67%	0.00%	99.33%	0.00%	0.00%
7	0.00%	0.00%	0.00%	0.00%	0.00%	2.67%	96.67%	0.67%
8	3.33%	0.00%	0.67%	0.67%	0.67%	0.00%	1.33%	93.33%

(b) $m = 17$ normal projections

Table A.3: Confusion matrix for 8-class classification with kNN

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	82.00%	0.00%	3.33%	3.33%	6.00%	0.67%	1.33%	3.33%
2	2.67%	44.67%	0.00%	0.67%	49.33%	0.67%	0.00%	2.00%
3	2.00%	0.00%	96.67%	0.00%	1.33%	0.00%	0.00%	0.00%
4	1.33%	0.67%	1.33%	80.67%	3.33%	0.67%	0.67%	11.33%
5	4.00%	0.00%	0.00%	5.33%	89.33%	0.00%	0.00%	1.33%
6	0.00%	0.67%	0.67%	0.67%	2.00%	96.00%	0.00%	0.00%
7	3.33%	0.00%	0.00%	0.00%	82.00%	5.33%	8.00%	1.33%
8	4.67%	0.67%	4.00%	0.67%	2.67%	0.00%	1.33%	86.00%

(a) Original signals

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	74.67%	2.00%	0.00%	3.33%	4.00%	0.00%	10.67%	5.33%
2	0.00%	96.00%	0.00%	3.33%	0.00%	0.00%	0.67%	0.00%
3	2.00%	0.00%	94.67%	0.00%	0.00%	0.00%	2.67%	0.67%
4	2.67%	0.00%	0.00%	82.67%	4.00%	1.33%	4.00%	5.33%
5	3.33%	0.00%	0.00%	1.33%	86.00%	0.00%	4.00%	5.33%
6	0.00%	0.00%	0.00%	0.67%	0.00%	98.00%	1.33%	0.00%
7	0.67%	0.67%	0.00%	2.00%	1.33%	2.67%	92.67%	0.00%
8	5.33%	0.00%	2.67%	2.00%	0.67%	0.00%	1.33%	88.00%

(b) $m = 17$ normal projections

Table A.4: Confusion matrix for 8-class classification with MLP

	1	**2**
1	98.83%	1.17%
2	1.10%	98.90%

(a) Original signals

	1	**2**
1	97.67%	2.33%
2	1.97%	98.03%

(b) $m = 17$ normal projections

Table A.5: Confusion matrix for normal/abnormal classification with kNN

	1	**2**
1	96.80%	3.20%
2	4.70%	95.30%

(a) Original signals

	0	**1**
1	96.40%	3.60%
2	5.33%	94.67%

(b) $m = 17$ normal projections

Table A.6: Confusion matrix for normal/abnormal classification with MLP

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	89.33%	0.00%	0.00%	1.33%	5.33%	0.67%	1.33%	2.00%
2	0.00%	98.67%	0.00%	0.00%	0.67%	0.00%	0.67%	0.00%
3	0.67%	0.67%	98.67%	0.00%	0.00%	0.00%	0.00%	0.00%
4	0.67%	0.00%	1.33%	94.00%	2.67%	1.33%	0.00%	0.00%
5	0.00%	1.33%	0.00%	2.00%	95.33%	0.00%	1.33%	0.00%
6	0.00%	0.00%	0.00%	0.00%	0.00%	99.33%	0.67%	0.00%
7	0.00%	0.00%	0.00%	0.00%	0.00%	2.00%	98.00%	0.00%
8	4.00%	0.00%	0.67%	1.33%	0.67%	0.00%	0.00%	93.33%

(a) Original signals

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	85.33%	0.00%	0.67%	0.00%	6.00%	0.67%	5.33%	2.00%
2	0.67%	98.67%	0.00%	0.00%	0.67%	0.00%	0.00%	0.00%
3	0.67%	0.00%	98.67%	0.67%	0.00%	0.00%	0.00%	0.00%
4	0.67%	0.00%	0.00%	93.33%	2.00%	0.67%	0.00%	3.33%
5	2.00%	2.00%	0.00%	1.33%	94.67%	0.00%	0.00%	0.00%
6	0.00%	0.00%	0.00%	1.33%	0.67%	97.33%	0.67%	0.00%
7	0.00%	0.67%	1.33%	0.00%	0.00%	1.33%	96.67%	0.00%
8	4.67%	0.67%	0.00%	2.67%	0.67%	0.67%	0.00%	90.67%

(b) $m = 17$ normal projections

Table A.7: Confusion matrix for 8-class classification with kNN

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	82.00%	0.00%	3.33%	3.33%	6.00%	0.67%	1.33%	3.33%
2	2.67%	44.67%	0.00%	0.67%	49.33%	0.67%	0.00%	2.00%
3	2.00%	0.00%	96.67%	0.00%	1.33%	0.00%	0.00%	0.00%
4	1.33%	0.67%	1.33%	80.67%	3.33%	0.67%	0.67%	11.33%
5	4.00%	0.00%	0.00%	5.33%	89.33%	0.00%	0.00%	1.33%
6	0.00%	0.67%	0.67%	0.67%	2.00%	96.00%	0.00%	0.00%
7	3.33%	0.00%	0.00%	0.00%	82.00%	5.33%	8.00%	1.33%
8	4.67%	0.67%	4.00%	0.67%	2.67%	0.00%	1.33%	86.00%

(a) Original signals

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	82.67%	0.67%	0.00%	2.67%	3.33%	0.00%	7.33%	3.33%
2	0.00%	96.67%	0.00%	2.67%	0.00%	0.00%	0.67%	0.00%
3	3.33%	0.00%	96.00%	0.00%	0.00%	0.00%	0.67%	0.00%
4	0.67%	0.67%	0.67%	79.33%	4.67%	0.67%	0.00%	13.33%
5	2.67%	0.67%	0.00%	2.00%	90.67%	0.00%	3.33%	0.67%
6	0.00%	0.00%	0.00%	2.00%	0.00%	98.00%	0.00%	0.00%
7	1.33%	3.33%	0.00%	0.00%	0.00%	4.00%	91.33%	0.00%
8	6.00%	0.00%	1.33%	1.33%	0.00%	0.67%	0.67%	90.00%

(b) $m = 17$ normal projections

Table A.8: Confusion matrix for 8-class classification with MLP

List of publications

Journal papers

- M. Fira, L. Goras, **N. Cleju**, and C. Barabasa, "On the projection matrices influence in the classifcation of compressed sensed ECG signals," International Journal of Advanced Computer Science and Applications, vol. 3, no. 8, pp. 141-145, 2012.

- **N. Cleju**, N. Thomos, and P. Frossard, "Selection of network coding nodes for minimal playback delay in streaming overlays,"IEEE Transactions on Multimedia, vol. 13, no. 5, pp. 1103-1115, oct. 2011.

- M. Fira, L. Goras, C. Barabasa, and **N. Cleju**, "On ECG Compressed Sensing using Specific Overcomplete Dictionaries," Advances in Electrical and Computer Engineering, vol. 10, no. 4, pp. 23-28, 2010.

Conference papers

- **N. Cleju**, M. Jafari, and M. D. Plumbley, "Choosing analysis or synthesis recovery for sparse reconstruction," in Proc. EUSIPCO 2012, 2012, pp.869-873.

- **N. Cleju**, M. Jafari, and M. D. Plumbley, "Analysis-based sparse reconstruction with synthesis-based solvers," in Proc. ICASSP 2012, Kyoto, Japan, pp. 5401 5404.

- C. M. Fira, L. Goras, C. Barabasa, and **N. Cleju**, "ECG compressed sensing based on classification in compressed space and specified dictionaries," in Proc. 20th European Signal Processing Conference EUSIPCO 2011, 2011, pp. 1573 - 1577.

- **N. Cleju**, C. M. Fira, C. Barabasa, and L. Goras, "Robust reconstruction of compressively sensed ECG signals," in Proc. ISSCS 2011, Iasi, Romania, pp. 507 - 510.

- M. Fira, L. Goras, **N. Cleju**, and C. Barabasa, "On the classification of compressed sensed signals," in Proc. International Symposium on Signals, Circuits and Systems ISSCS 2011, 2011.

- **N. Cleju**, N. Thomos, and P. Frossard, "Network coding node placement for delay minimization in streaming overlays," in Proc. IEEE International Conference on Communications ICC 2010, 2010.

- M. Fira, L. Goras, **N. Cleju**, and C. Barabasa, "On the possibilities of ECG signals compressed sensing," in Proc. 6th European Conference on Intelligent Systems and Technologies ECIT 2010, 2010.