

DISSERTATION

Titel der Dissertation

Compressive Channel Estimation Compressed Sensing Methods for Estimating Doubly Selective Channels in Multicarrier Systems

Verfasser

Mag. Daniel Eiwien

angestrebter akademischer Grad

Doktor der Naturwissenschaften (Dr.rer.nat)

Wien, Mai 2012

Studienkennzahl lt. Studienblatt: A 091 405

Matrikelnummer: 0107290

Studienrichtung lt. Studienblatt: Mathematik

Betreuer: a.o. Prof. Dr. Hans G. Feichtinger

Acknowledgments

I want to thank my advisor, Prof. Hans G. Feichtinger, for offering me to become a part of the NuHAG team. Your great motivation and vision, your inimitable way of explaining even the most difficult mathematical facts in terms of everyday life metaphors, and your patience even if you have to explain the same fact over and over again, have made it a great pleasure to work under your guidance. Thank you for all your effort to manage a wonderful group, for the countless discussions, words of advice, and all the support you gave to me.

Also, I am very grateful to my colleague Dr. Georg Tauböck, who introduced me to the ideas and concepts of wireless communications and compressed sensing when he allowed me to take part in his WWTF research project *Sparse Signals and Operators: Theory, Methods, and Applications* (SPORTS), which mainly funded my research. Thank you for all your support, all the hours of discussions, all the practical help you gave to me. You have been the most wonderful colleague to work with, meeting with you has always been the highlight of the week (or month...). I have learned so much from you. Thank you for everything!

Moreover, I want to thank Prof. Franz Hlawatsch for taking me under his wing and inviting me into his group as well. Thank you for all your great advice and support, all your help with the papers, and also for the pleasant hours during our conference trips.

I am also very grateful to all my colleagues at NuHAG and at the University of Technology, especially to Elmar Pauwels, Sebastian Schmutzhard, Saptarshi Das, Alex Jung and Christoph Neumann for many helpful discussions and suggestions. Also I want to thank the other project leader of SPORTS, Prof. Holger Rauhut, for many comments and ideas guiding me the way, and especially for inviting me to Bonn to learn from him, as well as some of the leading researchers in the field of compressed sensing.

Acknowledgments

There are hardly any words to describe how grateful I am to my wife Julie, who has supported me throughout my entire studies. Thank you for always being there for me, for loving me just the way I am, for helping me through the difficult times and celebrating the beautiful moments with me. You are everything I ever wished for! I love you!

Furthermore, I want to thank my entire family for the wonderful life they have enabled me to live. Dad, you and Mum have always been there for me, you have really shown me a glimpse of God's unconditional love and comfort. I am so grateful for the privilege of being brought up by such wonderful parents. I love you! Also I want to thank my wonderful siblings Elisabeth, Brigitte, Angelika, Christian and Johannes as well as our wonderful Angie for making me part of a family that truly loves and cares for each other.

Most and for all I want to thank God, the creator of heaven, earth and every living being therein. Thank You for Your incredible love and grace that You have shown us by sending Your son Jesus to the earth in order to enable us to live in a relationship with You from heart to heart! Thank You for the wonderful life You have given me and all the great people I get to share my life with! I love You!

Abstract

In this thesis we investigate *compressive channel estimation* (CCE), i.e. the application of the theory and methodology of Compressed Sensing (CS) to the problem of estimating doubly selective channels in multicarrier systems. After a brief introduction to wireless communications and a short survey of CS and some of its variations, we review the basic compressive channel estimator that was introduced in [1]. We analyze its performance as well as its computational complexity, and we explore the basic assumption underlying the compressive estimator, namely the delay-Doppler sparsity of typical channels, in more detail.

Based on this analysis, we propose several variations and extensions of the conventional compressive channel estimator. First, we make use of the fact that typical channels can be considered group sparse as well. This is due to the so-called leakage effect, which actually impairs the performance of any channel estimator utilizing CS techniques and therefore is one of the main challenges in CCE. Then, we investigate the extension of the compressive estimators to the multi-antenna (MIMO) case. We show that the various cross-channels of a MIMO system can (approximately) be considered jointly sparse, even jointly group sparse, and that therefore the methodology of multichannel CS can be utilized. Last, by using the recently introduced concept of modified CS, we exploit the approximate sequential sparsity of the channel in order to track it over a period of several consecutive symbol blocks. This approach can yield an additional performance gain, but more importantly it can substantially reduce the computational complexity of the method.

Additionally, we adapt the basis optimization techniques introduced in [2, 3] to the various settings, and we present simulation results that demonstrate the performance gains that can be achieved by using each of the compressive estimators presented in this thesis.

Zusammenfassung

In dieser Arbeit untersuchen wir die *kompressive Kanalschätzung* (KKS), also die Anwendung der Theorie und Methodologie des Compressed Sensing (CS) auf das Problem der Kanalschätzung doppelt selektiver Kanäle in Multicarrier-Systemen. Nach einer kurzen Einführung in die kabellose Kommunikation und einem kleinen Überblick über CS und einigen seiner Varianten betrachten wir die in [1] präsentierte elementare kompressive Kanalschätzmethode. Wir analysieren ihre Leistungsfähigkeit sowie ihre Komplexität, und wir untersuchen die ihr zugrundeliegende Annahme, nämlich die "delay-Doppler sparsity" typischer Kanäle, genauer.

Aufbauend auf dieser Analyse stellen wir einige Varianten und Erweiterungen der kompressiven Kanalschätzmethode vor. Zuerst nutzen wir die Tatsache dass typische Kanäle auch als "group sparse" angesehen werden können. Dies ist eine Folge des sogenannten Leck Effekts, welcher die Leistung einer jeden kompressiven Kanalschätzmethode beeinträchtigt und daher eine enorme Herausforderungen für die KKS darstellt. Weiters betrachten wir die Erweiterung der kompressiven Schätzmethode auf Mehrantennensysteme (MIMO). Wir zeigen, dass die einzelnen Querkanäle eines solchen MIMO Systems (in etwa) als "jointly sparse", sogar als "jointly group sparse" angesehen, und daher Methoden des Multichannel CS (MCS) verwendet werden können. Letztens nutzen wir - unter Verwendung der Konzepte des Modified CS (MOD-CS)- die approximative "sequential sparsity" des Kanals zum Kanal-Tracking über mehrere aufeinanderfolgende Symbolblöcke hinweg. Diese Vorgehensweise kann die Leistung zusätzlich steigern, viel wichtiger jedoch, sie kann die Komplexität der Methode reduzieren.

Darüber hinaus adaptieren wir die Technik der Basis-Optimierung, welche in [2, 3] vorgestellt wurde, für die verschiedenen Szenarien, und wir präsentieren Simulationsergebnisse, welche die verbesserte Leistung all jener Kanalschätzmethoden demonstrieren, die in dieser Arbeit erklärt werden.

Contents

Acknowledgments	iii
Abstract	v
Zusammenfassung	vii
1 Introduction	1
1.1 Overview	1
1.2 Previous Work	3
1.3 Contributions	3
2 Mathematical Preliminaries	5
2.1 Vectors and Matrices	5
2.2 The Fourier Transform	8
2.3 Some Special Functions	9
2.4 Probability Theory	10
2.5 Landau Notation	12
3 Wireless Communications	13
3.1 Introduction	13
3.2 Transceiver Setup	13
3.3 The Wireless Channel	16
3.3.1 Multipath Propagation	16
3.3.2 Time-Invariant Channels	19
3.3.3 Time-Varying Channels	20

3.4	Equivalent Multicarrier System Model	21
3.5	Channel Estimation	23
3.6	Multiple-Input Multiple-Output Systems	26
3.6.1	Introduction	26
3.6.2	MC MIMO System Model	28
3.6.3	MIMO Channel Estimation	29
4	Compressed Sensing	31
4.1	Introduction	31
4.2	Conventional Compressed Sensing	32
4.3	Group Sparse Compressed Sensing	39
4.4	Multichannel Compressed Sensing	42
4.5	Multichannel Group Sparse Compressed Sensing	47
4.6	Modified Compressed Sensing	48
5	Compressive Channel Estimation in SISO Systems	51
5.1	Conventional Compressive Channel Estimation	51
5.1.1	The Method	52
5.1.2	Computational Complexity	57
5.1.3	Delay-Doppler Sparsity	58
5.1.4	Basis Optimization	64
5.1.5	Simulation Results	68
5.2	Compressive Channel Estimation using Group Sparsity Methods	71
5.2.1	The Method	72
5.2.2	Computational Complexity	75
5.2.3	Delay-Doppler Group Sparsity	75
5.2.4	Basis Optimization	78
5.2.5	Simulation Results	80
6	Compressive Channel Estimation in MIMO Systems	85
6.1	Multichannel Compressive Channel Estimation	85
6.1.1	The Method	86
6.1.2	Computational Complexity	92
6.1.3	Joint Delay-Doppler Sparsity	93
6.1.4	Basis Optimization	97

6.1.5	Simulation Results	100
6.2	Multichannel Compressive Channel Estimation using Group Sparsity Methods	103
6.2.1	The Method	104
6.2.2	Computational Complexity	107
6.2.3	Joint Delay-Doppler Group Sparsity	108
6.2.4	Basis Optimization	109
6.2.5	Simulation Results	111
7	Compressive Channel Tracking	115
7.1	Compressive Channel Tracking in SISO Systems	115
7.1.1	The Method	115
7.1.2	Computational Complexity	118
7.1.3	Sequential Delay-Doppler Sparsity	119
7.2	Extension to the MIMO Case	122
7.3	Simulation Results	124
8	Conclusion	129
	Bibliography	134

Abbreviations

ASK	Amplitude-Shift Keying
BEM	Basis Expansion Model
BP	Basis Pursuit
BPDN	Basis Pursuit Denoising
CCE	Compressive Channel Estimation
CoSaMP	Compressive Sampling Matching Pursuit
CP-OFDM	Cyclic Prefix - OFDM
CS	Compressed Sensing
DCS-SOMP	Distributed Compressive Sensing SOMP
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
FDKD	Frequency Domain Kronecker Delta
FFT	Fast Fourier Transform
FSK	Frequency-Shift Keying
G-BPDN	Group BPDN
G-CoSaMP	Group CoSaMP
G-DCS-SOMP	Group DCS-SOMP
G-OMP	Group OMP
G-RIC	Group RIC
G-RIP	Group RIP
GSCS	Group Sparse Compressed Sensing
ICI	Intercarrier Interference
IDFT	Inverse Discrete Fourier Transform

Abbreviations

IFFT	Inverse Fast Fourier Transform
ISI	Intersymbol Interference
LASSO	Least Absolute Shrinkage and Selection Operator
LOS	Line of Sight
LS	Least Squares
LTI	Linear Time Invariant System
LTV	Linear Time-Varying System
M-BPDN	Multichannel BPDN
M-CoSaMP	Multichannel CoSaMP
MC	Multicarrier
MIMO	Multiple-Input Multiple-Output
MISO	Multiple-Input Single-Output
MMSE	Minimum Mean Square Error
MOD-CoSaMP	Modified CoSaMP
MOD-CS	Modified CS
MOD-G-CoSaMP	Modified G-CoSaMP
MOD-G-DCS-SOMP	Modified G-DCS-SOMP
MOD-OMP	Modified OMP
MPC	Multipath Component
MSE	Mean Square Error
OFDM	Orthogonal Frequency-Division Multiplexing
OMP	Orthogonal Matching Pursuit
pdf	probability density function
PSK	Phase-Shift Keying
QAM	Quadrature Amplitude Modulation
RIC	Restricted Isometry Constant
RIP	Restricted Isometry Property
SIMO	Single-Input Multiple-Output
SISO	Single-Input Single-Output
SNR	Signal-to-Noise Ratio
SOMP	Simultaneous OMP
WLAN	Wireless Local Area Network

Introduction

1.1 Overview

Over the last few decades wireless communications has become one of the fastest growing research fields in the engineering community. In many applications the estimation of the wireless channel connecting a transmitter with a receiver is indispensable. In order to do so, most modern systems employ training data to obtain estimates at some predefined time-frequency positions, and then use an interpolation technique, such as linear, quadratic or cubic spline interpolation, to gain information about the channel as a whole. In this thesis we study the application of the methodology of Compressed Sensing (CS) to that interpolation problem. CS is a rather young mathematical theory in which the concept of randomness is utilized to recover so-called sparse or compressible signals, i.e. signals that are only constituted of few essential contributions, from a very limited number of measurements.

The application of CS techniques to the problem of channel estimation was first proposed in 2008, and since then several similar channel estimators have been introduced, which we all subsume under the name of *compressive channel estimation*. They are based on the fact that in typical wireless transmission scenarios the signal takes many different paths from the transmitter to the receiver (for example due to reflections from large objects such as houses, trees, cars, mountains, etc.), but that often the number of such multipath components yielding significant contributions at the receiver is very limited. This fact yields that the description of the channel in the delay-Doppler re-

1 Introduction

gion exhibits some sparsity, which is mostly impaired by the so-called leakage effect due to the finite transmit bandwidth and blocklength of multicarrier transmission systems. In this thesis we investigate how this leakage affects the performance of compressive channel estimators. Moreover, we show how it can actually be utilized to improve the performance by using group-sparsity methods.

Furthermore, we study the adaption of compressive channel estimation methods to the setting where the transmitter and the receiver use more than one antenna. We show that the wireless channels connecting the various transmit and receive antennas in such a multiple-input multiple-output (MIMO) system exhibit certain similarities. In order to utilize this so-called joint sparsity the methodology of multichannel CS can be used, which leads to the multichannel compressive estimator that is demonstrated to outperform the conventional compressive channel estimator experimentally. In addition, we introduce a more general form of the multichannel compressive estimator by combining the ideas of exploiting group sparsity and joint sparsity for channel estimation. This estimator can take all the available sparsity-structure of the channel into account, and it can easily be adapted to fit most practical settings.

All the compressive estimators mentioned above are analyzed in terms of their performance and computational complexity, accompanied by extensive numerical simulations demonstrating their superior performance compared to standard estimation techniques. Furthermore, a basis optimization technique that considerably improves the performance of the conventional compressive channel estimator and that was also introduced in 2008, is adapted to incorporate the respective sparsity structure for the various methods.

Finally, we demonstrate that by utilizing the ideas and methodology of modified CS a simple adaption of the compressive channel estimators presented before can be used to track wireless channels, i.e. to keep some information about the channel at a given point in time in order to support its estimation at the next one. We show that the sparsity structure of the channel does not change very quickly in typical scenarios, even in fast varying environments, and that it therefore yields information appropriate for channel tracking.

1.2 Previous Work

The problem of channel estimation in wireless multicarrier systems has been studied comprehensively. As explained in Section 3.5 in more detail, many different techniques have been proposed, all of which can be classified as either pilot-aided schemes (where some training data is used to gain information about the channel [4, 5, 6, 7, 8]), blind schemes (where mostly statistical information about the channel is utilized [9, 10, 11, 12, 13]), or hybrid schemes, that are also denoted as semiblind schemes ([14, 15, 16]). In this thesis we only consider pilot-aided estimation schemes, and we restrict ourselves to the case of doubly selective channels (see Section 3.3.3 for details).

The use of CS methods for the estimation of channels using pilot symbols was first introduced in 2008 in [1] and [17, 18] independently, and since then has been investigated by several other authors. For example, [19, 20] studied the case of single-carrier signaling, and the results were mainly based on numerical simulations, without any CS theoretical background. The techniques presented in [19, 17, 21] are limited to sparsity in the delay domain only, i.e. they do not exploit Doppler sparsity. In contrast, the work in [18] considered a similar setting as we do in this thesis, merely using different CS techniques. Additionally, the authors of [18] were the first to consider the MIMO case [22, 23], where conventional CS methods are used in order to exploit the sparsity in the angular domain. Also, they only considered frequency selective channels.

Time-sequential estimation, i.e. tracking, of doubly selective channels has also been studied for some time now. In [24, 25], for example, subspace tracking methods are utilized, whereas the authors in [26, 27] make use of the so-called Kalman filter to track the channel. The application of CS, or more precisely modified CS, to the tracking problem, on the contrary, seems to be new.

1.3 Contributions

In this work we study the basic compressive channel estimator introduced in [1] in more detail, and we propose several variations and extensions. Most of this work has already been published in [28, 29, 30].

1 Introduction

The main contributions of this thesis can be summarized as follows.

- We analyze the performance of the conventional compressive channel estimator of [1] in terms of an upper bound on the mean square error of channel estimation in Theorem 5.1.1. In order to do so we use some basic linear algebra and results from the theory of CS.
- We give an estimate of the computational complexity of the compressive estimator in Section 5.1.2.
- We investigate the sparsity of the channel coefficients, or more precisely of the so-called leakage kernels, in Section 5.1.3 in more detail than it was done in [1, 3].
- We propose and analyze a variation of the compressive estimator that can take the group sparsity of the channel into account in Section 5.2.1. We also investigate its computational complexity (Section 5.2.2), as well as the group sparsity of the channel (Section 5.2.3), and we adapt the basis optimization technique from [2, 3] to this setting (Section 5.2.4). Most of this work has been published in [28].
- We propose and analyze an extension of the compressive estimator to the MIMO case in Section 6.1.1. In addition, its computational complexity (Section 6.1.2), as well as the joint sparsity of the various cross-channels (Section 6.1.3) is studied, and the adaption of the basis optimization technique is presented (Section 6.1.4). Most of this work has been published in [29].
- We propose and analyze the adaption of the multichannel compressive estimator that can also take group sparsity into account in Section 6.2.1. The analysis of its computational complexity, the joint group sparsity of the cross-channels as well as the adaption of the basis optimization technique is again presented (Section 6.2.2, Section 6.2.3 and Section 6.2.4, respectively).
- Moreover we propose a compressive channel tracker in Section 7.1.1. We analyze its computational complexity (Section 7.1.2), investigate the approximate sequential sparsity of the channel (Section 7.1.3), and finally discuss its extension to the MIMO case (Section 7.2). Most of this work has been published in [30].
- We present simulation results demonstrating the performance gain that can be achieved by the proposed compressive channel estimators (Section 5.1.5, Section 5.2.5, Section 6.1.5, Section 6.2.5 and Section 7.3).

Mathematical Preliminaries

In this chapter we introduce the notation and the basic mathematical tools that we use throughout this thesis. Although most of the definitions and results are well-known we still present them for the sake of completeness.

2.1 Vectors and Matrices

First of all note that throughout this thesis the imaginary unit is denoted by $j := \sqrt{-1}$, following the convention in the engineering literature, whereas the variable i is used as counting index. Furthermore, we denote the complex conjugate of a complex number x by x^* . We understand all the vectors as column vectors, and we denote them by bold lower case letters. Moreover, we start counting their indices at zero, and denote the $(n + 1)$ -th entry of a vector \mathbf{x} by $[\mathbf{x}]_n$, i.e. $\mathbf{x} = [[\mathbf{x}]_0, \dots, [\mathbf{x}]_{N-1}]^T$ for a vector $\mathbf{x} \in \mathbb{C}^N$, where $[\]^T$ denotes the transposed vector. Similarly, we denote matrices by bold upper case letters, and the entry in the $(m + 1)$ -th row and the $(n + 1)$ -th column of a matrix \mathbf{A} by $[\mathbf{A}]_{m,n}$. We also denote the Hermitian, i.e. the complex conjugated transpose of a vector \mathbf{x} or a matrix \mathbf{A} by \mathbf{x}^H or \mathbf{A}^H , respectively. The vector of length N containing only zeros is denoted by $\mathbf{0}_N$, and the $M \times N$ matrix containing only zeros by $\mathbf{0}_{M \times N}$. Moreover, the $N \times N$ identity matrix is denoted by \mathbf{I}_N . Note that we omit the indices whenever the size of the vectors or matrices is clear from the context.

2 Mathematical Preliminaries

Definition 2.1.1. For a vector $\mathbf{x} \in \mathbb{C}^N$ and some $p \geq 1$ we define the ℓ_p -norm of \mathbf{x} as

$$\|\mathbf{x}\|_p := \left(\sum_{n=0}^{N-1} |[\mathbf{x}]_n|^p \right)^{1/p}.$$

Here, $|[\mathbf{x}]_n|$ denotes the absolute value of $[\mathbf{x}]_n$. Note that throughout this thesis we also denote the number of elements contained in a set \mathcal{S} , i.e. its magnitude, by $|\mathcal{S}|$. Furthermore, we denote the *inner product* on \mathbb{C}^N by

$$\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{y}^H \mathbf{x} = \sum_{n=0}^{N-1} [\mathbf{x}]_n [\mathbf{y}]_n^*.$$

Definition 2.1.2. A matrix $\mathbf{U} \in \mathbb{C}^{N \times N}$ is called *unitary*, if for all $\mathbf{x}, \mathbf{y} \in \mathbb{C}^N$

$$\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle. \quad (2.1)$$

Note that condition (2.1) immediately yields $\|\mathbf{U}\mathbf{x}\|_2 = \|\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{C}^N$. Also, unitary matrices are invertible and satisfy $\mathbf{U}\mathbf{U}^{-1} = \mathbf{I}$, and therefore their columns can be interpreted as an orthonormal basis for \mathbb{C}^N .

Proposition 2.1.3. Each unitary matrix $\mathbf{U} \in \mathbb{C}^{N \times N}$ can be represented in terms of a Hermitian matrix \mathbf{V} (i.e., $\mathbf{V}^H = \mathbf{V}$) as $\mathbf{U} = e^{j\mathbf{V}}$.

Here, $e^{j\mathbf{V}}$ denotes the matrix exponential, i.e.

$$e^{j\mathbf{V}} = \sum_{k=0}^{\infty} \frac{j^k}{k!} \mathbf{V}^k,$$

which always converges. Therefore, it can be approximated by the partial sum

$$e^{j\mathbf{V}} \approx \mathbf{I}_N + j\mathbf{V}. \quad (2.2)$$

Obviously, this approximation can only be useful if \mathbf{V} does not contain quite large entries, i.e. if $\|\mathbf{V}\|_{\infty}$ is small, where

$$\|\mathbf{V}\|_{\infty} := \max_{n,k} |[\mathbf{V}]_{n,k}|.$$

These facts and further details can be found in [31].

Definition 2.1.4. For a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ and some $p, q \geq 1$ we define the ℓ_p/ℓ_q -norm of \mathbf{A} as

$$\|\mathbf{A}\|_{p,q} := \left(\sum_{m=0}^{M-1} \left(\sum_{n=0}^{N-1} |[\mathbf{A}]_{m,n}|^p \right)^{q/p} \right)^{1/q}.$$

Taking a closer look at this ℓ_p/ℓ_q -norm it can be interpreted as first taking the ℓ_p -norms of all the rows of \mathbf{A} , followed by taking the ℓ_q -norm of the vector of the ℓ_p -norms of the rows, i.e.

$$\|\mathbf{A}\|_{p,q} = \left(\sum_{m=0}^{M-1} \left(\|\mathbf{A}\|_p^{(m)} \right)^q \right)^{1/q}, \quad (2.3)$$

where $\|\mathbf{A}\|_p^{(m)}$ denotes the ℓ_p -norm of the m -th row of \mathbf{A} . In the special case where $p = q = 2$, this norm is often called *Frobenius norm*, which is denoted by

$$\|\mathbf{A}\|_F = \left(\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |[\mathbf{A}]_{m,n}|^2 \right)^{1/2}.$$

In the case of square matrices it is sub-multiplicative, i.e.

$$\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F. \quad (2.4)$$

Furthermore, it is compatible with the ℓ_2 -norm, i.e.

$$\|\mathbf{Ax}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{x}\|_2.$$

Definition 2.1.5. The support of a vector $\mathbf{x} \in \mathbb{C}^N$ is defined as the set of indices corresponding to the non-zero entries of \mathbf{x} , i.e.

$$\text{supp}\{\mathbf{x}\} := \{n \mid [\mathbf{x}]_n \neq 0\}.$$

We denote the number of non-zero entries of a vector by

$$\|\mathbf{x}\|_0 := |\text{supp}\{\mathbf{x}\}|,$$

although $\|\cdot\|_0$ actually is not a norm. Analogously, we define the *row support* of a matrix \mathbf{A} by

$$\text{supp}\{\mathbf{A}\} := \{m \mid \|\mathbf{A}\|_2^{(m)} > 0\},$$

i.e. the indices corresponding to the rows that are not identically zero, and we denote its magnitude by

$$\|\mathbf{A}\|_0 := |\text{supp}\{\mathbf{A}\}|.$$

At several points in this thesis it is necessary to stack an $M \times N$ matrix \mathbf{A} into a vector \mathbf{a} of length MN . This stacking is typically performed columnwise, i.e. $[\mathbf{a}]_{mM+n} = [\mathbf{A}]_{m,n}$, and we write $\mathbf{a} = \text{vec}\{\mathbf{A}\}_{m,n}$.

2.2 The Fourier Transform

We state the definition and very basic properties of the (continuous and discrete) Fourier transform here merely for the sake of clarity, since various, though equivalent, definitions can be found in the literature. All the results presented here can be found in [32].

Let $\mathcal{L}^1(\mathbb{R}^d) := \{f : \mathbb{R}^d \rightarrow \mathbb{C} \mid \int_{\mathbb{R}^d} |f(t)| dt < \infty\}$ be the set of all Lebesgue-integrable functions for some integer d .

Definition 2.2.1. The *Fourier transform* of a function $f \in \mathcal{L}^1(\mathbb{R}^d)$ is defined as

$$\mathcal{F}f(\xi) = \int_{\mathbb{R}^d} f(t) e^{-j2\pi\xi \cdot t} dt, \quad \xi \in \mathbb{R}^d.$$

The *inverse Fourier transform* of f is defined as

$$\mathcal{F}^{-1}f(t) = \int_{\mathbb{R}^d} f(\xi) e^{j2\pi t \cdot \xi} d\xi, \quad t \in \mathbb{R}^d.$$

Obviously, if both f and $\mathcal{F}f$ are in \mathcal{L}^1 , then $\mathcal{F}(\mathcal{F}^{-1}f) = f$. In this context t and ξ are often interpreted as representing time and frequency, respectively, and therefore the Fourier transform allows the transition from the description of a function in the time-domain to the description in the frequency domain, and vice versa.

Theorem 2.2.2. (*Convolution Theorem*). For $f, g \in \mathcal{L}^1(\mathbb{R}^d)$ we have

$$\mathcal{F}(f * g) = \mathcal{F}f \cdot \mathcal{F}g,$$

where $(f * g)(t) := \int_{\mathbb{R}^d} f(\tau)g(t - \tau)d\tau$ denotes the convolution of f and g .

For functions that depend on two variables the Fourier transform can be performed componentwise.

Definition 2.2.3. The *partial Fourier transform* of a function $F \in \mathcal{L}^1(\mathbb{R}^{2d})$ with respect to the second variable is defined as

$$\mathcal{F}_2 F(x, \xi) = \int_{\mathbb{R}^d} F(x, t) e^{-j2\pi\xi \cdot t} dt, \quad (x, \xi) \in \mathbb{R}^{2d}.$$

Similarly, \mathcal{F}_1 is defined as the partial Fourier transform with respect to the first variable.

In the discrete setting, the analog to the Fourier transform is the discrete Fourier transform (see [33] for more details).

Definition 2.2.4. For a vector $\mathbf{x} \in \mathbb{C}^N$ its *discrete Fourier transform* (DFT) $\mathbb{F}(\mathbf{x})$ is defined via

$$[\mathbb{F}(\mathbf{x})]_k := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} [\mathbf{x}]_n e^{-j2\pi \frac{nk}{N}}, \quad k = 0, \dots, N-1.$$

Proposition 2.2.5. *The DFT is unitary, i.e. $\|\mathbb{F}(\mathbf{x})\|_2 = \|\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{C}^N$.*

Therefore, the DFT is invertible, and the *inverse DFT* (IDFT) $\mathbb{F}^{-1}(\mathbf{x})$ of \mathbf{x} can be calculated via

$$[\mathbb{F}^{-1}(\mathbf{x})]_k := \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} [\mathbf{x}]_n e^{j2\pi \frac{nk}{N}}, \quad k = 0, \dots, N-1. \quad (2.5)$$

The main reason for the frequent use of the DFT in various applications probably is the fact that it can be implemented extremely fast by using the *fast Fourier transform* (FFT) [34].

2.3 Some Special Functions

Here we introduce some special functions that appear several times throughout this thesis. Note that we always use brackets in the discrete case and parentheses in the continuous setting.

The *Kronecker-delta* is defined as

$$\delta[n] := \begin{cases} 1 & n = 0 \\ 0 & \text{else} \end{cases}. \quad (2.6)$$

The continuous analog is the so-called *Dirac-delta*, which is defined as

$$\delta(t) := \begin{cases} 1 & t = 0 \\ 0 & \text{else} \end{cases}, \quad (2.7)$$

where the integral over the real line is 1. Thus, the Dirac-delta is not strictly a function, but is defined as a distribution (see [32] for more details). In this distributional sense we then have

$$\int_{\mathbb{R}} f(t) \delta(t - t_0) dt = f(t_0). \quad (2.8)$$

Furthermore, it follows that

$$\mathcal{F}(e^{j2\pi\xi_1 t})(\xi_2) = \delta(\xi_1 - \xi_2). \quad (2.9)$$

Another function that arises naturally in the description of wireless communication systems is the so-called *sinc-function*, which is defined as

$$\text{sinc}(x) = \begin{cases} \frac{\sin(\pi x)}{\pi x} & x \neq 0 \\ 1 & x = 0 \end{cases} . \quad (2.10)$$

Note that we use the normalized definition of the sinc-function. In the literature it is often referred to as the *ideal filter*. This is due to the fact that its Fourier transform is the rect-function, i.e. the function that is 1 between $-1/2$ and $1/2$ and zero everywhere else (to be precise, it takes the value $1/2$ at $-1/2$ and $1/2$). Thus, keeping the Convolution Theorem 2.2.2 in mind, by convolving a signal with an appropriately normalized sinc-function corresponds to cutting off all frequencies outside a symmetric region about the origin. Furthermore, the following property can be very useful (see [35] for details).

Proposition 2.3.1. *For any smooth function f with compact support we have*

$$\lim_{a \rightarrow 0} \int_{\mathbb{R}} \frac{1}{a} \text{sinc}(x/a) f(x) dx = f(0) . \quad (2.11)$$

2.4 Probability Theory

In this section we present some useful facts from the theory of probability, which are described in more detail in [36], for example. First, we only mention the following two probability distributions for the discrete case, where \mathbb{P} denotes the probability.

- A random variable X that only takes values in a discrete set Ω of size $|\Omega| = N$ is *uniformly distributed*, if each value has the same probability of being observed, i.e.

$$\mathbb{P}(X = \omega) = \frac{1}{N} \quad \text{for all } \omega \in \Omega .$$

- A discrete random variable X that can only take the value A or B is *Bernoulli distributed* for some parameter $0 < p < 1$, if

$$\mathbb{P}(X = A) = p \quad \text{and} \quad \mathbb{P}(X = B) = 1 - p .$$

For the continuous case we first give some basic definitions.

Definition 2.4.1. A random variable X has *probability density function* (pdf) $\varphi(x)$, if the probability that X takes values between some constants $a < b$ is given as

$$\mathbb{P}(a \leq X \leq b) = \int_a^b \varphi(x) dx .$$

Definition 2.4.2. The *expectation* of a random variable X is defined as

$$\mathbb{E}\{X\} := \int_{\mathbb{R}} x\varphi(x)dx,$$

whereas its *variance* is given by

$$\sigma_X^2 := \mathbb{E}\{(X - \mathbb{E}\{X\})^2\}.$$

Note that the expectation is linear, i.e.

$$\mathbb{E}\left\{\sum_{n=0}^{N-1} c_n X_n\right\} = \sum_{n=0}^{N-1} c_n \mathbb{E}\{X_n\} \quad (2.12)$$

for some random variables X_n and constants $c_n \in \mathbb{R}$.

Definition 2.4.3. An ensemble of random variables $\{X_n\}_{n=0}^{N-1}$ has *joint probability density function* $\varphi(x_0, \dots, x_{N-1})$, if the probability that each X_n takes a value between some constants $a_n < b_n$ is given as

$$\mathbb{P}(a_n \leq X_n \leq b_n, n = 0, \dots, N-1) = \int_{a_0}^{b_0} \cdots \int_{a_{N-1}}^{b_{N-1}} \varphi(x_0, \dots, x_{N-1}) dx_0 \dots dx_{N-1}.$$

Furthermore, the *conditional pdf* for $\{X_n\}_{n=0}^{N-1}$, given the values $\{y_m\}_{m=0}^{M-1}$ of $\{Y_m\}_{m=0}^{M-1}$, is defined as

$$\varphi_{\mathbf{X}}(\mathbf{x} \mid \mathbf{Y} = \mathbf{y}) := \frac{\varphi_{\mathbf{X}, \mathbf{Y}}(\mathbf{x}, \mathbf{y})}{\varphi_{\mathbf{Y}}(\mathbf{y})},$$

where $\mathbf{X} := [X_0, \dots, X_{N-1}]$, $\mathbf{Y} := [Y_0, \dots, Y_{M-1}]$, $\mathbf{x} := [x_0, \dots, x_{N-1}]$, $\mathbf{y} := [y_0, \dots, y_{M-1}]$. Note that here (and only here) we used bold *upper* case letters for the *vectors* \mathbf{X} and \mathbf{Y} .

Next, we mention the following two continuous probability distributions.

- A continuous random variable X is *uniformly distributed* in some interval $[a; b]$, if each subinterval of equal length has equal probability, i.e.

$$\varphi_X(x) = \begin{cases} \frac{1}{b-a} & \text{for } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases}.$$

- A random variable X is *Gaussian (normal) distributed* with mean μ and variance σ^2 , if

$$\varphi_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

Sometimes it is difficult to compute the exact expectation of a random variable X , or - more generally - of a transformed variable $Y = f(X)$ for some transformation f . In this case it can be approximated by

$$\mathbb{E}\{f(X)\} \approx \frac{1}{K} \sum_{k=0}^{K-1} f(x_k), \quad (2.13)$$

where the x_k are K samples of the random variable X . Obviously, the approximation improves the larger K becomes. This approach is called *Monte Carlo approximation*, and further details about it can be found in [37].

2.5 Landau Notation

Throughout this thesis we measure the computational complexity of an algorithm by the number of (multiplicative) operations that have to be performed. In order to do so we use the so-called *big-O notation* (also known as *Landau notation*).

Definition 2.5.1. Let $f, g : \mathbb{R} \rightarrow \mathbb{R}$. Then one refers to f as being *big-O of g* , written

$$f \in \mathcal{O}(g),$$

if there is a positive constant $C > 0$ and some $x_0 \in \mathbb{R}$, such that

$$|f(x)| \leq C|g(x)| \quad \text{for all } x > x_0.$$

For example, if an algorithm with input of size n can solve a problem using $an^2 + bn + c$ operations we refer to it as having complexity $\mathcal{O}(n^2)$, since with increasing n the term n^2 will dominate the terms of smaller order as well as the coefficient a by far.

Wireless Communications

3.1 Introduction

From the very first concepts, such as jungle drums or smoke signs, to the groundbreaking work of Maxwell and Hertz firstly fully describing the transmission of electromagnetic waves in the late 19th century, from the first wireless transmissions actually utilizing these waves by Tesla and Marconi at the turn of that century, over the first two-way communication systems developed for the military in the first half of the 20th century, to satellite television broadcasting, cell phones and wireless local area networks (WLANs), wireless communications has been one of the fastest growing areas in engineering, especially over the last decades, where its importance in every day life made it an important research field that seems never to be exhausted. In this chapter we first introduce the basic ideas and important concepts of wireless communications in Section 3.2. In Section 3.3 we describe the physical phenomena occurring during a wireless transmission, and introduce mathematical models describing them. Furthermore, we review an equivalent discrete system model in Section 3.4, and introduce the idea of channel estimation in Section 3.5. Finally, we describe the basic concepts of multi-antenna systems in Section 3.6 For all the details, see [38].

3.2 Transceiver Setup

The basic task in wireless communications is to transmit a message from a transmitter to a receiver. In modern communication systems the message is encoded into a transmit

signal, which then is transmitted from one or several antennas via electromagnetic waves. At the receiver, one or several antennas receive a distorted version of the transmit signal, from which the receiver tries to decode the original message.

Nowadays, the information typically is digitized (if it is analog) and compressed first to be able to apply mathematical tools such as error-correcting codes to make the data link more reliable, which is referred to as *source coding*. Then, the information bits are encoded using either convolutional codes or block codes [39], which is called *channel coding*. The *modulator* then maps the encoded bits to the *transmit symbols*, which are taken from a finite, typically complex alphabet \mathcal{A} . Commonly used schemes are *phase-shift keying* (PSK), *amplitude-shift keying* (ASK), and *frequency-shift keying* (FSK), where phase, amplitude or frequency of a base signal (often a sinusoid) are altered to represent the data symbol, respectively, or *quadrature amplitude modulation* (QAM), which can be viewed as a combination of PSK and ASK.

In the following we describe a *multicarrier* (MC) modulation scheme, which is typically used in broadband wireless communication systems because of its advantages over singlecarrier modulation schemes [40, 41]. In this setting, the frequency band of width B_0 available for transmission is divided into K subbands (*subcarriers*) of width $\frac{B_0}{K}$ each. The discrete-time transmit signal can then be written as

$$s[n] = \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} a_{l,k} g_{l,k}[n], \quad (3.1)$$

where $a_{l,k} \in \mathcal{A}$ denotes the l^{th} symbol transmitted at subcarrier k , $g_{l,k}[n] = e^{j2\pi k(n-lN)/K} g[n-lN]$ is the time-frequency shift of a discrete transmit pulse $g[n]$, with $N \geq K$ denoting the (discrete) symbol duration. Recall that in this thesis we use brackets in the discrete case and parentheses in the continuous case. Afterwards, $s[n]$ is converted to the continuous-time transmit signal

$$s(t) = \sum_{n \in \mathbb{Z}} s[n] f_1(t - nT_s) \quad (3.2)$$

using an interpolation filter with impulse response $f_1(t)$ and sampling period $T_s := 1/B_0$. This *baseband* signal, i.e. the bandlimited signal with frequencies starting close to zero, is then upconverted to the *passband* signal in the desired frequency band, i.e. $s_p(t) = e^{j2\pi f_c t} s(t)$, where f_c denotes the center frequency, which then is finally transmitted.

At the receiver, the received signal $r_p(t)$ is downconverted to the baseband signal $r(t)$

again, i.e. $r(t) = e^{-j2\pi f_c t} r_p(t)$, and transformed into the discrete-time signal

$$r[n] = \int_{\mathbb{R}} r(t) f_2(nT_s - t) dt \quad (3.3)$$

using an anti-aliasing filter with impulse response $f_2(t)$. Subsequently, the MC *demodulator* calculates the demodulated symbols

$$r_{l,k} = \sum_{n \in \mathbb{Z}} r[n] \gamma_{l,k}^*[n] \quad (3.4)$$

for every $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$, where $\gamma_{l,k}[n] = e^{j2\pi k(n-lN)/K} \gamma[n-lN]$ is the time-frequency shift of a receive pulse $\gamma[n]$. The design of the transmit and receive pulses is very important because they need to prevent subsequently transmitted symbols from interfering (i.e. intersymbol interference (ISI), which is discussed in Section 3.3.1). The demodulated symbols are then equalized, i.e. the effect of the channel is reverted, and quantized according to the symbol alphabet \mathcal{A} . Finally, the symbols are mapped back to the corresponding bits, which are then decoded and decompressed.

In practical systems it is of great importance that modulator and demodulator allow computationally extremely efficient implementations. A very popular and widely used MC modulation scheme is *orthogonal frequency-division multiplexing* (OFDM), or more precisely *cyclic prefix* (CP) OFDM [42, 43]. Here, transmit and receive pulses are both chosen rectangular as

$$g_{\text{cp}}[n] := \begin{cases} \frac{1}{\sqrt{K}} & n \in \{-L_{\text{cp}}, \dots, K-1\} \\ 0 & \text{else,} \end{cases} \quad \text{and} \quad \gamma_{\text{cp}}[n] := \begin{cases} \frac{1}{\sqrt{K}} & n \in \{0, \dots, K-1\} \\ 0 & \text{else,} \end{cases} \quad (3.5)$$

where $L_{\text{cp}} := N - K$ denotes the length of the guard interval that is used to avoid ISI. Then, the discrete-time transmit signal (3.1) can be written as $s[n] = \sum_{l=0}^{L-1} s_l[n-lN]$, with

$$s_l[n] = \begin{cases} \frac{1}{\sqrt{K}} \sum_{k=0}^{K-1} a_{l,k} e^{j2\pi \frac{kn}{K}} & n \in \{-L_{\text{cp}}, \dots, K-1\} \\ 0 & \text{else.} \end{cases}$$

Defining $\mathbf{a}_l := [a_{l,0}, \dots, a_{l,K-1}]^T$ for $l = 0, \dots, L-1$, we thus have $s_l[n] = [\mathbb{F}^{-1}(\mathbf{a}_l)]_{(n)_K}$ for $n \in \{-L_{\text{cp}}, \dots, K-1\}$, where \mathbb{F}^{-1} denotes the IDFT (see (2.5)), and $(n)_K := n \pmod{K}$ denotes the modulo K operation. Here, the guard interval at the beginning of s_l is filled with the last L_{cp} symbols of $\mathbb{F}^{-1}(\mathbf{a}_l)$ as cyclic prefix, an explanation for which can be found for example in [43]. Therefore, the modulator can be implemented using the FFT,

or more precisely the inverse FFT (IFFT), which has very low computational complexity (see Section 2.2 and [34]). At the receiver, the cyclic prefix is removed since $\gamma[n] = 0$ in the guard interval $n \in \{-L_{\text{cp}}, \dots, -1\}$, and the demodulated symbols (3.4) can then be written as

$$r_{l,k} = \frac{1}{\sqrt{K}} \sum_{m=0}^{K-1} r[m + lN] e^{-j2\pi \frac{mk}{K}} = \mathbb{F}(\mathbf{r}_l),$$

where $\mathbf{r}_l = [r[lN], \dots, r[K-1 + lN]]^T$ and \mathbb{F} denotes the DFT (see Definition 2.2.4), which can be implemented using the FFT. Since the IDFT and DFT in modulator and demodulator, respectively, are of length K , the number of subcarriers is typically chosen as a power of 2 to allow the fastest possible implementation.

As an alternative to the cyclic prefix, zero padding at the end of s_l can also be used. For more details see for example [44, 45].

3.3 The Wireless Channel

Ideally, the wireless channel leaves the transmitted signal identical so that the received signal equals the transmit signal, i.e. $r(t) = s(t')$, where $t' = t - \tau_0$ describes the time shift corresponding to the time τ_0 it takes for the signal to reach the receiver from the transmitter. Assuming perfect alignment of transmitter and receiver (i.e. assuming τ_0 to be known) the demodulated symbols $r_{l,k}$ equal the transmitted symbols $a_{l,k}$ if and only if the receive devices do not introduce noise and if the transmit pulse g and the receive pulse γ satisfy the biorthogonality condition

$$\langle g_{l,k}, \gamma_{l',k'} \rangle = \sum_{n \in \mathbb{Z}} g_{l,k}[n] \gamma_{l',k'}^*[n] = \delta[l - l'] \delta[k - k'],$$

where $\delta[l]$ is the Kronecker-delta (see (2.6)). Of course, an ideal channel never occurs in practice and in the following we describe the most important effects influencing the transmission.

3.3.1 Multipath Propagation

There are many different phenomena affecting the transmit signal on its way through the radio channel. First of all, since electromagnetic waves weaken as they propagate,

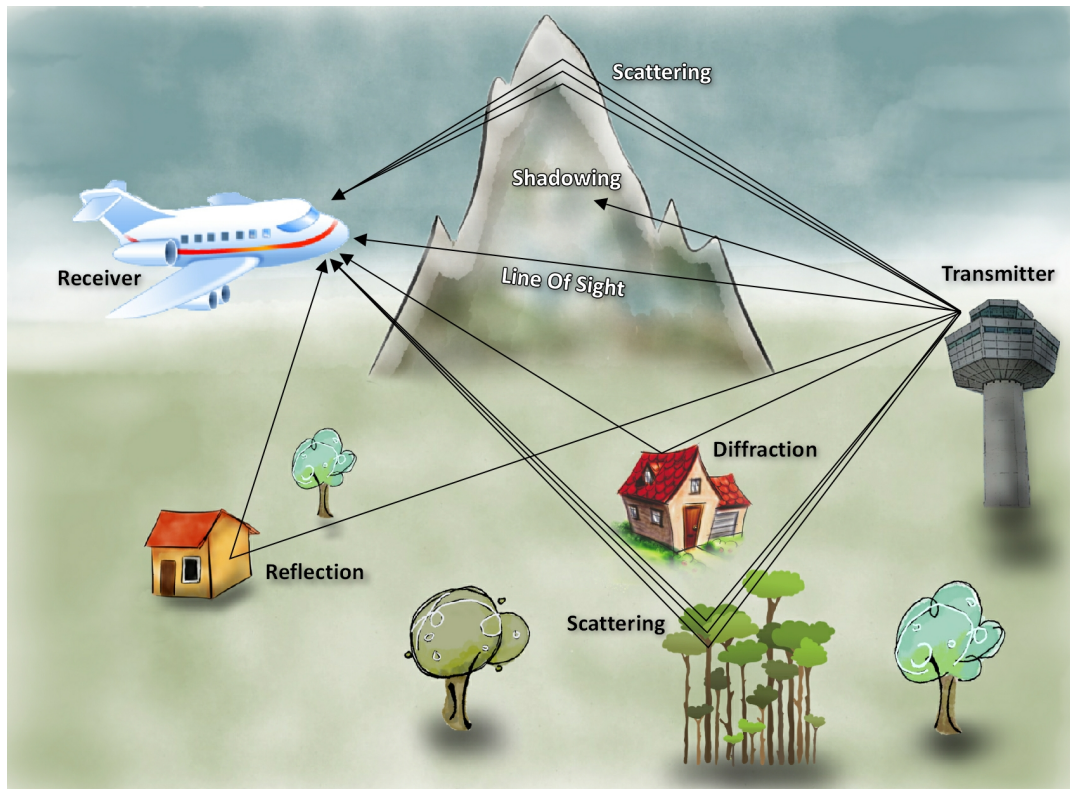


Figure 3.1: Illustration of multipath propagation [46]

the received signal strength decreases as the distance between transmitter and receiver grows, which is known as *free space path loss*. Furthermore, in addition to the direct *Line of Sight* (LOS) connection between transmitter and receiver, which might be blocked by walls, buildings, etc., the spherical propagation of electromagnetic waves yields that the signal also takes several other paths. This *multipath propagation* is due to the fact that copies of the signal typically are reflected by large, smooth surfaces (e.g. walls of a building), scattered by smaller, rougher surfaces (e.g. leaves of a tree), diffracted on the edges of some objects (e.g. houses, cars, mountains) or refracted (e.g. by the ionosphere). An example is depicted in Fig. 3.1. Therefore, the receiver sees a superposition of several *multipath components* (MPCs) with distinct amplitude, phase shift and time delay, leading to constructive or destructive interference at the receiver. This, in turn, causes the signal power seen at the receiver to fluctuate over time, frequency and/or space, which is known as *fading*. More specifically, fading caused by the interference of different MPCs is called *small-scale fading*, since the power fluctuations occur on a scale comparable to one wavelength, i.e. even changes as small as one wavelength in

the spatial geometry can cause huge changes in the receive signal power. This effect is known to all users of cell phones, probably, since sometimes taking one step in any direction can yield a dramatic change of the signal quality. *Large-scale fading*, on the other hand, is caused by *shadowing*, i.e. huge objects like buildings, mountains, etc., blocking whole MPCs. These effects, although completely different by nature, overlap and can yield a received signal power looking like the one depicted in Fig. 3.2, where the fast power fluctuations indicated by the blue, solid line can be explained by small-scale fading effects, whereas the trend shown by the averaged power level, visualized by the red, dashed line, is caused by large-scale fading.

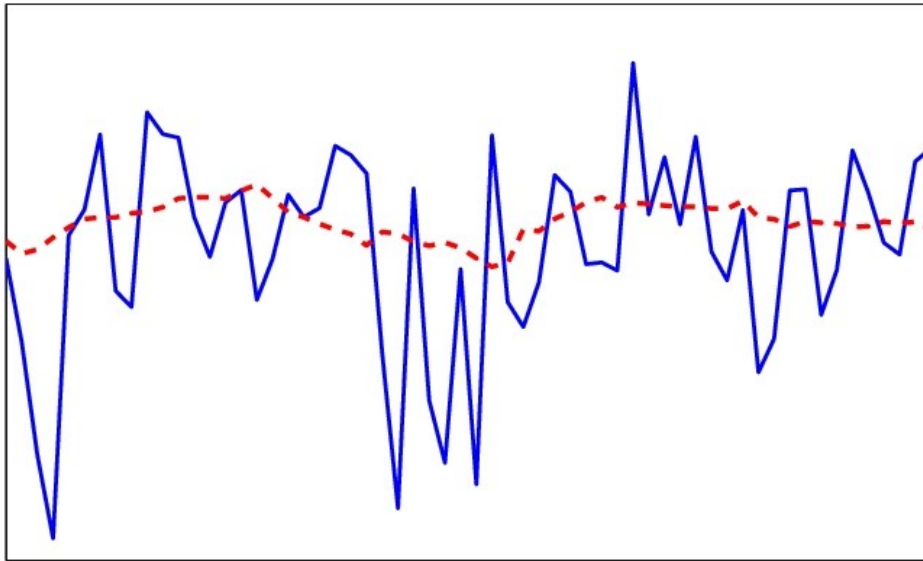


Figure 3.2: Typical fading characteristics. The blue, solid line shows the received signal power at each sampling point, whereas as the red, dashed line shows the average over 15 sampling points.

Another consequence of multipath propagation is signal dispersion, i.e. the fact that due to the different path lengths of the different MPCs several copies of the signal arrive at the receiver at different times. This can lead to *intersymbol interference* (ISI): Consecutive symbols interfere at the receiver because MPCs with long runtime carrying information from one symbol arrive at the receiver at the same time MPCs with shorter runtime carrying information from the next symbol do. ISI can therefore be described by the ratio of the symbol duration and the maximal time delay of an MPC occurring. Note that in principle infinitely many MPCs arrive at the receiver with arbitrarily large

time delay, but that all but a few will have lost most of their energy due to very long distances and/or absorption along the way. Therefore, it is reasonable to speak of a *maximal* time delay among those MPCs giving a *significant contribution* to the receive signal. By *significant* we mean that it can still be distinguished from the noise. The amount of noise present in a transmission system can be measured by means of the *signal-to-noise ratio* (SNR), which is defined as

$$\text{SNR} = 10 \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right),$$

where P_{signal} and P_{noise} denote the average power of the noise-free signal at the receive side and of the noise, respectively. Obviously a large SNR is desirable, since in that case the signal can be distinguished from the noise much better, and thus the reliability of the transmission link is increased.

3.3.2 Time-Invariant Channels

If transmitter, receiver and all the objects reflecting, diffracting, scattering or refracting the signal (from here on subsumed as *scatterers*) are static, the channel can be modeled as a *linear time-invariant system* (LTI), i.e.

$$r(t) = \int_{\mathbb{R}} h(\tau) s(t - \tau) d\tau + z(t), \quad (3.6)$$

where s is the time-domain transmit signal, r is the time-domain received signal, and h is the response of the channel to a unit pulse in terms of the delay-variable τ , and thus is called the *impulse response* of the channel. Here, z subsumes all the unaccountable noise such as spurious emissions from other transmitting devices, thermal noise, or noise introduced by amplifiers or mixers at the receiver.

In a typical multipath scenario, the impulse response can be modeled as

$$h(\tau) = \sum_{p=1}^P \eta_p \delta(\tau - \tau_p),$$

where P is the number of MPCs giving a "significant" contribution to the received signal, η_p are the attenuation coefficients, τ_p are the time delays of the MPCs proportional to their path lengths, and δ denotes the Dirac-delta (2.7). Taking the Fourier transform of the impulse response yields an equivalent representation of the channel in terms of

the frequency-varying *transfer function* $H(f) = (\mathcal{F}h)(f)$, and the input-output relation (3.6) can then be written as

$$r(t) = \int_{\mathbb{R}} H(f)(\mathcal{F}s)(f)e^{j2\pi ft} df$$

for a baseband signal s . Here, another effect of multipath propagation can be seen, namely that different frequency components of the transmit signal are attenuated differently, which is also known as *frequency selective fading*. Accordingly, a channel experiencing multipath propagation is called *frequency selective*.

Time-invariant channels occur, for example, in WLANs, where typically there is (almost) no mobility between transmitter and receiver.

3.3.3 Time-Varying Channels

In many other scenarios, transmitter, receiver and/or scatterers are moving, and therefore the wireless channel typically varies over time and has to be modeled as a *linear time-varying system* (LTV). The impulse response therefore additionally depends on the time, and we have

$$r(t) = \int_{\mathbb{R}} h(t, \tau)s(t - \tau)d\tau + z(t). \quad (3.7)$$

If the impulse response does not change too quickly, we call the channel *slowly time-varying* or *quasi-static*, and we can interpret $h(t, \tau)$ as the impulse response of an LTI system that is "valid" at time t . To model such an impulse response, we also have to take into account the fact that since transmitter, receiver and/or scatterers are moving, the electromagnetic waves will experience a frequency shift due to the Doppler effect. Therefore, we have

$$h(t, \tau) = \sum_{p=1}^{P(t)} \eta_p(t)\delta(\tau - \tau_p(t))e^{j2\pi\nu_p(t)t}, \quad (3.8)$$

where $\nu_p(t)$ denotes the Doppler-frequency shift, and all the parameters $P(t)$, $\eta_p(t)$, $\tau_p(t)$, and $\nu_p(t)$ will change over time. A frequency selective, time-varying channel is called *doubly selective*. Multicarrier systems applied to doubly selective channels not only suffer from ISI due to multipath propagation, but also from *intercarrier interference* (ICI): if the Doppler frequency shift of an MPC becomes as large as the frequency separation between two adjacent subcarriers, symbols of those two subcarriers arriving at the same time will interfere.

Taking the partial Fourier transform of the impulse response with respect to τ again yields the time- and frequency-varying transfer function $H(t, f) = (\mathcal{F}_2 h)(t, f)$. The input-output relation (3.7) then reads

$$r(t) = \int_{\mathbb{R}} H(t, f) (\mathcal{F}s)(f) e^{j2\pi ft} df + z(t),$$

and $H(t, f)$ can be interpreted as the transfer function "valid" at time t . Alternatively, we can take the partial Fourier transform with respect to t , which gives the *spreading function* $S(\nu, \tau) = (\mathcal{F}_1 h)(\nu, \tau)$ (throughout the rest of this thesis we will interchange the order of ν and τ , i.e. write $S(\tau, \nu)$, to be conform with the literature). By rewriting (3.7) as

$$r(t) = \int_{\mathbb{R}} \int_{\mathbb{R}} S(\tau, \nu) s(t - \tau) e^{j2\pi\nu t} d\nu d\tau + z(t), \quad (3.9)$$

it is easily interpreted as the factor by which an instance of the transmit signal s at delay τ and Doppler shift ν contributes to the received signal r .

As explained in Section 3.3.1, wireless channels do not experience infinite time delays and Doppler shifts in practice, so the integrals in (3.9) only have to be taken from 0 to τ_{\max} and from $-\nu_{\max}$ to ν_{\max} , respectively, where τ_{\max} and ν_{\max} denote the maximal time delay and Doppler frequency shift occurring, respectively. Their product $\tau_{\max}\nu_{\max}$ obviously is an upper bound for the size of the (effective) support of the spreading function, and thus measures how much the channel spreads the signal in the delay-Doppler region. Both τ_{\max} and ν_{\max} are inversely proportional to the propagation speed c (see Section 6.1.3 for the exact formulas), and therefore $\tau_{\max}\nu_{\max}$ behaves like $1/c^2$. Since in typical over-the-air transmissions the speed of light is $c \approx 3 \cdot 10^8$, we will always have $\tau_{\max}\nu_{\max} \ll 1$. Such channels are called *underspread* [47, 48]. Throughout the rest of this thesis we only consider underspread doubly selective channels.

3.4 Equivalent Multicarrier System Model

By combining (3.2), (3.3) and the input-output relation (3.7) of a doubly selective channel, we obtain a discrete-time channel relating the discrete-time transmit signal $s[n]$ to the discrete-time receive signal $r[n]$ as

$$r[n] = \sum_{m \in \mathbb{Z}} h[n, m] s[n - m] + z[n], \quad n \in \mathbb{Z}, \quad (3.10)$$

where

$$h[n, m] = \int_{\mathbb{R}} \int_{\mathbb{R}} h(t + nT_s, \tau) f_1(t - \tau + mT_s) f_2(-t) dt d\tau \quad (3.11)$$

is a discrete-time equivalent of the impulse response of the channel, and $z[n] = \int_{\mathbb{R}} z(t) f_2(nT_s - t) dt$ is some discrete-time noise. Inserting this relation into (3.4) and using (3.1) yields

$$r_{l,k} = \sum_{l'=0}^{L-1} \sum_{k'=0}^{K-1} H_{l,k;l',k'} a_{l',k'} + z_{l,k} \quad (3.12)$$

for all $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$, with equivalent noise terms $z_{l,k} = \sum_{n \in \mathbb{Z}} z[n] \gamma_{l,k}^*[n]$ and *system channel coefficients* $H_{l,k;l',k'} = \sum_{n \in \mathbb{Z}} \sum_{m \in \mathbb{Z}} h[n, m] g_{l',k'}[n - m] \gamma_{l,k}^*[n]$, which subsume the effects of modulator, interpolation filter, physical channel, anti-aliasing filter and demodulator. The coefficient $H_{l,k;l',k'}$ can easily be interpreted as describing ISI for $l \neq l'$ and ICI for $k \neq k'$. The input-output relation (3.12) can then be rewritten as

$$r_{l,k} = H_{l,k} a_{l,k} + \tilde{z}_{l,k}, \quad (3.13)$$

where $\tilde{z}_{l,k} := z_{l,k} + \sum_{l' \neq l} \sum_{k' \neq k} H_{l,k;l',k'} a_{l',k'}$ now subsumes the noise and the interference terms. If the amount of ISI and ICI introduced by the channel is not too large, i.e. if the channel dispersion is not too strong, the interference coefficients $H_{l,k;l',k'}$ for $l \neq l'$ and $k \neq k'$ will be very small, and therefore the $\tilde{z}_{l,k}$ will not differ too much from the $z_{l,k}$. Note that ISI and ICI can also be combated by the design of the transmission setup. To be precise, ISI can be reduced by choosing the discrete-time transmit pulse $g[n]$ such that a larger guard interval is given in between two consecutive OFDM symbols, whereas ICI can be reduced by choosing the subcarrier spacing large enough. Obviously, these parameters cannot be chosen arbitrarily large due to the need for high data rates on the one hand, and the limited bandwidth on the other hand.

In the following we assume $\gamma[n]$ to be zero outside $\{0, \dots, L_\gamma - 1\}$, since typically the receive pulse γ is compactly supported. Thus, to compute $r_{l,k}$ in (3.4) for all $l = 0, \dots, L-1$, the discrete-time received signal $r[n]$ has to be known for $n = 0, \dots, N_0 - 1$, where $N_0 := (L-1)N + L_\gamma + 1$. Taking the (non-unitary) discrete Fourier transform of appropriate length of the impulse response $h[n, m]$ with respect to the first variable n , we obtain the *discrete-delay-Doppler spreading function*

$$S_h[m, i] = \frac{1}{N_0} \sum_{n=0}^{N_0-1} h[n, m] e^{-j2\pi \frac{in}{N_0}}, \quad m, i \in \mathbb{Z}, \quad (3.14)$$

which now describes the channel by means of a discrete delay (or time shift) m and a discrete Doppler frequency shift i (here, again, we interchange the order of m and i to be conform with the literature). Therefore, we can rewrite (3.10) as

$$r[n] = \sum_{m \in \mathbb{Z}} \sum_{i=0}^{N_0-1} S_h[m, i] s[n-m] e^{j2\pi \frac{ni}{N_0}} + z[n]. \quad (3.15)$$

Combining (3.4), (3.15) and (3.1), we reobtain (3.12), where the system channel coefficients $H_{l,k;l',k'}$ can be written as

$$H_{l,k;l',k'} = \sum_{m \in \mathbb{Z}} \sum_{i=0}^{N_0-1} S_h[m, i] \sum_{n \in \mathbb{Z}} \gamma_{l,k}^*[n] g_{l',k'}[n-m] e^{j2\pi \frac{in}{N_0}}.$$

Next, we assume that the channel is causal with maximum delay $\tau_{\max} \leq (K-1)T_s$, i.e. $h[n, m] = 0$ for all $m \notin \{0, \dots, K-1\}$. Then, specializing the equation above to the case $l' = l$ and $k' = k$, we can write the *diagonal* channel coefficients $H_{l,k} := H_{l,k;l,k}$ as

$$H_{l,k} = \sum_{m=0}^{K-1} \sum_{i=0}^{N_0-1} S_h[m, i] A_{\gamma,g}^* \left(m, \frac{i}{N_0} \right) e^{-j2\pi \left(\frac{km}{K} - \frac{li}{L} \right)}$$

with the *cross-ambiguity function* $A_{\gamma,g}(m, \xi) := \sum_{n \in \mathbb{Z}} \gamma[n] g^*[n-m] e^{-j2\pi \xi n}$ of $\gamma[n]$ and $g[n]$ [49]. Using the approximation $N_0 \approx NL$, which actually is exact for CP-OFDM, we finally obtain the 2D-DFT expression

$$H_{l,k} = \sum_{m=0}^{K-1} \sum_{i=-L/2}^{L/2-1} F_{m,i} e^{-j2\pi \left(\frac{km}{K} - \frac{li}{L} \right)} \quad (3.16)$$

for all $l = 0, \dots, L-1$ (L is assumed even for mathematical convenience) and $k = 0, \dots, K-1$, where the Fourier coefficients $F_{m,i}$ can be written as

$$F_{m,i} = \sum_{q=0}^{N-1} S_h[m, i+qL] A_{\gamma,g}^* \left(m, \frac{i+qL}{N_0} \right). \quad (3.17)$$

3.5 Channel Estimation

The problem of reobtaining the transmit symbols from the demodulated symbols at the receiver, which are disturbed by the wireless channel as explained in the previous sections, is called *equalization*. The most natural approach to channel equalization, which

is widely used in practice, is to estimate its effects and revert them. This gives rise to the formulation of the important subproblem of *channel estimation*, for which many different techniques have been proposed. The goal is to estimate any of the representations of the channel (impulse response, transfer function, spreading function), or equivalently the system channel coefficients $H_{l,k;l',k'}$ in (3.12). One approach to do so is to use *training data*, i.e. symbols that are not used to transmit actual data but that are fixed before data transmission and known to both transmitter and receiver [4, 5, 6, 7, 8]. If no such training data is used, we speak about *blind* channel estimation [9, 10, 11, 12, 13]. Here, typically some statistical information about the channel or the transmit signal is used. There are also hybrid estimation schemes, which are subsumed under the name of *semiblind* channel estimation [14, 15, 16]. After an estimate of the channel has been calculated, its effect is reverted in order to obtain estimates of the transmit symbols.

In this thesis, we only consider *pilot-aided* estimation schemes, where *pilot symbols* $p_{l,k} = a_{l,k}$, which are also known to the receiver, are transmitted at the time-frequency positions $(l, k) \in \mathcal{P}$ in the *pilot set* \mathcal{P} . The choice of this set is crucial. For slowly varying channels, i.e. a channel that hardly changes during a block of several OFDM symbols, typically a *block-type* pilot arrangement like the one depicted in Fig. 3.3(a) is used. Here, pilots are transmitted at *all* subcarriers at some equispaced OFDM symbols, i.e. $\mathcal{P} = \{(l_0 + ml_B, k) \mid m = 0, \dots, L/l_B - 1; k = 0, \dots, K - 1\}$ for some $l_0 \geq 0$ and some block number l_B dividing the total number L of OFDM blocks. From the pilots, the channel for the considered block of OFDM symbols can then be estimated using for example the *least squares* (LS) approach. Since the pilot symbols $a_{l,k} = p_{l,k}$ transmitted at the pilot positions $(l, k) \in \mathcal{P}$ are known to the receiver, estimates of the channel coefficients $H_{l,k}$ can easily be calculated by simply dividing the received symbol by the pilot symbol, i.e.

$$\hat{H}_{l,k} = \frac{r_{l,k}}{p_{l,k}} = H_{l,k} + \frac{\tilde{z}_{l,k}}{p_{l,k}}, \quad (l, k) \in \mathcal{P}, \quad (3.18)$$

using the input-output relation (3.13). If the receiver has some knowledge about the second order statistics of the channel and the noise variance, a *minimum mean square error* (MMSE) approach can be utilized. The MMSE estimator typically outperforms the LS estimator, but has considerably higher computational complexity which makes it impractical for many applications. A block-type pilot arrangement is used, for example, in the WLAN standard IEEE 802.11a [50], and is often combined with a decision feedback equalizer to enhance the performance [51].

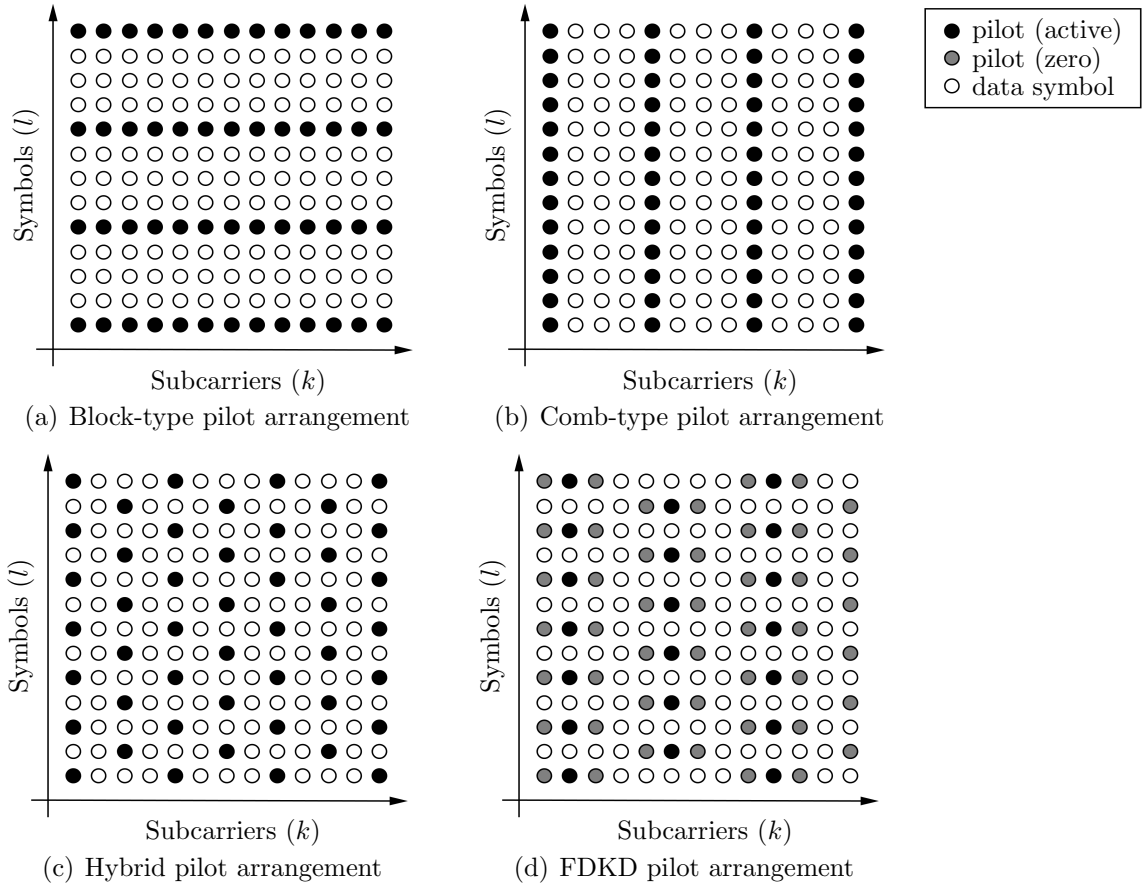


Figure 3.3: Pilot arrangements in SISO systems

If the channel changes within such a block of several OFDM symbols, pilots have to be inserted into *each* OFDM symbol as shown in Fig. 3.3(b). Such a pilot arrangement is called *comb-type*. The pilot set can then be written as $\mathcal{P} = \{(l, k_0 + mk_B) | l = 0, \dots, L-1; m = 0, \dots, K/k_B - 1\}$ with some $k_0 \geq 0$ and some number k_B dividing the number K of subcarriers. Here, channel estimates can only be calculated at the subcarriers bearing pilots, again using an LS or an MMSE approach, but then efficient interpolation algorithms have to be utilized to obtain estimates at all the other subcarriers. Commonly used interpolation techniques are piecewise-constant functions, linear functions, quadratic functions or cubic splines [51]. WLAN standard IEEE 802.11g, for example, uses such a comb-type pilot arrangement [52]. In other standards, like for example multiuser WiMAX standard IEEE 802.16a [53], a *hybrid* pilot arrangement like the one depicted in Fig. 3.3(c) is used, which combines the benefits of both arrangements.

The arguments given above yield that block-type pilot arrangements are not appropriate for the estimation of doubly selective wireless channels. If the Doppler spread of

the channel is large, i.e. if the channel introduces severe ICI, often a so-called *frequency domain Kronecker delta* (FDKD) arrangement is used (Fig. 3.3(d)). Here, the pilot sub-carriers at each OFDM symbol are arranged in blocks, where only the center pilot has non-zero power, and the neighboring pilots are set to zero to guard this center pilot from interference with the data symbols [54, 55]. FDKD is used in applications like DVB-T for channel estimation [8].

A very useful tool for estimating doubly selective channels is to use a *basis expansion model* (BEM) [56, 57, 58, 59, 8]. Here, the discrete-time channel impulse response $h[n, m]$ in (3.10) is modeled as a linear combination of the orthonormal basis functions ψ_i for $i = 0, \dots, N_0 - 1$, i.e.

$$h[n, m] = \sum_{i=0}^{N_0-1} T_h[m, i] \psi_i[n], \quad n = 0, \dots, N_0 - 1, \quad (3.19)$$

with m -dependent expansion coefficients

$$T_h[m, i] = \sum_{n=0}^{N_0-1} h[n, m] \psi_i^*[n].$$

This coefficient function $T_h[m, i]$ clearly is a generalization of the discrete-delay-Doppler spreading function $S_h[m, i]$ in (3.14) which is obtained in the special case where $\psi_i[n] = \frac{1}{\sqrt{N_0}} e^{-j2\pi \frac{in}{N_0}}$. The task then is to approximately calculate this coefficient function, from which an estimate of the channel then can be obtained via (3.19).

Many different ideas for choosing the basis functions have been proposed. Among them are for example the complex exponential BEM [56], which uses a truncated Fourier series for modeling the channel, and the Generalized CE-BEM [58], where the complex exponentials are oversampled in the frequency domain. Also, the use of discrete prolate spheroidal basis functions [59] as well as polynomial BEMs [57] have been studied.

3.6 Multiple-Input Multiple-Output Systems

3.6.1 Introduction

In 1993/94, Paulraj and Kailath first proposed to use multiple antennas in wireless communication systems [60]. Soon it turned out that such *multiple-input multiple-output*

(MIMO) systems have many useful properties that outweigh the drawbacks that arise. Therefore, MIMO systems have been studied extensively in recent years and are already part of several wireless standards such as WIFI standard IEEE 802.11n [61], WiMAX standard IEEE 802.16e [53], 3GPP HSPA+ [62], 3GPP LTE [63] as well as future cellular communication systems (3GPP LTE Advanced, 4G) [64, 65]. Note, that we can differentiate between *single-input multiple-output* (SIMO) systems, where only one transmit antenna but several receive antennas are used, *multiple-input single-output* (MISO) systems, where several transmit antennas but only one receive antenna is used, and MIMO systems utilizing several antennas at transmit and receive side, which we consider exclusively throughout the rest of this thesis.

As explained in the previous sections, doubly selective channels in *single-input single-output* (SISO) systems introduce time and frequency diversity, which means that several, (more or less) independently fading copies of the signal arrive at the receiver. If, on the other hand, the same signal is transmitted from several antennas in a MIMO system, an additional *spatial diversity* can be achieved, if the antennas are not too close to each other and therefore each copy of the signal fades independently. This increases the reliability of the data transmission without sacrificing spectral efficiency. Alternatively, different signals can be transmitted from different transmit antennas increasing the data throughput. This is known as *multiplexing*, and is probably the main reason why MIMO systems are as attractive. Here, the receiver has a far more difficult task to fulfill, namely to separate the received signal, which is a superposition of all the transmitted signals. Obviously, there is a fundamental tradeoff between the gain achieved by spatial diversity and by multiplexing, between increased reliability and higher data rate. It therefore depends on the actual setting which approach is to be followed.

Additionally, it is possible to increase the effective SNR by coherently combining signals on multiple transmit or receive antennas. Maximum ratio combining, for example, is one possibility to do so if the receiver has some knowledge about the channel [66]. If, on the other hand, the transmitter knows the channel approximately (for example by feedback from the receiver), beamforming can be applied to increase the SNR significantly [67]. As mentioned in Section 3.2, the reliability of a wireless communication link is not only affected by the SNR, but also by the use of error-correcting codes. In MIMO systems, the concept of space-time-coding is an additional way of increasing this reliability [68].

On the downside, MIMO systems need a quite complex hardware since each antenna needs a radio-frequency unit and the digital signal processing (DSP) units have to be very powerful. Also, the DSP algorithms involved have much higher computational complexity as the ones in SISO systems. Therefore, MIMO systems have a much higher power consumption which is a severe challenge for many applications such as handheld devices with limited battery power. Also, as mentioned above, the antennas have to be well separated to achieve maximal diversity, which might also be difficult if the transmit/receive devices are rather small. For more information on the basic concepts of MIMO systems, their advantages and disadvantages, and a detailed description of such systems, see for example [69, 70].

3.6.2 MC MIMO System Model

For later use we briefly review the MIMO system model, although it mostly parallels the one for SISO systems presented in Section 3.4. To do so, we denote the number of transmit and receive antennas by N_T and N_R , respectively. Again, we focus on MC transmission schemes with K subcarriers and symbol duration $N \geq K$. Here and throughout the rest of this thesis, we index transmit antennas by s , receive antennas by r , pairs of transmit and receive antennas by $\theta = (r, s)$, and the set of all possible pairs of antennas by $\Theta = \{\theta = (r, s) \mid r = 1, \dots, N_R; s = 1, \dots, N_T\}$. Each pair $\theta \in \Theta$ of transmit and receive antennas is connected by a doubly selective channel with impulse response $h^{(\theta)}(t, \tau)$. The MC MIMO system model can then be easily described in the same way as in the SISO case (see Section 3.4), using a simple vector-matrix notation as follows. Let $\mathbf{s}[n] = [s^{(1)}[n], \dots, s^{(N_T)}[n]]^T$ denote the vector-valued discrete-time transmit signal, where $s^{(s)}$ denotes the discrete-time signal corresponding to transmit antenna s for $s = 1, \dots, N_T$. Equivalently, let $\mathbf{r}[n] = [r^{(1)}[n], \dots, r^{(N_R)}[n]]^T$ denote the vector-valued discrete-time receive signal, and let $\mathbf{z}[n] = [z^{(1)}[n], \dots, z^{(N_R)}[n]]^T$ be the noise vector. Then, (3.10) becomes

$$\mathbf{r}[n] = \sum_{m \in \mathbb{Z}} \mathbf{H}[n, m] \mathbf{s}[n - m] + \mathbf{z}[n], \quad n \in \mathbb{Z},$$

with the $N_R \times N_T$ matrix $\mathbf{H}[n, m] = \int_{\mathbb{R}} \int_{\mathbb{R}} \mathbf{H}(t + nT_s, \tau) f_1(t - \tau + mT_s) f_2(-t) dt d\tau$, where $\mathbf{H}(t, \tau)$ is the matrix-valued channel impulse response with entries $[\mathbf{H}(t, \tau)]_{r,s} = h^{(r,s)}(t, \tau)$, and $f_1(t)$ and $f_2(t)$ are the impulse responses of the interpolation and anti-

aliasing filters, respectively. Denoting the symbol transmitted at time-frequency position (l, k) from transmit antenna s by $a_{l,k}^{(s)}$, we define the transmit symbol vector as $\mathbf{a}_{l,k} := [a_{l,k}^{(1)}, \dots, a_{l,k}^{(N_T)}]^T$. Equivalently, we define the received symbol vector as $\mathbf{r}_{l,k} := [r_{l,k}^{(1)}, \dots, r_{l,k}^{(N_R)}]^T$, a noise vector as $\mathbf{z}_{l,k} := [z_{l,k}^{(1)}, \dots, z_{l,k}^{(N_R)}]^T$, and the channel coefficient matrices $\mathbf{H}_{l,k;l'k'}$ with entries $[\mathbf{H}_{l,k;l'k'}]_{r,s} = H_{l,k;l',k'}^{(r,s)}$, where $H_{l,k;l',k'}^{(r,s)}$ denotes the system channel coefficient for the cross-channel between transmit antenna s and receive antenna r . Then, (3.12) becomes

$$\mathbf{r}_{l,k} = \sum_{l' \neq l} \sum_{k' \neq k} \mathbf{H}_{l,k;l'k'} \mathbf{a}_{l',k'} + \mathbf{z}_{l,k},$$

which, subsuming the ISI/ICI terms $\mathbf{H}_{l,k;l'k'}$ for $l' \neq l$ and $k' \neq k$ and the noise terms $\mathbf{z}_{l,k}$, can be written as

$$\mathbf{r}_{l,k} = \mathbf{H}_{l,k} \mathbf{a}_{l,k} + \tilde{\mathbf{z}}_{l,k}, \quad (3.20)$$

where $\mathbf{H}_{l,k} := \mathbf{H}_{l,k;l,k}$ and $\tilde{\mathbf{z}}_{l,k} := \mathbf{z}_{l,k} + \sum_{l' \neq l} \sum_{k' \neq k} \mathbf{H}_{l,k;l'k'} \mathbf{a}_{l',k'}$. If we use the approximation $N_0 \approx LK$ and assume the channel to be causal with maximum delay $\tau_{\max} \leq (K-1)T_s$, these "diagonal" coefficient matrices can be expressed as

$$\mathbf{H}_{l,k} = \sum_{m=0}^{K-1} \sum_{i=-L/2}^{L/2-1} \mathbf{F}_{m,i} e^{-j2\pi(\frac{km}{K} - \frac{li}{L})}, \quad (3.21)$$

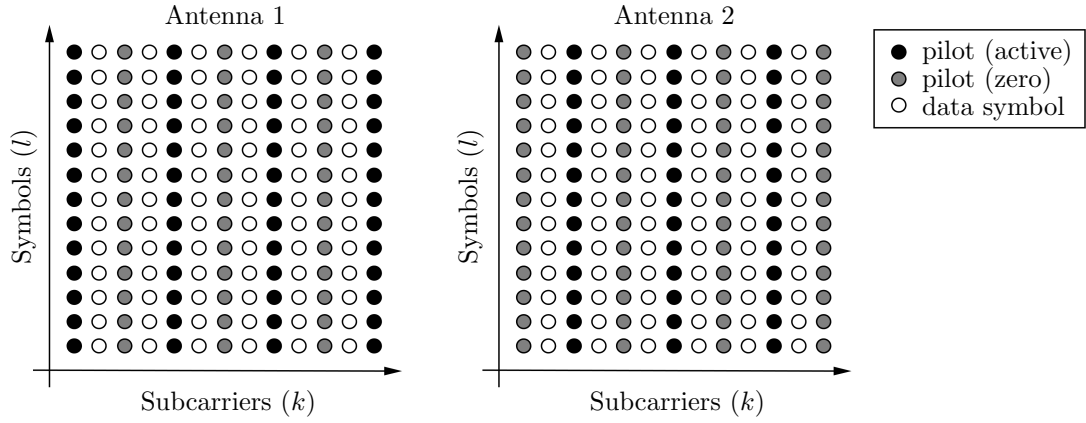
where the Fourier coefficient matrices $\mathbf{F}_{m,i}$ can be written as

$$\mathbf{F}_{m,i} = \sum_{q=0}^{N-1} \mathbf{S}_h[m, i + qL] A_{\gamma,g}^* \left(m, \frac{i + qL}{N_0} \right). \quad (3.22)$$

Here, $\mathbf{S}_h[m, i]$ denotes the discrete-delay-Doppler spreading function matrix with entries $[\mathbf{S}_h[m, i]]_{r,s} = S_h^{(r,s)}[m, i]$, where $S_h^{(r,s)}[m, i]$ denotes the discrete-delay-Doppler spreading function of the cross-channel $\theta = (r, s)$.

3.6.3 MIMO Channel Estimation

Channel estimation in MC MIMO systems is even more difficult than in the SISO case, since there are $N_T N_R$ cross channels that have to be estimated simultaneously. Therefore, the amount of training data has to be increased. Also, the arrangement and design of the pilot symbols is crucial since each receive antenna sees a superposition of all the signals transmitted over several cross channels. Typically, all antennas transmit

Figure 3.4: Comb-type pilot arrangement in a 2×2 MIMO system

pilot symbols at the same time-frequency positions $(l, k) \in \mathcal{P}$, and these *pilot vectors* $\mathbf{a}_{l,k} = \mathbf{p}_{l,k}$ are again arranged in a block-type, comb-type or hybrid arrangement. In many practical applications a *zero pilot pattern* is used where only one transmit antenna actually transmits a pilot symbol at a given pilot position, whereas the other antennas remain silent, i.e. each pilot vector $\mathbf{p}_{l,k}$ has only one non-zero entry. Then, the input-output relation (3.20) at a pilot position $(l, k) \in \mathcal{P}$ becomes

$$\mathbf{r}_{l,k} = \mathbf{H}_{l,k} \mathbf{p}_{l,k} + \tilde{\mathbf{z}}_{l,k} = p_{l,k}^{(s)} \mathbf{h}_{l,k}^{(s)} + \tilde{\mathbf{z}}_{l,k}, \quad (3.23)$$

where the pilot vector $\mathbf{p}_{l,k} = [0, \dots, p_{l,k}^{(s)}, \dots, 0]^T$ has its only non-zero entry $p_{l,k}^{(s)}$ at position s , and $\mathbf{h}_{l,k}^{(s)}$ is the s -th column of $\mathbf{H}_{l,k}$. Rewriting (3.23) coordinate wise, we have

$$r_{l,k}^{(r)} = H_{l,k}^{(\theta)} p_{l,k}^{(s)} + \tilde{z}_{l,k}^{(r)},$$

and we can calculate an estimate of the channel coefficient of the cross channel connecting each receive antenna $r = 1, \dots, N_R$ with the active transmit antenna s like in the SISO case (see Section 3.5). If no additional information is available at the receiver, the overall amount of training data obviously has to be increased by a factor of N_T to obtain the same quality of channel estimation as in the SISO case. A comb-type pilot arrangement for a 2×2 MIMO OFDM system using this approach is depicted in Fig. 3.4.

Alternatively, using a *non-zero pilot pattern*, non-zero pilots are transmitted from all antennas simultaneously, but then they have to be designed orthogonally such that the cross-channels can be separated at the receiver. Examples for estimation schemes using this type of pilot pattern can be found in [71, 72]. Also, estimation methods using zero and non-zero pilot patterns are compared in [73]. The MC MIMO channel estimators presented in this thesis can be used with both of these pilot constellations.

Compressed Sensing

4.1 Introduction

In our digital age it has become a very important task to transform natural signals into digital ones that can then be further processed by computational devices. This is done by *sampling* the analog (continuous) signal at as many sampling points as it takes to capture the essence of the signal. The theoretical foundation was built in the first half of the 20th century by Kotelnikov [74], Nyquist [75], Shannon [76], and Whittaker [77], who demonstrated that continuous-time, band-limited signals can be exactly recovered from twice as many samples per second as the highest frequency present in the signal, which is known as the *Nyquist rate*. Since then, an enormous amount of work has been invested into the development of sampling theory on the one hand, and digital sensing and processing systems on the other hand, that are typically faster, cheaper and more robust than conventional analog systems.

Although with the ever-growing computational capabilities of modern systems the amount of data that can be acquired has grown enormously, too, the sampling rate in many practical applications including highly complex signals might be so high that it still is too costly or not even possible to build an appropriate sampling device. In other cases it might not be possible to store all the data because of limited storage, or the data amount might be too large to be processed in real-time, so that it becomes necessary to *compress* the data. Here, often correlations or redundancies in the signal are exploited to reduce the amount of data that has to be stored to capture the essence

of the signal up to a reasonably good approximation level. The widely used technique called *transform coding*, in contrast, exploits some knowledge about the nature of the signal to extract the most important information (from an application point of view) and simply ignore the rest of the data. Well known examples are the JPEG and MPEG standards [78, 79]. JPEG, for example, utilizes the fact that the human eye is much more sensitive to small fluctuations in brightness than in color in order to reduce the number of colors and thereby reduce the amount of data that is to be stored. The image that is reconstructed from its JPEG encoding clearly differs from the original image, but this difference is barely noticeable.

A very closely related process is called *sparse approximation*, where a basis or frame that allows *sparse* or *compressible* representations of signals in a certain class of interest is desired [80, 81, 82, 83]. By sparse we mean that only very few of the representation coefficients are non-zero, whereas compressible signals are the ones that are well approximated by sparse signals. Sparse signals can be well compressed by just storing the non-zero coefficients and their locations.

The fundamental weakness of this approach is that most of the data that is acquired at a very high rate in the beginning is discarded later for compression. But since typically the location of the important coefficients is not known before sampling, there seemed to be no way out of this dilemma until the theory of *compressed sensing* came along.

In the following, we give a short overview over the basic notions, methods and results of conventional compressed sensing in Section 4.2. In Sections 4.3, 4.4, 4.5 and 4.6 we review the adaption of this methodology to the case of group sparsity, joint sparsity, joint group sparsity and sequential sparsity, respectively, which are all defined within the corresponding sections.

4.2 Conventional Compressed Sensing

The theory and methodology of compressed sensing (CS), also known as *compressive sampling*, emerged from the groundbreaking work of Candès, Romberg and Tao [84, 85] and Donoho [86, 87] in 2006, who showed that a finite dimensional, sparse signal can be recovered from far fewer measurements than predicted by the Nyquist rate. Since then, a variety of new theoretical results as well as computationally tractable algorithms have

been developed, some of which are reviewed in this section. As already mentioned, we only consider finite dimensional signals here, and therefore we use the terms *vector* and *signal* interchangeably. Let us start with some basic definitions.

Definition 4.2.1. A vector $\mathbf{x} \in \mathbb{C}^N$ is said to be S -sparse for some $S \in \mathbb{N}$, if at most S of its entries are non-zero, i.e. if

$$\|\mathbf{x}\|_0 \leq S.$$

We denote the set of all s -sparse vectors in \mathbb{C}^N by Σ_S^N .

Actually, we drop the index indicating the dimension and write Σ_S whenever the dimension of the signals is clear from the context.

Definition 4.2.2. For a vector $\mathbf{x} \in \mathbb{C}^N$, some $S \in \mathbb{N}$ and some $p \geq 1$, we define the S -term approximation error of \mathbf{x} with respect to the p -norm as

$$\sigma_S(\mathbf{x})_p := \min_{\tilde{\mathbf{x}} \in \Sigma_S} \|\mathbf{x} - \tilde{\mathbf{x}}\|_p.$$

Proposition 4.2.3. Let $\mathbf{x} \in \mathbb{C}^N$, and let $\mathbf{x}_S^{(p)}$ be such that $\sigma_S(\mathbf{x})_p = \|\mathbf{x} - \mathbf{x}_S^{(p)}\|_p$. Then, for $p \geq 1$, $\mathbf{x}_S^{(p)} \equiv \mathbf{x}_S$, where \mathbf{x}_S is the vector that coincides with \mathbf{x} at those S coefficients with largest absolute values, and is zero everywhere else.

Proof. Let π be the permutation of the indices $\{0, \dots, N-1\}$ that sorts the entries of \mathbf{x} in descending order according to their magnitudes, i.e. $|\mathbf{x}_{\pi(0)}| \geq |\mathbf{x}_{\pi(1)}| \geq \dots \geq |\mathbf{x}_{\pi(N-1)}|$, and let $\tilde{\mathbf{x}} \in \Sigma_S$ be arbitrary. Then we have

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_p^p = \sum_{i=0}^{N-1} |\mathbf{x}_i - [\tilde{\mathbf{x}}]_i|^p = \sum_{i \in \text{supp } \tilde{\mathbf{x}}} |\mathbf{x}_i - [\tilde{\mathbf{x}}]_i|^p + \sum_{i \notin \text{supp } \tilde{\mathbf{x}}} |\mathbf{x}_i|^p \geq \sum_{i \notin \text{supp } \tilde{\mathbf{x}}} |\mathbf{x}_i|^p.$$

On the other hand, we can write

$$\|\mathbf{x} - \mathbf{x}_S\|_p^p = \sum_{i=0}^{N-1} |\mathbf{x}_i - [\mathbf{x}_S]_i|^p = \sum_{i=S}^{N-1} |\mathbf{x}_{\pi(i)}|^p.$$

Since by the definition of π this last sum is taken over the $N - S$ smallest entries of \mathbf{x} , and since $|\text{supp } \tilde{\mathbf{x}}| = S$, we certainly have

$$\sum_{i \notin \text{supp } \tilde{\mathbf{x}}} |\mathbf{x}_i|^p \geq \sum_{i=S}^{N-1} |\mathbf{x}_{\pi(i)}|^p,$$

which concludes the proof since $\tilde{\mathbf{x}}$ was chosen arbitrarily in Σ_S . □

4 Compressed Sensing

Obviously, for all S -sparse signals $\mathbf{x} \in \Sigma_S$ we have $\sigma_S(\mathbf{x})_p = 0$ for all $p \geq 1$. Unfortunately, many signals that arise in practical scenarios are not exactly sparse, though. If a signal \mathbf{x} can be reasonably well approximated by a sparse signal, i.e. if its S -term approximation error $\sigma_S(\mathbf{x})_p$ decays fast in S for some $p \geq 1$, we call it *compressible*. This rather general notion can be quantified by using the permutation π defined in the proof of Proposition 4.2.3 again. If these coefficients follow a power law decay, i.e.

$$|[\mathbf{x}]_{\pi(i)}| \leq c_0 i^{-q}$$

for some $q > 0$ and some constant $c_0 > 0$, there exists some $r > 0$ and some constant $c_1 > 0$ such that [88]

$$\sigma_S(\mathbf{x})_2 \leq c_1 S^{-r}.$$

Obviously the compressibility of the signal is better the larger r (or equivalently q) is. Moreover, we use the rather general term *essential support* to refer to the part of the support of a signal corresponding to the "large" entries.

Now let us return to the basic problem of data acquisition. Typically, noisy linear measurements $\mathbf{y} \in \mathbb{C}^M$ of an unknown signal $\mathbf{x} \in \mathbb{C}^N$ are taken according to the measurement equation

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} + \mathbf{z}, \tag{4.1}$$

where $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ is the *measurement matrix*, and $\mathbf{z} \in \mathbb{C}^M$ is a noise vector. Let us consider the noise-free case first, i.e. let us assume $\mathbf{z} = \mathbf{0}_M$. In this case N linearly independent measurements need to be taken to reconstruct an arbitrary signal $\mathbf{x} \in \mathbb{C}^N$ exactly, i.e. $\mathbf{\Phi}$ needs to be quadratic and invertible, and we cannot expect to succeed with less measurements in the general case. For sparse signals, on the other hand, we can at least hope to do better. If the positions of the non-zero coefficients of an S -sparse signal \mathbf{x} were known in advance, we could reconstruct it from exactly S linear measurements if each row of $\mathbf{\Phi}$ was to be zero everywhere but at the position of a non-zero entry of \mathbf{x} . Unfortunately, this information is not available in typical scenarios, and therefore we can only hope to succeed with some intermediate number M of measurements, i.e. $S \leq M \leq N$.

This is exactly where the theory of CS comes into play. It introduces measurement and reconstruction strategies so that $M \ll N$ linear measurements are sufficient for exact reconstruction of S -sparse signals (where of course still $S \leq M$). The measurement matrix $\mathbf{\Phi}$ then is a "fat" matrix, i.e. it has much fewer rows than columns, and the

system (4.1) of linear equations is massively underdetermined. Therefore, there are infinitely many solutions in general. Restricting ourselves to the case of sparse signals, an intuitive strategy would be to simply choose the sparsest vector that is consistent with the measurements, i.e. to solve

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\mathbf{x}'\|_0 \quad \text{subject to } \Phi \mathbf{x}' = \mathbf{y}.$$

Unfortunately, $\|\cdot\|_0$ is not convex. Actually, this problem is NP hard [89] and therefore cannot be solved in a reasonable amount of time. The way out is to study the convex relaxation of this problem, namely to solve

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\mathbf{x}'\|_1 \quad \text{subject to } \Phi \mathbf{x}' = \mathbf{y},$$

which is widely known as the *basis pursuit* (BP) [90]. An explanation for the use of the ℓ_1 -norm for promoting sparsity can be found in [91].

In the noisy case, on the other hand, this strategy can easily be adapted as

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\mathbf{x}'\|_1 \quad \text{subject to } \|\Phi \mathbf{x}' - \mathbf{y}\|_2 \leq \epsilon, \quad (4.2)$$

where ϵ is an upper bound on the noise level. This problem is known as *basis pursuit denoising* (BPDN) [92], and can equivalently be formulated as [90]

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \frac{1}{2} \|\Phi \mathbf{x}' - \mathbf{y}\|_2^2 + \tau \|\mathbf{x}'\|_1$$

with some parameter $\tau > 0$, or, as is typically the case if referred to as *least absolute shrinkage and selection operator* (LASSO) [93],

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\Phi \mathbf{x}' - \mathbf{y}\|_2 \quad \text{subject to } \|\mathbf{x}'\|_1 \leq \epsilon.$$

Intuitively it is clear that the prementioned strategies will not succeed for arbitrary measurement matrices Φ . Therefore, we define the following.

Definition 4.2.4. A matrix $\Phi \in \mathbb{C}^{M \times N}$ is said to satisfy the *restricted isometry property* (RIP) of order S for some $S \in \mathbb{N}$, if there is a constant $0 \leq \delta < 1$ such that

$$(1 - \delta_S) \|\mathbf{x}\|_2^2 \leq \|\Phi \mathbf{x}\|_2^2 \leq (1 + \delta_S) \|\mathbf{x}\|_2^2$$

holds for every S -sparse $\mathbf{x} \in \Sigma_S$. The smallest such constant is denoted by δ_S , and it is called the *restricted isometry constant* (RIC) of Φ .

4 Compressed Sensing

If the RIC of a matrix Φ is small, the restriction of Φ to any subset of S columns behaves more or less like an isometry. Note that sometimes in the literature the RIP is called restricted isometry *condition* and therefore is abbreviated as RIC. We choose to follow the more frequently used nomenclature as defined in Definition 4.2.4 in this thesis.

With this definition at hand it is possible to show that solving BPDN indeed is a robust and stable reconstruction strategy for sparse or compressible signals [92].

Theorem 4.2.5. *Let $\mathbf{x} \in \mathbb{C}^N$ be arbitrary, and let $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$ be noisy measurements with $\mathbf{y} \in \mathbb{C}^M$ and $\|\mathbf{z}\|_2 \leq \epsilon$. If $\Phi \in \mathbb{C}^{M \times N}$ satisfies the RIP with RIC*

$$\delta_{2S} < \sqrt{2} - 1 \quad (4.3)$$

for some $S \in \mathbb{N}$, then the solution $\hat{\mathbf{x}}$ of BPDN satisfies

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_S(\mathbf{x})_1}{\sqrt{S}} + c_1 \epsilon,$$

where $c_0 = 2 \frac{1-(1-\sqrt{2})\delta_{2S}}{1-(1+\sqrt{2})\delta_{2S}}$ and $c_1 = 4 \frac{\sqrt{1+\delta_{2S}}}{1-(1+\sqrt{2})\delta_{2S}}$.

Here, the first term is due to the fact that \mathbf{x} might not be exactly sparse, whereas the second term is due to the measurement noise. Therefore, this theorem ensures that any S -sparse signal can be reconstructed exactly in the noiseless case, i.e. $\epsilon = 0$, by solving BPDN under condition (4.3). The next question therefore is, which matrices satisfy the RIP under this condition. Unfortunately it turns out that it is computationally intractable to calculate the RIC of large matrices, which makes it hard to design an appropriate measurement matrix in practice. In 2006 though, the groundbreaking work of Candès, Romberg and Tao [84, 85] and Donoho [86, 87] came to the rescue. By using the concept of randomness they were able to define a class of matrices that, with very high probability, will satisfy the RIP. Since then, many such classes of matrices have been identified, but we only mention some of them here.

For example, consider *Gaussian random matrices*, i.e. matrices where the entries are identically and independently distributed Gaussian random variables with mean 0 and variance $1/M$, or *Bernoulli random matrices*, where the entries take the value $+1/\sqrt{M}$ or $-1/\sqrt{M}$ with equal probability (see Section 2.4). It can be shown that such matrices satisfy the RIP of some order S with small RIC δ_S with very high probability if the number of rows is chosen large enough [94]. In many of the theory-building work in

CS Gaussian or Bernoulli random matrices are used because of their nice theoretical properties. In practice, though, they are of limited use for several reasons. In some applications the design of the measurement matrix is constrained by physical or other conditions. Sometimes, as in the compressive channel estimation methods presented in this thesis, we cannot design it at all because it is fixed from the application. Most importantly, though, these matrices do not allow a fast matrix multiplication, which typically has to be performed quite often in practical applications.

Therefore, *structured* random matrices are often preferred [95]. Such matrices can, among others, be constructed from general bounded orthonormal systems, from circulant matrices or from Toeplitz matrices. For example, let $\mathbf{U} \in \mathbb{C}^{N \times N}$ be a unitary matrix. Then define the $M \times N$ matrix Φ by choosing M rows of \mathbf{U} uniformly at random, i.e. each row of \mathbf{U} has equal probability of being chosen. Then, the following holds [95].

Theorem 4.2.6. *Let $\Phi \in \mathbb{C}^{M \times N}$ be constructed from a unitary matrix $\mathbf{U} \in \mathbb{C}^{N \times N}$ as described above. If*

$$\frac{M}{\ln(5M)} \geq c\delta^{-2}K^2s \log^2(100S) \log(4N) \ln(8\epsilon^{-1})$$

for some $\epsilon, \delta > 0$, $K := \sqrt{N} \max_{k,l} |U_{k,l}|$, and some constant $c < 84800$, then, with probability at least $1 - \epsilon$, $\sqrt{\frac{N}{M}}\Phi$ satisfies the RIP with RIC $\delta_S \leq \delta$.

The so-called *partial random Fourier matrices* certainly form the most prominent example, where $\mathbf{U} = \mathbb{F}_N$ is chosen to be the $N \times N$ DFT matrix. This setup has been the first to be studied in [84, 85].

By now there are many different algorithms solving BPDN quite fast [96, 97, 91], each of which has different computational complexity. For example, BPDN can usually be solved with $\mathcal{O}(MN^2)$ operations using interior point methods [98].

As an alternative reconstruction strategy, *greedy algorithms* have obtained a lot of attention because of their ease of implementation and low computational complexity. Here, the support of the unknown signal is recovered iteratively. The most prominent example is the so-called *orthogonal matching pursuit* (OMP) [99, 100]. Given the noisy measurements $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$ of the signal \mathbf{x} , it starts by initializing a residual $\mathbf{r}_{(0)} = \mathbf{y}$, the support set as $\mathcal{S}_{(0)} = \emptyset$, and the iteration count as $i = 0$. Then, at iteration $i \geq 1$, a signal proxy $\mathbf{u}_{(i)} = |\Phi^H \mathbf{r}_{(i-1)}|$ is calculated, the entries of which correspond to the correlations of the columns of Φ with the previous residual. Then, the support set is

4 Compressed Sensing

updated as $\mathcal{S}_{(i)} = \mathcal{S}_{(i-1)} \cup \{n_{(i)}\}$, where $n_{(i)}$ is the index corresponding to the entry of $\mathbf{u}_{(i)}$ with maximal absolute value, i.e. the column of Φ maximally correlated with the residual. Finally, the LS problem $\mathbf{x}_{(i)} = \arg \min_{\text{supp}\{\mathbf{x}'\} \in \mathcal{S}_{(i)}} \|\mathbf{y} - \Phi \mathbf{x}'\|_2$ is solved to obtain a new signal estimate $\mathbf{x}_{(i)}$, and, in turn, a new residual $\mathbf{r}_{(i)} = \mathbf{y} - \Phi \mathbf{x}_{(i)}$. The process is stopped if either the maximal number of iterations has been reached or if the ℓ_2 -norm of the residual falls below a prescribed threshold. Note that by definition each signal estimate $\mathbf{x}_{(i)}$ is i -sparse.

For this algorithm, it is possible to obtain the following result [101].

Theorem 4.2.7. *Let $\mathbf{x} \in \mathbb{C}^N$ be arbitrary, and let $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ be noisy measurements with $\mathbf{y} \in \mathbb{C}^N$ and $\|\mathbf{z}\|_2 \leq \epsilon$. If $\Phi \in \mathbb{C}^{M \times N}$ satisfies the RIP with RIC*

$$\delta_S + (1 + \delta)\delta_{\alpha S} < \delta \tag{4.4}$$

with $\alpha = \lceil 16 + 15\delta \rceil$ for some $0 < \delta \leq 1$, then the result $\hat{\mathbf{x}}$ of OMP after $2(\alpha - 1)S$ steps satisfies

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_S(\mathbf{x})_1}{\sqrt{S}} + c_1 \epsilon,$$

where $c_0 = 2 + c_1$ and $c_1 = 2(1 + \delta)(\sqrt{11 + 20\delta} + 1) + 1$.

Note that this result is of limited use in practice since the sparsity level S has to be quite small for (4.4) to hold.

Let n_{OMP} denote the number of OMP steps that are taken (note that by construction of the algorithm $n_{\text{OMP}} \leq M$). Following the implementation of OMP described in [102] based on the QR-factorization, its computational complexity is $\mathcal{O}(n_{\text{OMP}}(Mn_{\text{OMP}} + \Phi))$, where $\mathcal{O}(\Phi)$ denotes the computational complexity of applying Φ or Φ^H to a vector of appropriate length. Here, the term Mn_{OMP} is due to the update of the QR-factorization that is performed in each step, whereas the term $\mathcal{O}(\Phi)$ is a consequence of the calculation of the signal proxy. Note that for a completely unstructured matrix Φ we will have $\mathcal{O}(\Phi) = MN$, but it can also be as small as $\mathcal{O}(\Phi) = M \log(N)$, if for example Φ is constructed from the DFT matrix as described above and therefore FFT techniques can be used.

Next to OMP, the greedy algorithm *compressive sampling matching pursuit* (CoSaMP) [103] has also gained a lot of attention. Similarly to OMP, it starts by initializing a residual as $\mathbf{r}_{(0)} = \mathbf{y}$, the support set as $\mathcal{S}_{(0)} = \emptyset$, and the iteration count as $i = 0$, and

additionally sets $\mathbf{x}_{(0)} = \mathbf{0}$. Then, at iteration $i \geq 1$, the signal proxy $\mathbf{u}_{(i)} = |\Phi^H \mathbf{r}_{(i-1)}|$ is calculated just as in OMP, but here a set $\tilde{\mathcal{S}}_{(i)}$ is defined as the augmentation of the support of the previous estimate $\mathbf{x}_{(i-1)}$ with the $2S$ indices corresponding to the entries of $\mathbf{u}_{(i)}$ with largest absolute values. Here S is an estimate of the sparsity of \mathbf{x} that has to be given before the start of the algorithm. Then, a temporary signal estimate is calculated as $\tilde{\mathbf{x}}_{(i)} = \arg \min_{\text{supp}\{\mathbf{x}'\} \in \tilde{\mathcal{S}}_{(i)}} \|\mathbf{y} - \Phi \mathbf{x}'\|_2$, and the support estimate $\mathcal{S}_{(i)}$ is defined as the S indices corresponding to the entries of $\tilde{\mathbf{x}}_{(i)}$ with largest absolute values. Finally, a new signal estimate $\mathbf{x}_{(i)} = (\tilde{\mathbf{x}}_{(i)})_{\mathcal{S}_{(i)}}$ and a new residual $\mathbf{r}_{(i)} = \mathbf{y} - \Phi \mathbf{x}_{(i)}$ are calculated (by $(\tilde{\mathbf{x}}_{(i)})_{\mathcal{S}_{(i)}}$ we again mean the vector coinciding with $\tilde{\mathbf{x}}_{(i)}$ at the S "largest" coefficients and vanishing everywhere else). The algorithm stops when the maximal number of iterations has been reached. In [103] the following result about the quality of the solution has been proven.

Theorem 4.2.8. *Let $\mathbf{x} \in \mathbb{C}^N$ be arbitrary, and let $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ be noisy measurements with $\mathbf{z} \in \mathbb{C}^N$ and $\|\mathbf{z}\|_2 \leq \epsilon$. If $\Phi \in \mathbb{C}^{M \times N}$ satisfies the RIP with RIC $\delta_{4S} < 0.1$, then the result $\hat{\mathbf{x}}$ of CoSaMP after n_{CoSaMP} steps satisfies*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq 20\sigma_S(\mathbf{x})_2 + \frac{20\sigma_S(\mathbf{x})_1}{\sqrt{S}} + 20\epsilon + 2^{-n_{\text{CoSaMP}}} \|\mathbf{x}\|_2.$$

Also in [103] the computational complexity of CoSaMP is given as $\mathcal{O}(n_{\text{CoSaMP}} \Phi)$, where n_{CoSaMP} is the maximal number of iterations.

Finally note that there are many other algorithms for sparse reconstruction, such as Iterative Hard Thresholding (IHT) [104], Stagewise OMP (StOMP) [105], Regularized OMP (ROMP) [106, 107] and many more, which we do not describe in detail here.

4.3 Group Sparse Compressed Sensing

In many practical scenarios the non-zero components of sparse signals tend to appear in clusters. This and more general forms of *structured sparsity* arise naturally for example when dealing with multi-band signals [108, 109], in measurements of gene expression levels [110], or in magnetoencephalography [111, 112]. To exploit this structure for improved reconstruction quality, the methodology of *group sparse CS* (GSCS) has been introduced [113]. GSCS is closely related to *block sparse CS* [114, 115], *model-based CS* [116], and *recovery of signals from a structured union of subspaces* [117]. To be exact,

4 Compressed Sensing

let $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ be a partition of the set $\{0, \dots, N-1\}$, i.e. $\bigcup_{b=0}^{B-1} I_b = \{0, \dots, N-1\}$ and $\sum_{b=0}^{B-1} |I_b| = N$. Furthermore, we define $\mathbf{x}[b] \in \mathbb{C}^{|I_b|}$ to be the subvector of a signal $\mathbf{x} \in \mathbb{C}^N$ comprising the elements $[\mathbf{x}]_i$ corresponding to the indices $i \in I_b$.

Definition 4.3.1. A signal $\mathbf{x} \in \mathbb{C}^N$ is called *group S -sparse with respect to a partition \mathcal{J}* , if at most S of the subvectors $\mathbf{x}[b]$ are not identically zero. The set of all such signals is denoted by $\Sigma_{S|\mathcal{J}}$.

First of all, note that a group S -sparse signal \mathbf{x} is also S' -sparse, where S' is the sum of the cardinalities of the groups associated with the non-zero entries of \mathbf{x} , or, more generally, S'' -sparse, where S'' is the sum of the cardinalities of the S largest groups. Therefore, if we are given some noisy linear measurements \mathbf{y} of a group sparse signal \mathbf{x} according to (4.1), we can reconstruct it by using any of the methods described in the previous section like BPDN, OMP or CoSaMP. Obviously, this approach does not take the structure inherent to the signal into account. To do so, the notions and techniques have been adapted to this setting.

Definition 4.3.2. A matrix $\Phi \in \mathbb{C}^{M \times N}$ satisfies the *group restricted isometry property (G-RIP) of order S* , if there is a constant $0 < \delta < 1$ such that

$$(1 - \delta)\|\mathbf{x}\|_2^2 \leq \|\Phi\mathbf{x}\|_2^2 \leq (1 + \delta)\|\mathbf{x}\|_2^2$$

holds for every group S -sparse $\mathbf{x} \in \Sigma_{S|\mathcal{J}}$. The smallest such constant, denoted by $\delta_{S|\mathcal{J}}$, is called the *group restricted isometry constant (G-RIC) of Φ* .

In [117] it is shown that Gaussian and Bernoulli random matrices also satisfy the G-RIP with very high probability if the number of rows is chosen large enough. For measurement matrices constructed from unitary matrices as described in Section 4.2 there are no such results, to the best of our knowledge, yet. Nevertheless, since obviously $\delta_{S|\mathcal{J}} \leq \delta_{S''}$ (where S'' is defined as above), Theorem 4.2.6 can be used here also to guarantee a small G-RIC under the assumptions made therein.

Next, let us define the mixed norm

$$\|\mathbf{x}\|_{p|\mathcal{J}} := \sum_{b=0}^{B-1} \|\mathbf{x}[b]\|_p.$$

Here, the content of each group subvector $\mathbf{x}[b]$ is measured by the ℓ_p -norm, whereas the sum of these norms can be interpreted as an ℓ_1 -norm, which serves as a measure of sparsity as mentioned in the previous section.

Definition 4.3.3. For a vector $\mathbf{x} \in \mathbb{C}^N$, a partition \mathcal{J} , some $S \in \mathbb{N}$ and some norm $\|\cdot\|$ on \mathbb{C}^N , we define the S -group approximation error of \mathbf{x} with respect to \mathcal{J} and the norm $\|\cdot\|$ as

$$\sigma_{S|\mathcal{J}}(\mathbf{x})_{\|\cdot\|} := \min_{\tilde{\mathbf{x}} \in \Sigma_{S|\mathcal{J}}} \|\mathbf{x} - \tilde{\mathbf{x}}\|. \quad (4.5)$$

The adaption of the BPDN problem, which is called the *group BPDN* (G-BPDN), is defined as

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\mathbf{x}'\|_{2|\mathcal{J}} \quad \text{subject to} \quad \|\Phi \mathbf{x}' - \mathbf{y}\|_2 \leq \epsilon.$$

In [117], the following result has been obtained, which shows that G-BPDN is also stable with respect to compressibility and robust to noise.

Theorem 4.3.4. *Let $\mathbf{x} \in \mathbb{C}^N$ be arbitrary, and let $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ be noisy measurements with $\mathbf{y}, \mathbf{z} \in \mathbb{C}^M$ and $\|\mathbf{z}\|_2 \leq \epsilon$. If $\Phi \in \mathbb{C}^{M \times N}$ satisfies the G-RIP with respect to \mathcal{J} with G-RIC*

$$\delta_{2S|\mathcal{J}} < \sqrt{2} - 1,$$

then the solution $\hat{\mathbf{x}}$ of G-BPDN satisfies

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_{S|\mathcal{J}}(\mathbf{x})_{2|\mathcal{J}}}{\sqrt{S}} + c_1 \epsilon,$$

where $c_0 = \frac{2(1-\delta_{2S|\mathcal{J}})}{1-(1+\sqrt{2})\delta_{2S|\mathcal{J}}}$ and $c_1 = \frac{4\sqrt{1+\delta_{2S|\mathcal{J}}}}{1-(1+\sqrt{2})\delta_{2S|\mathcal{J}}}$.

Note that this theorem was originally formulated for the special case of block sparsity, but can be transferred to this general setting without any change. To the best of our knowledge there are no results analyzing the computational complexity of algorithms solving G-BPDN in the literature.

Similarly, OMP can easily be adapted to the group sparse setting [118, 114]. Instead of adding only one single index to the support set $\mathcal{S}_{(i)}$ in each iteration, an entire group is added, i.e. $\mathcal{S}_{(i)} = \mathcal{S}_{(i-1)} \cup I_b$, where $b = \arg \max_{b'} \|\mathbf{u}_{(i)}[b']\|_2$ for $\mathbf{u}_{(i)} = |\Phi^H \mathbf{r}_{(i-1)}|$. Apart from that, *group OMP* (G-OMP) proceeds exactly like conventional OMP. G-OMP will be faster in practice, since the signal proxy has to be calculated less often and the search space for the next group is smaller if the groups contain more than one index. The computational complexity of G-OMP is $\mathcal{O}(M(n'_{\text{G-OMP}})^2 + n_{\text{G-OMP}} \Phi)$ following again the implementation in [102], where $n_{\text{G-OMP}}$ denotes the number of steps that are taken, and $n'_{\text{G-OMP}}$ is the sum of the magnitudes of the chosen groups. If

$M(n'_{\text{G-OMP}})^2 \geq n_{\text{OMP}} \Phi$, this formally is the same complexity as for conventional OMP. Unfortunately, the performance of G-OMP has not been studied very well, yet (see [118, 114] for some partial results).

Moreover, by specializing the modelbased CoSaMP algorithm described in [116] to the group sparse model we get the *group CoSaMP* (G-CoSaMP) algorithm. Its computational complexity is given by $\mathcal{O}(n_{\text{G-CoSaMP}} \Phi)$, where $n_{\text{G-CoSaMP}}$ denotes the number of iterations. Adapting the result in [116] regarding block sparse signals we get the following theorem, which holds for partitions \mathcal{J} with groups of equal size.

Theorem 4.3.5. *Let $\mathbf{x} \in \mathbb{C}^N$ be arbitrary, and let $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ be noisy measurements with $\mathbf{y}, \mathbf{z} \in \mathbb{C}^M$ and $\|\mathbf{z}\|_2 \leq \epsilon$. If $\Phi \in \mathbb{C}^{M \times N}$ satisfies the G-RIP with G-RIC $\delta_{4S|\mathcal{J}} < 0.1$ for some $S \in \mathbb{N}$, then the result $\hat{\mathbf{x}}$ of CoSaMP after $n_{\text{G-CoSaMP}}$ steps satisfies*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq 2^{-n_{\text{G-CoSaMP}}} \|\mathbf{x}\|_2 + 20 \left(1 + \frac{1}{\sqrt{S}}\right) \sigma_S(\mathbf{x})_{2|\mathcal{J}} + 20\epsilon.$$

To obtain this result we used that $\|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_{2|\mathcal{J}}$, since

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{i=0}^{N-1} |[\mathbf{x}]_i|^2} = \sqrt{\sum_{b=0}^{B-1} \sum_{i \in I_b} |[\mathbf{x}]_i|^2} \leq \sum_{b=0}^{B-1} \sqrt{\sum_{i \in I_b} |[\mathbf{x}]_i|^2} = \sum_{b=0}^{B-1} \|\mathbf{x}[b]\|_2 = \|\mathbf{x}\|_{2|\mathcal{J}}.$$

Furthermore, note that the matrix X_K^S from the result in [116] is defined such that its (group sparse) vectorized form $\text{vec}\{X_K^S\}$ minimizes $\|\mathbf{x} - \tilde{\mathbf{x}}\|_2$ for $\tilde{\mathbf{x}} \in \Sigma_{S|\mathcal{J}}$. It can be shown by the exact same arguments as in the proof of Proposition 4.2.3 that this is the vector coinciding with \mathbf{x} at the positions corresponding to the S groups with largest ℓ_2 -norm. Obviously this is exactly the vector minimizing $\|\mathbf{x} - \tilde{\mathbf{x}}\|_{2|\mathcal{J}}$ appearing in the definition of $\sigma_S(\mathbf{x})_{2|\mathcal{J}}$.

Finally, note that in the special case where each group contains only one index G-BPDN, G-OMP and G-CoSaMP obviously coincide with their conventional counterparts.

4.4 Multichannel Compressed Sensing

Next, we consider the setting where not only one, but a whole ensemble of signals is measured. This naturally arises in many practical applications, for example in sensor

networks where multiple sensors acquire information about a physical or environmental condition such as temperature, pressure or sound [119]. If a centralized architecture is used, all the measurements of the sensors are transmitted to a central node which then processes all the signals. Since all the sensors observe the same phenomenon the different signals are likely to share certain structures, like sparsity.

To be more precise, let us begin with a few definitions.

Definition 4.4.1. An ensemble of vectors $\{\mathbf{x}^{(\theta)}\}_{\theta \in \Theta}$ with $\mathbf{x}^{(\theta)} \in \mathbb{C}^N$, is said to be *jointly S -sparse* for some $S \in \mathbb{N}$, if they share a common S -sparse support, i.e. if each of the vectors is S -sparse and the non-zero entries occur at the same positions. Stacking the vectors into a matrix columnwise, i.e. writing $\mathbf{X} := [\mathbf{x}^{(1)} \dots \mathbf{x}^{(|\Theta|)}] \in \mathbb{C}^{N \times |\Theta|}$, this means

$$\|\mathbf{X}\|_0 \leq S,$$

where $\|\mathbf{X}\|_0$ counts the number of rows of \mathbf{X} that are not identically zero.

Note that we use the ensemble notation $\{\mathbf{x}^{(\theta)}\}_{\theta \in \Theta}$ and the matrix notation \mathbf{X} interchangeably whenever it is clear from the context. Furthermore, recall the definition of the ℓ_p/ℓ_q -norm $\|\mathbf{X}\|_{p,q}$ of a matrix $\mathbf{X} \in \mathbb{C}^{N \times |\Theta|}$, i.e.

$$\|\mathbf{X}\|_{p,q} = \left(\sum_{i=0}^{N-1} \left(\|\mathbf{X}\|_p^{(i)} \right)^q \right)^{1/q},$$

where $\|\mathbf{X}\|_p^{(i)}$ denotes the ℓ_p -norm of the i -th row of \mathbf{X} (see (2.3)).

Definition 4.4.2. For an ensemble $\mathbf{X} \in \mathbb{C}^{N \times |\Theta|}$, some $S \in \mathbb{N}$ and some $p, q \geq 1$, we define the joint S -term approximation error of \mathbf{X} with respect to the ℓ_p/ℓ_q -norm as

$$\sigma_S(\mathbf{x})_{p,q} := \min_{\|\tilde{\mathbf{X}}\|_0 \leq S} \|\mathbf{X} - \tilde{\mathbf{X}}\|_{p,q}. \quad (4.6)$$

As a measure of the joint sparsity of a signal ensemble we use the ℓ_2/ℓ_1 -norm, i.e.

$$\|\mathbf{X}\|_{2,1} := \sum_{i=0}^{N-1} \|\mathbf{X}\|_2^{(i)}.$$

Here, the ℓ_2 -norms measure if the corresponding index is part of the joint support or not, whereas the sum of the prementioned ℓ_2 -norms can be viewed as an ℓ_1 -norm, which is a measure of sparsity as mentioned in Section 4.2. Therefore, this norm is a valid measure of joint sparsity (see also [120, 121] and the references therein).

In the most general setting, noisy linear measurements of an ensemble of jointly sparse signals $\{\mathbf{x}^{(\theta)}\}_{\theta \in \Theta}$ are taken according to

$$\mathbf{y}^{(\theta)} = \mathbf{\Phi}^{(\theta)} \mathbf{x}^{(\theta)} + \mathbf{z}^{(\theta)}, \quad \theta \in \Theta. \quad (4.7)$$

Since each of the signals is sparse itself, any conventional CS reconstruction strategy like for example BPDN, OMP or CoSaMP could be used to recover each signal individually. With this approach, though, the joint structure of the signals is ignored completely. Most of the algorithms have been adapted to this multiple signal problem, but they typically consider the special case where the measurement procedure is the same for all the different signals, i.e. that $\mathbf{\Phi} \equiv \mathbf{\Phi}^{(\theta)}$ for all $\theta \in \Theta$. In this case we can rewrite the measurement process as $\mathbf{Y} = \mathbf{\Phi} \mathbf{X} + \mathbf{Z}$, where $\mathbf{Y} \in \mathbb{C}^{M \times |\Theta|}$ and $\mathbf{Z} \in \mathbb{C}^{M \times |\Theta|}$ are the columnwise stackings of the $\mathbf{y}^{(\theta)}$ and $\mathbf{z}^{(\theta)}$, respectively. The adaptation of BPDN to this setting is called *multichannel BPDN* (M-BPDN) and is formulated as [122, 123, 124]

$$\arg \min_{\mathbf{X}' \in \mathbb{C}^{N \times |\Theta|}} \|\mathbf{X}'\|_{2,1} \quad \text{subject to} \quad \|\mathbf{\Phi} \mathbf{X}' - \mathbf{Y}\|_2 \leq \epsilon.$$

The multichannel version of OMP is called *simultaneous OMP* (SOMP) [125, 126], and here the signal proxy is calculated as $\mathbf{U}_{(i)} = |\mathbf{\Phi}^H \mathbf{R}_{(i-1)}|$, where $\mathbf{R}_{(i-1)}$ is the residual of the previous step, with $\mathbf{R}_{(0)} = \mathbf{Y}$. Then, the index $n_{(i)}$ corresponding to the row of $\mathbf{U}_{(i)}$ with maximal ℓ_2 -norm is added to the support $\mathcal{S}_{(i-1)}$ in order to obtain $\mathcal{S}_{(i)}$. Finally, a new signal estimate $\mathbf{X}_{(i)} = \arg \min_{\text{supp}_r\{\mathbf{X}'\} \in \mathcal{S}_{(i)}} \|\mathbf{Y} - \mathbf{\Phi} \mathbf{X}'\|_2$ and a new residual $\mathbf{R}_{(i)} = \mathbf{Y} - \mathbf{\Phi} \mathbf{X}_{(i)}$ are calculated, where $\text{supp}_r\{\mathbf{X}'\}$ denotes the rowwise support of \mathbf{X}' (i.e. the indices of the rows of \mathbf{X}' that are not identically zero). The process is stopped if either the maximal number of iterations has been reached or if the Frobenius norm of the residual falls below a prescribed threshold.

For the general case where the measurement matrices $\mathbf{\Phi}^{(\theta)}$ are not equal, the algorithm *distributed compressive sensing SOMP* (DCS-SOMP) has been proposed in [127]. Here a vector $\mathbf{u}_{(i)} = \sum_{\theta \in \Theta} |\mathbf{\Phi}^{(\theta)H} \mathbf{r}_{(i-1)}^{(\theta)}|$ is calculated in step i , where the $\mathbf{r}_{(i-1)}^{(\theta)}$ are the $|\Theta|$ residuals of the previous step, with $\mathbf{r}_{(0)}^{(\theta)} = \mathbf{y}^{(\theta)}$. Then the index $n_{(i)}$ corresponding to the largest entry of $\mathbf{u}_{(i)}$ is added to the support estimate. Afterwards, new signal estimates $\mathbf{x}_{(i)}^{(\theta)}$ and residuals $\mathbf{r}_{(i)}^{(\theta)}$ are calculated exactly like in conventional OMP. The algorithm stops after a fixed number of iterations or if the ℓ_2 -norm of one of the residuals falls below a prescribed threshold.

The CoSaMP algorithm has been adapted to this case as well, where it is called CoSOMP [128]. We prefer to call it *Multichannel CoSaMP* (M-CoSaMP) for the reason

of uniformity within this thesis. The adaption is straight forward, where the signal proxy now is calculated exactly like in SOMP, the support is again understood rowwise, and throughout the algorithm the individual signals from conventional CoSaMP are replaced by the stacked signal matrices.

The theoretical analysis of these algorithms is quite hard. Although it seems reasonable that as the number of jointly sparse signals grows the estimation of the support should become easier, it is impossible to show any performance gain in this case for all possible signals. This is due to the fact that in the worst case scenario all the signals $\mathbf{x}^{(\theta)}$ are equal and therefore no additional information is available as compared to the single signal case. In practice, though, this case hardly ever occurs, and the average case analysis performed for M-BPDN and S-OMP in [126, 129] yields a considerable performance gain for these algorithms with a growing number of jointly sparse signals with very high probability.

As for the computational complexity, it is again hard to give an exact statement for M-BPDN, whereas SOMP can be implemented using $\mathcal{O}(n_{\text{SOMP}}(Mn_{\text{SOMP}} + |\Theta|\Phi))$ operations and M-CoSaMP has complexity $\mathcal{O}(|\Theta|n_{\text{M-CoSaMP}}\Phi)$. Furthermore, following again the implementation of OMP in [102], DCS-SOMP can be implemented using $\mathcal{O}(n_{\text{DCS-SOMP}}(|\Theta|Mn_{\text{DCS-SOMP}} + \sum_{\theta \in \Theta} \Phi^{(\theta)}))$ operations. Here n_{SOMP} , $n_{\text{M-CoSaMP}}$ and $n_{\text{DCS-SOMP}}$ denote the number of SOMP, M-CoSaMP and DCS-SOMP iterations, respectively, and $\mathcal{O}(\Phi)$ denotes the complexity of applying Φ or Φ^H to a vector.

Another way of tackling the multichannel reconstruction problem is by using the well-known fact that it can actually be reinterpreted as a group sparse problem as described in Section 4.3 [117]. To be precise, we first chose an arbitrary but fixed ordering $\{\theta_1, \dots, \theta_{|\Theta|}\}$ of Θ . Then, we define the block-diagonal matrix

$$\Phi := \begin{pmatrix} \Phi^{(\theta_1)} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \Phi^{(\theta_2)} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \Phi^{(\theta_{|\Theta|})} \end{pmatrix} \quad (4.8)$$

of size $|\Theta|M \times |\Theta|N$. Furthermore, we stack the signals $\mathbf{x}^{(\theta)} \in \mathbb{C}^N$, $\mathbf{y}^{(\theta)} \in \mathbb{C}^M$ and $\mathbf{z}^{(\theta)} \in \mathbb{C}^M$ into the vectors $\mathbf{x} := [\mathbf{x}^{(\theta_1)T} \dots \mathbf{x}^{(\theta_{|\Theta|})T}]^T$ of length $|\Theta|N$, $\mathbf{y} := [\mathbf{y}^{(\theta_1)T} \dots \mathbf{y}^{(\theta_{|\Theta|})T}]^T$ of length $|\Theta|M$ and $\mathbf{z} := [\mathbf{z}^{(\theta_1)T} \dots \mathbf{z}^{(\theta_{|\Theta|})T}]^T$ again of length $|\Theta|M$. Then, we can gather the equations $\mathbf{y}^{(\theta)} = \Phi^{(\theta)}\mathbf{x}^{(\theta)} + \mathbf{z}^{(\theta)}$, $\theta \in \Theta$, into one measurement equation $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$.

Now consider the partition $\mathcal{J} := \{I_b\}_{b=0}^{N-1}$ of the set $\{0, \dots, |\Theta|N-1\}$ with groups

$$I_b := \{b + iN | i = 0, \dots, |\Theta|-1\}, \quad b = 0, \dots, N-1. \quad (4.9)$$

If the vectors $\mathbf{x}^{(\theta)}$ are jointly sparse, the stacked signal vector \mathbf{x} is group S -sparse with respect to \mathcal{J} . Therefore, the methodology of GSCS can be utilized for the stacked system. For theoretical analysis it is therefore important to get an idea about the G-RIC of Φ .

Proposition 4.4.3. *Assume that the matrices $\Phi^{(\theta)} \in \mathbb{C}^{M \times N}$, $\theta \in \Theta$, satisfy the RIP of order S with RIC $\delta_S^{(\theta)}$. Then the block-diagonal matrix Φ defined in (4.8) satisfies the G-RIP of order S with respect to the partition \mathcal{J} defined as above with G-RIC $\delta_{S|\mathcal{J}} = \max_{\theta \in \Theta} \delta_S^{(\theta)}$.*

Proof. First note that any $\mathbf{x} \in \mathbb{C}^{|\Theta|N}$ can be viewed as a stacking of $|\Theta|$ vectors $\mathbf{x}^{(\theta)} \in \mathbb{C}^N$. Then we have $\|\mathbf{x}\|_2^2 = \sum_{\theta \in \Theta} \|\mathbf{x}^{(\theta)}\|_2^2$ and $\|\Phi \mathbf{x}\|_2^2 = \sum_{\theta \in \Theta} \|\Phi^{(\theta)} \mathbf{x}^{(\theta)}\|_2^2$. Next, define $\tilde{\delta} := \max_{\theta \in \Theta} \delta_S^{(\theta)}$ for easier notation. If \mathbf{x} is group S -sparse with respect to \mathcal{J} , then each $\mathbf{x}^{(\theta)}$ is S -sparse itself, and therefore

$$\begin{aligned} \|\Phi \mathbf{x}\|_2^2 &= \sum_{\theta \in \Theta} \|\Phi^{(\theta)} \mathbf{x}^{(\theta)}\|_2^2 \leq \sum_{\theta \in \Theta} (1 + \delta_S^{(\theta)}) \|\mathbf{x}^{(\theta)}\|_2^2 \\ &\leq (1 + \max_{\theta \in \Theta} \delta_S^{(\theta)}) \sum_{\theta \in \Theta} \|\mathbf{x}^{(\theta)}\|_2^2 = (1 + \tilde{\delta}) \|\mathbf{x}\|_2^2. \end{aligned} \quad (4.10)$$

Exactly the same reasoning can be used for the lower inequality, i.e. $\|\Phi \mathbf{x}\|_2^2 \geq (1 - \tilde{\delta}) \|\mathbf{x}\|_2^2$. Therefore we have $\delta_{S|\mathcal{J}} \leq \tilde{\delta}$, since $\mathbf{x} \in \Sigma_{S|\mathcal{J}}$ was arbitrary and $\delta_{S|\mathcal{J}}$ by definition is the minimal value satisfying those two inequalities for arbitrary $\mathbf{x} \in \Sigma_{S|\mathcal{J}}$ simultaneously. Now consider the case where only the block $\mathbf{x}^{(\theta_0)}$ of \mathbf{x} with $\theta_0 = \arg \max_{\theta \in \Theta} \delta_S^{(\theta)}$ actually has non-zero entries (note that then technically \mathbf{x} is still group sparse with respect to \mathcal{J} , although even the non-zero group-subvectors have many zero-entries). Then we obviously have $\|\mathbf{x}\|_2 = \|\mathbf{x}^{(\theta_0)}\|_2$ and $\|\Phi \mathbf{x}\|_2 = \|\Phi^{(\theta_0)} \mathbf{x}^{(\theta_0)}\|_2$, and therefore

$$\|\Phi^{(\theta_0)} \mathbf{x}^{(\theta_0)}\|_2^2 = \|\Phi \mathbf{x}\|_2^2 \leq (1 + \delta_{S|\mathcal{J}}) \|\mathbf{x}\|_2^2 = (1 + \delta_{S|\mathcal{J}}) \|\mathbf{x}^{(\theta_0)}\|_2^2.$$

In the same way we obtain $\|\Phi^{(\theta_0)} \mathbf{x}^{(\theta_0)}\|_2^2 \geq (1 - \delta_{S|\mathcal{J}}) \|\mathbf{x}^{(\theta_0)}\|_2^2$. Since $\mathbf{x}^{(\theta_0)}$ is S -sparse, and since these inequalities hold for arbitrary $\mathbf{x}^{(\theta_0)} \in \Sigma_S$, we have $\delta_{S|\mathcal{J}} \geq \delta_S^{(\theta_0)} = \tilde{\delta}$, which finally yields $\delta_{S|\mathcal{J}} = \tilde{\delta}$. \square

With this fact at hand, the theoretical results presented in Section 4.3 can easily be applied to analyze the performance of G-BPDN and G-CoSaMP in this setting.

Finally, for practical implementations it is noteworthy that the application of Φ or Φ^H can be performed block-wise, which reduces its computational complexity to $\mathcal{O}(\Phi) = \mathcal{O}(\sum_{\theta \in \Theta} \Phi^{(\theta)})$.

4.5 Multichannel Group Sparse Compressed Sensing

Combining the notions of group sparsity and joint sparsity we get the following definition.

Definition 4.5.1. A collection of vectors $\mathbf{x}^{(\theta)}$, $\theta \in \Theta$, is called *jointly group S -sparse with respect to the partition* $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$, if the vectors $[\|\mathbf{x}^{(\theta)}[0]\|_2, \dots, \|\mathbf{x}^{(\theta)}[B-1]\|_2]^T$, $\theta \in \Theta$, are jointly S -sparse.

Given noisy measurements $\mathbf{y}^{(\theta)} = \Phi^{(\theta)}\mathbf{x}^{(\theta)} + \mathbf{z}^{(\theta)}$ of such jointly group sparse (or compressible) signals, one could use any conventional CS method like BPDN, CoSaMP or OMP for reconstruction of each signal individually. Also, GSCS methods could be used since each signal is group sparse, or even MCS methods, since the signals are jointly sparse. But to take the full structure into account, we can proceed as described in the previous section, where we identified the multichannel reconstruction problem as a special case of the group sparse problem. We again chose an arbitrary but fixed ordering $\{\theta_1, \dots, \theta_{|\Theta|}\}$ of Θ and consider the stacked vectors $\mathbf{x} := [\mathbf{x}^{(\theta_1)T} \dots \mathbf{x}^{(\theta_{|\Theta|})T}]^T$ of length $|\Theta|N$, $\mathbf{y} := [\mathbf{y}^{(\theta_1)T} \dots \mathbf{y}^{(\theta_{|\Theta|})T}]^T$ of length $|\Theta|M$ and $\mathbf{z} := [\mathbf{z}^{(\theta_1)T} \dots \mathbf{z}^{(\theta_{|\Theta|})T}]^T$ again of length $|\Theta|M$. Then, recalling (4.8), we can again gather the equations $\mathbf{y}^{(\theta)} = \Phi^{(\theta)}\mathbf{x}^{(\theta)} + \mathbf{z}^{(\theta)}$, $\theta \in \Theta$, into the one measurement equation $\mathbf{y} = \Phi\mathbf{x} + \mathbf{z}$. If the signals $\mathbf{x}^{(\theta)}$ are jointly *group S -sparse with respect to \mathcal{J}* we can consider the partition $\tilde{\mathcal{J}} := \{\tilde{I}_b\}_{b=0}^{B-1}$ of the set $\{0, \dots, |\Theta|N-1\}$ with groups

$$\tilde{I}_b := \{k + iN | k \in I_b, i = 0, \dots, |\Theta|-1\}, \quad b = 0, \dots, B-1. \quad (4.11)$$

Then, the stacked signal vector \mathbf{x} is *group S -sparse with respect to $\tilde{\mathcal{J}}$* . Therefore, the whole available structure can be taken into account by again utilizing the methodology of GSCS for the stacked system. Note that Proposition 4.4.3 can easily be adapted to this setting.

Corollary 4.5.2. Assume that the matrices $\Phi^{(\theta)} \in \mathbb{C}^{M \times N}$, $\theta \in \Theta$, satisfy the G -RIP of order S with respect to the same partition \mathcal{J} with G -RIC $\delta_{S|\mathcal{J}}^{(\theta)}$. Then the block-diagonal

matrix Φ defined in (4.8) satisfies the *G-RIP* of order S with respect to the partition $\tilde{\mathcal{J}}$ as defined above with *G-RIC* $\delta_{S|\tilde{\mathcal{J}}} = \max_{\theta \in \Theta} \delta_{S|\mathcal{J}}^{(\theta)}$.

Proof. For the proof we set $\tilde{\delta} := \max_{\theta \in \Theta} \delta_{S|\mathcal{J}}^{(\theta)}$ and note that if \mathbf{x} is group S -sparse with respect to $\tilde{\mathcal{J}}$, then each $\mathbf{x}^{(\theta)}$ is group S -sparse with respect to \mathcal{J} , and therefore (4.10) again holds. Then we have $\delta_{S|\tilde{\mathcal{J}}} \leq \tilde{\delta}$. Equality again follows from choosing a vector \mathbf{x} for which only one subvector $\mathbf{x}^{(\theta_0)}$ with $\theta_0 = \arg \max_{\theta \in \Theta} \delta_{S|\mathcal{J}}^{(\theta)}$ is non-zero, and following the exact same reasoning as in the proof of Proposition 4.4.3. \square

4.6 Modified Compressed Sensing

In a general sparse reconstruction setting the support of the unknown signal \mathbf{x} is completely unknown. However, there are practical applications where some prior information about this support is available. It might, for example, be known that the signals of interest must have significant entries in some region due to physical constraints, whereas other regions are completely undetermined. In image analysis it might be known that the image that is to be reconstructed belongs to a certain class of images which can yield some information about the support. In this thesis we consider the setting where a time-varying signal is to be reconstructed at several different points in time. If the support of the signal changes only slowly between two consecutive points in time, we can use the estimated support from one point in time, or at least a part of it, as support information for the next one. To take the prior support information into account, all the methods introduced in the previous sections can easily be adapted. Because of the nomenclature in the first papers describing this setting we subsume these methods under the name of *modified CS* (MOD-CS) [130, 131]. In the following we merely explain the changes that have to be made in the conventional CS methods. In the same way the GSCS and MCS methods can then be adapted straight forwardly.

To do so we consider the following setting. Let $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ be noisy measurements of an unknown S -sparse signal $\mathbf{x} \in \mathbb{C}^N$, let $\mathcal{S} \subseteq \text{supp}\{\mathbf{x}\}$ be the known part of the support of \mathbf{x} , and let \mathcal{S}^c be its complement within $\{0, \dots, N-1\}$. With this notation, the *modified BPDN* problem that was introduced in [131], is given as

$$\arg \min_{\mathbf{x}' \in \mathbb{C}^N} \|\mathbf{x}'|_{\mathcal{S}^c}\|_1 \quad \text{subject to} \quad \|\mathbf{y} - \Phi \mathbf{x}'\|_2 \leq \epsilon.$$

Here we look for a signal that is as sparse as possible outside of \mathcal{S} . As for the reconstruction quality the following result has been obtained in [132].

Theorem 4.6.1. *Let noisy linear measurements $\mathbf{y} = \Phi \mathbf{x} + \mathbf{z}$ of a signal $\mathbf{x} \in \mathbb{C}^N$ with $\Phi \in \mathbb{C}^{M \times N}$, and a set $\mathcal{S} \subseteq \{0, \dots, N-1\}$ of size $|\mathcal{S}| = S_0$ be given. If Φ satisfies the RIP of order $S_0 + 2S$ with RIC $\delta_{2S}^2 + \delta_{S_0+2S} < 1$, then the solution $\hat{\mathbf{x}}$ of MOD-BPDN satisfies*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_S(\mathbf{x} - \mathbf{x}|_{\mathcal{S}})_1}{\sqrt{S}} + c_1 \epsilon,$$

where $c_0 := 2 \frac{1+\mu-\delta_{S_0+2S}}{1-\delta_{S_0+2S}-\mu}$ and $c_1 := 4 \frac{\sqrt{1+\delta_{S_0+2S}}}{1-\delta_{S_0+2S}-\mu}$, with $\mu := \sqrt{\delta_{S_0+2S}^2 + \delta_{2S}^2}$.

Here, $\mathbf{x}|_{\mathcal{S}}$ denotes the vector coinciding with \mathbf{x} on \mathcal{S} and vanishing everywhere else. Note that in this result the term $\sigma_S(\mathbf{x} - \mathbf{x}|_{\mathcal{S}})_1$ can be much smaller than the corresponding term $\sigma_S(\mathbf{x})_1$ in Theorem 4.2.5 analyzing conventional BPDN, but that at the same time the assumptions on Φ get more restrictive. Therefore this theoretical result is of limited use in practice.

The adaption of OMP is straight forward. In the *modified OMP* (MOD-OMP) we start with an initialization step, where an initial signal estimate $\mathbf{x}_{(0)}$ is calculated on the partially known support, i.e. $\mathbf{x}_{(0)} := \arg \min_{\text{supp}\{\mathbf{x}'\} \subseteq \mathcal{S}} \|\mathbf{y} - \Phi \mathbf{x}'\|_2$, and the residual $\mathbf{r}_{(0)} = \mathbf{y} - \Phi \mathbf{x}_{(0)}$ as well as the support set $\mathcal{S}_{(0)} = \mathcal{S}$ are initialized. After that, we proceed just like in conventional OMP, where only the maximal number of iterations is reduced by $|\mathcal{S}|$. MOD-OMP has, to the best of our knowledge, firstly been analyzed in [133] in a different context, where it is called *Orthogonal Greedy Algorithm*. In this work an exact recovery result for the noiseless case has been obtained. Also, the arguments used in the proof of the exact recovery result for conventional OMP in [134] can easily be adapted to yield an analogous result for MOD-OMP. Unfortunately, there are no results for the case of noisy measurements yet. The computational complexity of MOD-OMP clearly reduces to $\mathcal{O}(M(n_{\text{MOD-OMP}} + |\mathcal{S}|)^2 + n_{\text{MOD-OMP}} \Phi)$, where $n_{\text{MOD-OMP}}$ denotes the number of MOD-OMP iterations.

The adaption of CoSaMP to the MOD-CS setting is again very simple. In the first iteration we simply augment the temporary support estimate $\tilde{\mathcal{S}}_{(1)}$ with the partially known support set \mathcal{S} . Then, at each iteration $i \geq 1$, we calculate a temporary signal estimate $\tilde{\mathbf{x}}_{(i)}$ as usual, and define the support estimate $\mathcal{S}_{(i)}$ as the augmentation of \mathcal{S} with the $S - |\mathcal{S}|$ indices of \mathcal{S}^c corresponding to the entries of $\tilde{\mathbf{x}}_{(i)}$ with largest absolute values.

Finally, the new signal estimate $\mathbf{x}_{(i)}$ and the residual $\mathbf{r}_{(i)}$ are calculated as usual. With this simple adaption we make sure that \mathcal{S} is part of the support of each signal estimate $\mathbf{x}_{(i)}$. For obvious reasons we dub this algorithm the *modified CoSaMP* (MOD-CoSaMP). To the best of our knowledge MOD-CoSaMP has not appeared in the literature yet. Its computational complexity is the same as the one for conventional CoSaMP, namely $\mathcal{O}(n_{\text{MOD-CoSaMP}}\Phi)$, where $n_{\text{MOD-CoSaMP}}$ denotes the maximal number of iterations.

Note that in the special case where $\mathcal{S} = \emptyset$, i.e. where no prior support information is available, MOD-BPDN, MOD-OMP and MOD-CoSaMP obviously coincide with their conventional counterparts. Finally, note that for a signal \mathbf{x} that is only compressible almost any set \mathcal{S} is part of the support of \mathbf{x} . Thus, we cannot expect to gain anything if \mathcal{S} is allowed to be an arbitrary part of the support in this setting. It might make things even worse if, for example, \mathbf{x} only has small entries on \mathcal{S} . Nevertheless, we can still expect improved performance if the set \mathcal{S} is part of the essential support of \mathbf{x} , i.e. the part that corresponds to its "large" entries.

Compressive Channel Estimation in SISO Systems

5.1 Conventional Compressive Channel Estimation

In this section we explain how the methodology of CS can be utilized to estimate the mobile radio channel in wireless communication systems. The idea is not new, it was first introduced by different authors independently in [1, 17, 18] in 2008. CS techniques have also been applied to various other settings, such as ultra-wideband (UWB) systems ([135, 136]) or radar systems ([137, 138, 139]). For the general case of doubly selective channels in wireless systems a good survey have been given, for example, in [140]. Note that in the literature there does not appear to be an expression subsuming all the channel estimation techniques using CS methods. We choose to use the expression "Compressive Channel Estimation" (CCE) to express that *compressive* sampling methods are used for channel estimation, although there is no compression whatsoever performed in those estimation techniques.

This section is organized as follows. We start by presenting the general method introduced in [1] and analyzed in more detail in [3] in Section 5.1.1, and extend this analysis by giving a new error estimate for two of the CS methods in Theorem 5.1.1. In Section 5.1.2 we analyze the computational complexity of the presented method, which has, to the best of our knowledge, not been done before. Then, in Section 5.1.3, we explain in what sense typical wireless channels can be considered compressible following

our analysis given in [1, 3]. Section 5.1.4 is devoted to explaining how the use of a different basis than the complex exponentials can greatly improve the performance of the compressive estimator, and how such a basis can be found (again following [2, 3]). Finally, extensive simulation results presented in Section 5.1.5 demonstrate the superior performance of the presented compressive channel estimator as compared to conventional LS methods.

5.1.1 The Method

As explained in Section 3.5 the basic task of channel estimation is to identify any of the different channel representations. Our method allows the calculation of the channel coefficients $H_{l,k}$ in (3.13) via the 2D DFT relation (3.16), i.e. the coefficients $F_{m,i}$ in (3.17). As mentioned in Section 3.3.3 we only consider underspread channels. For such channels and practical transmit and receive pulses, these coefficients $F_{m,i}$ are effectively supported in some small rectangular region about the origin. We therefore assume the support of $F_{m,i}$ to be contained in $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ for some constants D and J chosen such that $\Delta K := K/D$ and $\Delta L := L/J$ are integers, and J (and therefore also L) is even for mathematical convenience. Then, because of the 2D DFT relation (3.16), it is sufficient to know the values of the channel coefficients $H_{l,k}$ (or equivalently the 2D DFT coefficients $F_{m,i}$) at JD positions to find the values of $H_{l,k}$ at all time-frequency positions (l, k) for $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$. Therefore we define a *subsampled time-frequency grid* $\mathcal{G} := \{(\lambda\Delta L, \kappa\Delta K) \mid \lambda = 0, \dots, J-1; \kappa = 0, \dots, D-1\}$, on which (3.16) becomes

$$H_{\lambda\Delta L, \kappa\Delta K} = \sum_{m=0}^{K-1} \sum_{i=-L/2}^{L/2-1} F_{m,i} e^{-j2\pi(\frac{\kappa\Delta K m}{K} - \frac{\lambda\Delta L i}{L})} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} F_{m,i} e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}. \quad (5.1)$$

Note that of course the limiting cases $D = K$ and/or $J = L$, i.e. no subsampling in time and/or frequency direction, are included. The key observation for the use of CS methods for channel estimation is that the 2D DFT coefficients $F_{m,i}$ are compressible. This fact is explained in detail in Section 5.1.3, where it is also shown that their compressibility is strongly affected by the *leakage effect*. Analogously to the BEMs mentioned in Section 3.5, we therefore generalize (5.1) to

$$H_{\lambda\Delta L, \kappa\Delta K} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} G_{m,i} u_{m,i}[\lambda, \kappa], \quad (5.2)$$

where $\{u_{m,i}[\lambda, \kappa]\}$ is an orthonormal 2D basis appropriately chosen such that the expansion coefficients $G_{m,i}$ are even "more compressible" than the 2D DFT coefficients $F_{m,i}$. In Section 5.1.4 we explain one way of constructing such a basis. Obviously the 2D DFT expansion (5.1) is a special case of (5.2) with basis $u_{m,i}[\lambda, \kappa] = (1/\sqrt{JD})e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}$ and coefficients $G_{m,i} = \sqrt{JD}F_{m,i}$. Now let us define the JD -dimensional vectors \mathbf{h}_Δ , \mathbf{g} and $\mathbf{u}_{m,i}$ as $[\mathbf{h}_\Delta]_{\kappa J + \lambda} := H_{\lambda\Delta L, \kappa\Delta K}$, $[\mathbf{g}]_{(i+J/2)D+m} := G_{m,i}$, and $[\mathbf{u}_{m,i}]_{\kappa J + \lambda} := u_{m,i}[\lambda, \kappa]$, respectively (with $\lambda = 0, \dots, J-1$; $\kappa = 0, \dots, D-1$; $m = 0, \dots, D-1$; $i = -J/2, \dots, J/2-1$). This corresponds to a columnwise stacking of $H_{\lambda\Delta L, \kappa\Delta K}$, $G_{m,i}$, and $u_{m,i}[\lambda, \kappa]$, respectively, each viewed as a $J \times D$ matrix. Furthermore, we define the $JD \times JD$ matrix \mathbf{U} as $[\mathbf{U}]_{\kappa J + \lambda, (i+J/2)D+m} := u_{m,i}[\lambda, \kappa]$. Here the $((i+J/2)D+m+1)$ -th column corresponds to the stacked basis vector $\mathbf{u}_{m,i}$, and thus \mathbf{U} is a unitary matrix. With these definitions at hand we can rewrite (5.2) as

$$\mathbf{h}_\Delta = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} G_{m,i} \mathbf{u}_{m,i} = \mathbf{U} \mathbf{g}. \quad (5.3)$$

For pilot-aided channel estimation, as explained in Section 3.5, first a pilot set \mathcal{P} of size $Q := |\mathcal{P}|$ is chosen. Following the LS-approach described in that section, estimates $\hat{H}_{l,k}$ of the channel coefficients at all the pilot positions $(l, k) \in \mathcal{P}$ can be calculated by $\hat{H}_{l,k} = r_{l,k}/p_{l,k}$ (see (3.18)), where $r_{l,k}$ are the received symbols and $p_{l,k}$ are the pilot symbols. If we now choose the pilot set \mathcal{P} as a subset of the subsampled time-frequency grid \mathcal{G} , we get an estimate $\hat{\mathbf{h}}_{(p)}$ of the subvector $\mathbf{h}_{(p)} := (\mathbf{h}_\Delta)|_{\mathcal{P}}$ of \mathbf{h}_Δ constituted of the entries of \mathbf{h}_Δ corresponding to \mathcal{P} , with $\hat{\mathbf{h}}_{(p)} = \mathbf{h}_{(p)} + \mathbf{z}_{(p)}$ (again see (3.18)), where $\mathbf{z}_{(p)}$ has entries $\tilde{z}_{l,k}/p_{l,k}$ with $(l, k) \in \mathcal{P}$. Constraining the matrix \mathbf{U} to the same rows corresponding to \mathcal{P} and denoting the resulting matrix by $\mathbf{U}_{(p)}$ (which then is a $Q \times JD$ matrix), we obtain $\mathbf{h}_{(p)} = \mathbf{U}_{(p)} \mathbf{g}$, which in turn yields $\hat{\mathbf{h}}_{(p)} = \mathbf{U}_{(p)} \mathbf{g} + \mathbf{z}_{(p)}$. Since the expansion coefficients $G_{m,i}$, and thus the vector \mathbf{g} , were assumed compressible, this can be identified as an instance of the basic CS measurement equation (4.1). As explained in Section 4.2 we therefore have to examine the "measurement matrix" $\mathbf{U}_{(p)}$ closer. This matrix is constructed from a unitary matrix by choosing the rows corresponding to the pilot positions. Thus, following Theorem 4.2.6, if we choose the pilot positions uniformly at random (from the grid \mathcal{G}), and if we choose the number of pilots Q large enough, the renormalized matrix $\sqrt{JD/Q} \mathbf{U}_{(p)}$ will have a small RIC with very high probability, which typically yields good reconstruction properties. Therefore, we rewrite

the measurement equation above as

$$\hat{\mathbf{h}}_{(p)} = \mathbf{\Phi}\mathbf{x} + \mathbf{z}_{(p)}, \quad (5.4)$$

where $\mathbf{\Phi} := \sqrt{JD/Q}\mathbf{U}_{(p)}$ and $\mathbf{x} = \sqrt{Q/JD}\mathbf{g}$.

Now we can formulate the compressive channel estimator. Let \mathcal{P} be a pilot set that is chosen uniformly at random from the grid \mathcal{G} and communicated to the receiver once and for all *before* the start of data transmission (it stays fixed therein).

Step 1. Calculate channel estimates at the pilot positions and stack them in order to obtain the measurement equation (5.4).

Step 2. Run any CS algorithm to obtain an estimate $\hat{\mathbf{x}}$ of \mathbf{x} , and rescale $\hat{\mathbf{x}}$ with $\sqrt{JD/Q}$ for an estimate $\hat{\mathbf{g}}$ of \mathbf{g} , the entries of which are exactly the expansion coefficients $G_{m,i}$.

Step 3. Calculate estimates of the subsampled channel coefficients $H_{\lambda\Delta L, \kappa\Delta K}$ from (5.2).

Step 4. Invert (5.1) to obtain estimates of the 2D DFT coefficients $F_{m,i}$ for $m = 0, \dots, D-1$ and $i = -J/2, \dots, J/2-1$. Note that by assumption $F_{m,i}$ vanishes for all other indices.

Step 5. Calculate estimates of all the channel coefficients $H_{l,k}$ by using the 2D DFT expansion (3.16).

Note that in the special case where the 2D DFT basis is used we have $G_{m,i} = \sqrt{JD}F_{m,i}$, and therefore Steps 3 and 4 can be omitted.

Concerning the quality of channel estimation we can say the following. Assume that the expansion coefficients $G_{m,i}$, and therefore also the vector $\mathbf{x} = \sqrt{Q/JD}\mathbf{g}$, are compressible in the sense explained in Section 4.2. Then we obtain the following result regarding the estimation error of the compressive channel estimator, which obviously depends on the CS recovery method that is used. We only present results for BPDN and CoSaMP here. To the best of our knowledge this is a new result that has not yet appeared in the literature.

Theorem 5.1.1. *Let $\mathcal{S} \subseteq \{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ be any set of size $|\mathcal{S}| = S$. Furthermore, let*

$$C_{G,\mathcal{S}} := \sum_{(m,i) \notin \mathcal{S}} |G_{m,i}|,$$

as well as $\epsilon_n := \|\mathbf{z}_{(p)}\|_2$ and $\epsilon > 0$.

1. Take BPDN to be the CS reconstruction method used in Step 2 of the compressive channel estimator. If Φ satisfies the RIP with RIC $\delta_{2S} < \sqrt{2}-1$, and if $\epsilon_n < \epsilon$, we can bound the estimation error as

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq C'_0 S^{-1/2} C_{G,S} + C'_1 \epsilon, \quad (5.5)$$

with the constants $C'_0 := c_0 \sqrt{\frac{LK}{JD}}$ and $C'_1 := c_1 \sqrt{\frac{LK}{Q}}$, where c_0 and c_1 are the constants from Theorem 4.2.5.

2. Use CoSaMP with i steps for CS reconstruction in Step 2. If the RIC δ_{4S} of Φ satisfies $\delta_{4S} < 0.1$ and if $\epsilon_n < \epsilon$, then

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq \tilde{C}_0 (1 + S^{-1/2}) C_{G,S} + \tilde{C}_1 \epsilon + \tilde{C}_2,$$

with $\tilde{C}_0 := 20 \sqrt{\frac{LK}{JD}}$, $\tilde{C}_1 := 20 \sqrt{\frac{LK}{Q}}$ and $\tilde{C}_2 := 2^{-i} \sqrt{\frac{LK}{JD}} \left(\sum_{(m,i)} |G_{m,i}|^2 \right)^{1/2}$. Note that since the number of CoSaMP iterations can be chosen arbitrarily, \tilde{C}_2 can be made arbitrarily small.

Proof. We begin by defining $\mathbf{h} := \text{vec}\{H_{l,k}\}_{l,k}$, where $\text{vec}\{\cdot\}$ denotes columnwise stacking of the $L \times K$ "matrix" given by the coefficients $H_{l,k}$, i.e. $[\mathbf{h}]_{kL+l} = H_{l,k}$, $\mathbf{f} := \text{vec}\{F_{m,i}\}_{m,i}$, i.e. $[\mathbf{f}]_{(i+L/2)K+m} = F_{m,i}$, and the unitary matrix \mathbf{U}_F with entries $[\mathbf{U}_F]_{kL+l, (i+L/2)K+m} = (1/\sqrt{LK}) e^{-j2\pi(\frac{km}{K} - \frac{li}{L})}$ (here $l = 0, \dots, L-1$; $k = 0, \dots, K-1$; $m = 0, \dots, K-1$; $i = -L/2, \dots, L/2-1$). Then, (3.16) can be rewritten as $\mathbf{h} = \sqrt{LK} \mathbf{U}_F \mathbf{f}$, which yields $\|\mathbf{h}\|_2 = \sqrt{LK} \|\mathbf{f}\|_2$ since \mathbf{U}_F is a unitary matrix.

Next, we define $\tilde{\mathbf{f}}$ as the restriction of \mathbf{f} to $[0, \dots, D-1] \times [-J/2, \dots, J/2-1]$, i.e. $[\tilde{\mathbf{f}}]_{(i+J/2)D+m} = F_{m,i}$, and the "small" version $\tilde{\mathbf{U}}_F$ of \mathbf{U}_F via $[\tilde{\mathbf{U}}_F]_{\kappa J+\lambda, (i+J/2)D+m} = (1/\sqrt{JD}) e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}$ (here $\lambda = 0, \dots, J-1$; $\kappa = 0, \dots, D-1$; $m = 0, \dots, D-1$; $i = -J/2, \dots, J/2-1$). Then we can rewrite (5.1) as $\mathbf{h}_\Delta = \sqrt{JD} \tilde{\mathbf{U}}_F \tilde{\mathbf{f}}$, which now yields $\|\mathbf{h}_\Delta\|_2 = \sqrt{JD} \|\tilde{\mathbf{f}}\|_2 = \sqrt{JD} \|\mathbf{f}\|_2$, since again $\tilde{\mathbf{U}}_F$ is unitary and by assumption the restriction of \mathbf{f} to $\tilde{\mathbf{f}}$ only deletes zero-entries. On the other hand, following (5.3), we have $\mathbf{h}_\Delta = \mathbf{U} \mathbf{g}$, which yields $\|\mathbf{h}_\Delta\|_2 = \|\mathbf{g}\|_2$ by the unitarity of \mathbf{U} . Putting all this together, we thus obtain

$$\|\mathbf{h}\|_2 = \sqrt{LK} \|\mathbf{f}\|_2 = \sqrt{\frac{LK}{JD}} \|\mathbf{h}_\Delta\|_2 = \sqrt{\frac{LK}{JD}} \|\mathbf{g}\|_2.$$

Because of the way the channel estimates $\hat{H}_{l,k}$ are calculated from the estimated expansion coefficients $\hat{G}_{m,i}$, the exact same arguments as before yield $\|\mathbf{h} - \hat{\mathbf{h}}\|_2 = \sqrt{LK/JD} \|\mathbf{g} - \hat{\mathbf{g}}\|_2$, where $\hat{\mathbf{f}}$ and $\hat{\mathbf{g}}$ denote the estimates of \mathbf{f} and \mathbf{g} , respectively. We thus have

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} = \|\mathbf{h} - \hat{\mathbf{h}}\|_2 = \sqrt{\frac{LK}{JD}} \|\mathbf{g} - \hat{\mathbf{g}}\|_2. \quad (5.6)$$

Furthermore, since $\mathbf{g} = \sqrt{JD/Q} \mathbf{x}$ and $\hat{\mathbf{g}} = \sqrt{JD/Q} \hat{\mathbf{x}}$, we have $\|\mathbf{g} - \hat{\mathbf{g}}\|_2 = \sqrt{JD/Q} \|\mathbf{x} - \hat{\mathbf{x}}\|_2$, which yields

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq \sqrt{\frac{LK}{Q}} \|\mathbf{x} - \hat{\mathbf{x}}\|_2. \quad (5.7)$$

Next, let us consider part 1 of the theorem. Under the assumptions on Φ and ϵ_n Theorem 4.2.5 yields

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_S(\mathbf{x})_1}{\sqrt{S}} + c_1 \epsilon$$

with the constants given therein. Now, we define the one-to-one mapping

$$\begin{aligned} \mathbf{S} : \{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\} &\rightarrow \{0, \dots, JD-1\}, \\ \mathbf{S}(m, i) &:= (i + J/2)D + m, \end{aligned} \quad (5.8)$$

which actually corresponds to the columnwise stacking operation $\mathbf{g} := \text{vec}\{G_{m,i}\}_{m,i}$, i.e. $[\mathbf{g}]_{(i+J/2)D+m} = [\mathbf{g}]_{\mathbf{S}(m,i)} = G_{m,i}$. Then, we can obviously write

$$\sigma_S(\mathbf{x})_1 \leq \sum_{(m,i) \notin \mathcal{S}} |[\mathbf{x}]_{\mathbf{S}(m,i)}| = \sqrt{\frac{Q}{JD}} \sum_{(m,i) \notin \mathcal{S}} |G_{m,i}|, \quad (5.9)$$

where the inequality is due to the definition of $\sigma_S(\mathbf{x})_1$. Finally, we thus get

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq c_0 \sqrt{\frac{LK}{JD}} \frac{C_{G,S}}{\sqrt{S}} + c_1 \sqrt{\frac{LK}{Q}} \epsilon.$$

Setting $C'_0 := c_0 \sqrt{LK/JD}$ and $C'_1 := c_1 \sqrt{LK/Q}$ gives the desired result.

For part 2 the assumptions on Φ and ϵ_n together with Theorem 4.2.8 yield

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq 20\sigma_S(\mathbf{x})_2 + \frac{20\sigma_S(\mathbf{x})_1}{\sqrt{S}} + 20\epsilon + 2^{-i} \|\mathbf{x}\|_2,$$

which then gives

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq 20\sqrt{\frac{LK}{Q}}(1 + S^{-1/2})\sigma_S(\mathbf{x})_1 + 20\sqrt{\frac{LK}{Q}}\epsilon + \sqrt{\frac{LK}{Q}}2^{-i}\|\mathbf{x}\|_2,$$

where we have used that $\|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1$, and therefore $\sigma_S(\mathbf{x})_2 \leq \sigma_S(\mathbf{x})_1$. Finally, using (5.9) and setting the constants $\tilde{C}_0 := 20\sqrt{LK/JD}$, $\tilde{C}_1 := 20\sqrt{LK/Q}$ and $\tilde{C}_2 := 2^{-i}\sqrt{LK/JD}\left(\sum_{(m,i)} |G_{m,i}|^2\right)^{1/2}$ finishes the proof. \square

Note that the theorem holds for any set $\mathcal{S} \subseteq \{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$, but that $C_{G,\mathcal{S}}$ will only be small if \mathcal{S} covers the essential support of the coefficients $G_{m,i}$. In the case where the 2D DFT basis is used, $C_{G,\mathcal{S}}$ somehow characterizes the leakage occurring if \mathcal{S} is chosen according to the analysis given in Section 5.1.3. Furthermore, note that the second term in the estimates above is due to the noise, whereas the third term in the error estimate of part 2 of the theorem can be made arbitrarily small since the number of CoSaMP iterations can be chosen arbitrarily (although the computational cost increases with an increasing number of iterations, see Section 4.2).

As mentioned above, this result gives a strict estimate for the estimation error of the compressive channel estimator. Nevertheless, it is well known that in most practical scenarios the CS methods actually give much better results than suggested by the provable error estimates. Also, many measurement matrices work well for CS reconstruction although they do not suffice the conditions needed for the provable results.

5.1.2 Computational Complexity

For the analysis of the computational complexity of the compressive estimator we use the \mathcal{O} -notation explained in Section 2.5. We proceed step by step and only count multiplicative operations.

Step 1. To calculate channel estimates following (3.18) we need Q divisions.

Step 2. Here the computational complexity strongly depends on the CS algorithm that is used, which we denote by $\mathcal{O}(\text{CS})$. The rescaling takes $\mathcal{O}(JD)$ operations.

Step 3. For the calculation of the estimates of the subsampled channel coefficients $H_{\lambda\Delta L, \kappa\Delta K}$ from (5.2), or equivalently (5.3), one vector-matrix product with the unitary $JD \times JD$ matrix \mathbf{U} has to be performed, which typically comes at the cost

of $(JD)^2$ multiplications. Note that some matrices \mathbf{U} allow a faster implementation of the vector-matrix product.

Step 4. Following the proof of Theorem 5.1.1 and utilizing the same notation as therein, (5.1) can be rewritten as $\mathbf{h}_\Delta = \sqrt{JD}\tilde{\mathbf{U}}\tilde{\mathbf{F}}\tilde{\mathbf{f}}$. Inverting this equation can be done very efficiently by using the IFFT in $\mathcal{O}(JD \log(JD))$ operations.

Step 5. Using the same reasoning as before the 2D DFT expansion (3.16), which can be rewritten as $\mathbf{h} = \sqrt{LK}\mathbf{U}_F\mathbf{f}$, can be implemented with just $\mathcal{O}(LK \log(LK))$ operations by using the FFT.

All in all, the compressive channel estimator can be implemented using

$$\mathcal{O}(\text{CS}) + \mathcal{O}((JD)^2) + \mathcal{O}(LK \log(LK)) \quad (5.10)$$

multiplicative operations. In a typical setting the third term will be dominated by the second one by far. As explained in Section 4.2 it is hard to give explicit bounds for the computational complexity of the CS algorithms because there are so many different implementations available. Nevertheless, using the bounds for the implementations mentioned therein, and keeping in mind that the measurement matrix Φ in (5.4) is a $Q \times JD$ matrix, we have $\mathcal{O}(\text{CS}) = \mathcal{O}(Q(JD)^2)$ for BPDN, $\mathcal{O}(\text{CS}) = \mathcal{O}(n_{\text{OMP}}(Qn_{\text{OMP}} + \Phi))$ for OMP, and $\mathcal{O}(\text{CS}) = \mathcal{O}(n_{\text{CoSaMP}}(JD)^2)$ for CoSaMP, where n_{OMP} and n_{CoSaMP} denote the numbers of OMP and CoSaMP iterations, respectively. All in all, in a typical setting $\mathcal{O}(\text{CS})$ will dominate the other terms, no matter which of these CS algorithms is used.

Finally, note that if the 2D DFT basis is used throughout, Steps 3 and 4 can be omitted, which eliminates the term $\mathcal{O}((JD)^2)$. More importantly, the complexity of the CS algorithms typically is reduced too, because then FFT-methods can be used to calculate the vector-matrix products.

5.1.3 Delay-Doppler Sparsity

In this section we analyze the compressibility of the 2D DFT expansion coefficients $F_{m,i}$ in (3.16), which is the fundamental assumption underlying the compressive channel estimator presented in the previous section. To do so, recall the representation (3.17) of these coefficients, namely

$$F_{m,i} = \sum_{q=0}^{N-1} S_h[m, i + qL] A_{\gamma,g}^* \left(m, \frac{i + qL}{N_0} \right)$$

for the discrete-delay variables $m = 0, \dots, K-1$ and the discrete Doppler shift variable $i = -L/2, \dots, L/2-1$. For typical transmit and receive pulses g and γ the absolute value of the cross-ambiguity function $A_{\gamma,g}$ looks somewhat like a plateau function. For classical CP-OFDM, for example, with transmit and receive pulses given in (3.5), we get

$$A_{\gamma,g}(m, \xi) = \frac{1}{K} \sum_{n=n_m}^{N_m} e^{-j2\pi\xi n} \quad (5.11)$$

for $n_m := \max(0, m - L_{cp})$ and $N_m := \min(K-1, K-1+m)$, where we define the sum to be 0 if $N_m < n_m$. For each ξ its absolute value looks like a trapezoid with maximum 1. Therefore, the support of $F_{m,i}$ is governed by the support of the discrete-delay-Doppler spreading function $S_h[m, i]$, which we analyze in the following.

To do so, recall the model (3.8) of the impulse response of a time-varying channel. As an even simpler model we approximate the number of propagation paths $P(t)$ corresponding to $P(t)$ specular scatterers, as well as the attenuation parameter $\eta_p(t)$, the time delay $\tau_p(t)$ and the Doppler shift $\nu_p(t)$ by their initial values P , η_p , τ_p and ν_p , i.e.

$$h(t, \tau) = \sum_{p=1}^P \eta_p \delta(\tau - \tau_p) e^{j2\pi\nu_p t}, \quad (5.12)$$

which often is a good approximation to real mobile radio channels [141, 142]. Note that this model is only used to analyze the sparsity of the channel in the delay-Doppler region as well as to motivate the basis optimization techniques described in Section 5.1.4, but it is not necessary for the compressive channel estimator. Combining equations (3.14), (3.11) and (5.12), we can calculate the discrete-delay-Doppler spreading function as

$$\begin{aligned} S_h[m, i] &= \frac{1}{N_0} \sum_{n=0}^{N_0-1} e^{-j2\pi\frac{in}{N_0}} \iint_{\mathbb{R}\mathbb{R}} \sum_{p=1}^P \eta_p \delta(\tau - \tau_p) e^{j2\pi\nu_p(t+nT_s)} f_1(t+nT_s - \tau) f_2(-t) dt d\tau \\ &= \frac{1}{N_0} \sum_{n=0}^{N_0-1} e^{-j2\pi\frac{n}{N_0}(i - \nu_p T_s N_0)} \sum_{p=1}^P \eta_p \int_{\mathbb{R}} e^{j2\pi\nu_p t} f_2(-t) \int_{\mathbb{R}} f_1(t+nT_s - \tau) \delta(\tau - \tau_p) d\tau dt \\ &= \sum_{p=1}^P \eta_p \int_{\mathbb{R}} e^{-j2\pi\nu_p t} f_1\left(T_s\left(m - \frac{\tau_p}{T_s}\right) - t\right) f_2(t) dt \frac{1}{N_0} \sum_{n=0}^{N_0-1} e^{-j2\pi\frac{n}{N_0}(i - \nu_p T_s N_0)}, \end{aligned}$$

where first the linearity of the integral has been used and the order of the (finite) sums was exchanged, and then the integral with regard to τ was evaluated (by using (2.8)), as well as the variable t of integration was replaced by $-t$ in the other integral. Next,

we define the functions

$$\phi_p(x) := \int_{\mathbb{R}} e^{-j2\pi\nu_p t} f_1(T_s x - t) f_2(t) dt \quad \text{and} \quad (5.13)$$

$$\begin{aligned} \psi(y) &:= \frac{1}{N_0} e^{j\pi \frac{y}{N_0} (N_0-1)} \sum_{n=0}^{N_0-1} e^{-j2\pi \frac{y}{N_0} n} \\ &= \frac{\sin(\pi y)}{N_0 \sin(\pi y/N_0)}. \end{aligned} \quad (5.14)$$

Then we can rewrite the discrete-delay-Doppler spreading function as

$$S_h[m, i] = \sum_{p=1}^P \eta_p e^{j\pi (\nu_p T_s - \frac{i}{N_0}) (N_0-1)} \Lambda_p[m, i], \quad (5.15)$$

with the *shifted leakage kernels*

$$\Lambda_p[m, i] := \phi_p\left(m - \frac{\tau_p}{T_s}\right) \psi\left(i - \nu_p T_s N_0\right). \quad (5.16)$$

Therefore, analyzing the sparsity of $S_h[m, i]$ reduces to analyzing these shifted leakage kernels $\Lambda_p[m, i]$, which will occupy most of the rest of this section.

Before we begin to do so let us take a short detour to understand these leakage kernels better. Using the simplified model (5.12), the continuous equivalent of the discrete-delay-Doppler spreading function, given in (3.9), can be calculated as

$$\begin{aligned} S(\nu, \tau) &= (\mathcal{F}_1 h)(\nu, \tau) = \left(\mathcal{F}_t \left(\sum_{p=1}^P \eta_p \delta(\tau - \tau_p) e^{j2\pi\nu_p t} \right) \right) (\nu, \tau) \\ &= \sum_{p=1}^P \eta_p \delta(\tau - \tau_p) \mathcal{F}_t \left(e^{j2\pi\nu_p t} \right) (\nu) = \sum_{p=1}^P \eta_p \delta(\tau - \tau_p) \delta(\nu - \nu_p), \end{aligned}$$

where we used the linearity of the Fourier transform and (2.9). So in this setting the spreading function is constituted of Dirac-deltas at the delay-Doppler points (τ_p, ν_p) for the time delay τ_p and Doppler frequency shift ν_p corresponding to scatterer p . In the discrete case, however, the Dirac-deltas are replaced by the functions ϕ_p in the m -direction (the discrete delay) and ψ in the i -direction (the discrete Doppler frequency shift). If we assume ideal filters, i.e. $f_1(t) = f_2(t) = \sqrt{1/T_s} \text{sinc}(t/T_s)$, with the sinc-function defined in (2.10), the discrete-time impulse response (3.11) becomes $h[n, m] \approx \int_{\mathbb{R}} h(nT_s, \tau) \text{sinc}(m - \tau/T_s) d\tau$ (using (2.11) and the fact that T_s typically is very small),

which is seen to be $h[n, m] \approx \sum_{p=1}^P \eta_p e^{j\pi\nu_p n T_s} \text{sinc}(m - \tau_p/T_s)$ using the channel model (5.12). Then, the shifted leakage kernels (5.16) become

$$\Lambda_p[m, i] \approx \text{sinc}\left(m - \frac{\tau_p}{T_s}\right) \psi\left(i - \nu_p T_s N_0\right).$$

These leakage kernels would reduce to simple (discrete) Dirac-deltas if the time delays τ_p and Doppler shifts ν_p were such that τ_p/T_s and $\nu_p T_s N_0$ are integers, since both the sinc and the ψ -function vanish at all integers except at zero, where they are one. In every other case, though, this is not true anymore, and $\Lambda_p[m, i]$ will have non-zero entries for all m and i because the sinc and the ψ -functions both have infinite support (but at least with decreasing intensity due to the decay of sinc and ψ). In other words, the main peak in the discrete delay-Doppler domain will "leak" into the surrounding (discrete) delay-Doppler bins.

This so-called *leakage effect* is a fundamental phenomenon in signal processing, and it actually also occurs in the continuous setting. It is due to the fact that in practical scenarios a signal is only observed during a finite amount of time, which obviously can be viewed as a pointwise product with a rectangular function that cuts off all contributions outside a given time span. Therefore, the spectrum of the observed signal is the convolution of the original spectrum with the spectrum of the rectangular function, which is a sinc-function (see Section 2.2 and Section 2.3 for details). So each individual frequency component present in the original signal is spread out over the entire spectrum due to the infinite support of the sinc-function.

Note that the leakage in m -direction, described by the function ϕ_p , can be reduced by using filters f_1 and f_2 different from the ideal low-pass filter. For example, using for $f_1(t) = f_2(t)$ a root-raised cosine filter with roll-off factor α , the function ϕ_p decays polynomially of order 3 [3] (at least approximately), whereas using the ideal low-pass filter only leads to a polynomial decay of order 1. Here, by polynomial decay of order s we mean that $|f(t)| \leq C(1+|t/t_0|)^{-s}$ for some positive constants C and t_0 . Nevertheless, due to the finite transmit bandwidth ($B_0 = 1/T_s$) and the finite blocklength ($N_0 \approx NL$), the discrete-delay-Doppler spreading function (5.15) always suffers from the leakage effect (characterized by the shifted leakage kernels (5.16)), which obviously affects its sparsity. Note that a larger blocklength reduces the leakage effect, but at the same time the accuracy of the simple channel model (5.12) with constant parameters decreases, and thus the continuous-time delay-Doppler spreading function becomes less compressible.

Now let us return to analyzing the general leakage kernels $\Lambda_p[m, i]$ given in (5.16). As already mentioned above, the function ϕ_p is mostly governed by the interpolation and anti-aliasing filters f_1 and f_2 , and can thus be assumed to decay polynomially of some order $s \geq 1$ away from zero, i.e. $|\phi_p(x)| \leq C(1 + |x/x_0|)^{-s}$. The function ψ is also centered about zero and decays away from zero. Therefore, each leakage kernel $\Lambda_p[m, i]$ is centered about the delay-Doppler point $\xi_p = (\tau_p/T_s, \nu_p T_s N_0)$ and decays away from it, which means that it can be considered compressible. To be more precise, we now bound the ℓ_2 -norm of all the samples $\Lambda_p[m, i]$ which are outside a box of size at most $(2\Delta m + 1) \times (2\Delta i + 1)$ around ξ_p for some $\Delta m \in \{2, \dots, K/2 - 1\}$ and $\Delta i \in \{2, \dots, N_0/2 - 1\}$. To do so, we define $\mathcal{M}_p := \{m \in \{0, \dots, K-1\} \mid |m - \tau_p/T_s| \leq \Delta m\}$ and $\mathcal{I}_p := \{i \in \{0, \dots, N_0-1\} \mid |(i - \nu_p T_s N_0)_{N_0}| \leq \Delta i\}$, where we define $(i-x)_{N_0} := i + d_x N_0 - x$ with $d_x \in \mathbb{Z}$ such that $-N_0/2 \leq (i + d_x N_0 - x) \leq N_0/2$. Then we have

$$\begin{aligned} \sum_{m \notin \mathcal{M}_p} \left| \phi_p\left(m - \frac{\tau_p}{T_s}\right) \right|^2 &\leq C^2 \sum_{m \notin \mathcal{M}_p} \left(1 + \left| \frac{m - \tau_p/T_s}{x_0} \right| \right)^{-2s} \\ &\leq 2C^2 \int_{\Delta m - 1}^{\infty} \left(1 + \frac{x}{|x_0|}\right)^{-2s} dx \\ &= \frac{2C^2 |x_0|}{2s - 1} \left(1 + \frac{\Delta m - 1}{|x_0|}\right)^{-2s+1}, \end{aligned}$$

where we have interpreted the sum as a Riemann sum and therefore bound it by the improper Riemann integral from above. As for the function ψ , first note that ψ is N_0 -periodic by definition (see (5.14)), and therefore $\psi(i - \nu_p T_s N_0) = \psi((i - \nu_p T_s N_0)_{N_0})$. Then we can easily derive

$$\begin{aligned} \sum_{i \notin \mathcal{I}_p} |\psi(i - \nu_p T_s N_0)|^2 &\leq \sum_{i \notin \mathcal{I}_p} \frac{1}{N_0^2 \sin^2(\pi(i - \nu_p T_s N_0)_{N_0}/N_0)} \\ &\leq \frac{1}{N_0^2} \sum_{i \notin \mathcal{I}_p} |2(i - \nu_p T_s N_0)_{N_0}/N_0|^{-2} \\ &\leq \frac{1}{2} \int_{\Delta i - 1}^{\infty} x^{-2} dx \\ &= \frac{1}{2(\Delta i - 1)}. \end{aligned}$$

Here we have used $|\sin(x)| \leq 1$ for the first inequality. The second one follows from the facts that $|(i - \nu_p T_s N_0)_{N_0}| \leq N_0/2$ and that $|\sin(\pi x)| \geq 2|x|$ for all $|x| \leq 1/2$, whereas for the third inequality we again interpreted the sum as a Riemann sum and bound it by the improper Riemann integral. Moreover, we can bound $\sum_{m=0}^{K-1} |\phi_p(m - \tau_p/T_s)|^2 \leq D_0$

and $\sum_{i=0}^{N_0-1} |\psi(i - \nu_p T_s N_0)|^2 \leq D_1$ for all $p = 1, \dots, P$, where D_0 and D_1 are some (small) positive constants. Putting all this together, we get

$$\begin{aligned} \sum_{(m,i) \notin \mathcal{M}_p \times \mathcal{I}_p} \left| \Lambda_p[m, i] \right|^2 &= \sum_{m \notin \mathcal{M}_p} \sum_{i \in \mathcal{I}_p} \left| \Lambda_p[m, i] \right|^2 + \sum_{m \in \mathcal{M}_p} \sum_{i \notin \mathcal{I}_p} \left| \Lambda_p[m, i] \right|^2 + \sum_{m \notin \mathcal{M}_p} \sum_{i \notin \mathcal{I}_p} \left| \Lambda_p[m, i] \right|^2 \\ &\leq \tilde{D}_0 \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1} + \tilde{D}_1 (\Delta i - 1)^{-1} \\ &\quad + \tilde{D}_2 (\Delta i - 1)^{-1} \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1}, \end{aligned} \quad (5.17)$$

with $\tilde{D}_0 := (2D_1 C^2 |x_0|) / (2s - 1)$, $\tilde{D}_1 := D_0 / 2$ and $\tilde{D}_2 := (C^2 |x_0|) / (2s - 1)$. The last expression bounds the error for approximating Λ_p only by its values inside the box $\mathcal{M}_p \times \mathcal{I}_p$, and it obviously decreases with increasing Δm and Δi . In other words, each leakage kernel is mainly supported inside a box of size most at $(2\Delta m + 1) \times (2\Delta i + 1)$ around ξ_p .

Now let $\mathbf{\Lambda}_p := \text{vec}\{\Lambda_p[m, i]\}_{m,i}$. Since $|\mathcal{M}_p| \leq 2\Delta m + 1$ and $|\mathcal{I}_p| \leq 2\Delta i + 1$, setting $S = (2\Delta m + 1)(2\Delta i + 1)$ and combining Definition 4.2.2 of the S -term approximation error with (5.17) yields

$$\begin{aligned} (\sigma_S(\mathbf{\Lambda}_p)_2)^2 &\leq \sum_{(m,i) \notin \mathcal{M}_p \times \mathcal{I}_p} \left| \Lambda_p[m, i] \right|^2 \\ &\leq \tilde{D}_0 \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1} + \tilde{D}_1 (\Delta i - 1)^{-1} \\ &\quad + \tilde{D}_2 (\Delta i - 1)^{-1} \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1}. \end{aligned} \quad (5.18)$$

Since the expression on the right-hand-side decreases quite fast with increasing Δm and Δi , and therefore for increasing S , we can consider each leakage kernel $\Lambda_p[m, i]$ compressible as defined in Section 4.2. Following (5.15) the discrete-delay-Doppler spreading function $S_h[m, i]$ is basically the sum of the shifted leakage kernels and therefore can be considered compressible, too. Because of (3.17), in turn, the same is true for the 2D DFT coefficients $F_{m,i}$, which is what we wanted to indicate in this section. Finally, note that in typical practical scenarios the supports of the shifted leakage kernels $\Lambda_p[m, i]$ will overlap to some degree, so that the overall sparsity typically will not grow linearly with the number P of scatterers.

5.1.4 Basis Optimization

In this section, following [2, 3], we adapt the general basis expansion model (5.2) to the channel model (5.12) (but not to the specific channel parameters P , η_p , τ_p and ν_p), and we describe a method for finding such a basis. Obviously, the performance of the compressive channel estimator described in Section 5.1.1 depends on the sparsity (or, more precisely, the compressibility) of the expansion coefficients $G_{m,i}$. For the "natural" 2D DFT basis these coefficients are given by $G_{m,i} = \sqrt{JD}F_{m,i}$, the compressibility of which has been shown to be impaired mostly by the only slowly decaying factor $\psi(i - \nu_p T_s N_0)$ in the previous section (in general the factor $\phi_p(m - \tau_p/T_s)$ can be designed to decay faster).

Let us start by taking a closer look at these coefficients $F_{m,i}$. Following (3.17) and the channel model (5.12) we get

$$\begin{aligned} F_{m,i} &= \sum_{q=0}^{N-1} S_h[m, i + qL] A_{\gamma,g}^* \left(m, \frac{i + qL}{N_0} \right) \\ &= \sum_{p=1}^P \phi^{(p)} \left(m - \frac{\tau_p}{T_s} \right) \alpha_{m,i}^{(p)}, \end{aligned}$$

where we have set

$$\alpha_{m,i}^{(p)} := \sum_{q=0}^{N-1} \eta_p e^{j\pi \left(\nu_p T_s - \frac{i+qL}{N_0} \right) (N_0-1)} \psi \left(i + qL - \nu_p T_s N_0 \right) A_{\gamma,g}^* \left(m, \frac{i + qL}{N_0} \right). \quad (5.19)$$

Note that $\alpha_{m,i}^{(p)}$ does not depend on the time delay τ_p . With this at hand (5.1) becomes

$$\begin{aligned} H_{\lambda\Delta L, \kappa\Delta K} &= \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} F_{m,i} e^{-j2\pi \left(\frac{\kappa m}{D} - \frac{\lambda i}{J} \right)} \\ &= \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \sum_{p=1}^P \phi_p \left(m - \frac{\tau_p}{T_s} \right) \alpha_{m,i}^{(p)} e^{-j2\pi \left(\frac{\kappa m}{D} - \frac{\lambda i}{J} \right)} \\ &= \sum_{m=0}^{D-1} \sum_{p=1}^P \sqrt{D} \phi_p \left(m - \frac{\tau_p}{T_s} \right) C^{(p)}[m, \lambda] \frac{1}{\sqrt{D}} e^{-j2\pi \frac{\kappa m}{D}} \end{aligned}$$

with

$$C^{(p)}[m, \lambda] := \sum_{i=-J/2}^{J/2-1} \alpha_{m,i}^{(p)} e^{j2\pi \frac{\lambda i}{J}}. \quad (5.20)$$

As explained above, the coefficient function $\phi_p(m - \tau_p/T_s)$ decays quite fast, whereas the sparsity of the coefficient function $\alpha_{m,i}^{(p)}$ is impaired by the slow decay of ψ . Therefore, we only replace the DFT representation (5.20) by a general basis expansion, i.e.

$$C^{(p)}[m, \lambda] = \sum_{i=-J/2}^{J/2-1} \beta_{m,i}^{(p)} v_{m,i}[\lambda]. \quad (5.21)$$

With this at hand, we finally get

$$H_{\lambda\Delta L, \kappa\Delta K} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \sum_{p=1}^P \phi_p\left(m - \frac{\tau_p}{T_s}\right) \beta_{m,i}^{(p)} v_{m,i}[\lambda] e^{-j2\pi \frac{\kappa m}{D}},$$

which is of the form (5.2) as desired, with

$$u_{m,i}[\lambda, \kappa] = v_{m,i}[\lambda] \frac{1}{\sqrt{D}} e^{-j2\pi \frac{\kappa m}{D}} \quad \text{and} \quad G_{m,i} = \sum_{p=1}^P \sqrt{D} \phi_p\left(m - \frac{\tau_p}{T_s}\right) \beta_{m,i}^{(p)}. \quad (5.22)$$

Thus, our goal is to find a family of orthonormal bases $\{v_{m,i}[\lambda]\}_{i=-J/2}^{J/2-1}$, $m = 0, \dots, D-1$, i.e. $\sum_{\lambda} v_{m,i}[\lambda] v_{m,i'}^*[\lambda] = \delta[i - i']$, such that the coefficients $G_{m,i}$ are as sparse as possible. To make sure that these bases do not depend on the actual channel parameters τ_p , ν_p and η_p , we assume these parameters to be random, with (τ_p, ν_p) distributed according to the probability density function (pdf) $\wp(\tau_p, \nu_p)$ (see Section 2.4). Note that here we set $\eta_p = 1$, but that the distribution of η_p can also be taken into account [3]. The coefficients $G_{m,i}$ then become random variables, and our goal is for these coefficients to be maximally sparse *on average*. For algorithmic simplicity we measure the sparsity of the coefficient vector $\mathbf{g} = \text{vec}\{G_{m,i}\}_{m,i}$ by its ℓ_1 -norm, i.e. we want $\mathbb{E}\{\|\mathbf{g}\|_1\}$ to be as small as possible. Note that in practice \wp might be known (approximately) from some prior information about the channel. However, for the case where it is not known a non-statistical design is easily obtained by formally using a uniform distribution (see Section 2.4).

To formulate the optimization problem in a more convenient way we define the vectors $\mathbf{c}_m^{(p)} := [C^{(p)}[m, 0], \dots, C^{(p)}[m, J-1]]^T$, as well as the unitary $J \times J$ matrices \mathbf{V}_m with entries $[\mathbf{V}_m]_{\lambda, i+J/2} = v_{m,i}[\lambda]$ for $m = 0, \dots, D-1$, $i = -J/2, \dots, J/2-1$ and $\lambda = 0, \dots, J-1$, which corresponds to a columnwise stacking of the basis vectors for each m . Then, (5.21) can obviously be rewritten as $\mathbf{c}_m^{(p)} = \mathbf{V}_m \boldsymbol{\beta}_m^{(p)}$ with the vector $\boldsymbol{\beta}_m^{(p)} := [\beta_{m,-J/2}^{(p)}, \dots, \beta_{m, J/2-1}^{(p)}]^T$. Since each \mathbf{V}_m is unitary, we can thus calculate the coefficient

vector as $\beta_m^{(p)} = \mathbf{V}_m^H \mathbf{c}_m^{(p)}$. Therefore, following (5.22), we can write the vectors $\mathbf{g}_m := [G_{m,-J/2}, \dots, G_{m,J/2-1}]^T$ as

$$\mathbf{g}_m = \mathbf{V}_m^H \left(\sum_{p=1}^P \tilde{\mathbf{c}}_m^{(p)} \right) \quad (5.23)$$

for each $m = 0, \dots, D-1$, where $\tilde{\mathbf{c}}_m^{(p)} := \sqrt{D} \phi_p(m - \tau_p/T_s) \mathbf{c}_m^{(p)}$. To simplify the problem we set $P = 1$, i.e. we consider a single-scatterer channel, which yields $\mathbf{g}_m = \mathbf{V}_m^H \tilde{\mathbf{c}}_m^{(1)}$. Note that if the coefficients are compressible on average for a single-scatterer channel they will also be compressible on average for the general channel model with $P > 1$ scatterers.

As explained above our task is to find basis matrices \mathbf{V}_m for $m = 0, \dots, D-1$ such that $\mathbb{E}\{\|\mathbf{g}\|_1\}$ becomes minimal. Since $\mathbb{E}\{\|\mathbf{g}\|_1\} = \mathbb{E}\{\sum_{m=0}^{D-1} \|\mathbf{g}_m\|_1\} = \sum_{m=0}^{D-1} \mathbb{E}\{\|\mathbf{g}_m\|_1\}$ by the linearity of the expectation (see (2.12)), we can minimize $\mathbb{E}\{\|\mathbf{g}\|_1\}$ by minimizing each of the summands $\mathbb{E}\{\|\mathbf{g}_m\|_1\}$ separately, and thus solve the problem

$$\hat{\mathbf{V}}_m = \arg \min_{\mathbf{V}_m \in \mathcal{U}} \mathbb{E}\{\|\mathbf{V}_m^H \tilde{\mathbf{c}}_m^{(1)}\|_1\}$$

for each $m = 0, \dots, D-1$ individually, where \mathcal{U} denotes the set of all unitary $J \times J$ matrices. Using a Monte-Carlo approximation (see Section 2.4) we finally state the optimization problem

$$\hat{\mathbf{V}}_m = \arg \min_{\mathbf{V}_m \in \mathcal{U}} \sum_{\rho} \|\mathbf{V}_m^H (\tilde{\mathbf{c}}_m^{(1)})_{\rho}\|_1, \quad m = 0, \dots, D-1, \quad (5.24)$$

where $(\tilde{\mathbf{c}}_m^{(1)})_{\rho}$ denotes the value of $\tilde{\mathbf{c}}_m^{(1)}$ for a sample of the random vector (τ_1, ν_1) drawn from its pdf $\varphi(\tau_1, \nu_1)$.

Unfortunately, the set \mathcal{U} is not convex, and therefore the optimization problem (5.24) cannot be solved using standard optimization tools. In [2, 3] the following algorithm for approximating a solution was proposed. It is based on Proposition 2.1.3, namely that each unitary matrix $\mathbf{V} \in \mathcal{U}$ can be represented as $\mathbf{V} = e^{j\mathbf{A}}$, where \mathbf{A} is a Hermitian matrix (see Section 2.1). This yields that in the optimization problem (5.24) the non-convex set \mathcal{U} of unitary matrices can actually be replaced by the convex set \mathcal{H} of all Hermitian matrices of the same dimension. Since again the matrix exponential is not convex, it is approximated by $\mathbf{V} \approx \mathbf{I}_J + j\mathbf{A}$ following (2.2) (note that the approximation error will be small if $\|\mathbf{A}\|_{\infty}$ is small). This finally yields a convex optimization problem that can be solved efficiently by using standard optimization tools. Therefore, the algorithm proceeds as follows.

- **Input.** Initialization matrix $\mathbf{V}_m^{(0)}$, pdf $\wp(\tau_1, \nu_1)$, initial threshold $\gamma^{(0)}$.
- **Initialization.** $n = 0$
- **while** stopping criterion not met **do**
 1. Solve the convex problem $\hat{\mathbf{A}}_m^{(n)} = \arg \min_{\mathbf{A} \in \mathcal{H}_n} \sum_{\rho} \|(\mathbf{I}_J + J\mathbf{A})\mathbf{V}_m^{(n)}(\tilde{\mathbf{c}}_m^{(1)})_{\rho}\|_1$, where \mathcal{H}_n denotes the set of all Hermitian matrices \mathbf{A} with $\|\mathbf{A}\|_{\infty} < \gamma^{(n)}$
 2. Set $\hat{\mathbf{V}}_m^{(n)} = e^{J\hat{\mathbf{A}}_m^{(n)}} \mathbf{V}_m^{(n)}$
 3. **if** $\sum_{\rho} \|\hat{\mathbf{V}}_m^{(n)}(\tilde{\mathbf{c}}_m^{(1)})_{\rho}\|_1 < \sum_{\rho} \|\mathbf{V}_m^{(n)}(\tilde{\mathbf{c}}_m^{(1)})_{\rho}\|_1$
 update $\mathbf{V}_m^{(n+1)} = \hat{\mathbf{V}}_m^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}$
else
 update $\mathbf{V}_m^{(n+1)} = \mathbf{V}_m^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}/2$
 4. Iterate $n \mapsto n + 1$
- **Output.** $\mathbf{V}_m = \mathbf{V}_m^{(n)}$

As mentioned above, the algorithm proceeds by performing just a "small" step in each iteration ("small" meaning that the matrix $\hat{\mathbf{A}}_m^{(n)}$ has small entries). Since the product of two unitary matrices is again unitary, the iterate $\hat{\mathbf{V}}_m^{(n)}$ in step 2 of the loop is guaranteed to be unitary for each n . In step 3 of the loop it is checked if an improvement over the previous iterate has been made. If so, the iterate is accepted as the new update. Otherwise, it is rejected, the previous iterate is kept and the "acceptable size" for the following step is reduced. The algorithm is stopped if either the threshold $\gamma^{(n)}$ falls below a prescribed value or if the maximal number of iterations is reached.

Since the analysis in the previous section has shown that the use of the 2D DFT basis already yields compressible coefficients $G_{m,i} = \sqrt{JD}F_{m,i}$, we choose the $J \times J$ DFT matrix as initialization matrix $\mathbf{V}_m^{(0)}$ for each $m = 0, \dots, D-1$. Because of the way the algorithm is constructed, its output \mathbf{V}_m is ensured to outperform the DFT matrix (as measured by (5.24)). Note that the use of this basis will, on the other hand, increase the computational complexity of the method, since in general no fast FFT-methods can be used (also see Section 5.1.2 for more details). Nevertheless, if J is not too large, this will not matter too much. Furthermore note that this basis optimization only has to be performed once before the start of data transmission.

5.1.5 Simulation Results

In order to compare the performance of the proposed compressive channel estimator to standard LS estimation methods we have conducted extensive numerical experiments, the results of which are presented in this section.

MC system parameters. We simulated a CP-OFDM system with $K = 512$ subcarriers, CP-length $L_{\text{cp}} = N - K = 128$, center frequency $f_0 = 5$ GHz, and bandwidth $1/T_s = 5$ MHz. We used 4-QAM symbols, and the interpolation and anti-aliasing filters $f_1(t) = f_2(t)$ were chosen as root-raised cosine filters with roll-off factor $\alpha = 1/4$.

Channel. During blocks of $L = 32$ OFDM symbols we generated doubly selective channels using the geometry based channel simulation tool `IlmProp` [143]. The transmitter and the receiver were separated by about 1500 m. In an area not too tightly surrounding them, 7 clusters of 10 specular scatterers each were distributed randomly. Furthermore, 3 clusters of 10 specular scatterers each were distributed within a circle of radius 100 m surrounding the receiver. In Fig. 5.1 an example of such a scenario is depicted. The clusters had diameters between 5 m and 30 m, and each of them, as well as the receiver, had random velocity and acceleration vectors with uniformly distributed directions, and velocities and accelerations that were distributed uniformly up to 50 m/s and 7 m/s², respectively. The noise $z[n]$ (see (3.10)) was zero-mean complex white Gaussian with component variance σ_z^2 (i.e. $z[n]$ were Gaussian random variables with mean 0 and variance σ^2 that were uncorrelated for different n , see Section 2.4) such that a prescribed SNR was achieved. Here we define the SNR as the mean received signal power averaged over one block of length LN , divided by σ_z^2 , i.e. $\sum_{n=0}^{N_0-1} \mathbb{E}\{|r[n] - z[n]|^2\} / \sum_{n=0}^{N_0-1} \mathbb{E}\{|z[n]|^2\}$.

Subsampling and pilot setup. For compressive channel estimation the pilot set was chosen uniformly at random from a subsampled time-frequency grid \mathcal{G} with spacing $\Delta L = 1$ and $\Delta K = 4$ as explained above. The number of pilots was $Q_1 = 1024$ and $Q_2 = 2048$, corresponding to 6.25% and 12.5% of all the symbols, respectively. For classic LS channel estimation we used again $Q_2 = 2048$, and now $Q_3 = 4096$ pilots, corresponding to 12.5% and 25% of all the symbols, respectively, using a comb-type pilot arrangement as described in Section 3.5.

Performance measure. For all the simulations the performance is measured by the mean square error (MSE) normalized by the mean energy of the channel coefficients, i.e. $\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2 / \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k}|^2$.

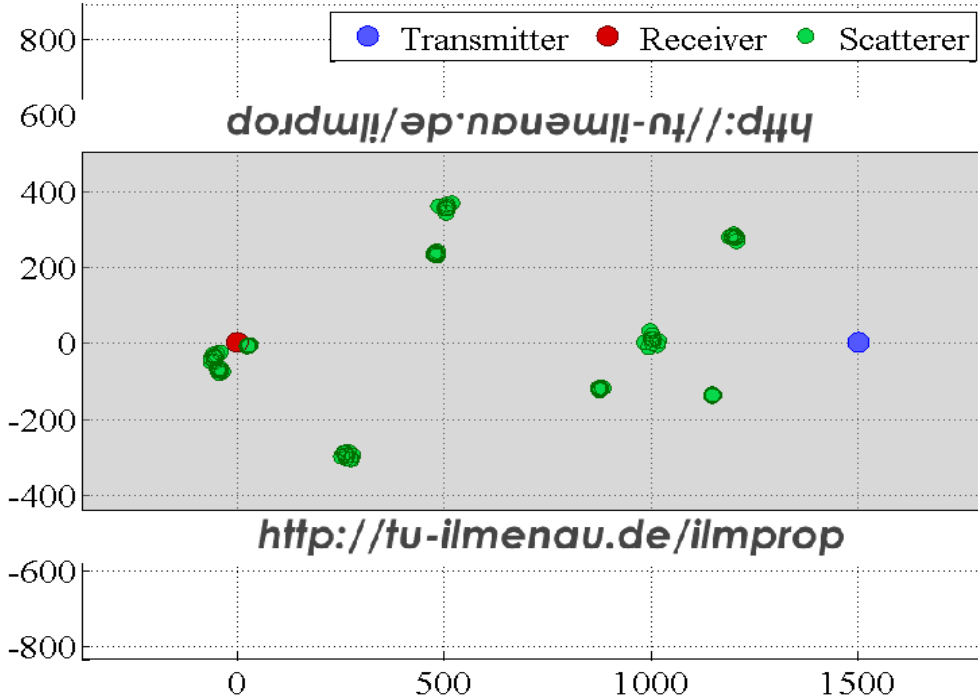
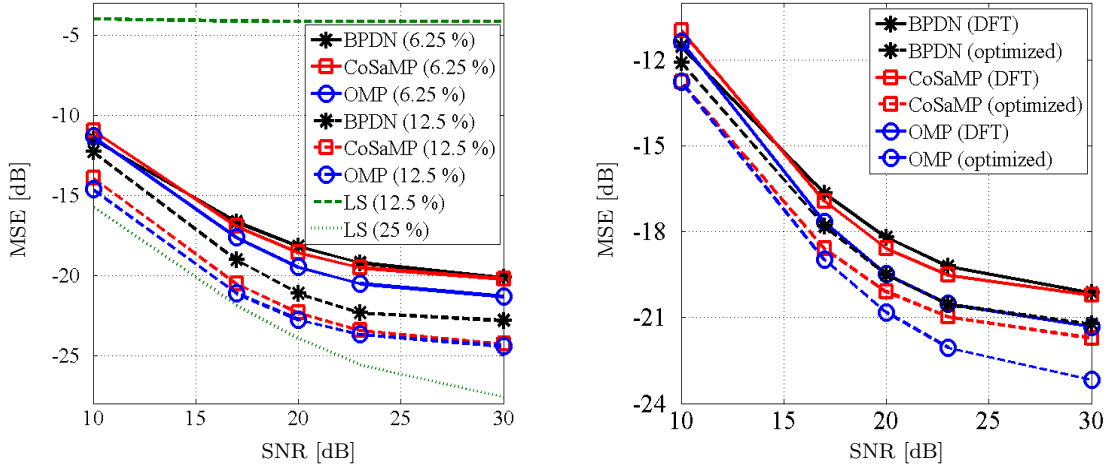


Figure 5.1: Typical simulation scenario

Channel Estimation. We compared the classic LS channel estimator to the compressive channel estimator described above. For the LS approach we used cubic splines to interpolate between the channel estimates at the pilot positions, whereas for sparse reconstruction in Step 2 of the compressive estimation method BPDN, CoSaMP and OMP were used. For BPDN, the noise parameter ϵ (see (4.2)) was chosen as $\epsilon = 10^{-2.6}$. The number of CoSaMP iterations was $n_{\text{CoSaMP}} = 16$, whereas the sparsity estimate was $S = 100$. For OMP we chose the number of iterations $n_{\text{OMP}} = 120$. All these parameters were found experimentally. Finally, note that we used the toolbox SPGL1 [144] to solve BPDN. For CoSaMP we applied the routine `cosamp.m` written by David Mary [145], whereas we implemented OMP ourselves.

Basis Optimization. We constructed an optimized basis according to Section 5.1.4 using a non-statistical approach. Here, $\wp(\tau, \nu) = 1/(2\tau_{\max}\nu_{\max})$ for $(\tau, \nu) \in [0; \tau_{\max}] \times [-\nu_{\max}; \nu_{\max}]$, and $\wp(\tau, \nu) = 0$ everywhere else, where $\tau_{\max} = L_{\text{cp}}T_s = 25.6 \mu\text{s}$ and $\nu_{\max} = 0.03NL/K \approx 293 \text{ Hz}$ (corresponding to 3 % of the subcarrier spacing). Since in our case $L_{\text{cp}} \geq D-1$ (actually $L_{\text{cp}} = D-1 = 128$), (5.11) yields $A_{\gamma,g}(m, \xi) = A_{\gamma,g}(0, \xi)$ for



(a) Compressive channel estimation using 6.25% and 12.5% pilots compared to LS channel estimation using 12.5% and 25% pilots

(b) Comparison of the use of the 2D DFT basis to that of an optimized basis in CCE (using 6.25% pilots)

Figure 5.2: Performance of the compressive channel estimator: MSE versus SNR

all $m = 0, \dots, D-1$. Therefore, following (5.20) and (5.19), also $C^{(p)}[m, \lambda] = C^{(p)}[0, \lambda]$, and in turn $\mathbf{c}_m^{(p)} = \mathbf{c}_0^{(p)}$ for each $m = 0, \dots, D-1$, which yields that only one basis \mathbf{V} has to be calculated (instead of the D different bases \mathbf{V}_m for $m = 0, \dots, D-1$). As mentioned in Section 5.1.4 we used the $J \times J$ DFT matrix as initialization matrix $\mathbf{V}^{(0)}$.

Using this optimized basis, the sparsity of the coefficients $G_{m,i}$ is "better" than that of the 2D DFT coefficients $F_{m,i}$, and therefore the parameters of the CS reconstruction methods have to be adapted. For BPDN we again used $\epsilon = 10^{-2.6}$, and the number of CoSaMP iterations was still $n_{\text{CoSaMP}}^{\text{opt}} = 16$. On the contrary, the sparsity estimate was then chosen as $S^{\text{opt}} = 80$, whereas the number of OMP iterations was $n_{\text{OMP}}^{\text{opt}} = 100$.

Results. In Fig. 5.2(a) we plot the MSE of channel estimation versus the SNR for 500 different channel realizations, each of them generated independently as described above. It is obvious that in the case where 12.5% of the symbols are pilots the compressive channel estimator outperforms the LS estimator by far, no matter which CS technique is used. Actually, even if only half as many pilots are used the compressive channel estimator still performs much better. The extremely bad performance of the LS estimator in this setting is due to the fact that the Shannon sampling criterion is not met. In contrast, the compressive channel estimator can be seen to produce reliable channel estimates even far below the Shannon sampling rate. Furthermore, the proposed method

does not perform much worse than the LS estimator using 25% of the symbols as pilots, especially in the low-to-medium SNR regime. Therefore, it is an attractive alternative achieving reasonable reconstruction quality despite using very few pilots.

From Fig. 5.2(b) it can be concluded that utilizing the optimized basis as described above instead of the 2D DFT basis also yields a considerable performance gain for the compressive estimator. This is due to the "better" sparsity of the expansion coefficients $G_{m,i}$ for this basis compared to the 2D DFT coefficients $F_{m,i}$. On the contrary, recall that its use increases the computational complexity of the compressive estimator.

Moreover, note that using OMP gives the best results in practice, followed by CoSaMP and BPDN. Nevertheless, BPDN is quite stable with regard to the choice of the noise parameter ϵ , whereas the performance of CoSaMP and OMP quite heavily depends on the sparsity estimate S and the number n_{OMP} of OMP iterations, respectively. In our experiments the running time of BPDN was also shorter compared to CoSaMP and OMP, but this could be due to the fact that we used a well-optimized routine to solve BPDN, whereas our own implementation of OMP has not been optimized yet.

Conclusion. The compressive channel estimator is seen to be a very good alternative to classical LS channel estimation, yielding good estimation quality despite using only a very little amount of pilots. Furthermore, the use of an optimized basis can substantially improve the performance, coming at the cost of an increased computational complexity.

5.2 Compressive Channel Estimation using Group Sparsity Methods

In the following we explain in what way the methodology of GSCS, as explained in Section 4.3, can be used to improve the performance of the basic compressive channel estimator presented in the previous section. In Section 5.1.3 we have shown that the 2D DFT coefficients $F_{m,i}$ from (3.16) are compressible in a typical scenario, but that the sparsity is impaired by the so-called leakage effect. Due to the finite bandwidth and the finite number of transmit symbols, the optimal Dirac-deltas, that actually constitute the ideal continuous spreading function, are transformed into broader "bumps" in the discrete spreading function. The decay of these is governed by the functions ϕ_p and ψ (see (5.13) and (5.14)), which in turn impairs the performance of any compressive chan-

nel estimator. Nevertheless, we show that by using GSCS techniques this leakage can actually be utilized to improve the performance of the compressive estimator presented in Section 5.1.1 and at the same time reduce its running time.

The remainder of this section is organized as follows. In Section 5.2.1 we introduce the extension of the method reviewed in Section 5.1.1 that is based on the fact that the 2D DFT coefficients $F_{m,i}$ typically are group compressible. This fact is reviewed in Section 5.2.3, after a brief discussion of the computational complexity in Section 5.2.2. Then, in Section 5.2.4, the basis optimization technique explained in Section 5.1.4 is adapted to this setting. Finally, we present simulation results demonstrating the performance gain of the proposed method compared to the conventional compressive channel estimator in Section 5.2.5.

5.2.1 The Method

Here we consider the same setting as in Section 5.1.1. The proposed method merely differs from the basic method presented there in the Step 2, where GSCS methods are used instead of conventional CS methods. This is feasible since the 2D DFT coefficients $F_{m,i}$ (see (3.16)) can be seen to be group compressible (see Section 5.2.3).

To be more precise, recall that we assume the coefficients $F_{m,i}$ to be supported inside the box $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ for J and D as before. Now, we partition this set into small blocks of size $\Delta\tilde{m} \times \Delta\tilde{i}$ for some $\Delta\tilde{m} \in \mathbb{Z}$ and $\Delta\tilde{i} \in \mathbb{Z}$, for which $B_D := D/\Delta\tilde{m}$ and $B_J := (J/2)/\Delta\tilde{i}$ are integers, and such that the blocks are of the form $\mathcal{B}_b = \{k_b\Delta\tilde{m}, \dots, (k_b+1)\Delta\tilde{m}-1\} \times \{l_b\Delta\tilde{i}, \dots, (l_b+1)\Delta\tilde{i}-1\}$ for $k_b \in \{0, \dots, B_D-1\}$ and $l_b \in \{-B_J, \dots, B_J-1\}$. Furthermore, we define the partition $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ of $\{0, \dots, JD-1\}$ with $B := 2B_DB_J$ and the groups $I_b = \mathbf{S}(\mathcal{B}_b)$, where \mathbf{S} is the one-to-one mapping \mathbf{S} of (5.8) corresponding to columnwise stacking. In Section 5.2.3 and Section 5.2.4 we explore the fact that the 2D DFT coefficients $F_{m,i}$, or the more general coefficients $G_{m,i}$ in (5.2), are group compressible with regard to the blocks \mathcal{B}_b , which yields that the vectors $\mathbf{f} = \text{vec}\{F_{m,i}\}_{m,i}$ and $\mathbf{g} = \text{vec}\{G_{m,i}\}_{m,i}$ obviously are group compressible over this partition \mathcal{J} . Fixing the pilot set \mathcal{P} of size $Q := |\mathcal{P}|$, which is again chosen uniformly at random from the grid \mathcal{G} and communicated to the receiver once and for all *before* the start of data transmission and stays fixed therein, the compressive channel estimator using GSCS techniques can be stated as follows.

- Step 1.** Calculate channel estimates at the pilot positions and stack them to obtain the measurement equation (5.4).
- Step 2.** Run any GSCS algorithm to obtain an estimate $\hat{\mathbf{x}}$ of \mathbf{x} , and rescale $\hat{\mathbf{x}}$ with $\sqrt{JD/Q}$ for an estimate $\hat{\mathbf{g}}$ of \mathbf{g} , the entries of which are exactly the expansion coefficients $G_{m,i}$.
- Step 3.** Calculate estimates of the subsampled channel coefficients $H_{\lambda\Delta L, \kappa\Delta K}$ from (5.2).
- Step 4.** Invert (5.1) to obtain estimates of the 2D DFT coefficients $F_{m,i}$ for $m = 0, \dots, D-1$ and $i = -J/2, \dots, J/2-1$. Note that by assumption $F_{m,i}$ vanishes for all other indices.
- Step 5.** Calculate estimates of all the channel coefficients $H_{l,k}$ by using the 2D DFT expansion (3.16).

Note that in the special case where the 2D DFT basis is used we have $G_{m,i} = \sqrt{JD}F_{m,i}$, and therefore Steps 3 and 4 can be omitted.

As mentioned before, the only difference to the basic method is that GSCS techniques are used in Step 2. Therefore, the performance guarantee given below is very similar to the one given in Theorem 5.1.1. Assuming that the expansion coefficients $G_{m,i}$ are group compressible in the sense explained in Section 4.3, the estimation error of the compressive estimator utilizing GSCS techniques can be bound in dependence on the GSCS method that is used. Here we only present results for G-BPDN and G-CoSaMP.

Theorem 5.2.1. *Let $\mathcal{S} \subseteq \{0, \dots, B-1\}$ be any set of size $|\mathcal{S}| = S$. Furthermore, let*

$$C_{G,\mathcal{S},\mathcal{J}} := \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} |G_{m,i}|^2 \right)^{1/2},$$

as well as $\epsilon_n := \|\mathbf{z}_{(p)}\|_2$ and $\epsilon > 0$.

1. *Let G-BPDN be the GSCS reconstruction method used in Step 2 of the presented compressive channel estimator. If Φ satisfies the G-RIP with G-RIC $\delta_{2S|\mathcal{J}} < \sqrt{2} - 1$, and if $\epsilon_n < \epsilon$, we can bound the estimation error as*

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq C'_0 S^{-1/2} C_{G,\mathcal{S},\mathcal{J}} + C'_1 \epsilon,$$

with the constants $C'_0 := c_0 \sqrt{\frac{LK}{JD}}$ and $C'_1 := c_1 \sqrt{\frac{LK}{Q}}$, where c_0 and c_1 are the constants from Theorem 4.3.4.

2. For the case where G-CoSaMP with i steps is used for GSCS reconstruction in Step 2, assume that Φ satisfies the G-RIP over \mathcal{J} with $\delta_{4S|\mathcal{J}} < 0.1$, and assume that $\epsilon_n < \epsilon$, then

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq \tilde{C}_0(1 + S^{-1/2})C_{G,S,\mathcal{J}} + \tilde{C}_1\epsilon + \tilde{C}_2,$$

with $\tilde{C}_0 := 20\sqrt{\frac{LK}{JD}}$, $\tilde{C}_1 := 20\sqrt{\frac{LK}{Q}}$ and $\tilde{C}_2 := 2^{-i}\sqrt{\frac{LK}{JD}}\left(\sum_{(m,i)} |G_{m,i}|^2\right)^{1/2}$. Note that since the number of G-CoSaMP iterations can be chosen arbitrarily, \tilde{C}_2 can be made arbitrarily small.

Proof. The proof follows the exact same lines as the proof of Theorem 5.1.1. Recall (5.7), namely

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k} - \hat{H}_{l,k}|^2} \leq \sqrt{\frac{LK}{Q}} \|\mathbf{x} - \hat{\mathbf{x}}\|_2.$$

Using Theorem 4.3.4 and Theorem 4.3.5, the assumptions of which are satisfied in our setting, and noting that by the definition of $\sigma_S(\mathbf{x})_{2|\mathcal{J}}$ we have

$$\sigma_S(\mathbf{x})_{2|\mathcal{J}} \leq \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} |[\mathbf{x}]_{\mathbf{s}(m,i)}|^2 \right)^{1/2} = \sqrt{\frac{Q}{JD}} \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} |G_{m,i}|^2 \right)^{1/2}$$

gives the results. \square

The set \mathcal{S} can again be chosen arbitrarily, but for a small error it should contain the group indices covering the essential support of the coefficients $G_{m,i}$. In the case where the 2D DFT basis is used $C_{G,S,\mathcal{J}}$ again somehow characterizes the leakage occurring, merely measured in a different norm than in the result for the conventional compressive channel estimator. The second term in the estimate above, however, is again due to the noise, whereas the third term in the error estimate of Part 2 of the theorem can be made arbitrarily small by choosing the number of G-CoSaMP iterations large enough (again, note that the computational cost of G-CoSaMP increases with an increasing number of iterations, see Section 4.3).

The major difference to the basic compressive channel estimator is that by using GSCS techniques we can actually utilize the leakage effect to a certain degree, which decreases the sparsity of the coefficients and therefore actually impairs the performance of all compressive channel estimators. To be more precise, the leakage effect still degrades

the performance of the proposed estimator, even if group sparsity methods are used. Nevertheless, the "broadness" of the peaks in the coefficients (as compared to the Dirac spikes in the continuous case) is utilized to locate the "important" contributions, which is the most difficult task for sparse reconstruction methods. Since this localization is performed group-wise, it is less likely to mistake noise contributions for contributions originating from actual scatterers. Therefore, using GSCS techniques typically improves the performance of conventional CCE, which we also demonstrate in Section 5.2.5.

5.2.2 Computational Complexity

The computational complexity of the proposed channel estimator only differs from that of the basic estimator in Step 2, where GSCS techniques are used instead of CS techniques. Therefore, it is given by

$$\mathcal{O}(\text{GSCS}) + \mathcal{O}((JD)^2) + \mathcal{O}(LK \log(LK)). \quad (5.25)$$

As mentioned in Section 4.3, the computational complexity $\mathcal{O}(\text{GSCS})$ is hard to determine because it strongly depends on the implementation. In that section we explained that G-OMP, for example, will typically have a shorter running time than OMP, but that the overall complexity $\mathcal{O}(M(n'_{\text{G-OMP}})^2 + n_{\text{G-OMP}}\Phi)$ remains the same, where $n_{\text{G-OMP}}$ denotes the number of G-OMP iterations, and $n'_{\text{G-OMP}}$ is the sum of the magnitudes of the chosen groups. For G-CoSaMP, we have $\mathcal{O}(n_{\text{G-CoSaMP}}\Phi)$, where $n_{\text{G-CoSaMP}}$ is the number of iterations. Since in typical scenarios $\mathcal{O}(\text{GSCS})$ will be the dominant term in (5.25), we conclude that the (theoretical) computational complexity of the proposed compressive channel estimator using GSCS techniques does not change compared to the conventional estimator, but that the running time of the GSCS algorithms will typically be shorter than that of their CS counterparts.

5.2.3 Delay-Doppler Group Sparsity

To analyze the group sparsity of the 2D DFT coefficients $F_{m,i}$ for $m = 0, \dots, D-1$ and $i = -J/2, \dots, J/2-1$, recall that their sparsity is mainly governed by the leakage kernels $\Lambda_p[m, i]$ with $p = 1, \dots, P$ (see (5.16)). It was shown in Section 5.1.3 that each of them is mainly supported inside a box of size at most $(2\Delta m + 1) \times (2\Delta i + 1)$ around a center point $\xi_p = (\tau_p/T_s, \nu_p T_s N_0)$, where τ_p and ν_p again denote the time

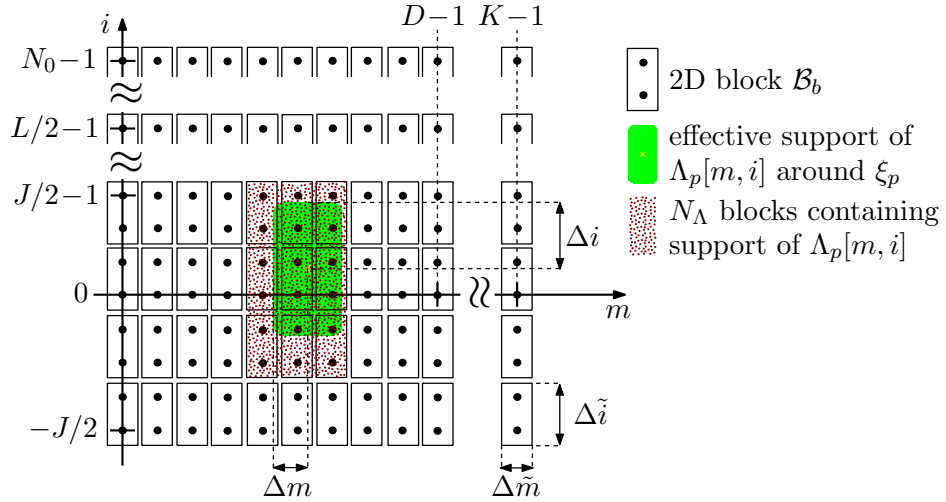


Figure 5.3: Illustration of the 2D block tiling $\{\mathcal{B}_b\}$, the effective support of a shifted leakage kernel $\Lambda_p[m, i]$, and the N_Λ blocks containing this effective support. Here, $\Delta m = 1$, $\Delta i = 2$, $\Delta \tilde{m} = 1$, $\Delta \tilde{i} = 2$, and $N_\Lambda = 9$.

delay and Doppler frequency shift corresponding to scatterer p , respectively, and Δm and Δi can be chosen to achieve a prescribed approximation quality. Since these center points (and therefore the locations of the support boxes) differ for each scatterer and each propagation scenario, it is impossible to choose a partition of the delay-Doppler plane such that these support boxes occur as groups for each scenario. Also, they may overlap for different scatterers, which is not allowed for the groups of a partition. Therefore, we fix the partition and analyze the group sparsity of a leakage kernel $\Lambda_p[m, i]$ for this partition. To do so, we extend the blocks $\mathcal{B}_b = \{k_b \Delta \tilde{m}, \dots, (k_b + 1) \Delta \tilde{m} - 1\} \times \{l_b \Delta \tilde{i}, \dots, (l_b + 1) \Delta \tilde{i} - 1\}$ introduced in Section 5.2.1 to $k_b \in \{0, \dots, (K/\Delta \tilde{m}) - 1\}$ and $l_b \in \{-L/(2\Delta \tilde{i}), \dots, ((N_0 - L/2)/\Delta \tilde{i}) - 1\}$, such that now these blocks form a partition of $\{0, \dots, K - 1\} \times \{-L/2, \dots, N_0 - (L/2) - 1\}$. Then, the support box of the leakage kernel $\Lambda_p[m, i]$ is contained in at most N_Λ blocks, where

$$N_\Lambda := \left(\left\lceil \frac{2\Delta m}{\Delta \tilde{m}} \right\rceil + 1 \right) \left(\left\lceil \frac{2\Delta i}{\Delta \tilde{i}} \right\rceil + 1 \right).$$

Fig. 5.3 shows an example of this situation. To be more precise we define the partition $\bar{\mathcal{J}} = \{\bar{I}_b\}_{b=0}^{\bar{B}-1}$ of $\{0, \dots, KN_0 - 1\}$ with $\bar{B} := (KN_0)/(\Delta \tilde{m} \Delta \tilde{i})$ and $\bar{I}_b = \bar{\mathbf{S}}(\mathcal{B}_b)$, where

$$\begin{aligned} \bar{\mathbf{S}}: \{0, \dots, K - 1\} \times \{-L/2, \dots, N_0 - (L/2) - 1\} &\rightarrow \{0, \dots, KN_0 - 1\} \\ \bar{\mathbf{S}}(m, i) &:= (i + L/2)K + m \end{aligned}$$

corresponds to the stacking $\mathbf{\Lambda}_p = \text{vec}\{\Lambda_p[m, i]\}_{m, i}$. We choose to bound the S -group approximation error $(\sigma_{S|\bar{\mathcal{J}}}(\mathbf{\Lambda}_p)_2)^2$ as defined in (4.5) for the ℓ_2 -norm. Now we can

proceed as in Section 5.1.3. Since by construction the set $\mathcal{M}_p \times \mathcal{I}_p$ defined therein is contained in the N_Λ blocks mentioned above, approximating Λ_p with $S = N_\Lambda$ groups cannot be worse than using the $(2\Delta m + 1)(2\Delta i + 1)$ indices used for the bound (5.18), which therefore also holds for $(\sigma_{N_\Lambda|\bar{\mathcal{J}}}(\Lambda_p)_2)^2$. In fact, the bound can be improved somewhat. To do so, we denote the indices contained in the N_Λ blocks containing the support box of $\Lambda_p[m, i]$ by $\mathcal{S}_p = \mathcal{S}_p^m \times \mathcal{S}_p^i$ for some sets \mathcal{S}_p^m and \mathcal{S}_p^i in the discrete m - and i -direction, respectively. Then, let $\Delta\bar{m}_p := \min\{||m - \tau_p/T_s|| \mid m \notin \mathcal{S}_p^m\}$ and $\Delta\bar{i}_p := \min\{||i - \nu_p T_s N_0|| \mid i \notin \mathcal{S}_p^i\}$, where $(i - x)_{N_0}$ is defined as in Section 5.1.3. Since, as mentioned above, $\mathcal{M}_p \times \mathcal{I}_p \subseteq \mathcal{S}_p^m \times \mathcal{S}_p^i$, we have $\Delta\bar{m}_p \geq \Delta m$ and $\Delta\bar{i}_p \geq \Delta i$. Therefore, defining the sets $\bar{\mathcal{M}}_p := \{m \in \{0, \dots, K-1\} \mid |m - \tau_p/T_s| \leq \Delta\bar{m}_p\}$ and $\bar{\mathcal{I}}_p := \{i \in \{0, \dots, N_0-1\} \mid |(i - \nu_p T_s N_0)_{N_0}| \leq \Delta\bar{i}_p\}$, we have $\mathcal{M}_p \times \mathcal{I}_p \subseteq \bar{\mathcal{M}}_p \times \bar{\mathcal{I}}_p \subseteq \mathcal{S}_p^m \times \mathcal{S}_p^i$, and the same reasoning as in Section 5.1.3 yields

$$\begin{aligned}
 (\sigma_{N_\Lambda|\bar{\mathcal{J}}}(\Lambda_p)_2)^2 &\leq \sum_{(m,i) \notin \bar{\mathcal{M}}_p \times \bar{\mathcal{I}}_p} |\Lambda_p[m, i]|^2 \\
 &\leq \tilde{D}_0 \left(1 + \frac{\Delta\bar{m}_p - 1}{|x_0|}\right)^{-2s+1} + \tilde{D}_1 (\Delta\bar{i}_p - 1)^{-1} \\
 &\quad + \tilde{D}_2 (\Delta\bar{i}_p - 1)^{-1} \left(1 + \frac{\Delta\bar{m}_p - 1}{|x_0|}\right)^{-2s+1}, \tag{5.26}
 \end{aligned}$$

where \tilde{D}_0 , \tilde{D}_1 and \tilde{D}_2 are the constants from (5.18). Since $\Delta\bar{m}_p \geq \Delta m$ and $\Delta\bar{i}_p \geq \Delta i$ (as mentioned above) the bound (5.18) is improved. All in all, we have shown that since the bound in (5.26) obviously decreases quite fast with increasing $\Delta\bar{m}_p$ and $\Delta\bar{i}_p$, each leakage kernel Λ_p can be considered group compressible with respect to $\bar{\mathcal{J}}$ as defined in Section 4.3. Therefore, following (5.15), the discrete-delay-Doppler spreading function $S_h[m, i]$ can be considered group compressible too. Since the blocks \mathcal{B}_b were chosen such that $D/\Delta\tilde{m}$ and $J/(2\Delta\tilde{i})$ are integers, they are compatible with the summation in (3.17) and, in turn, with the restriction of the support of the 2D DFT coefficients $F_{m,i}$ to $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-i\}$. Thus, these coefficients can be considered group compressible too (now with respect to \mathcal{J}), which is exactly what we wanted to show in this section. Finally note that in typical practical scenarios the scatterers tend to appear in clusters, so that the groups containing the supports of the shifted leakage kernels $\Lambda_p[m, i]$ will overlap to some degree (even more than the supports themselves) and the overall group sparsity will not typically grow linearly with the number P of scatterers. This is another (physical) reason for the inherent group compressibility of typical channels.

5.2.4 Basis Optimization

We can easily adapt the basis optimization technique presented in Section 5.1.4 to the group sparse case. The performance of the estimator described in Section 5.2.1 obviously depends on the group sparsity of the expansion coefficients $G_{m,i}$. For this setting we again use the channel model (5.12), and therefore proceed exactly as in Section 5.1.4 also using the same notation. Following Section 4.3, we measure the group sparsity of the coefficient vector \mathbf{g} by $\|\mathbf{g}\|_{2|\mathcal{J}}$, so the goal of this section is to find basis function $v_{m,i}[\lambda]$ that minimize $\mathbb{E}\{\|\mathbf{g}\|_{2|\mathcal{J}}\}$.

In order to do so, recall (5.23), namely $\mathbf{g}_m = \mathbf{V}_m^H \left(\sum_{p=1}^P \tilde{\mathbf{c}}_m^{(p)} \right)$ for each $m = 0, \dots, D-1$, where \mathbf{g}_m are the vectors with entries $G_{m,i}$, $\tilde{\mathbf{c}}_m^{(p)} := \sqrt{D} \phi_p(m - \tau_p/T_s) \mathbf{c}_m^{(p)}$, and \mathbf{V}_m is the columnwise stacking of the basis functions $v_{m,i}$. Next, we define the block-diagonal $JD \times JD$ matrix \mathbf{V} with the $J \times J$ blocks \mathbf{V}_m on its diagonal, and the JD -dimensional vector $\tilde{\mathbf{g}} = [\mathbf{g}_0^T \cdots \mathbf{g}_{D-1}^T]^T$. Again restricting ourselves to the single scatterer channel, i.e. $P = 1$, and defining $\tilde{\mathbf{c}}^{(1)} = [\tilde{\mathbf{c}}_0^{(1)} \cdots \tilde{\mathbf{c}}_{D-1}^{(1)}]^T$, we can summarize the equations (5.23) as

$$\tilde{\mathbf{g}} = \mathbf{V}^H \tilde{\mathbf{c}}^{(1)}.$$

Next, we define the one-to-one mapping

$$\begin{aligned} \mathbf{S}' : \{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\} &\rightarrow \{0, \dots, JD-1\}, \\ \mathbf{S}'(m, i) &:= mJ + i + J/2, \end{aligned} \quad (5.27)$$

which corresponds to rowwise stacking (as opposed to the columnwise stacking operation \mathbf{S} defined in (5.8)). Furthermore, we define the partition $\mathcal{J}' = \{I'_b\}_{b=0}^{B-1}$ of $\{0, \dots, JD-1\}$ with the groups $I'_b = \mathbf{S}'(\mathcal{B}_b)$. Recalling the partition $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ with groups $I_b = \mathbf{S}(\mathcal{B}_b)$, we clearly have $\|\mathbf{g}\|_{2|\mathcal{J}} = \|\tilde{\mathbf{g}}\|_{2|\mathcal{J}'}$, since \mathbf{g} corresponds to the columnwise stacking and $\tilde{\mathbf{g}}$ corresponds to the rowwise stacking of the coefficients $G_{m,i}$. Therefore, the optimization problem can be formulated as

$$\hat{\mathbf{V}} = \arg \min_{\mathbf{V} \in \tilde{\mathcal{U}}} \mathbb{E}\{\|\mathbf{V}^H \tilde{\mathbf{c}}^{(1)}\|_{2|\mathcal{J}'}\},$$

where $\tilde{\mathcal{U}}$ denotes the set of all blockdiagonal $JD \times JD$ matrices with unitary blocks of size $J \times J$ on the diagonal. Again using a Monte-Carlo approximation (see Section 2.4) we can redefine this problem as

$$\hat{\mathbf{V}} = \arg \min_{\mathbf{V} \in \tilde{\mathcal{U}}} \sum_{\rho} \|\mathbf{V}^H (\tilde{\mathbf{c}}^{(1)})_{\rho}\|_{2|\mathcal{J}'}, \quad (5.28)$$

where $(\tilde{\mathbf{c}}^{(1)})_\rho$ denotes the value of $\tilde{\mathbf{c}}^{(1)}$ for a sample of the random vector (τ_1, ν_1) drawn from its pdf $\wp(\tau_1, \nu_1)$. Here we again set $\eta_1 = 1$, but the distribution of η_1 can also be taken into account (extending [3] to this setting).

To solve (5.28) first note that the set $\tilde{\mathcal{U}}$ is not convex either. Therefore, we extend the algorithm presented in Section 5.1.4 to this setting in order to get an approximate solution. To do so, let $\tilde{\mathcal{H}}$ be the set of all blockdiagonal $JD \times JD$ matrices with Hermitian blocks of size $J \times J$ on the diagonal, which again is convex. Following the same reasoning as in Section 5.1.4, which can immediately be extended to the block-diagonal case, the algorithm can be stated as follows.

- **Input.** Initialization matrix $\mathbf{V}^{(0)}$, pdf $\wp(\tau_1, \nu_1)$, initial threshold $\gamma^{(0)}$.
- **Initialization.** $n = 0$
- **while** stopping criterion not met **do**
 1. Solve the convex problem $\hat{\mathbf{A}}^{(n)} = \arg \min_{\mathbf{A} \in \tilde{\mathcal{H}}_n} \sum_\rho \|(\mathbf{I}_{JD} + j\mathbf{A})\mathbf{V}^{(n)}(\tilde{\mathbf{c}}^{(1)})_\rho\|_{2|\mathcal{J}'}$,
where $\tilde{\mathcal{H}}_n$ denotes the set of all $\mathbf{A} \in \tilde{\mathcal{H}}$ with $\|\mathbf{A}\|_\infty < \gamma^{(n)}$
 2. Set $\hat{\mathbf{V}}^{(n)} = e^{j\hat{\mathbf{A}}^{(n)}}\mathbf{V}^{(n)}$
 3. **if** $\sum_\rho \|\hat{\mathbf{V}}^{(n)}(\tilde{\mathbf{c}}^{(1)})_\rho\|_{2|\mathcal{J}'} < \sum_\rho \|\mathbf{V}^{(n)}(\tilde{\mathbf{c}}^{(1)})_\rho\|_{2|\mathcal{J}'}$
update $\mathbf{V}^{(n+1)} = \hat{\mathbf{V}}^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}$
 - else**
update $\mathbf{V}^{(n+1)} = \mathbf{V}^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}/2$
 4. Iterate $n \mapsto n + 1$
- **Output.** $\mathbf{V} = \mathbf{V}^{(n)}$

In Section 5.2.3 we have shown that by using the 2D DFT basis the coefficients $G_{m,i} = \sqrt{JD}F_{m,i}$ are already group compressible. Therefore, we choose the initialization matrix $\mathbf{V}^{(0)}$ to be the blockdiagonal $JD \times JD$ matrix with the $J \times J$ DFT matrix repeated D times on the diagonal. The problem does not depend on the actual channel or the receive signal and therefore only has to be solved once before the start of data transmission.

Finally, note that the computational complexity of solving (5.28) can be substantially reduced by decomposing it into $B_D = D/\Delta\tilde{m}$ separate problems of dimension $J\Delta\tilde{m} \times J\Delta\tilde{m}$ each (as opposed to $JD \times JD$), where typically $\Delta\tilde{m}$ is quite small. In order to see that this is possible, recall that $\mathbf{V}^H\tilde{\mathbf{c}}^{(1)}$ is constituted of the subvectors $\mathbf{V}_m^H\tilde{\mathbf{c}}_m^{(1)}$ and

that $\|\mathbf{V}^H \tilde{\mathbf{c}}^{(1)}\|_{2|\mathcal{J}'} = \sum_{b=0}^{2B_D B_J - 1} \|(\mathbf{V}^H \tilde{\mathbf{c}}^{(1)})[b]\|_2$. Due to the way the groups $I'_b = \mathbf{S}'(\mathcal{B}_b)$ were defined, each of the group subvectors $(\mathbf{V}^H \tilde{\mathbf{c}}^{(1)})[b]$ only involves the $\Delta\tilde{m}$ subvectors $\mathbf{V}_m^H \tilde{\mathbf{c}}_m^{(1)}$ of $\mathbf{V}^H \tilde{\mathbf{c}}^{(1)}$ for those m for which $(m, i) \in \mathcal{B}_b$. Therefore, sub-dividing the sum into the corresponding parts yields the statement given above.

5.2.5 Simulation Results

In this section we experimentally demonstrate the advantage of the proposed compressive channel estimator utilizing group sparsity in comparison to the conventional compressive estimator of Section 5.1.1.

Simulation setup. We used the same simulation setup as described in Section 5.1.5, and even the same channel realizations generated by the channel simulator IlmProp [143]. The number of pilots was chosen as $Q = 1024$, corresponding to 6.25% of all the symbols. For the group sparse reconstruction in Step 2 of the proposed method G-BPDN, G-CoSaMP and G-OMP were used. The groups were defined as described in Section 5.2.1, i.e. $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ with $I_b = \mathbf{S}(\mathcal{B}_b)$, where the blocks \mathcal{B}_b were of size $\Delta\tilde{m} \times \Delta\tilde{i}$. For our simulations we used $\Delta\tilde{m}, \Delta\tilde{i} \in \{1, 2, 4, 8\}$. Note that for $\Delta\tilde{m} = \Delta\tilde{i} = 1$ the algorithms G-BPDN, G-CoSaMP and G-OMP correspond to their conventional counterparts and the method coincides with the conventional compressive estimator of Section 5.1.1.

We also generated an optimized basis according to Section 5.2.4 using a non-statistical design. The basic parameters were also chosen as in Section 5.1.5. Note that the same reasoning as in that section yields that only one basis has to be computed for each $\Delta\tilde{i} \in \{1, 2, 4, 8\}$ since $\mathbf{c}_m^{(p)} = \mathbf{c}_0^{(p)}$ for each $m = 0, \dots, D-1$ in this setting. The initialization matrix was chosen blockdiagonal with the $J \times J$ DFT matrix as blocks.

For each group size and both the 2D DFT and the optimized basis the noise parameter for G-BPDN was chosen as $\epsilon = 10^{-2.6}$, and the number of G-CoSaMP iterations was $n_{\text{G-CoSaMP}} = 16$. The group sparsity estimates S and S^{opt} for G-CoSaMP as well as the numbers of group iterations $n_{\text{G-OMP}}$ and $n_{\text{G-OMP}}^{\text{opt}}$ for G-OMP obviously had to be chosen differently for the various group sizes and bases. They are subsumed in Table 5.1. All these parameters were found experimentally. In order to solve G-BPDN we again used the toolbox SPGL1 [144], whereas we adapted the routine `cosamp.m` by David Mary [145] and our own OMP routine to solve G-CoSaMP and G-OMP, respectively.

Finally, the performance of all the channel estimation methods was measured using the normalized MSE.

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	100	65	42	24
2	75	38	22	13
4	49	24	12	7
8	26	13	6	4

 (a) Sparsity estimate S for G-CoSaMP using the 2D DFT basis

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	120	80	51	38
2	100	44	28	21
4	69	37	16	12
8	45	23	11	6

 (b) Number $n_{\text{G-OMP}}$ of G-OMP iterations using the 2D DFT basis

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	80	50	34	24
2	70	28	20	14
4	48	21	10	7
8	26	12	6	3

 (c) Sparsity estimate S^{opt} for G-CoSaMP using an optimized basis

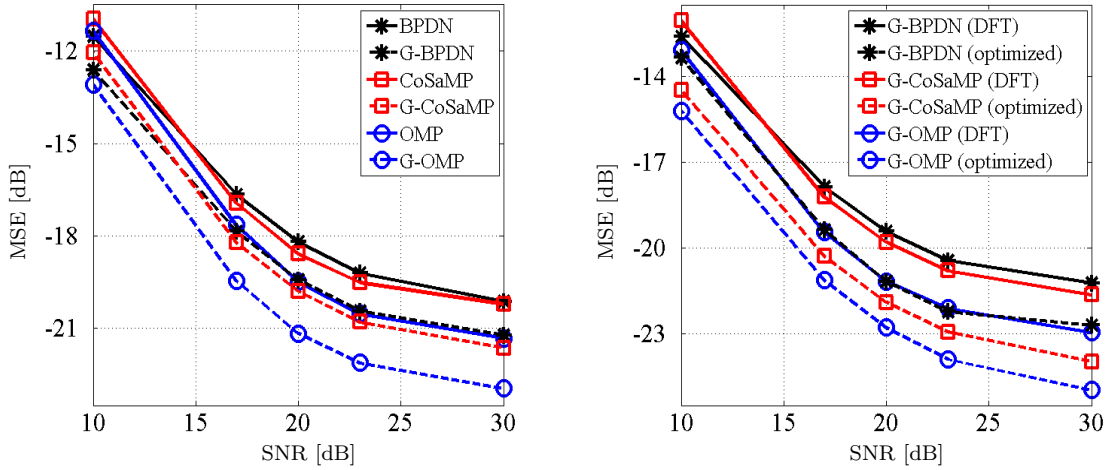
$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	100	55	42	35
2	85	31	24	20
4	67	26	13	10
8	46	20	10	5

 (d) Number $n_{\text{G-OMP}}^{\text{opt}}$ of G-OMP iterations using an optimized basis

Table 5.1: Simulation parameters for G-CoSaMP and G-OMP

Results. In Fig. 5.4(a) we plot the MSE of channel estimation versus the SNR for the proposed method using G-BPDN, G-CoSaMP and G-OMP with block size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$, comparing it to the conventional compressive estimator using BPDN, CoSaMP and OMP, i.e. $\Delta\tilde{m} \times \Delta\tilde{i} = 1 \times 1$. In both cases only the 2D DFT basis was used. It can clearly be seen that utilizing the group compressibility can improve the performance of the compressive estimator substantially. Note that if G-OMP is used the running time decreases at the same time. Furthermore, from Fig. 5.4(b) it becomes obvious that the use of a basis optimized as described above yields an additional performance gain (again keeping in mind that the computational complexity is increased at the same time).

In Fig. 5.5(a) we again plot the MSE of channel estimation versus the SNR, but here we plot the result for G-OMP using the 2D DFT basis for some of the block sizes. It can be seen that by choosing the block sizes appropriately the performance is improved compared to the case $\Delta\tilde{m} \times \Delta\tilde{i} = 1 \times 1$. However, a bad choice can also degrade the



(a) Compressive channel estimation utilizing group sparsity compared to conventional compressive channel estimation

(b) Comparison of the use of the 2D DFT basis to that of an optimized basis

Figure 5.4: Performance of the compressive channel estimator utilizing group sparsity for block size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$: MSE versus SNR

results. This is due to the fact that if the blocks are chosen too large in any direction many entries that do not belong to the essential support of \mathbf{x} (for example entries that are mostly due to noise) will still be part of the support of the estimate $\hat{\mathbf{x}}$, because they belong to a block also containing large entries. If the blocks are small enough only few of these "non-essential" entries will be chosen together with the large ones.

Furthermore, we show the performance of G-OMP using the 2D DFT basis for the different $\Delta\tilde{i} \in \{1, 2, 4, 8\}$ for fixed $\Delta\tilde{m}$ in Fig. 5.5(b), and the one for different $\Delta\tilde{m} \in \{1, 2, 4, 8\}$ for fixed $\Delta\tilde{i}$ in Fig. 5.5(c), both at a fixed SNR of 21 dB. Note that varying $\Delta\tilde{i}$ does not affect the performance as much as varying $\Delta\tilde{m}$, especially as $\Delta\tilde{m}$ becomes larger. This is due to the fact that the sparsity of the coefficients in the discrete m -direction is quite good (as explained in Section 5.1.3) and therefore $\Delta\tilde{m}$ should be chosen relatively small. On the contrary, the sparsity in the discrete i -direction is not as good, and therefore $\Delta\tilde{i}$ can be chosen larger.

Finally note that G-OMP gives the best results in practice again, whereas in our experiments G-BPDN had the shortest running time. This could be due to the implementations we used. Also, G-BPDN turned out to be more stable with regard to the choice of parameters than G-CoSaMP and G-OMP.

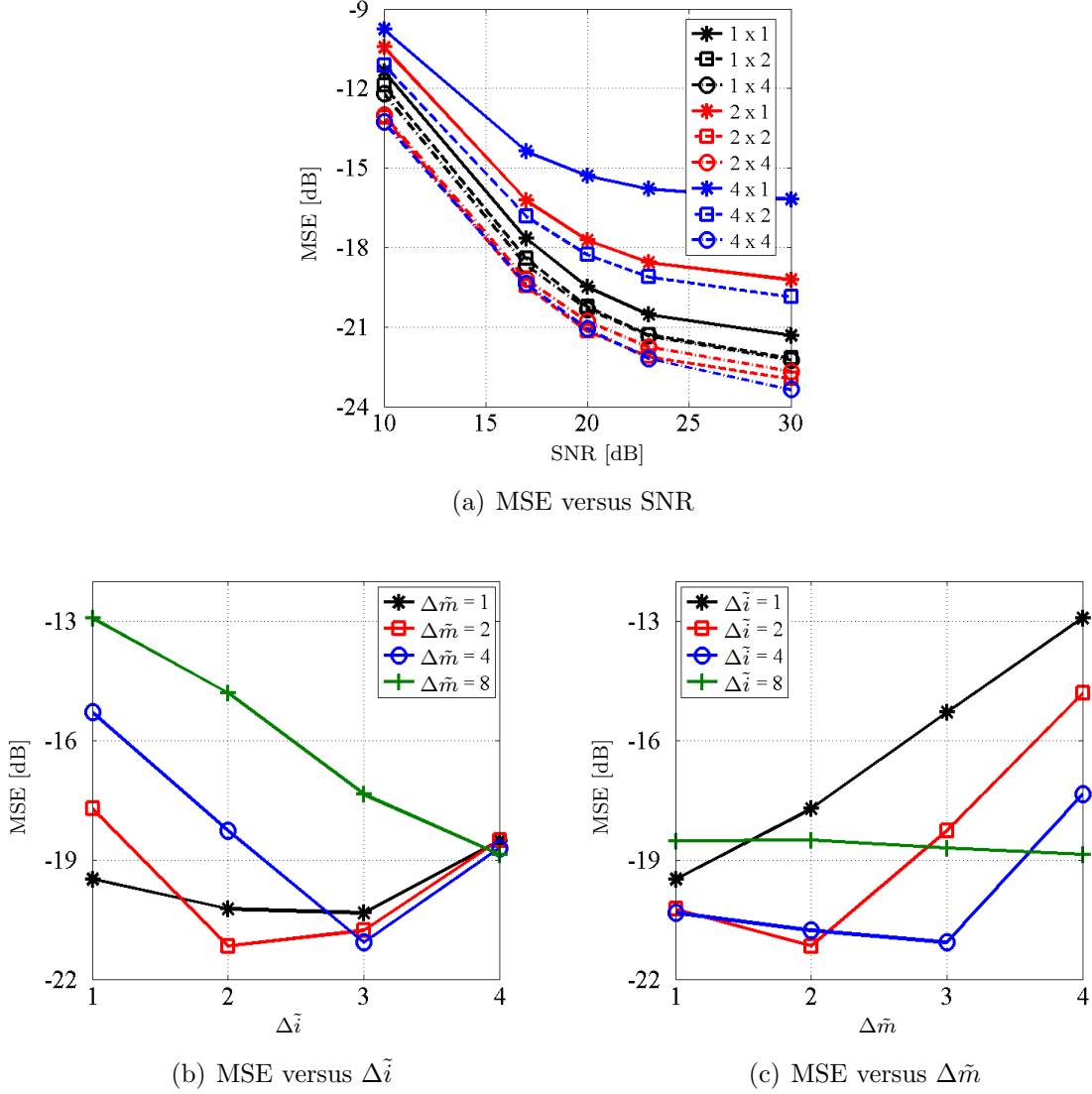


Figure 5.5: Performance of the compressive channel estimator utilizing group sparsity for various block sizes $\Delta \tilde{m} \times \Delta \tilde{i}$ using G-OMP

Conclusion. Obviously, utilizing the inherent group sparsity of the channel can considerably improve the performance of the compressive channel estimator. Nevertheless, an appropriate choice of the block size is also important, since otherwise the performance deteriorates again. Using a basis optimized as described above yields an additional performance gain due to the "better" sparsity of the expansion coefficients. Moreover, using G-OMP as reconstruction technique, the running time decreases the larger the groups are chosen.

Compressive Channel Estimation in MIMO Systems

6.1 Multichannel Compressive Channel Estimation

In this section we explain how the compressive channel estimator presented in Section 5.1.1 can be adapted to the MIMO case. In typical MIMO channel estimators each of the cross-channels connecting a transmit and a receive antenna is estimated individually, as explained in Section 3.6.3. The only publications referring to the use of CS methods in MIMO channel estimation seem to be [22, 23], where conventional CS methods are used in order to utilize the sparsity in the angular domain, and only frequency selective channels were considered. In this section, on the contrary, we present an estimation scheme for doubly selective channels utilizing the fact that the individual cross-channels can be considered jointly sparse. It is applicable with both zero and non-zero pilot patterns.

The rest of this section is organized as follows. In Section 6.1.1 we formulate the compressive channel estimator utilizing MCS methods, the computational complexity of which is analyzed in Section 6.1.2. The joint sparsity of the cross-channels in the delay-Doppler region is explored in Section 6.1.3. Furthermore, the basis optimization technique explained in Section 5.1.4 is adapted to the MIMO case in Section 6.1.4. Finally, the simulation results presented in Section 6.1.5 demonstrate the performance gain of the proposed MIMO channel estimator compared to the application of the conventional compressive estimator of Section 5.1.1 for each cross-channel individually.

6.1.1 The Method

The following exposition largely parallels the one in Section 5.1.1, since we make the same assumptions about each cross-channel of the MIMO channel, keeping in mind that each of them is in fact a SISO channel. Recall that the number of transmit and receive antennas is denoted by N_T and N_R , respectively, and that transmit and receive antennas are indexed by s and r , respectively. Moreover, each pair of transmit/receive antennas is denoted by $\theta = (r, s)$, where $\Theta = \{\theta = (r, s) \mid r = 1, \dots, N_R; s = 1, \dots, N_T\}$ denotes the set of all such pairs. As before we only consider underspread channels, for which the Fourier coefficients $\mathbf{F}_{m,i}$ in (3.21), which are $N_R \times N_T$ matrices with entries $F_{m,i}^{(\theta)}$ here, are effectively supported in $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$, by which we mean that outside of this box the matrices $\mathbf{F}_{m,i}$ are all-zero, i.e. $\mathbf{F}_{m,i} \equiv \mathbf{O}$. The constants D and J are again chosen such that $\Delta K := K/D$ and $\Delta L := L/J$ are integers, and J (and therefore also L) is even for mathematical convenience. Once more, the limiting cases of $D = K$ and/or $J = L$ are included. Therefore, following (3.21), we can again subsample the values of the channel coefficients $\mathbf{H}_{l,k}$ using the grid $\mathcal{G} := \{(\lambda\Delta L, \kappa\Delta K) \mid \lambda = 0, \dots, J-1; \kappa = 0, \dots, D-1\}$ to gather enough information to calculate the values of the matrices $\mathbf{H}_{l,k}$ at all time-frequency positions (l, k) for $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$. Then we get

$$\mathbf{H}_{\lambda\Delta L, \kappa\Delta K} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \mathbf{F}_{m,i} e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}, \quad (6.1)$$

which is equivalent to (5.1) entrywise. In Section 6.1.3 we explain in detail that the coefficients $F_{m,i}^{(\theta)}$ can be considered jointly compressible, but that their joint compressibility is again impaired by the leakage effect. Therefore, we generalize (6.1) to

$$\mathbf{H}_{\lambda\Delta L, \kappa\Delta K} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \mathbf{G}_{m,i} u_{m,i}[\lambda, \kappa], \quad (6.2)$$

where $\{u_{m,i}[\lambda, \kappa]\}$ is an orthonormal 2D basis chosen such that the expansion coefficients $G_{m,i}^{(r,s)} := [\mathbf{G}_{m,i}]_{r-1, s-1}$ are even "more jointly compressible" than the Fourier coefficients $F_{m,i}^{(\theta)}$. One way of constructing such a basis is explained in Section 6.1.4. Evidently, the 2D DFT expansion (6.1) is a special case with $u_{m,i}[\lambda, \kappa] = (1/\sqrt{JD})e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}$ and coefficients $\mathbf{G}_{m,i} = \sqrt{JD}\mathbf{F}_{m,i}$.

From here on we have to proceed differently. Let $\mu := (l, k)$ index the time-frequency positions. For pilot-aided channel estimation we now choose N_T linearly independent

pilot vectors $\mathbf{p}^{(s)} = [p^{(1,s)}, \dots, p^{(N_T, s)}]^T$ for $s = 1, \dots, N_T$, as well as N_T pairwise disjoint subsets $\mathcal{P}^{(s)} \subseteq \mathcal{G}$ of the subsampled time-frequency grid \mathcal{G} , each of the same size $|\mathcal{P}^{(s)}| = Q$. This corresponds to the non-zero pilot pattern and obviously includes the zero pattern as a special case (see Section 3.6.3). Then, each pilot vector $\mathbf{p}^{(s)}$ is transmitted at all pilot positions $\mu_q^{(s)} \in \mathcal{P}^{(s)}$, $q = 0, \dots, Q-1$, so that in total $N_T Q$ pilot vectors are transmitted. Now let $\tilde{G}_{m,i}^{(\theta)} := \mathbf{g}_{m,i}^{(r)T} \mathbf{p}^{(s)}$, where $\mathbf{g}_{m,i}^{(r)}$ denotes the r -th row of the matrix $\mathbf{G}_{m,i}$. Then, by writing $\mu_q^{(s)} = (\lambda_q^{(s)} \Delta L, \kappa_q^{(s)} \Delta K)$, restricting (6.2) to the pilot positions, and inserting it into the approximate input-output relation (3.20), we obtain

$$r_{\mu_q^{(s)}}^{(r)} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \tilde{G}_{m,i}^{(\theta)} u_{m,i}[\lambda_q^{(s)}, \kappa_q^{(s)}] + \tilde{z}_{\mu_q^{(s)}}^{(r)} \quad (6.3)$$

for all $\mu_q^{(s)} \in \mathcal{P}^{(s)}$, $q = 0, \dots, Q-1$, and all $s = 1, \dots, N_T$, $r = 1, \dots, N_R$.

In order to rewrite this in matrix notation recall the definition of \mathbf{U} of Section 5.1.1 as the $JD \times JD$ matrix with columns $\mathbf{u}_{m,i} = \text{vec}\{u_{m,i}[\lambda, \kappa]\}_{\lambda, \kappa}$. Furthermore, we define $\Phi^{(s)} := \sqrt{JD/Q} \mathbf{U}^{(s)}$, where $\mathbf{U}^{(s)}$ denotes the $Q \times JD$ submatrix of \mathbf{U} constituted of the Q rows corresponding to the pilot positions $\mathcal{P}^{(s)}$. Finally, let $\mathbf{x}^{(\theta)} := \sqrt{Q/JD} \text{vec}\{\tilde{G}_{m,i}^{(\theta)}\}_{m,i}$, $\mathbf{y}^{(\theta)} := [r_0^{(\theta)}, \dots, r_{Q-1}^{(\theta)}]^T$ and $\mathbf{z}^{(\theta)} := [z_0^{(\theta)}, \dots, z_{Q-1}^{(\theta)}]^T$, with $r_q^{(\theta)} := r_{\mu_q^{(s)}}^{(r)}$ and $z_q^{(\theta)} := z_{\mu_q^{(s)}}^{(r)}$ (where $\theta = (r, s)$). Then we can rewrite (6.3) as

$$\mathbf{y}^{(\theta)} = \Phi^{(s)} \mathbf{x}^{(\theta)} + \mathbf{z}^{(\theta)}, \quad \theta = (r, s) \in \Theta. \quad (6.4)$$

Since by assumption the coefficients $G_{m,i}^{(\theta)}$ are jointly compressible, the rows $\mathbf{g}_{m,i}^{(r)T}$ of $\mathbf{G}_{m,i}$ are either all "small" for (m, i) outside of the essential joint support, or all "large". Therefore, the coefficients $\tilde{G}_{m,i}^{(\theta)} = \mathbf{g}_{m,i}^{(r)T} \mathbf{p}^{(s)}$, and in turn the vectors $\mathbf{x}^{(\theta)}$ are also jointly compressible, which yields that (6.4) can be identified as an instance of the MCS problem (4.7) and can therefore be solved by using MCS techniques (see Section 4.4). Each measurement matrix $\Phi^{(s)}$ is again constructed from the renormalized matrix $\sqrt{JD/Q} \mathbf{U}$ by taking the rows corresponding to the pilot positions $\mathcal{P}^{(s)}$. Therefore, motivated by Theorem 4.2.6, we choose these pilot positions uniformly at random from the subsampled grid \mathcal{G} , so that we can expect $\Phi^{(s)}$ to have a small RIC with high probability, which typically induces good reconstruction quality. Note that in fact Theorem 4.2.6 cannot be applied to this case, because the fact that the pilot sets $\mathcal{P}^{(s)}$ have to be chosen pairwise disjoint contradicts the basic assumption of the theorem, namely that each row of \mathbf{U} has equal probability of being chosen. Nevertheless, we use Theorem 4.2.6 as a motivation for our construction of the measurement matrices $\Phi^{(s)}$. An example of such a pilot scheme

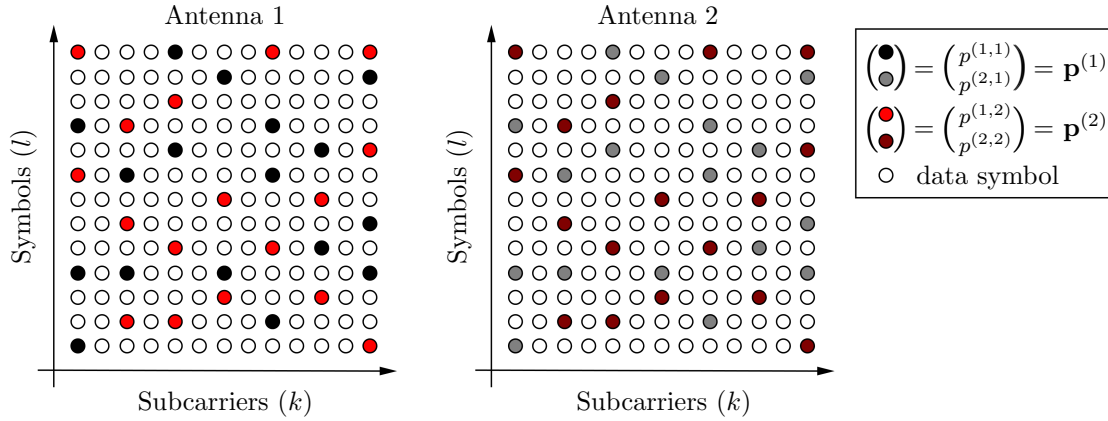


Figure 6.1: Pilot arrangement in a 2×2 MIMO system, where the grid \mathcal{G} has spacings $\Delta K = 2$ and $\Delta L = 1$.

is depicted in Fig. 6.1. Here, the pilot vector $\mathbf{p}^{(1)} = [p^{(1,1)}, p^{(2,1)}]^T$ is transmitted at all pilot positions in $\mathcal{P}^{(1)}$, i.e. $p^{(1,1)}$ (black disc) is transmitted from transmit antenna 1 and $p^{(2,1)}$ (gray disc) from transmit antenna 2 at all time-frequency positions $(l, k) \in \mathcal{P}^{(1)}$. In analogy, for $\mathbf{p}^{(2)} = [p^{(1,2)}, p^{(2,2)}]^T$, $p^{(1,2)}$ (red disc) is transmitted from transmit antenna 1 and $p^{(2,2)}$ (maroon disc) from transmit antenna 2 at all the pilot positions in $\mathcal{P}^{(2)}$.

Now we can formulate the multichannel compressive channel estimator as follows. For all $s = 1, \dots, N_T$ choose pilot sets $\mathcal{P}^{(s)} \subset \mathcal{G}$ uniformly at random from the subsampled grid \mathcal{G} , pairwise disjoint, and choose linearly independent pilot vectors $\mathbf{p}^{(s)}$, which are communicated to the receiver before the start of data transmission and stay fixed therein.

Step 1. For each $\theta = (r, s) \in \Theta$ stack the received symbols at the pilot positions corresponding to transmit antenna s into the vector $\mathbf{y}^{(\theta)}$ and run any MCS algorithm to obtain estimates $\hat{\mathbf{x}}^{(\theta)}$ of the vectors $\mathbf{x}^{(\theta)}$.

Step 2. Rescale these estimates $\hat{\mathbf{x}}^{(\theta)}$ with $\sqrt{JD/Q}$ to get estimates $\hat{G}_{m,i}^{(\theta)}$ of $\tilde{G}_{m,i}^{(\theta)}$.

Step 3. Calculate $\hat{\mathbf{g}}_{m,i}^{(r)} = \mathbf{P}^{-T} [\tilde{G}_{m,i}^{(r,1)}, \dots, \tilde{G}_{m,i}^{(r,N_T)}]^T$, where $\mathbf{P} := [\mathbf{p}^{(1)} \dots \mathbf{p}^{(N_T)}]$, which is non-singular since the pilot vectors $\mathbf{p}^{(s)}$ were chosen linearly independent.

Step 4. Calculate estimates of the subsampled matrices $\mathbf{H}_{\lambda\Delta L, \kappa\Delta K}$ from (6.2).

Step 5. Invert (6.1) to obtain estimates of the 2D DFT coefficient matrices $\mathbf{F}_{m,i}$ for $m = 0, \dots, D-1$ and $i = -J/2, \dots, J/2-1$. Note that by assumption $\mathbf{F}_{m,i}$ vanishes for all other indices.

Step 6. Calculate estimates of all the channel coefficient matrices $\mathbf{H}_{l,k}$ by using the 2D DFT expansion (3.21).

Note that in the special case where the 2D DFT basis is used Steps 4 and 5 can be omitted, since in that case we have $\mathbf{G}_{m,i} = \sqrt{JD}\mathbf{F}_{m,i}$.

For the multichannel sparse reconstruction in Step 1 any of the MCS methods described in Section 4.4 can be used, such as DCS-SOMP, M-CoSaMP as well as G-BPDN, G-OMP or G-CoSaMP (after reformulation as a group sparse problem). Alternatively, also M-BPDN or S-OMP can be used. In order to see this note that all the equations in (6.4) corresponding to cross-channels θ originating from the same transmit antenna share the same matrix $\Phi^{(s)}$, since this matrix only depends on the transmit antenna s . Therefore, Step 1 can be divided into N_T subproblems, each of which can be solved by using M-BPDN or S-OMP.

Regarding the quality of channel estimation we present results for G-BPDN and G-CoSaMP as reconstruction techniques in Step 1 (again for the reformulation as a group sparse problem). To do so, recall the definitions of $\mathbf{x} = [\mathbf{x}^{(\theta_1)T} \dots \mathbf{x}^{(\theta_{N_R N_T})T}]^T$ and $\mathbf{z} = [\mathbf{z}^{(\theta_1)T} \dots \mathbf{z}^{(\theta_{N_R N_T})T}]^T$ as the stackings of the vectors $\mathbf{x}^{(\theta)}$ and $\mathbf{z}^{(\theta)}$ for some ordering $\{\theta_1, \dots, \theta_{N_R N_T}\}$ of Θ , respectively, as well as the partition $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ from (4.9), which gathers the entries of the subvectors $\mathbf{x}^{(\theta)}$ with the same index into one group. Then we can obtain the following result.

Theorem 6.1.1. *Let $\mathcal{S} \subseteq \{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ be any set of size $|\mathcal{S}| = S$. Furthermore, let*

$$C_{\mathbf{G},\mathcal{S}} := \sum_{(m,i) \notin \mathcal{S}} \|\mathbf{G}_{m,i}\|_F,$$

as well as $\epsilon_n := \|\mathbf{z}\|_2$ and $\epsilon > 0$.

1. Use G-BPDN to solve the reformulated problem in Step 1 of the multichannel compressive channel estimator. If all the matrices $\Phi^{(s)}$ satisfy the RIP with RIC $\delta_{2S}^{(s)} < \sqrt{2}-1$, respectively, and if $\epsilon_n < \epsilon$, we can bound the estimation error as

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)}|^2} \leq C'_0 S^{-1/2} C_{\mathbf{G},\mathcal{S}} + C'_1 \epsilon,$$

with the constants $C'_0 := c_0 \sqrt{\frac{LK}{JD}} C_{\mathbf{P}}$ and $C'_1 := c_1 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F$, where $C_{\mathbf{P}} := \|\mathbf{P}\|_F \|\mathbf{P}^{-1}\|_F$, and c_0 and c_1 are the constants from Theorem 4.3.4.

2. Using G-CoSaMP with i steps in Step 1, assume that each matrix $\Phi^{(s)}$ satisfies the RIP with RIC $\delta_{4S}^{(s)} < 0.1$, and assume that $\epsilon_n < \epsilon$. Then

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} \leq \tilde{C}_0 (1 + S^{-1/2}) C_{\mathbf{G},S} + \tilde{C}_1 \epsilon + \tilde{C}_2,$$

with the constants $\tilde{C}_0 := 20\sqrt{\frac{LK}{JD}} C_{\mathbf{P}}$, $\tilde{C}_1 := 20\sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F$ and $\tilde{C}_2 := 2^{-i} \sqrt{\frac{LK}{Q}} C_{\mathbf{P}} \left(\sum_{(m,i),\theta} |G_{m,i}^{(\theta)}|^2 \right)^{1/2}$. Note that again since the number of G-CoSaMP iterations can be chosen arbitrarily, \tilde{C}_2 can be made arbitrarily small.

Proof. The proof is an easy adaption of the proof of Theorem 5.1.1. To be more precise, the first part of that proof can be adopted without any change for each cross-channel θ individually. Defining $\mathbf{h}^{(\theta)} = \text{vec}\{H_{l,k}^{(\theta)}\}_{l,k}$ and $\mathbf{g}^{(\theta)} = \text{vec}\{G_{m,i}^{(\theta)}\}_{m,i}$ and the associated vectors of estimates $\hat{\mathbf{h}}^{(\theta)} = \text{vec}\{\hat{H}_{l,k}^{(\theta)}\}_{l,k}$ and $\hat{\mathbf{g}}^{(\theta)} = \text{vec}\{\hat{G}_{m,i}^{(\theta)}\}_{m,i}$, (5.6) yields

$$\sqrt{\sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} = \|\mathbf{h}^{(\theta)} - \hat{\mathbf{h}}^{(\theta)}\|_2 = \sqrt{\frac{LK}{JD}} \|\mathbf{g}^{(\theta)} - \hat{\mathbf{g}}^{(\theta)}\|_2$$

for each $\theta \in \Theta$. Next, let $\tilde{\mathbf{G}}_{m,i}$ be the $N_R \times N_T$ matrix with entries $[\tilde{\mathbf{G}}_{m,i}]_{r-1,s-1} = G_{m,i}^{(r,s)}$ ($r = 1, \dots, N_R; s = 1, \dots, N_T$). Then, by definition of $\tilde{G}_{m,i}^{(\theta)}$, we have $\tilde{\mathbf{G}}_{m,i} = \mathbf{G}_{m,i} \mathbf{P}$, which yields $\mathbf{G}_{m,i} = \tilde{\mathbf{G}}_{m,i} \mathbf{P}^{-1}$. Furthermore, denoting its estimate by $\hat{\tilde{\mathbf{G}}}_{m,i}$, using $\tilde{\mathbf{g}}^{(\theta)} = \sqrt{JD/Q} \mathbf{x}^{(\theta)}$ and the submultiplicativity of the Frobenius norm (see (2.4)), we have

$$\begin{aligned} \sum_{\theta \in \Theta} \|\mathbf{g}^{(\theta)} - \hat{\mathbf{g}}^{(\theta)}\|_2^2 &= \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \|\mathbf{G}_{m,i} - \hat{\mathbf{G}}_{m,i}\|_F^2 = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \|(\tilde{\mathbf{G}}_{m,i} - \hat{\tilde{\mathbf{G}}}_{m,i}) \mathbf{P}^{-1}\|_F^2 \\ &\leq \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \|\tilde{\mathbf{G}}_{m,i} - \hat{\tilde{\mathbf{G}}}_{m,i}\|_F^2 \|\mathbf{P}^{-1}\|_F^2 = \|\mathbf{P}^{-1}\|_F^2 \sum_{\theta \in \Theta} \|\tilde{\mathbf{g}}^{(\theta)} - \hat{\tilde{\mathbf{g}}}^{(\theta)}\|_2^2 \\ &\leq \|\mathbf{P}^{-1}\|_F^2 \frac{JD}{Q} \sum_{\theta \in \Theta} \|\mathbf{x}^{(\theta)} - \hat{\mathbf{x}}^{(\theta)}\|_2^2, \end{aligned}$$

where the first and the second-to-last equalities are easily obtained by reordering the sums. Noting that $\sum_{\theta \in \Theta} \|\mathbf{x}^{(\theta)} - \hat{\mathbf{x}}^{(\theta)}\|_2^2 = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$, where $\hat{\mathbf{x}}$ is the estimate of \mathbf{x} , we therefore get

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} = \sqrt{\sum_{\theta \in \Theta} \frac{LK}{JD} \|\mathbf{g}^{(\theta)} - \hat{\mathbf{g}}^{(\theta)}\|_2^2} \leq \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F \|\mathbf{x} - \hat{\mathbf{x}}\|_2. \quad (6.5)$$

For part 1 of the theorem note that if all the matrices $\Phi^{(s)}$ satisfy the RIP with RIC $\delta_{2S}^{(s)} < \sqrt{2} - 1$, respectively, then, following Proposition 4.4.3, the stacked matrix Φ of (4.8) satisfies the G-RIP with G-RIC $\delta_{2S|\mathcal{J}} < \sqrt{2} - 1$. Therefore, Theorem 4.3.4 yields

$$\|\mathbf{x} - \hat{\mathbf{x}}\| \leq c_0 \frac{\sigma_S(\mathbf{x})_{2|\mathcal{J}}}{\sqrt{S}} + c_1 \epsilon.$$

By definition of $\sigma_S(\mathbf{x})_{2|\mathcal{J}}$ and \mathbf{x} we can easily proceed as

$$\begin{aligned} \sigma_S(\mathbf{x})_{2|\mathcal{J}} &\leq \sum_{(m,i) \notin \mathcal{S}} \sqrt{\frac{Q}{JD}} \left(\sum_{\theta \in \Theta} \left| [\tilde{\mathbf{g}}^{(\theta)}]_{\mathbf{s}(m,i)} \right|^2 \right)^{1/2} \\ &= \sqrt{\frac{Q}{JD}} \sum_{(m,i) \notin \mathcal{S}} \|\tilde{\mathbf{G}}_{m,i}\|_F. \end{aligned} \quad (6.6)$$

Since by definition $\tilde{\mathbf{G}}_{m,i} = \mathbf{G}_{m,i} \mathbf{P}$ we have $\|\tilde{\mathbf{G}}_{m,i}\|_F \leq \|\mathbf{G}_{m,i}\|_F \|\mathbf{P}\|_F$ again by (2.4), which finally yields

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} \leq c_0 \sqrt{\frac{LK}{JD}} \|\mathbf{P}^{-1}\|_F \|\mathbf{P}\|_F \frac{C_{\mathbf{G},\mathcal{S}}}{\sqrt{S}} + c_1 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F \epsilon.$$

The assumptions of part 2 yield that the stacked matrix Φ satisfies the G-RIP with G-RIC $\delta_{4S|\mathcal{J}} < 0.1$ (again using Proposition 4.4.3), which together with Theorem 4.3.5 and (6.6) gives

$$\begin{aligned} \sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} &\leq 20 \sqrt{\frac{LK}{JD}} C_{\mathbf{P}} (1 + S^{-1/2}) C_{\mathbf{G},\mathcal{S}} + 20 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F \epsilon \\ &\quad + 2^{-i} \sqrt{\frac{LK}{Q}} C_{\mathbf{P}} \left(\sum_{(m,i),\theta} |G_{m,i}^{(\theta)}|^2 \right)^{1/2}, \end{aligned}$$

which is exactly what we wanted to show. \square

Note again that the set \mathcal{S} can be chosen arbitrarily, but that the error will be small if \mathcal{S} covers the essential joint support of the coefficient matrices $\mathbf{G}_{m,i}$. If the 2D DFT basis is used, $C_{\mathbf{G},\mathcal{S}}$ again somehow characterizes the leakage occurring, whereas the second term is due to the noise and the third term in the result of part 2 of the theorem can be made arbitrarily small by choosing the number of G-CoSaMP iterations large enough (which at the same time increases the computational complexity).

Finally note that the SISO case is included as the special case where $N_T = N_R = 1$, where the proposed estimator coincides with the conventional compressive estimator of Section 5.1.1.

6.1.2 Computational Complexity

In order to analyze the computational complexity we again proceed step by step.

Step 1. We denote the complexity of solving the MCS problem by $\mathcal{O}(\text{MCS})$.

Step 2. This rescaling takes $\mathcal{O}(N_{\text{R}}N_{\text{T}}JD)$ operations.

Step 3. The matrix \mathbf{P} has size $N_{\text{T}} \times N_{\text{T}}$, and therefore the complexity of the computation of the products is $\mathcal{O}(N_{\text{R}}N_{\text{T}}^2JD)$. Note that \mathbf{P} only has to be inverted once *before* starting data transmission.

Step 4. Calculating the subsampled channel coefficient matrices $\mathbf{H}_{\lambda\Delta L, \kappa\Delta K}$ according to (6.2) can be done entrywise, i.e. channel-by-channel, and therefore typically comes at the cost of $\mathcal{O}(N_{\text{R}}N_{\text{T}}(JD)^2)$ operations, unless the matrix \mathbf{U} allows a faster implementation of the vector-matrix product.

Step 5. Proceeding entrywise, inverting (6.1) can be done very efficiently by using the FFT in $\mathcal{O}(N_{\text{R}}N_{\text{T}}JD \log(JD))$ operations.

Step 6. By the same arguments (3.21) can be implemented with $\mathcal{O}(N_{\text{R}}N_{\text{T}}LK \log(LK))$ operations using the FFT.

Thus, the multichannel compressive estimator can be implemented using

$$\mathcal{O}(\text{MCS}) + \mathcal{O}(N_{\text{R}}N_{\text{T}}(JD)^2) + \mathcal{O}(N_{\text{R}}N_{\text{T}}LK \log(LK))$$

operations, since typically $N_{\text{T}} \ll JD$ in practice. Recall from Section 4.4 that explicit values for the computational complexity of the MCS algorithms heavily depend on the implementation. For the implementations mentioned in Section 4.3 and Section 4.4, recall that the measurement matrices $\Phi^{(s)}$ are $Q \times JD$ matrices and that only N_{T} different matrices are involved, and thus $\mathcal{O}(\text{MCS}) = \mathcal{O}(n_{\text{DCS-SOMP}}(N_{\text{T}}JDn_{\text{DCS-SOMP}} + N_{\text{R}} \sum_s \Phi^{(s)}))$ for DCS-SOMP (following the implementation of OMP in [102]). For the reformulation as a group sparse problem, note that the application of Φ or Φ^H can be implemented blockwise because of its block-diagonal structure. Therefore, we have $\mathcal{O}(\Phi) \leq N_{\text{R}} \sum_s \mathcal{O}(\Phi^{(s)})$, and thus $\mathcal{O}(\text{MCS}) = \mathcal{O}(n_{\text{G-CoSaMP}}N_{\text{R}} \sum_s \Phi^{(s)})$ for G-CoSaMP. Here $n_{\text{DCS-SOMP}}$ and $n_{\text{G-CoSaMP}}$ denote the number of DCS-SOMP and G-CoSaMP iterations, respectively. Altogether, in a typical setting $\mathcal{O}(\text{MCS})$ will dominate the overall complexity. Also, the computational complexity of the multichannel compressive channel estimator basically equals that of the conventional compressive channel estimator used channel-per-channel.

Finally, note that if the 2D DFT basis is used throughout, Steps 4 and 5 can be omitted. which reduces the term in the middle to $\mathcal{O}(N_R N_T^2 J D)$. More importantly, the complexity of the MCS algorithms is also typically reduced, because then FFT-methods can be used to calculate the vector-matrix products.

6.1.3 Joint Delay-Doppler Sparsity

In order to analyze the joint compressibility of the MIMO channel, we use the same channel model (5.12) as in the SISO case for each cross-channel individually, i.e.

$$h^{(\theta)}(t, \tau) = \sum_{p=1}^P \eta_p^{(\theta)} \delta(\tau - \tau_p^{(\theta)}) e^{j2\pi\nu_p^{(\theta)} t}, \quad \theta \in \Theta, \quad (6.7)$$

where $\eta_p^{(\theta)}$, $\tau_p^{(\theta)}$ and $\nu_p^{(\theta)}$ denote the path gain, time delay and Doppler frequency shift for the cross channel indexed by θ corresponding to scatterer p , respectively. Following (5.15), this yields the discrete-delay-Doppler spreading functions

$$S_h^{(\theta)}[m, i] = \sum_{p=1}^P \eta_p^{(\theta)} e^{j\pi(\nu_p^{(\theta)} T_s - \frac{i}{N_0})} \binom{N_0-1}{i} \Lambda_p^{(\theta)}[m, i], \quad (6.8)$$

with the shifted leakage kernels

$$\Lambda_p^{(\theta)}[m, i] := \phi_p^{(\theta)}\left(m - \frac{\tau_p^{(\theta)}}{T_s}\right) \psi\left(i - \nu_p^{(\theta)} T_s N_0\right) \quad (6.9)$$

for each $\theta \in \Theta$, where $\phi_p^{(\theta)}(x) = \int_{\mathbb{R}} e^{-j2\pi\nu_p^{(\theta)} t} f_1(T_s x - t) f_2(t) dt$. The analysis carried out in Section 5.1.3 yields that each of these leakage kernels $\Lambda_p^{(\theta)}$ is essentially concentrated around a center point $\xi_p^{(\theta)} = (\tau_p^{(\theta)}/T_s, \nu_p^{(\theta)} T_s N_0)$. In the following we show that in typical scenarios these points differ minimally for the various cross-channels θ (but fixed scatterer p), from which we conclude that the supports of the different leakage kernels mostly overlap.

To do so, we describe the propagation scenario geometrically (see Fig. 6.2). Following the very basic identity $t = d/v$, where t , d and v denote time, distance and velocity, respectively, we can calculate the time delay $\tau_p^{(\theta)}$ for a specific scatterer p and the pair of antennas $\theta = (r, s)$ as follows. Let $\mathbf{w}_{T,p}^{(s)}$ and $\mathbf{w}_{R,p}^{(r)}$ be the vectors connecting and let $w_{T,p}^{(s)} := \|\mathbf{w}_{T,p}^{(s)}\|_2$ and $w_{R,p}^{(r)} := \|\mathbf{w}_{R,p}^{(r)}\|_2$ denote their respective lengths. By denoting the speed of light by c , we get

$$\tau_p^{(\theta)} = \frac{w_{T,p}^{(s)} + w_{R,p}^{(r)}}{c}. \quad (6.10)$$

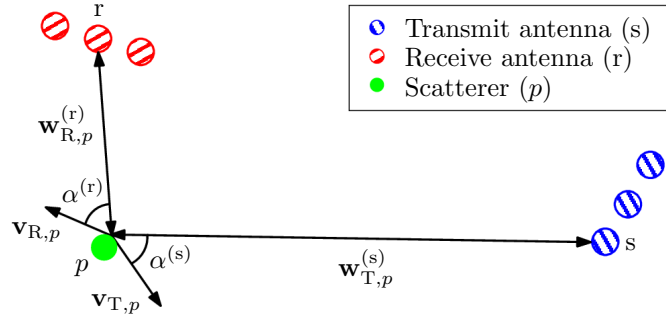


Figure 6.2: Geometric illustration of a propagation path from a transmit antenna s to a receive antenna r via a scatterer p in a 3×3 MIMO system

To describe the Doppler effect, on the other hand, assume that the source of an electromagnetic wave with frequency f_0 moves towards an observer with relative velocity v at an angle α relative to the direction from the observer to the source. Then the Doppler frequency shift ν that occurs is approximately given by $\nu = f_0 \frac{v}{c} \cos \alpha$, where we do not take relativistic effects into account since the velocities that are involved typically are very small compared to the speed of light. Now, we define $\mathbf{v}_{T,p}$ and $\mathbf{v}_{R,p}$ as the velocity vectors of scatterer p relative to transmitter and receiver, respectively, as well as $v_{T,p} := \|\mathbf{v}_{T,p}\|_2$ and $v_{R,p} := \|\mathbf{v}_{R,p}\|_2$. Note that in this geometric model we do not take rotations of the transmitter and/or the receiver into account, which would yield different velocity vectors for different transmit/receive antennas. Since such rotations, which are due to the rotation of the transmitting and/or receiving device, typically do not occur very quickly in practice, the velocity vectors will only differ very little. Although these rotations could easily be accounted for in the analysis we choose to ignore them for the sake of simplicity of notation. Furthermore note that since transmitter, receiver and scatterers are moving, the Doppler effect actually occurs twice. At the scatterer, the observed frequency will be $f_1 = f_0 + \nu_{T,p}^{(\theta)}$ with the Doppler frequency shift

$$\nu_{T,p}^{(\theta)} = f_0 \frac{v_{T,p}}{c} \frac{\mathbf{v}_{T,p}^T \mathbf{w}_{T,p}^{(s)}}{v_{T,p} w_{T,p}^{(s)}} = \frac{f_0}{c} \frac{\mathbf{v}_{T,p}^T \mathbf{w}_{T,p}^{(s)}}{w_{T,p}^{(s)}}, \quad (6.11)$$

where f_0 denotes the carrier frequency, and we have used the fact that $\cos \alpha = \mathbf{v}_{T,p}^T \mathbf{w}_{T,p}^{(s)} / (v_{T,p} w_{T,p}^{(s)})$, with α denoting the angle between $\mathbf{v}_{T,p}$ and $\mathbf{w}_{T,p}^{(s)}$. The observed frequency

at the receiver is then given by $f_2 = f_1 + \nu_{R,p}^{(\theta)}$ with

$$\nu_{R,p}^{(\theta)} = \frac{f_1}{c} \frac{\mathbf{v}_{R,p}^T \mathbf{w}_{R,p}^{(r)}}{w_{R,p}^{(r)}}.$$

Thus, by inserting f_1 we get $f_2 = f_0 + \nu_{T,p}^{(\theta)} + \nu_{R,p}^{(\theta)}$, which yields the overall Doppler frequency shift

$$\nu_p^{(\theta)} = \nu_{T,p}^{(\theta)} + \nu_{R,p}^{(\theta)}. \quad (6.12)$$

Now we can bound the differences $\Delta\tau_p^{(\theta_1, \theta_2)} := |\tau_p^{(\theta_1)} - \tau_p^{(\theta_2)}|$ and $\Delta\nu_p^{(\theta_1, \theta_2)} := |\nu_p^{(\theta_1)} - \nu_p^{(\theta_2)}|$ of the time delays and the Doppler frequency shifts, respectively, between two different cross-channels θ_1 and θ_2 . For the time delays we have

$$\begin{aligned} \Delta\tau_p^{(\theta_1, \theta_2)} &= \left| \tau_p^{(\theta_1)} - \tau_p^{(\theta_2)} \right| \\ &= \frac{1}{c} \left| w_{T,p}^{(s_1)} + w_{R,p}^{(r_1)} - w_{T,p}^{(s_2)} - w_{R,p}^{(r_2)} \right| \\ &\leq \frac{1}{c} \left(\left| w_{T,p}^{(s_1)} - w_{T,p}^{(s_2)} \right| + \left| w_{R,p}^{(r_1)} - w_{R,p}^{(r_2)} \right| \right), \end{aligned}$$

following (6.10). Obviously, the difference of the path lengths $\left| w_{T,p}^{(s_1)} - w_{T,p}^{(s_2)} \right|$ can not be larger than the distance between the two transmit antennas s_1 and s_2 . This distance is bounded by d_T , which denotes the maximal distance between any two transmit antennas. By using the same argument for the receiver side and denoting the maximal distance between any two receive antennas by d_R , we obtain $\Delta\tau_p^{(\theta_1, \theta_2)} \leq \tau_B$, where

$$\tau_B = \frac{d_T + d_R}{c}.$$

For the Doppler frequency shifts we can use (6.12) to get

$$\begin{aligned} \Delta\nu_p^{(\theta_1, \theta_2)} &= \left| \nu_p^{(\theta_1)} - \nu_p^{(\theta_2)} \right| \\ &= \left| \nu_{T,p}^{(\theta_1)} + \nu_{R,p}^{(\theta_1)} - \nu_{T,p}^{(\theta_2)} - \nu_{R,p}^{(\theta_2)} \right| \\ &\leq \left| \nu_{T,p}^{(\theta_1)} - \nu_{T,p}^{(\theta_2)} \right| + \left| \nu_{R,p}^{(\theta_1)} - \nu_{R,p}^{(\theta_2)} \right|. \end{aligned}$$

Again considering the transmitter side first and using (6.11) we get

$$\begin{aligned}
 \left| \nu_{T,p}^{(\theta_1)} - \nu_{T,p}^{(\theta_2)} \right| &= \frac{f_0}{c} \left| \frac{\mathbf{v}_{T,p}^T \mathbf{w}_{T,p}^{(s_1)}}{w_{T,p}^{(s_1)}} - \frac{\mathbf{v}_{T,p}^T \mathbf{w}_{T,p}^{(s_2)}}{w_{T,p}^{(s_2)}} \right| \\
 &\leq \frac{f_0}{c} \left| \frac{\mathbf{v}_{T,p}^T (\mathbf{w}_{T,p}^{(s_1)} - \mathbf{w}_{T,p}^{(s_2)})}{w_{T,p,\min}^{(s_1,s_2)}} \right| \\
 &\leq \frac{f_0}{c} \frac{v_{T,p} d_T}{w_{T,p,\min}^{(s_1,s_2)}},
 \end{aligned}$$

where we have set $w_{T,p,\min}^{(s_1,s_2)} := \min_{s_1,s_2} \{w_{T,p}^{(s_1)}, w_{T,p}^{(s_2)}\}$, and we have used the Cauchy-Schwarz inequality, as well as the fact that $\|\mathbf{w}_{T,p}^{(s_1)} - \mathbf{w}_{T,p}^{(s_2)}\|_2 \leq d_T$, to obtain the last inequality. Defining $w_{R,p,\min}^{(r_1,r_2)} := \min_{r_1,r_2} \{w_{R,p}^{(r_1)}, w_{R,p}^{(r_2)}\}$, and using the same arguments for the receiver side we get $\Delta \nu_p^{(\theta_1,\theta_2)} \leq \nu_{B,p}^{(\theta_1,\theta_2)}$ with

$$\nu_{B,p}^{(\theta_1,\theta_2)} := \frac{1}{c} \left(\frac{f_0 v_{T,p} d_T}{w_{T,p,\min}^{(s_1,s_2)}} + \frac{f_1 v_{R,p} d_R}{w_{R,p,\min}^{(r_1,r_2)}} \right).$$

Defining $\nu_B := \max_{p,\theta_1 \neq \theta_2} \{\nu_{B,p}^{(\theta_1,\theta_2)}\}$, the center points $\xi_p^{(\theta_1)}$ and $\xi_p^{(\theta_2)}$ of the supports of the shifted leakage kernels $\Lambda_p^{(\theta_1)}$ and $\Lambda_p^{(\theta_2)}$ for any two cross-channels θ_1 and θ_2 differ by at most τ_B/T_s in m -direction, and $\nu_B T_s N_0$ in i -direction. In a typical setting the antenna spacings d_T and d_R will be relatively small, and therefore τ_B will also be small. Furthermore, the velocities of transmitter, receiver and scatterers are very small compared to the speed of light, and typically the path lengths $w_{T,p}^{(s)}$ and $w_{R,p}^{(r)}$ are quite large compared to the antenna spacings d_T and d_R , which is why ν_B will be small, too. Thus, the center points $\xi_p^{(\theta_1)}$ and $\xi_p^{(\theta_2)}$ will be extremely close for each pair of cross-channels θ_1 and θ_2 .

Next, we bound the joint S -term approximation error $\sigma_S(\mathbf{\Lambda}_p)_F$ defined in (4.6), where $\mathbf{\Lambda}_p$ ($p = 1, \dots, P$) are now the $KN_0 \times N_R N_T$ matrices $\mathbf{\Lambda}_p := [\mathbf{\Lambda}_p^{(\theta_1)}, \dots, \mathbf{\Lambda}_p^{(\theta_{N_R N_T})}]$ for any ordering $\{\theta_1, \dots, \theta_{N_R N_T}\}$ of the cross-channels, with $\mathbf{\Lambda}_p^{(\theta)} = \text{vec}\{\Lambda_p^{(\theta)}[m, i]\}_{m,i}$. We choose the Frobenius norm as measure for the approximation error because it is the analog to the ℓ_2 -norm in the SISO case. Now let $\mathcal{M}_p^{(\theta)} := \{m \in \{0, \dots, K-1\} \mid |m - \tau_p^{(\theta)}/T_s| \leq \Delta m\}$ and $\mathcal{I}_p^{(\theta)} := \{i \in \{0, \dots, N_0-1\} \mid |(i - \nu_p^{(\theta)} T_s N_0)_{N_0}| \leq \Delta i\}$, where again $(i - x)_{N_0} := i + d_x N_0 - x$ such that $-N_0/2 \leq (i + d_x N_0 - x) \leq N_0/2$, and $\Delta m, \Delta i \geq 2$ are chosen as

in Section 5.1.3. Then, the analysis carried out in that section yields that

$$\begin{aligned} \sum_{(m,i) \notin \mathcal{M}_p^{(\theta)} \times \mathcal{I}_p^{(\theta)}} \left| \Lambda_p^{(\theta)}[m, i] \right|^2 &\leq \tilde{D}_0 \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1} + \tilde{D}_1 (\Delta i - 1)^{-1} \\ &\quad + \tilde{D}_2 (\Delta i - 1)^{-1} \left(1 + \frac{\Delta m - 1}{|x_0|} \right)^{-2s+1} \end{aligned} \quad (6.13)$$

for each cross-channel $\theta \in \Theta$ individually, where \tilde{D}_0 , \tilde{D}_1 and \tilde{D}_2 are the constants from (5.18). Next, let $\mathcal{M}'_p := \bigcup_{\theta \in \Theta} \mathcal{M}_p^{(\theta)}$ and $\mathcal{I}'_p := \bigcup_{\theta \in \Theta} \mathcal{I}_p^{(\theta)}$. Then, defining $\Delta m' := (2\Delta m + 1) + \lceil \tau_B / T_s \rceil$ and $\Delta i' := (2\Delta i + 1) + \lceil \nu_B T_s N_0 \rceil$, the analysis given above yields that $|\mathcal{M}'_p| \leq \Delta m'$ and $|\mathcal{I}'_p| \leq \Delta i'$. Moreover, we obviously have $\mathcal{M}_p^{(\theta)} \subseteq \mathcal{M}'_p$ and $\mathcal{I}_p^{(\theta)} \subseteq \mathcal{I}'_p$ for all $\theta \in \Theta$. Therefore, setting $S = \Delta m' \Delta i'$ and using (6.13) for each cross-channel yields

$$\begin{aligned} (\sigma_S(\Lambda_p)_F)^2 &\leq \sum_{(m,i) \notin \mathcal{M}'_p \times \mathcal{I}'_p} \sum_{\theta \in \Theta} \left| \Lambda_p^{(\theta)}[m, i] \right|^2 \\ &\leq N_R N_T \left(\tilde{D}_1 \left(1 + \frac{\Delta m' - 1}{|x_0|} \right)^{-2s+1} + \tilde{D}_2 (\Delta i' - 1)^{-1} \right. \\ &\quad \left. + \tilde{D}_3 (\Delta i' - 1)^{-1} \left(1 + \frac{\Delta m' - 1}{|x_0|} \right)^{-2s+1} \right). \end{aligned}$$

Since this bound decreases quite fast with increasing $\Delta m'$ and $\Delta i'$ (and therefore also $S = \Delta m' \Delta i'$), the shifted leakage kernels $\Lambda_p^{(\theta)}$ can be considered jointly compressible. As before, the same is then true for the spreading functions $S_h^{(\theta)}[m, i]$ (see (6.8)), and, in turn, for the 2D DFT coefficients $F_{m,i}^{(\theta)}$ (see (3.22)).

Finally note that since τ_B and ν_B are very small in typical scenarios, the joint sparsity parameter $\Delta m' \Delta i'$ of the leakage kernels $\Lambda_p^{(\theta)}$ will not be much larger than the individual sparsity parameters $(2\Delta m + 1)(2\Delta i + 1)$.

6.1.4 Basis Optimization

The basis optimization technique described in Section 5.1.4 can also be adapted to the multichannel setting to improve the performance of the multichannel compressive channel estimator. To do so, recall the channel model (6.7) specialized to the case $P = 1$, i.e. the single-scatterer channel $h^{(\theta)}(t, \tau) = \eta_1^{(\theta)} \delta(\tau - \tau_1^{(\theta)}) e^{j2\pi \nu_1^{(\theta)} t}$ for $\theta \in \Theta$, where $\tau_1 := (\tau_1^{(1,1)}, \dots, \tau_1^{(N_R, N_T)})$ and $\nu_1 := (\nu_1^{(1,1)}, \dots, \nu_1^{(N_R, N_T)})$ are random and distributed according to the pdf $\wp(\tau_1, \nu_1)$. A non-statistical design is easily obtained by constructing $\wp(\tau_1, \nu_1)$

from uniform distributions. From here on, we omit the index 1. Following Section 4.4, we measure the joint sparsity of the coefficient vectors $\mathbf{g}^{(\theta)}$ by $\|\mathbf{G}\|_{2,1}$, where \mathbf{G} is the $JD \times N_{\text{R}}N_{\text{T}}$ matrix $\mathbf{G} = [\mathbf{g}^{(\theta_1)} \dots \mathbf{g}^{(\theta_{N_{\text{R}}N_{\text{T}}})}]$ for an arbitrary but fixed ordering $\{\theta_1, \dots, \theta_{N_{\text{R}}N_{\text{T}}}\}$ of Θ . Recalling (6.2), the goal of this section therefore is to find orthonormal basis functions $u_{m,i}[\lambda, \kappa]$ for each $m = 0, \dots, D-1$ such that $\mathbb{E}\{\|\mathbf{G}\|_{2,1}\}$ is minimized. In order to do so, we again consider the special case (5.22), i.e. $u_{m,i}[\lambda, \kappa] = (1/\sqrt{D})v_{m,i}[\lambda]e^{-j2\pi\frac{\kappa m}{D}}$. Recall the definition of \mathbf{V}_m for $m = 0, \dots, D-1$ as the $J \times J$ matrices corresponding to the columnwise stacking of the basis functions $v_{m,i}$. Following Section 5.1.4, we define the vectors $\mathbf{c}_m^{(\theta)} := [C^{(\theta)}[m, 0], \dots, C^{(\theta)}[m, J-1]]^T$ for

$$C^{(\theta)}[m, \lambda] := \sum_{i=-J/2}^{J/2-1} \alpha_{m,i}^{(\theta)} e^{j2\pi\frac{\lambda i}{D}} = \sum_{i=-J/2}^{J/2-1} \beta_{m,i}^{(\theta)} v_{m,i}[\lambda]$$

according to (5.20) and (5.21), where

$$\alpha_{m,i}^{(\theta)} := \eta^{(\theta)} \sum_{q=0}^{N-1} e^{j\pi(\nu^{(\theta)}T_s - \frac{i+qL}{N_0})(N_0-1)} \psi\left(i + qL - \nu^{(\theta)}T_s N_0\right) A_{\gamma,g}^*\left(m, \frac{i+qL}{N_0}\right)$$

according to (5.19). Moreover, we define $\tilde{\mathbf{c}}_m^{(\theta)} := \sqrt{D}\phi_p^{(\theta)}(m - \tau^{(\theta)}/T_s)\mathbf{c}_m^{(\theta)}$. Then, by following (5.23), the vectors $\mathbf{g}_m^{(\theta)} := [G_{m,-J/2}^{(\theta)}, \dots, G_{m,J/2-1}^{(\theta)}]^T$ become $\mathbf{g}_m^{(\theta)} = \mathbf{V}_m^H \tilde{\mathbf{c}}_m^{(\theta)}$ for each $m = 0, \dots, D-1$ and $\theta \in \Theta$. Gathering these vectors into the $J \times N_{\text{R}}N_{\text{T}}$ matrices $\mathbf{G}_m := [\mathbf{g}_m^{(\theta_1)} \dots \mathbf{g}_m^{(\theta_{N_{\text{R}}N_{\text{T}}})}]$ and $\tilde{\mathbf{C}}_m := [\tilde{\mathbf{c}}_m^{(\theta_1)} \dots \tilde{\mathbf{c}}_m^{(\theta_{N_{\text{R}}N_{\text{T}}})}]$, we thus have

$$\mathbf{G}_m = \mathbf{V}_m^H \tilde{\mathbf{C}}_m, \quad m = 0, \dots, D-1. \quad (6.14)$$

Recalling (2.3) note that obviously

$$\|\mathbf{G}\|_{2,1} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \|\mathbf{G}\|_2^{(\mathbf{S}(m,i))} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} \|\mathbf{G}_m\|_2^{(i+J/2)} = \sum_{m=0}^{D-1} \|\mathbf{G}_m\|_{2,1},$$

where \mathbf{S} is the one-to-one mapping of (5.8), and $\|\mathbf{G}_m\|_2^{(i+J/2)}$ denotes the ℓ_2 -norm of the $(i+J/2)$ -th row of \mathbf{G}_m . Since the expectation is linear, we again can decompose the problem of minimizing $\mathbb{E}\{\|\mathbf{G}\|_{2,1}\}$ into the D smaller problems of minimizing $\mathbb{E}\{\|\mathbf{G}_m\|_{2,1}\}$, which can be written as

$$\hat{\mathbf{V}}_m = \arg \min_{\mathbf{V}_m \in \mathcal{U}} \mathbb{E}\{\|\mathbf{V}_m^H \tilde{\mathbf{C}}_m\|_{2,1}\}$$

for each $m = 0, \dots, D-1$, where \mathcal{U} denotes the set of all unitary $J \times J$ matrices. Once more, by using a Monte-Carlo approximation (see (2.13)), (6.14) and the linearity of the expectation we can redefine these problems as

$$\hat{\mathbf{V}}_m = \arg \min_{\mathbf{V}_m \in \mathcal{U}} \sum_{\rho} \sum_{i=0}^{J-1} \|\mathbf{V}_m^H (\tilde{\mathbf{C}}_m)_\rho\|_2^{(i)}, \quad m = 0, \dots, D-1,$$

where $(\tilde{\mathbf{C}}_m)_\rho$ denotes the value of $\tilde{\mathbf{C}}_m$ for a sample of the random vector (τ, ν) drawn from its pdf $\wp(\tau, \nu)$. As before we set $\eta^{(\theta)} = 1$ for all $\theta \in \Theta$ in this simple setting, although their distribution could also be taken into account (see [3] for the SISO case).

To solve these problems the algorithm presented in Section 5.1.4 only has to be adapted somewhat, but nevertheless we state the entire algorithm for the sake of completeness.

- **Input.** Initialization matrix $\mathbf{V}_m^{(0)}$, pdf $\wp(\tau, \nu)$, initial threshold $\gamma^{(0)}$.
- **Initialization.** $n = 0$
- **while** stopping criterion not met **do**
 1. Solve the convex problem $\hat{\mathbf{A}}_m^{(n)} = \arg \min_{\mathbf{A} \in \mathcal{H}_n} \sum_{\rho} \sum_{i=0}^{J-1} \|(\mathbf{I}_J + j\mathbf{A})\mathbf{V}_m^{(n)} (\tilde{\mathbf{C}}_m)_\rho\|_2^{(i)}$, where \mathcal{H}_n denotes the set of all Hermitian matrices \mathbf{A} with $\|\mathbf{A}\|_\infty < \gamma^{(n)}$
 2. Set $\hat{\mathbf{V}}_m^{(n)} = e^{j\hat{\mathbf{A}}_m^{(n)}} \mathbf{V}_m^{(n)}$
 3. **if** $\sum_{\rho} \sum_{i=0}^{J-1} \|\hat{\mathbf{V}}_m^{(n)} (\tilde{\mathbf{C}}_m)_\rho\|_2^{(i)} < \sum_{\rho} \sum_{i=0}^{J-1} \|\mathbf{V}_m^{(n)} (\tilde{\mathbf{C}}_m)_\rho\|_2^{(i)}$
 update $\mathbf{V}_m^{(n+1)} = \hat{\mathbf{V}}_m^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}$
else
 update $\mathbf{V}_m^{(n+1)} = \mathbf{V}_m^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}/2$
 4. Iterate $n \mapsto n + 1$
- **Output.** $\mathbf{V}_m = \mathbf{V}_m^{(n)}$

As before, the algorithm is stopped if either the maximal number of iterations is reached or if the threshold $\gamma^{(n)}$ falls below a prescribed value. Also, we choose the $J \times J$ DFT matrix as initialization matrix $\mathbf{V}^{(0)}$ because the analysis carried out in Section 6.1.3 yields that in this case the expansion coefficients $G_{m,i}^{(\theta)} = \sqrt{JD}F_{m,i}^{(\theta)}$ are already jointly compressible to a certain degree. Finally note again that the optimization problem does not depend on the actual channel parameters or the receive signal, and it therefore only has to be solved once before the start of data transmission.

6.1.5 Simulation Results

In the following we present simulation results demonstrating the performance gain obtained by the proposed multichannel compressive channel estimator in comparison to the conventional compressive channel estimator of Section 5.1.1. The simulation setup was similar to the one described in Section 5.1.5.

MC MIMO system parameters. We simulated a CP-MIMO-OFDM system with again $K = 512$ subcarriers, CP-length $L_{\text{cp}} = 128$, center frequency $f_0 = 5$ GHz and bandwidth $1/T_s = 5$ MHz. We also used 4-QAM symbols and root-raised cosine filters $f_1(t) = f_2(t)$ with roll-off factor $\alpha = 1/4$. The number of transmit and receive antennas was $N_T = N_R \in \{1, 2, 3, 4\}$.

Channel. Doubly selective channels during blocks of $L = 32$ OFDM symbols were again generated using `IlmProp` [143]. Actually, we used the same channel realizations for the SISO case as we did here, by simply reducing the channel to one of the pairs of transmit/receive antennas. For details see Section 5.1.5.

Subsampling and pilot setup. We again used a subsampled time-frequency grid \mathcal{G} with spacing $\Delta L = 1$ and $\Delta K = 4$. We fixed the number of pilots as $Q = |\mathcal{P}^{(s)}| = 1024$ ($s = 1, \dots, N_T$), meaning that $6.25 \cdot N_T\%$ of all the symbols were pilots. Then, we chose a set of size $N_T Q$ uniformly at random from \mathcal{G} and divided it into N_T sets of equal size Q in order to obtain the pilot sets $\mathcal{P}^{(s)}$ for $s = 1, \dots, N_T$. This way of choosing the pilot sets not only assures that they are pairwise disjoint, but also that the measurement matrices $\Phi^{(s)}$ are "close to" being constructed according to Theorem 4.2.6 (i.e. the pilot positions are "close to" being chosen uniformly at random). Therefore, we expected these matrices to have good reconstruction properties.

Performance measure. Again the performance is measured by the MSE normalized by the mean energy of all the channel coefficients, i.e. $\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)}|^2 / \sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k}^{(\theta)}|^2$.

Channel estimation. In [23] it was proposed to use the conventional compressive channel estimator for each cross-channel of a MIMO system individually. Therefore, we compared this method to the proposed multichannel compressive estimator. We used BPDN, CoSaMP and OMP for the channel-per-channel application of the conventional compressive channel estimator, and DCS-SOMP as well as G-BPDN and G-CoSaMP for the (reformulated) multichannel estimator. Furthermore, we constructed an optimized

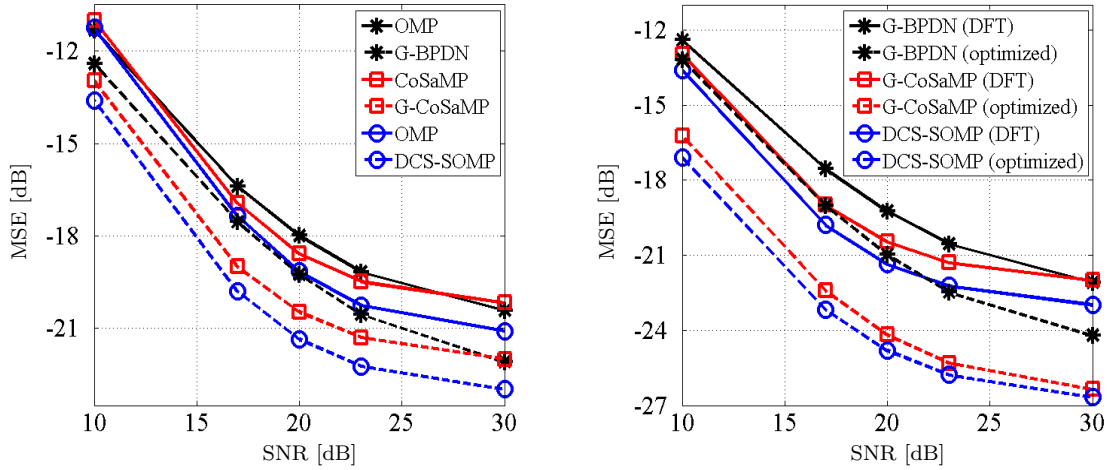
basis according to Section 6.1.4 again using a non-statistical design. The basic parameters were the same as in Section 5.1.5. The pdf was chosen as $\wp(\tau, \nu) = \wp(\tau^{(1,1)}, \nu^{(1,1)}) \wp(\tau^{(1,2)}, \nu^{(1,2)} | \tau^{(1,1)}, \nu^{(1,1)}) \dots \wp(\tau^{(N_R, N_T)}, \nu^{(N_R, N_T)} | \tau^{(1,1)}, \nu^{(1,1)})$, where each factor is uniform in a rectangular region. The region for the first factor was determined as in Section 5.1.5, whereas the regions for the remaining factors were determined by τ_B and ν_B (see Section 6.1.3). A difference of up to 3 bins for the discrete Doppler shift $\nu_p^{(\theta)}$ for the various cross-channels $\theta \in \Theta$ was accounted for, corresponding to approximately 1.4 Hz.

For BPDN and G-BPDN the noise parameter ϵ was again chosen as $\epsilon = 10^{-2.6}$. The number of CoSaMP and G-CoSaMP iterations was 16 for both the 2D DFT and the optimized basis. The sparsity estimates S and S^{opt} as well as the numbers of DCS-SOMP iterations $n_{\text{DCS-SOMP}}$ and $n_{\text{DCS-SOMP}}^{\text{opt}}$ were chosen according to Table 6.1 (note that the parameters for CoSaMP and OMP correspond to the first column, i.e. $N_T = 1$). All the parameters were found experimentally.

Par. \ N_T	1	2	3	4
S	100	130	140	150
S^{opt}	80	80	80	90
$n_{\text{DCS-SOMP}}$	120	150	170	180
$n_{\text{DCS-SOMP}}^{\text{opt}}$	100	80	90	90

Table 6.1: Simulation parameters S and S^{opt} for G-CoSaMP as well as $n_{\text{DCS-SOMP}}$ and $n_{\text{DCS-SOMP}}^{\text{opt}}$ for DCS-SOMP for the various numbers $N_T = N_R$ of transmit/receive antennas

Results. In Fig. 6.3(a) we plot the MSE of channel estimation versus the SNR for a MIMO system with $N_T = N_R = 2$ transmit/receive antennas. It can be readily seen that utilizing the joint compressibility of the expansion coefficients substantially improves the performance of the compressive channel estimator. Also, the use of an optimized basis yields an additional performance gain, as seen in Fig. 6.3(b). This is due to the "better" joint sparsity of the coefficient matrices $\mathbf{G}_{m,i}$ with respect to this basis in comparison to the 2D DFT coefficient matrices $\mathbf{F}_{m,i}$. But note again that the improved performance comes at the cost of an increased computational complexity.



(a) Multichannel compressive channel estimation compared to channel-per-channel conventional compressive channel estimation

(b) Comparison of the use of the 2D DFT basis to that of an optimized basis

Figure 6.3: Performance of the multichannel compressive channel estimator in a system with $N_T = N_R = 2$ transmit/receive antennas: MSE versus SNR

In Fig. 6.4 the MSE of channel estimation is plotted versus the number of transmit/receive antennas $N_T = N_R \in \{1, 2, 3, 4\}$ at a fixed SNR of 21 dB, and we only show the results for the 2D DFT basis. Note that for this figure we only used 150 different simulation runs. Clearly the performance of the proposed multichannel compressive estimator improves with an increasing number of antennas, since with a larger number of jointly compressible signals the estimation of the essential joint support should become more accurate. This behavior has been studied in [129] (although the authors only considered the algorithms M-BPDN and SOMP). The flattening of the curves, on the contrary, cannot easily be explained. We think it is due to the fact that the cross-channels are not only jointly compressible, but also similar to some degree. In other words, not only their essential supports mostly overlap, but their entries are also quite similar. In [129] it is explained that the worst case scenario for MCS is if all the jointly sparse (or compressible) signals are equal, because in this case no additional support information can be gained despite the larger number of signals. Accordingly, we are actually close to the worst case scenario, which could lead to the flattening of the curves. However, this effect is alleviated since in our case N_T different measurement matrices are involved, and therefore "different" information can be gained about the quite similar jointly compressible signals. In contrast, the conventional compressive channel estima-

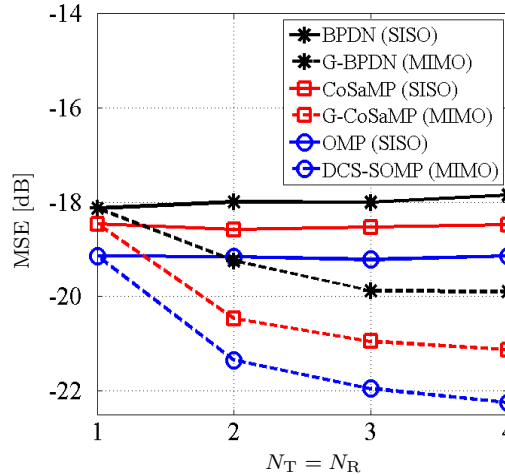


Figure 6.4: MSE of multichannel compressive channel estimation versus the number of transmit/receive antennas $N_T = N_R$

tor used channel-per-channel is not affected by the number of antennas, therefore its performance stays quite constant.

In analogy to the SISO case, DCS-SOMP outperforms G-BPDN and G-CoSaMP, whereas G-BPDN is the fastest and most stable method regarding the choice of parameters. Finally note that the computational complexity of the proposed multichannel compressive estimator basically equals that of the conventional compressive estimator used channel-per-channel (taking into account that only N_T different measurement matrices are involved).

Conclusion. The multichannel compressive estimator has been seen to outperform the channel-per-channel use of the conventional compressive estimator at an equal computational cost. Furthermore, the use of an optimized basis yields an additional performance gain, though the computational complexity increases. All in all, the proposed estimator should clearly be favored in the MIMO case.

6.2 Multichannel Compressive Channel Estimation using Group Sparsity Methods

In Section 6.1.3 we have shown that the shifted leakage kernels of the different cross-channels in a MIMO system can be considered jointly compressible. Moreover, the

analysis given in Section 5.2.3 indicates that each of these cross-channels can additionally be considered group compressible. Therefore, we adapt the method presented in Section 6.1.1 by using MGSCS techniques to also take the group compressibility into account. We first present the general method in Section 6.2.1, followed by a short note on its computational complexity in Section 6.2.2. Then, the joint delay-Doppler group compressibility of the individual cross-channels is investigated in Section 6.2.3. After adapting of the basis optimization method described in Section 6.1.4 to this setting in Section 6.2.4, we finally present experimental results indicating that the proposed method outperforms the previously described methods in Section 6.2.5.

6.2.1 The Method

We consider the same setting as in Section 6.1.1. The only difference actually occurs in Step 2, where now MGSCS techniques are used.

In order to describe the method very briefly, first recall that we assume the 2D DFT coefficient matrices $\mathbf{F}_{m,i}$ to be supported inside the box $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ for J and D as in Section 6.1.1. To incorporate the group sparsity of these matrices (see Section 6.2.3), recall the partition of this set into the small blocks $\mathcal{B}_b = \{k_b \Delta \tilde{m}, \dots, (k_b + 1) \Delta \tilde{m}\} \times \{l_b \Delta \tilde{i}, \dots, (l_b + 1) \Delta \tilde{i}\}$ for $k_b \in \{0, \dots, B_D - 1\}$ as well as $l_b \in \{-B_J, \dots, B_J - 1\}$ as described in Section 5.2.1, which are again of size $\Delta \tilde{m} \times \Delta \tilde{i}$ for some integers $\Delta \tilde{m}$ and $\Delta \tilde{i}$ such that $B_D = D/\Delta \tilde{m}$ and $B_J = (J/2)/\Delta \tilde{i}$ are integers. Then, following Section 6.2.3, we can consider the 2D DFT coefficients $F_{m,i}^{(\theta)}$, or more generally the coefficients $G_{m,i}^{(\theta)}$ of (6.2), jointly group compressible with respect to these blocks. Recalling the partition $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ of $\{0, \dots, JD-1\}$ with the groups $I_b = \mathbf{S}(\mathcal{B}_b)$, where \mathbf{S} is the one-to-one mapping from (5.8), this can again be translated into the joint group compressibility of the vectors $\mathbf{x}^{(\theta)}$ in (6.4).

Now we can state the multichannel estimator utilizing group sparsity methods. Before starting data transmission pairwise disjoint pilot sets $\mathcal{P}^{(s)} \subset \mathcal{G}$ of equal size $|\mathcal{P}^{(s)}| \equiv Q$ are chosen uniformly at random from the subsampled grid \mathcal{G} (see Section 5.1.1) for all $s = 1, \dots, N_T$. Furthermore linearly independent pilot vectors $\mathbf{p}^{(s)}$ are chosen and communicated to the receiver. They stay fixed during data transmission. The receiver then proceeds as follows.

- Step 1.** For each $\theta = (r, s) \in \Theta$ stack the received symbols at the pilot positions corresponding to the transmit antenna s into the vector $\mathbf{y}^{(\theta)}$ and run any MGSCS algorithm to obtain estimates $\hat{\mathbf{x}}^{(\theta)}$ of the vectors $\mathbf{x}^{(\theta)}$.
- Step 2.** Rescale these estimates $\hat{\mathbf{x}}^{(\theta)}$ with $\sqrt{JD/Q}$ to get estimates $\hat{G}_{m,i}^{(\theta)}$ of $G_{m,i}^{(\theta)}$.
- Step 3.** Calculate $\hat{\mathbf{g}}_{m,i}^{(r)} = \mathbf{P}^{-T} [\tilde{G}_{m,i}^{(r,1)}, \dots, \tilde{G}_{m,i}^{(r,N_T)}]^T$, where $\mathbf{P} := [\mathbf{p}^{(1)} \dots \mathbf{p}^{(N_T)}]$, which is non-singular since the pilot vectors $\mathbf{p}^{(s)}$ were chosen linearly independent.
- Step 4.** Calculate estimates of the subsampled channel coefficient matrices $\mathbf{H}_{\lambda\Delta L, \kappa\Delta K}$ from (6.2).
- Step 5.** Invert (6.1) to obtain estimates of the 2D DFT coefficient matrices $\mathbf{F}_{m,i}$ for $m = 0, \dots, D-1$ and $i = -J/2, \dots, J/2-1$. Note that by assumption $\mathbf{F}_{m,i}$ vanishes for all other indices.
- Step 6.** Calculate estimates of all the channel coefficient matrices $\mathbf{H}_{l,k}$ by using the 2D DFT expansion (3.21).

Note that in the special case where the 2D DFT basis is used Steps 4 and 5 can be omitted, since in that case we have $\mathbf{G}_{m,i} = \sqrt{JD}\mathbf{F}_{m,i}$.

As already mentioned before, the only difference to the method presented in Section 6.1.1 is the reconstruction technique used in Step 1, where now MGSCS methods can be used. To do so, the MCS algorithm DCS-SOMP can easily be adapted to also take group sparsity into account. Alternatively, as explained in Section 4.5, we can transform the problem into a quite large GSCS problem by defining a new partition $\tilde{\mathcal{J}}$ according to (4.11). Then, recalling the vectors \mathbf{x} and \mathbf{z} from Section 6.1.1 as the stackings of the vectors $\mathbf{x}^{(\theta)}$ and $\mathbf{z}^{(\theta)}$ ($\theta \in \Theta$), respectively, we can obtain a bound for the estimation error of the proposed method for G-BPDN and G-CoSaMP.

Theorem 6.2.1. *Let $\mathcal{S} \subseteq \{0, \dots, 2B_D B_J - 1\}$ be any set of size $|\mathcal{S}| = S$. Furthermore, let*

$$C_{\mathbf{G}, \mathcal{S}, \mathcal{J}} := \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} \|\mathbf{G}_{m,i}\|_F^2 \right)^{1/2},$$

as well as $\epsilon_n := \|\mathbf{z}\|_2$ and $\epsilon > 0$.

1. Use G-BPDN to solve the reformulated problem in Step 1 of the multichannel compressive channel estimator. If all the matrices $\Phi^{(s)}$ satisfy the G-RIP with G-RIC $\delta_{2S|\mathcal{J}}^{(s)} < \sqrt{2} - 1$, respectively, and if $\epsilon_n < \epsilon$, we can bound the estimation error

as

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} \leq C'_0 S^{-1/2} C_{\mathbf{G}, \mathcal{S}, \mathcal{J}} + C'_1 \epsilon,$$

with the constants $C'_0 := c_0 \sqrt{\frac{LK}{JD}} C_{\mathbf{P}}$ and $C'_1 := c_1 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F$, where $C_{\mathbf{P}} := \|\mathbf{P}\|_F \|\mathbf{P}^{-1}\|_F$, while c_0 and c_1 are the constants from Theorem 4.3.4.

2. Using G-CoSaMP with i steps in Step 2, assume that each matrix $\Phi^{(s)}$ satisfies the G-RIP over \mathcal{J} with G-RIC $\delta_{4S|\mathcal{J}}^{(s)} < 0.1$, and assume that $\epsilon_n < \epsilon$. Then

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} \leq \tilde{C}_0 S^{-1/2} C_{\mathbf{G}, \mathcal{S}, \mathcal{J}} + \tilde{C}_1 \epsilon + \tilde{C}_2,$$

with the constants $\tilde{C}_0 := 20 \sqrt{\frac{LK}{JD}} C_{\mathbf{P}}$, $\tilde{C}_1 := 20 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F$ and $\tilde{C}_2 := 2^{-i} \sqrt{\frac{LK}{Q}} C_{\mathbf{P}} \left(\sum_{(m,i), \theta} |G_{m,i}^{(\theta)}|^2 \right)^{1/2}$. Note that again since the number of G-CoSaMP iterations can be chosen arbitrarily, \tilde{C}_2 can be made arbitrarily small.

Proof. The proof follows the proof of Theorem 6.1.1. First recall (6.5), namely

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \left| H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)} \right|^2} \leq \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F \|\mathbf{x} - \hat{\mathbf{x}}\|_2.$$

For part 1 of this theorem, note that the assumptions on $\Phi^{(s)}$ together with Corollary 4.5.2 yield that the G-RIC $\delta_{2S|\tilde{\mathcal{J}}}$ with respect to the modified partition $\tilde{\mathcal{J}}$ (see (4.11)) of the stacked measurement matrix Φ , which is constructed from N_R copies of each matrix $\Phi^{(s)}$ on its diagonal, satisfies $\delta_{2S|\tilde{\mathcal{J}}} < \sqrt{2} - 1$. This and the assumption on ϵ_n allow us to use Theorem 4.3.4, which yields

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq c_0 \frac{\sigma_S(\mathbf{x})_{2|\tilde{\mathcal{J}}}}{\sqrt{S}} + c_1 \epsilon$$

with the constants given therein. By definition of $\sigma_S(\mathbf{x})_{2|\tilde{\mathcal{J}}}$ we can again derive

$$\begin{aligned} \sigma_S(\mathbf{x})_{2|\tilde{\mathcal{J}}} &\leq \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} \sqrt{\frac{Q}{JD}} \sum_{\theta \in \Theta} \left| [\tilde{\mathbf{g}}^{(\theta)}]_{\mathbf{s}(m,i)} \right|^2 \right)^{1/2} \\ &\leq \sqrt{\frac{Q}{JD}} \|\mathbf{P}\|_F \sum_{b \notin \mathcal{S}} \left(\sum_{(m,i) \in \mathcal{B}_b} \|\mathbf{G}_{m,i}\|_F^2 \right)^{1/2}. \end{aligned}$$

which finally gives

$$\sqrt{\sum_{\theta \in \Theta} \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} |H_{l,k}^{(\theta)} - \hat{H}_{l,k}^{(\theta)}|^2} \leq c_0 \sqrt{\frac{LK}{JD}} \|\mathbf{P}^{-1}\|_F \|\mathbf{P}\|_F \frac{C_{\mathbf{G},\mathcal{S},\mathcal{J}}}{\sqrt{S}} + c_1 \sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F \epsilon.$$

For part 2 the assumptions on $\Phi^{(s)}$ (and thus on the stacked matrix Φ) and ϵ_n together with Theorem 4.3.5 yield

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq 20(1 + S^{-1/2}) \sigma_S(\mathbf{x})_{2|\tilde{\mathcal{J}}} + 20\epsilon + 2^{-i} \|\mathbf{x}\|_2.$$

Finally setting the constants $\tilde{C}_0 := 20\sqrt{\frac{LK}{JD}} C_{\mathbf{P}}$, $\tilde{C}_1 := 20\sqrt{\frac{LK}{Q}} \|\mathbf{P}^{-1}\|_F$ and $\tilde{C}_3 := \sqrt{\frac{LK}{Q}} C_{\mathbf{P}} \left(\sum_{(m,i),\theta} |G_{m,i}^{(\theta)}|^2 \right)^{1/2}$ finishes the proof. \square

As in all the previous results \mathcal{S} can be chosen arbitrarily, but in order to achieve a small error it should contain the groups covering the essential joint support of the coefficient matrices $\mathbf{G}_{m,i}$. In the case where the 2D DFT basis is used, $C_{\mathbf{G},\mathcal{S},\mathcal{J}}$ again characterizes the leakage occurring, the second term is due to the noise, and the third term in the error estimate of part 2 of the theorem can be made arbitrarily small by choosing the number of G-CoSaMP iterations large enough. Again, note that the computational cost of G-CoSaMP increases with a larger number of iterations (see Section 4.3).

6.2.2 Computational Complexity

Obviously, the computational complexity of the proposed method equals that of the multichannel estimator presented in Section 6.1.1 in each step but the first. Thus, it is given by

$$\mathcal{O}(\text{MGSCS}) + \mathcal{O}(N_{\text{R}} N_{\text{T}} (JD)^2) + \mathcal{O}(LK \log(LK)).$$

As already mentioned several times throughout this thesis, it is hard to give explicit values for $\mathcal{O}(\text{MGSCS})$, since it depends on the implementation. Nevertheless, it will dominate the overall complexity of the estimator in typical scenarios. Applying for example G-CoSaMP to the reformulated system yields $\mathcal{O}(\text{MGSCS}) = \mathcal{O}(n_{\text{G-CoSaMP}} N_{\text{R}} \sum_s \Phi^{(s)})$, where $n_{\text{G-CoSaMP}}$ denotes the maximal number of iterations. Note that the vector-matrix product by the stacked matrix Φ can again be implemented taking the block-diagonal structure into account, and thus reducing its complexity. In a straight forward way

the algorithm DCS-SOMP can be adapted to also incorporate group sparsity. Instead of adding only single indices to the joint support in each iteration, *Group DCS-SOMP* (G-DCS-SOMP) adds entire groups corresponding to the maximal ℓ_2 -norm of the group-subvectors $\mathbf{u}_{(i)}[b]$ of $\mathbf{u}_{(i)}$. Following the implementation of OMP in [102] the computational complexity of G-DCS-SOMP in this special setting (where only N_T different matrices are involved) is $\mathcal{O}(\text{G-DCS-SOMP}) = \mathcal{O}(N_T JD(n'_{\text{G-DCS-SOMP}})^2 + n_{\text{G-DCS-SOMP}} N_R \sum_s \Phi^{(s)})$, where again $n_{\text{G-DCS-SOMP}}$ and $n'_{\text{G-DCS-SOMP}}$ denote the number of G-DCS-SOMP iterations and the sum of the magnitudes of the chosen groups, respectively.

Furthermore, note that although the actual bounds do not change, taking the joint *group* compressibility into account will reduce the running time compared to the conventional multichannel estimator which only accounts for the joint compressibility (at least for some of the algorithms).

Finally, the proposed method obviously coincides with the compressive estimator utilizing group sparsity of Section 5.2.1 in the SISO case, i.e. $N_T = N_R = 1$, and with the multichannel compressive estimator of Section 6.1.1 in the case where each group only contains one index, i.e. $\Delta\tilde{m} = \Delta\tilde{i} = 1$.

6.2.3 Joint Delay-Doppler Group Sparsity

In Section 6.1.3 we have shown that the shifted leakage kernels $\Lambda_p^{(\theta)}$ defined in (6.9) can be considered jointly compressible. Furthermore, in Section 5.2.3 we have seen that each such leakage kernel can also be considered group compressible. Therefore, they are jointly group compressible as described in Section 4.5.

To be more precise, we again consider the same tiling of $\mathbb{Z} \times \mathbb{Z}$ into the blocks \mathcal{B}_b of size $\Delta\tilde{m} \times \Delta\tilde{i}$ as defined in Section 5.2.3. Then, the exact same arguments yield that the joint support of the leakage kernels $\Lambda_p^{(\theta)}$ for $\theta \in \Theta$ is contained in at most N'_Λ blocks with

$$N'_\Lambda := \left(\left\lceil \frac{2\Delta m'}{\Delta\tilde{m}} \right\rceil + 1 \right) \left(\left\lceil \frac{2\Delta i'}{\Delta\tilde{i}} \right\rceil + 1 \right),$$

where $\Delta m'$ and $\Delta i'$ are as defined in Section 6.1.3. Then, following the reasoning given in Section 5.2.3 and Section 6.1.3, the spreading functions $S_h^{(\theta)}[m, i]$ and, in turn, the 2D DFT coefficients $F_{m,i}^{(\theta)}$ are jointly group compressible, too.

Finally note that in this case also the coefficients $\tilde{G}_{m,i}^{(\theta)}$ are jointly group sparse over the partition \mathcal{J} , which is again defined as in Section 5.2.1 (see Section 6.1.1 for an explanation).

6.2.4 Basis Optimization

It is straight forward to combine the basis optimization methods from Section 6.1.4 and Section 5.2.4. To do so, recall the modified partition $\tilde{\mathcal{J}} := \{\tilde{I}_b\}_{b=0}^{B-1}$ from (4.11) with groups $\tilde{I}_b = \{k + iJD | k \in I_b, i = 0, \dots, N_R N_T - 1\}$, where $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ with $I_b = \mathbf{S}(\mathcal{B}_b)$ is the partition defined in Section 5.2.1. Then, following Section 4.3, the joint group sparsity of the coefficient vectors $\mathbf{g}^{(\theta)}$ can be measured by $\|\mathbf{g}\|_{2|\tilde{\mathcal{J}}}$, where $\mathbf{g} = [\mathbf{g}^{(\theta_1)T} \dots \mathbf{g}^{(\theta_{N_R N_T})T}]^T$ is the stacked coefficient vector, i.e.

$$\|\mathbf{g}\|_{2|\tilde{\mathcal{J}}} = \sum_{b=0}^{B-1} \left(\sum_{(m,i) \in I_b} \sum_{\theta \in \Theta} |G_{m,i}^{(\theta)}|^2 \right)^{1/2}. \quad (6.15)$$

To better fit the notation we have used in Section 6.1.4, recall the $JD \times N_R N_T$ matrix $\mathbf{G} = [\mathbf{g}^{(\theta_1)} \dots \mathbf{g}^{(\theta_{N_R N_T})}]$. Then we define $\mathbf{G}[b]$ as the $|I_b| \times N_R N_T$ submatrix constituted of the rows of \mathbf{G} corresponding to the indices in I_b for each $b = 0, \dots, B-1$. With this at hand, we can define $\|\mathbf{G}\|_{F|\mathcal{J}} := \sum_{b=0}^{B-1} \|\mathbf{G}[b]\|_F$, which is easily seen to coincide with the right side of (6.15). This, in turn, yields that $\|\mathbf{g}\|_{2|\tilde{\mathcal{J}}} = \|\mathbf{G}\|_{F|\mathcal{J}}$. Therefore, we can state the goal of this section as follows. For each $m = 0, \dots, D-1$ find orthonormal basis functions $u_{m,i}[\lambda, \kappa]$ such that $\mathbb{E}\{\|\mathbf{G}\|_{F|\mathcal{J}}\}$ is minimized, where the expectation is taken with respect to the random variables (τ, ν) , which are distributed according to a pdf $\wp(\tau, \nu)$.

Next, recall (6.14), namely $\mathbf{G}_m = \mathbf{V}_m^H \tilde{\mathbf{C}}_m$ for $m = 0, \dots, D-1$, with \mathbf{G}_m , \mathbf{V}_m and $\tilde{\mathbf{C}}_m$ defined as in Section 6.1.4, and the block-diagonal $JD \times JD$ matrix \mathbf{V} with the $J \times J$ blocks \mathbf{V}_m on its diagonal. Furthermore, in analogy to Section 5.2.4, we define the $JD \times N_R N_T$ matrices $\tilde{\mathbf{G}}$ and $\tilde{\mathbf{C}}$ as the "matrix-wise" stacking of the matrices \mathbf{G}_m and $\tilde{\mathbf{C}}_m$, respectively, "above" each other. Then, we can obviously write $\tilde{\mathbf{G}} = \mathbf{V}^H \tilde{\mathbf{C}}$. Now recall the one-to-one mapping \mathbf{S}' from (5.27), which corresponds to rowwise stacking of a matrix into a vector, and the partition $\mathcal{J}' = \{I'_b\}_{b=0}^{B-1}$ of $\{0, \dots, JD-1\}$ with the groups $I'_b = \mathbf{S}'(\mathcal{B}_b)$. The columns of $\tilde{\mathbf{G}}$ correspond to the rowwise stacking of the coefficients $G_{m,i}^{(\theta)}$, i.e. $[\tilde{\mathbf{G}}]_{\mathbf{S}'(m,i), \theta_n} = G_{m,i}^{(\theta_n)}$, whereas the columns of \mathbf{G} correspond to columnwise

stacking, i.e. $[\mathbf{G}]_{\mathbf{s}(m,i),\theta_n} = G_{m,i}^{(\theta_n)}$. Thus we have

$$\begin{aligned} \|\tilde{\mathbf{G}}\|_{F|\mathcal{J}'} &= \sum_{b=0}^{B-1} \left(\sum_{n=1}^{N_R N_T} \sum_{(m,i) \in \mathcal{B}_b} |[\tilde{\mathbf{G}}]_{\mathbf{s}'(m,i),\theta_n}|^2 \right)^{1/2} = \sum_{b=0}^{B-1} \left(\sum_{\theta \in \Theta} \sum_{(m,i) \in \mathcal{B}_b} |G_{m,i}^{(\theta)}|^2 \right)^{1/2} \\ &= \sum_{b=0}^{B-1} \left(\sum_{n=1}^{|\Theta|} \sum_{(m,i) \in \mathcal{B}_b} [\mathbf{G}]_{\mathbf{s}(m,i),\theta_n} \right)^{1/2} = \|\mathbf{G}\|_{F|\mathcal{J}}. \end{aligned}$$

Therefore, the optimization problem can be formulated as

$$\hat{\mathbf{V}} = \arg \min_{\mathbf{V} \in \tilde{\mathcal{U}}} \mathbb{E} \{ \|\mathbf{V}^H \tilde{\mathbf{C}}\|_{F|\mathcal{J}'} \},$$

where $\tilde{\mathcal{U}}$ again denotes the set of all blockdiagonal $JD \times JD$ matrices with unitary blocks of size $J \times J$ on the diagonal. As before, by using a Monte-Carlo approximation we finally state the problem as

$$\hat{\mathbf{V}} = \arg \min_{\mathbf{V} \in \tilde{\mathcal{U}}} \sum_{\rho} \|\mathbf{V}^H (\tilde{\mathbf{C}})_{\rho}\|_{F|\mathcal{J}'},$$

with $(\tilde{\mathbf{C}})_{\rho}$ denoting the value of $\tilde{\mathbf{C}}$ for a sample of the random vector $(\boldsymbol{\tau}, \boldsymbol{\nu})$ drawn from its pdf $\wp(\boldsymbol{\tau}, \boldsymbol{\nu})$. Once again, we have ignored the path gain, the distribution of which could also be taken into account (extending [3]).

Finally, we merely state the adaption of the optimization algorithm.

- **Input.** Initialization matrix $\mathbf{V}^{(0)}$, pdf $\wp(\boldsymbol{\tau}, \boldsymbol{\nu})$, initial threshold $\gamma^{(0)}$.
- **Initialization.** $n = 0$
- **while** stopping criterion not met **do**
 1. Solve the convex problem $\hat{\mathbf{A}}^{(n)} = \arg \min_{\mathbf{A} \in \tilde{\mathcal{H}}_n} \sum_{\rho} \|(\mathbf{I}_J + \mathbf{J}\mathbf{A})\mathbf{V}^{(n)}(\tilde{\mathbf{C}})_{\rho}\|_{F|\mathcal{J}'}$, where $\tilde{\mathcal{H}}_n$ denotes the set of all block-diagonal matrices \mathbf{A} with Hermitian blocks of size $J \times J$ on its diagonal, where $\|\mathbf{A}\|_{\infty} < \gamma^{(n)}$
 2. Set $\hat{\mathbf{V}}^{(n)} = e^{j\hat{\mathbf{A}}^{(n)}} \mathbf{V}_m^{(n)}$
 - if** $\sum_{\rho} \|\hat{\mathbf{V}}^{(n)}(\tilde{\mathbf{C}})_{\rho}\|_{F|\mathcal{J}'} < \sum_{\rho} \|\mathbf{V}^{(n)}(\tilde{\mathbf{C}})_{\rho}\|_{F|\mathcal{J}'}$
 - update $\mathbf{V}^{(n+1)} = \hat{\mathbf{V}}^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}$
 3. **else**
 - update $\mathbf{V}^{(n+1)} = \mathbf{V}^{(n)}$ and $\gamma^{(n+1)} = \gamma^{(n)}/2$
 4. Iterate $n \mapsto n + 1$
- **Output.** $\mathbf{V} = \mathbf{V}^{(n)}$

As always, the algorithm is stopped if $\gamma^{(n)}$ falls below a prescribed threshold or after the maximal number of iterations, and $\mathbf{V}^{(0)}$ is chosen with $J \times J$ DFT blocks on the diagonal. Finally, note that as explained in Section 5.2.4, this optimization problem can be decomposed into $B_J = D/\Delta\tilde{m}$ separate problems of dimension $J\Delta\tilde{m} \times J\Delta\tilde{m}$, each.

6.2.5 Simulation Results

We compared the proposed channel estimator to the compressive estimators of Section 5.1.1, Section 5.2.1 and Section 6.1.1.

Simulation setup. We used the same MC MIMO system parameters and pilot setup as described in Section 6.1.5, and we even used the same channel realizations generated by the simulation tool IlmProp. We chose the number of transmit/receive antennas to be $N_T = N_R = 2$, and we defined the groups as described in Section 6.2.1, i.e. $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ with $I_b = \mathbf{S}(\mathcal{B}_b)$, where the blocks \mathcal{B}_b were of size $\Delta\tilde{m} \times \Delta\tilde{i}$. For our simulations we used $\Delta\tilde{m}, \Delta\tilde{i} \in \{1, 2, 4, 8\}$. Note that for $\Delta\tilde{m} = \Delta\tilde{i} = 1$ the method coincides with the multichannel compressive estimator of Section 6.1.1, and for $N_T = N_R = 1$ it coincides with the estimator of Section 5.2.1 only utilizing group sparsity.

In addition, we generated an optimized basis according to Section 6.2.4, where once more a non-statistical design was used. The basic parameters were chosen as in Section 6.1.5. By the same reasoning as in that section and in Section 5.2.5 only one basis had to be computed for each $\Delta\tilde{i} \in \{1, 2, 4, 8\}$. As before, the optimization algorithm was initialized with a blockdiagonal matrix with $J \times J$ DFT matrices on its diagonal.

For the sparse reconstruction in Step 1 we used G-DCS-SOMP as well as G-BPDN and G-CoSaMP (for the reformulated problem). We chose the noise parameter for G-BPDN as $\epsilon = 10^{-2.6}$, and the number of G-CoSaMP iterations as $n_{\text{G-CoSaMP}} = n_{\text{G-CoSaMP}}^{\text{opt}} = 16$ for all group sizes and both the 2D DFT and the optimized basis. The sparsity estimates S and S^{opt} as well as the numbers $n_{\text{G-DCS-SOMP}}$ and $n_{\text{G-DCS-SOMP}}^{\text{opt}}$ were chosen according to Table 6.2. These parameters were again found experimentally.

Note that for the channel-per-channel use of the compressive estimator of Section 5.2.1 we used the parameters given in Table 5.1. Finally note that the performance is measured by the normalized MSE.

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	130	75	45	23
2	90	40	23	13
4	48	25	12	7
8	27	12	7	3

(a) Sparsity estimate S for G-CoSaMP using the 2D DFT basis

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	150	90	57	40
2	125	52	30	22
4	84	38	16	12
8	48	21	12	6

(b) Number $n_{\text{G-DCS-SOMP}}$ of G-DCS-SOMP iterations using the 2D DFT basis

$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	80	45	28	20
2	65	25	17	11
4	44	19	8	6
8	28	14	6	3

(c) Sparsity estimate S^{opt} for G-CoSaMP using an optimized basis

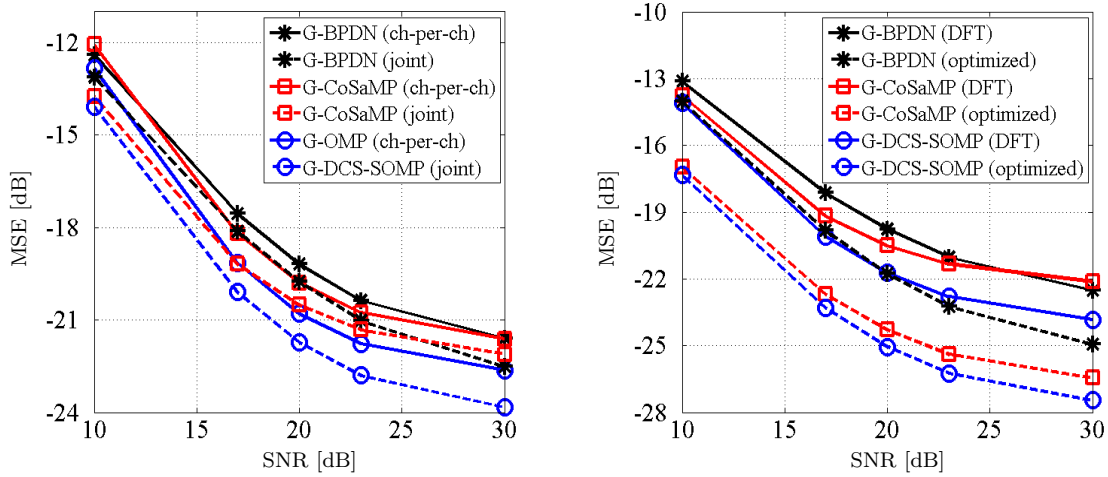
$\Delta\tilde{m} \backslash \Delta\tilde{i}$	1	2	4	8
1	80	49	32	22
2	75	28	17	14
4	62	24	9	8
8	43	19	8	4

(d) Number $n_{\text{G-DCS-SOMP}}^{\text{opt}}$ of G-DCS-SOMP iterations using an optimized basis

Table 6.2: Simulation parameters for the multichannel compressive estimator utilizing group sparsity using G-CoSaMP and G-DCS-SOMP

Results. In Fig. 6.5(a) we compare the performance of the proposed estimator to the compressive estimator of Section 5.2.1, used for each cross-channel individually, by plotting the MSE of channel estimation versus the SNR. Here the blocks defining the partition \mathcal{J} were of size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$. It is obvious that taking the joint group compressibility of the expansion coefficients into account considerably improves the performance of the compressive channel estimator. Moreover, the use of an optimized basis can be seen to yield another substantial improvement from Fig. 6.5(b), although the computational complexity is increased.

In Fig. 6.6 we compare the performance of the proposed method for different sizes of the blocks \mathcal{B}_b defining the partition \mathcal{J} . Here we only show the results for the use of G-DCS-SOMP and the 2D DFT basis. It can be seen that in this setting the block size



(a) Compressive channel estimation utilizing group sparsity for the 2D DFT basis: the multichannel version compared to the channel-per-channel version

(b) Comparison of the use of the 2D DFT basis to that of an optimized basis

Figure 6.5: Performance of the multichannel compressive estimator utilizing group sparsity for block size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$: MSE versus SNR

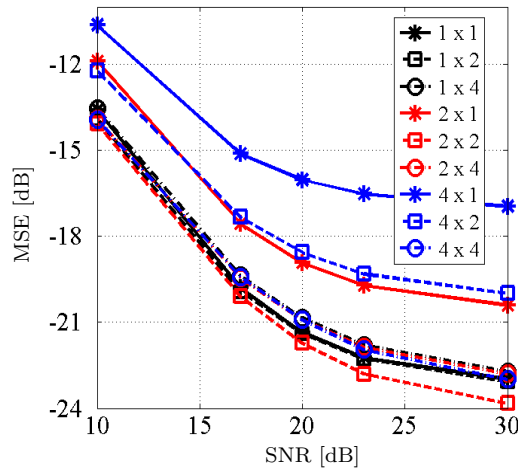


Figure 6.6: Performance of the multichannel compressive estimator utilizing group sparsity for various block sizes $\Delta\tilde{m} \times \Delta\tilde{i}$ using G-DCS-SOMP and the 2D DFT basis: MSE versus SNR

$\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$ renders the best performance, but that choosing $\Delta\tilde{m} \times \Delta\tilde{i} = 1 \times 1$ yields results quite similar to those obtained by most of the other block sizes. Thus, we conclude that utilizing the joint compressibility of the expansion coefficients is already close to optimal in this setting. Nevertheless, choosing the blocks somewhat larger yields a similar (or even better) performance, but reduces the running time of the MGSCS algorithm G-DCS-SOMP.

Conclusion. The multichannel compressive estimator utilizing group sparsity has been shown to outperform the conventional compressive estimator used channel-per-channel, and to perform comparable to the multichannel compressive estimator of Section 6.1.1, while having a shorter running time (if G-DCS-SOMP is used). On the one hand, using an optimized basis yields another performance gain, while on the other hand it leads to an increased computational complexity.

Compressive Channel Tracking

The basic idea of channel tracking is to utilize information about the channel from previous estimates to improve the quality of the present estimate. Several different tracking methods have been proposed for SISO and MIMO MC systems. Most of them make use of Kalman filter techniques, such as [26, 27]. In this chapter we adapt the compressive channel estimators presented before so that the channel can actually be tracked by utilizing MOD-CS methods. First, we consider the SISO case in Section 7.1 for notational convenience. Then, we briefly explain the extension to the MIMO case in Section 7.2, also taking the (joint) group compressibility into account. Finally, we present simulation results which indicate that the compressive channel tracker outperforms the previously introduced compressive channel estimators in Section 7.3.

7.1 Compressive Channel Tracking in SISO Systems

Here, we first present the general method in Section 7.1.1, followed by a short note on its computational complexity in Section 7.1.2. In Section 7.1.3 we finally show that typical channels can be considered approximately sequentially sparse, which motivates the use of MOD-CS techniques.

7.1.1 The Method

For the tracking scenario we assume that Ω blocks of L OFDM symbols each are transmitted. For each of these blocks we can use the system description given in Section 3.4,

7 Compressive Channel Tracking

adding an index $\omega = 0, \dots, \Omega-1$. If we again assume the 2D DFT coefficients $F_{m,i}^{(\omega)}$ from (3.17) for each ω to be supported inside the box $\{0, \dots, D-1\} \times \{-J/2, \dots, J/2-1\}$ with D and J such that $\Delta K = K/D$ and $\Delta L = L/J$ are integers, the channel coefficients $H_{l,k}^{(\omega)}$ are determined by their values on the subsampled grid $\mathcal{G} = \{(\lambda\Delta L, \kappa\Delta K) \mid \lambda = 0, \dots, J-1; \kappa = 0, \dots, D-1\}$. Then (5.1) yields

$$H_{\lambda\Delta L, \kappa\Delta K}^{(\omega)} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} F_{m,i}^{(\omega)} e^{-j2\pi(\frac{\kappa m}{D} - \frac{\lambda i}{J})}, \quad (7.1)$$

or once more using a basis yielding enhanced sparsity of the coefficients,

$$H_{\lambda\Delta L, \kappa\Delta K}^{(\omega)} = \sum_{m=0}^{D-1} \sum_{i=-J/2}^{J/2-1} G_{m,i}^{(\omega)} u_{m,i}[\lambda, \kappa]. \quad (7.2)$$

For pilot-aided channel estimation we first choose pilot sets $\mathcal{P}^{(\omega)} \subset \mathcal{G}$ of size $Q^{(\omega)} := |\mathcal{P}^{(\omega)}|$. Then, proceeding exactly as in Section 5.1.1 and using the same notation, we define $\hat{\mathbf{h}}_{(p)}^{(\omega)}$ to be the vector of the estimated channel coefficients at the pilot positions $\mathcal{P}^{(\omega)}$, $\mathbf{g}^{(\omega)} = \text{vec}\{G_{m,i}^{(\omega)}\}_{m,i}$, $\mathbf{z}_{(p)}^{(\omega)}$ with entries $\tilde{z}_{l,k}^{(\omega)} / p_{l,k}^{(\omega)}$, where $\tilde{z}_{l,k}$ are the noise terms from (3.13) and $p_{l,k}^{(\omega)}$ are the pilot symbols, and $\mathbf{U}_{(p)}^{(\omega)}$ the $Q^{(\omega)} \times JD$ matrix constituted of the rows of \mathbf{U} corresponding to the indices in $\mathcal{P}^{(\omega)}$, where \mathbf{U} is the columnwise stacking of the vectorized basis functions $\text{vec}\{u_{m,i}[\lambda, \kappa]\}_{\lambda, \kappa}$. Then, (5.4) yields

$$\hat{\mathbf{h}}_{(p)}^{(\omega)} = \mathbf{\Phi}^{(\omega)} \mathbf{x}^{(\omega)} + \mathbf{z}_{(p)}^{(\omega)} \quad (7.3)$$

for each $\omega = 0, \dots, \Omega-1$, where $\mathbf{\Phi}^{(\omega)} := \sqrt{JD/Q^{(\omega)}} \mathbf{U}_{(p)}^{(\omega)}$ and $\mathbf{x}^{(\omega)} = \sqrt{Q^{(\omega)}/JD} \mathbf{g}^{(\omega)}$.

As explained in Section 5.1.1 and Section 5.1.3, each vector $\mathbf{x}^{(\omega)}$ can be considered compressible. In Section 7.1.3 we show that in typical scenarios these coefficients, at least in the case where the 2D DFT basis is used, can also be considered approximately *sequentially sparse*, by which we mean that they are approximately sparse for all the symbol blocks considered, and that their essential support does not change very quickly. Although this notion of sequential sparsity is quite vague it seems to be the most fitting description in our setting. Note that it is not a standard terminology in the CS literature. Nevertheless it has been studied in the context of MOD-CS (see Section 4.6). Here a part of the support of the signal at one point in time is used as the "known part" of the support of the signal at the next one. This is feasible if the support does not change too much between two consecutive symbol blocks, i.e. if the signals are sequentially sparse as defined above.

Next, we adapt this strategy to our method. For the first symbol block, i.e. $\omega = 0$, we use the conventional compressive channel estimator as described in Section 5.1.1 to obtain estimates $\hat{H}_{l,k}^{(0)}$ of $H_{l,k}^{(0)}$. For the remaining blocks we proceed as follows. Given the estimate $\hat{\mathbf{x}}^{(\omega)}$ of $\mathbf{x}^{(\omega)}$ for some symbol block $\omega \in \{0, \dots, \Omega-2\}$, we want to find a subset of the support of $\hat{\mathbf{x}}^{(\omega)}$ that is expected to best match the support of $\mathbf{x}^{(\omega+1)}$. Unfortunately there is no way of knowing $\mathbf{x}^{(\omega+1)}$ or its support in advance. Recall that in the case where the 2D DFT basis is used the entries of the vectors $\mathbf{x}^{(\omega)}$ are simple scalings of the coefficients $F_{m,i}^{(\omega)}$. Since these coefficients are related to physical quantities (such as the complex path gains η_p of the individual propagation paths, see Section 7.1.3), also their magnitudes will only change minimally between two consecutive symbol blocks, which is why we can expect the entries of $\mathbf{x}^{(\omega)}$ with large absolute values not to become very small and thus still be part of the essential support of $\mathbf{x}^{(\omega+1)}$. Therefore, we define $\mathcal{S}^{(\omega)}$ as the $|\mathcal{S}^{(\omega)}|$ indices corresponding to the entries of $\hat{\mathbf{x}}^{(\omega)}$ with largest absolute values. Alternatively, we could define $\mathcal{S}^{(\omega)}$ as the indices corresponding to all entries of $\hat{\mathbf{x}}^{(\omega)}$ whose magnitude exceeds a prescribed threshold $\gamma > 0$. Either way, we can now use any MOD-CS algorithm to obtain an estimate of $\mathbf{x}^{(\omega+1)}$ using the set $\mathcal{S}^{(\omega)}$ as the known part of the support.

Let us now formulate the method. For $\omega = 0, \dots, \Omega-1$, fix the pilot sets $\mathcal{P}^{(\omega)}$, which are chosen uniformly at random from the subsampled grid \mathcal{G} before starting data transmission and stay fixed therein. For $\omega = 0$, perform conventional compressive channel estimation as described in Section 5.1.1 to obtain channel estimates $\hat{H}_{l,k}^{(0)}$ of $H_{l,k}^{(0)}$. Furthermore, determine the set $\mathcal{S}^{(0)}$ as described above via the estimate $\hat{\mathbf{x}}^{(0)}$ from Step 2 of the estimator. Then, for $\omega = 1, \dots, \Omega-1$, proceed as follows.

- Step 1.** Calculate channel estimates at the pilot positions $\mathcal{P}^{(\omega)}$ and stack them to obtain the measurement equation (7.3).
- Step 2.** Run any MOD-CS algorithm using $\mathcal{S}^{(\omega-1)}$ as the "known part" of the support to obtain an estimate $\hat{\mathbf{x}}^{(\omega)}$ of $\mathbf{x}^{(\omega)}$.
- Step 3.** Determine the set $\mathcal{S}^{(\omega)}$ as described above via $\hat{\mathbf{x}}^{(\omega)}$, and rescale $\hat{\mathbf{x}}^{(\omega)}$ with $\sqrt{JD/Q^{(\omega)}}$ for an estimate $\hat{\mathbf{g}}^{(\omega)}$ of $\mathbf{g}^{(\omega)}$.
- Step 4.** Calculate estimates of the subsampled channel coefficients $H_{\lambda\Delta L, \kappa\Delta K}^{(\omega)}$ from (7.2).
- Step 5.** Invert (7.1) to obtain estimates of $F_{m,i}^{(\omega)}$.
- Step 6.** Finally calculate estimates of all the channel coefficients $H_{l,k}^{(\omega)}$ by using (3.16).

Note that in the special case where the 2D DFT basis is used we have $G_{m,i}^{(\omega)} = \sqrt{JD}F_{m,i}^{(\omega)}$, and therefore Steps 4 and 5 can again be omitted.

As already mentioned in Section 4.6 Theorem 4.6.1 regarding the estimation error of MOD-BPDN is of very limited use in practice, and results for MOD-OMP and MOD-CoSaMP have not been reported, yet. Therefore, we do not give a bound for the estimation error for the proposed compressive channel tracker. Nevertheless, the simulation results presented in Section 7.3 indicate that tracking the channel in the way described above can significantly improve the performance compared to the conventional compressive estimator. More importantly, as indicated in the next section, the running time of the MOD-CS algorithms can be reduced significantly.

7.1.2 Computational Complexity

For the first symbol block ($\omega = 0$) the conventional channel estimator is used, the computational complexity of which has been studied in Section 5.1.2. In order to determine the set $\mathcal{S}^{(0)}$ we then simply have to order the entries of $\hat{\mathbf{x}}^{(0)}$ by their magnitude, which can be done by using mergesort, for example, with $\mathcal{O}(JD \log(JD))$ operations [146].

For the remaining symbol blocks we only have to take a closer look at Steps 2 and 3. We denote the complexity of Step 2 by $\mathcal{O}(\text{MOD-CS})$, since it again depends on the algorithm that is used to solve the MOD-CS problem. Finding the set $\mathcal{S}^{(\omega)}$ in Step 3 has complexity $\mathcal{O}(JD \log(JD))$ as explained above, whereas the rescaling can be done by using $\mathcal{O}((JD)^2)$ operations as before.

Therefore, the computational complexity for the first symbol block is given by

$$\mathcal{O}(\text{CS}) + \mathcal{O}((JD)^2) + \mathcal{O}(LK \log(LK))$$

following (5.10), whereas for the remaining symbol blocks it is

$$\mathcal{O}(\text{MOD-CS}) + \mathcal{O}((JD)^2) + \mathcal{O}(LK \log(LK)).$$

Note that the overall complexity will typically be governed by the term $\mathcal{O}(\text{CS})$ and $\mathcal{O}(\text{MOD-CS})$, respectively. As explained in Section 4.6, we have $\mathcal{O}(\text{MOD-CS}) = \mathcal{O}(JD(n_{\text{MOD-OMP}})^2 + (n_{\text{MOD-OMP}} - \min_{\omega} |\mathcal{S}^{(\omega)}|)\Phi)$ using MOD-OMP, or $\mathcal{O}(\text{MOD-CS}) = \mathcal{O}(n_{\text{MOD-CoSaMP}}\Phi)$ using MOD-CoSaMP, where $n_{\text{MOD-OMP}}$ and $n_{\text{MOD-CoSaMP}}$ denote the maximal numbers of MOD-OMP or MOD-CoSaMP iterations, respectively. Therefore, if MOD-OMP is used and if $\min_{\omega} |\mathcal{S}^{(\omega)}|$ is large, the complexity is reduced considerably.

7.1.3 Sequential Delay-Doppler Sparsity

For the analysis of the sequential compressibility of the 2D DFT coefficients $F_{m,i}^{(\omega)}$, we assume that for each symbol block $\omega \in \{0, \dots, \Omega-1\}$ the channel comprises P propagation paths corresponding to the same P specular scatterers with path gains $\eta_p^{(\omega)}$, time delays $\tau_p^{(\omega)}$ and Doppler frequency shifts $\nu_p^{(\omega)}$ for $p = 1, \dots, P$. In other words, we assume these time delays and Doppler shifts to be constant over the duration T_b of one symbol block and to only change from one symbol block to the next (where $T_b \approx T_s N_0$, see also [3]). As already mentioned in Section 5.1.3, assuming the parameters to be constant during one symbol block is merely an approximation that obviously gets worse the larger the symbol blocks are. Moreover, of course scatterers appear and/or disappear during transmissions in practice. In order to take this into account the number Ω of transmitted symbol blocks can be reduced and the proposed channel tracker can be started again to estimate the channel coefficients for the next Ω symbol blocks. However, this model is only used for analyzing the sequential sparsity and it is not necessary for the proposed channel tracker.

Now recall that following (5.15) each of the discrete-delay-Doppler spreading functions $S_h^{(\omega)}[m, i]$ is basically constituted of the shifted leakage kernels $\Lambda_p^{(\omega)}[m, i]$, which have been shown to be mainly supported inside a box of size $2\Delta m \times 2\Delta i$ around the center points $\xi_p^{(\omega)} = (\tau_p^{(\omega)}/T_s, \nu_p^{(\omega)}T_s N_0)$ (see Section 5.1.3 for the details). Consider symbol block ω . Then, in analogy to Section 6.1.3, we denote by $\mathbf{w}_{T,p}^{(\omega)}$ and $\mathbf{w}_{R,p}^{(\omega)}$ the vectors connecting a given scatterer p with the transmitter and the receiver, respectively, by $\mathbf{v}_{T,p}^{(\omega)}$ and $\mathbf{v}_{R,p}^{(\omega)}$ the velocity vectors of the scatterer relative to the transmitter and the receiver, respectively, and we define $w_{T,p}^{(\omega)} := \|\mathbf{w}_{T,p}^{(\omega)}\|_2$, $w_{R,p}^{(\omega)} := \|\mathbf{w}_{R,p}^{(\omega)}\|_2$, $v_{T,p}^{(\omega)} := \|\mathbf{v}_{T,p}^{(\omega)}\|_2$ and $v_{R,p}^{(\omega)} := \|\mathbf{v}_{R,p}^{(\omega)}\|_2$. By denoting the speed of light by c and the carrier frequency by f_0 , the same reasoning as in Section 6.1.3 yields that

$$\tau_p^{(\omega)} = \frac{w_{T,p}^{(\omega)} + w_{R,p}^{(\omega)}}{c}. \quad (7.4)$$

For the Doppler frequency shifts (6.12) yields

$$\nu_p^{(\omega)} = \nu_{T,p}^{(\omega)} + \nu_{R,p}^{(\omega)}, \quad (7.5)$$

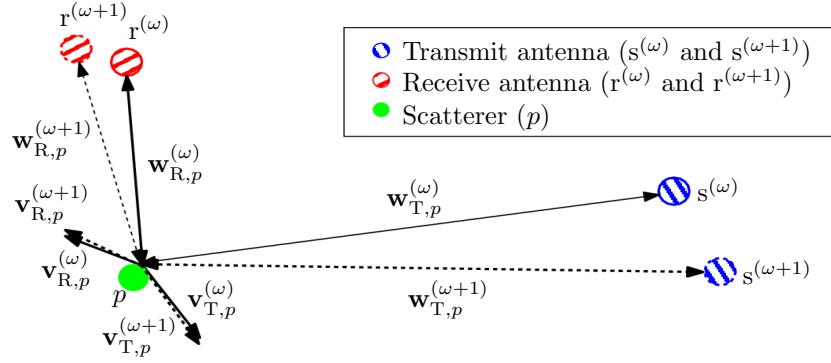


Figure 7.1: Geometric illustration of a propagation path for two consecutive symbol blocks ω and $\omega + 1$ from the scatterer's point of view: Solid lines depict the situation for symbol block ω , dashed lines that for symbol block $\omega + 1$.

where

$$\nu_{T,p}^{(\omega)} = \frac{f_0}{c} \frac{\mathbf{v}_{T,p}^{(\omega)T} \mathbf{w}_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} \quad \text{and} \quad (7.6)$$

$$\nu_{R,p}^{(\omega)} = \frac{f_1}{c} \frac{\mathbf{v}_{R,p}^{(\omega)T} \mathbf{w}_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}} \quad (7.7)$$

are the partial Doppler shifts, such that $f_1 = f_0 + \nu_p^{(\omega)}$.

Now let us consider two consecutive symbol blocks ω and $\omega + 1$ (see Fig. 7.1). To bound the difference $\Delta\tau_p^{(\omega)} := |\tau_p^{(\omega+1)} - \tau_p^{(\omega)}|$ of time delays we can use (7.4) and the triangle inequality to obtain

$$\begin{aligned} \Delta\tau_p^{(\omega)} &= \frac{1}{c} \left| w_{T,p}^{(\omega+1)} + w_{R,p}^{(\omega+1)} - w_{T,p}^{(\omega)} - w_{R,p}^{(\omega)} \right| \\ &\leq \frac{1}{c} \left(\left| w_{T,p}^{(\omega+1)} - w_{T,p}^{(\omega)} \right| + \left| w_{R,p}^{(\omega+1)} - w_{R,p}^{(\omega)} \right| \right). \end{aligned}$$

Now let $\Delta w_{T,p}^{(\omega)} := \|\mathbf{w}_{T,p}^{(\omega+1)} - \mathbf{w}_{T,p}^{(\omega)}\|_2$ and $\Delta w_{R,p}^{(\omega)} := \|\mathbf{w}_{R,p}^{(\omega+1)} - \mathbf{w}_{R,p}^{(\omega)}\|_2$. Then, since obviously $\left| w_{T,p}^{(\omega+1)} - w_{T,p}^{(\omega)} \right| \leq \Delta w_{T,p}^{(\omega)}$ and $\left| w_{R,p}^{(\omega+1)} - w_{R,p}^{(\omega)} \right| \leq \Delta w_{R,p}^{(\omega)}$, we have $\Delta\tau_p^{(\omega)} \leq \tau_{B,p}^{(\omega)}$, with

$$\tau_{B,p}^{(\omega)} := \frac{1}{c} \left(\Delta w_{T,p}^{(\omega)} + \Delta w_{R,p}^{(\omega)} \right).$$

As for the difference $\Delta\nu_p^{(\omega)} := |\nu_p^{(\omega+1)} - \nu_p^{(\omega)}|$ of Doppler frequency shifts, we can use

(7.5) and (7.6) to get

$$\begin{aligned} \Delta\nu_p^{(\omega)} &= \frac{1}{c} \left| f_0 \frac{\mathbf{v}_{T,p}^{(\omega+1)T} \mathbf{w}_{T,p}^{(\omega+1)}}{w_{T,p}^{(\omega+1)}} + f_1 \frac{\mathbf{v}_{R,p}^{(\omega+1)T} \mathbf{w}_{R,p}^{(\omega+1)}}{w_{R,p}^{(\omega+1)}} - f_0 \frac{\mathbf{v}_{T,p}^{(\omega)T} \mathbf{w}_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} - f_1 \frac{\mathbf{v}_{R,p}^{(\omega)T} \mathbf{w}_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}} \right| \\ &\leq \frac{1}{c} \left(f_0 \left| \frac{\mathbf{v}_{T,p}^{(\omega+1)T} \mathbf{w}_{T,p}^{(\omega+1)}}{w_{T,p}^{(\omega+1)}} - \frac{\mathbf{v}_{T,p}^{(\omega)T} \mathbf{w}_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} \right| + f_1 \left| \frac{\mathbf{v}_{R,p}^{(\omega+1)T} \mathbf{w}_{R,p}^{(\omega+1)}}{w_{R,p}^{(\omega+1)}} - \frac{\mathbf{v}_{R,p}^{(\omega)T} \mathbf{w}_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}} \right| \right). \end{aligned}$$

At first we consider the transmitter side. Inserting some help terms (by first adding them and afterwards subtracting them again) and using the triangle and the Cauchy-Schwarz inequalities yields

$$\begin{aligned} \left| \frac{\mathbf{v}_{T,p}^{(\omega+1)T} \mathbf{w}_{T,p}^{(\omega+1)}}{w_{T,p}^{(\omega+1)}} - \frac{\mathbf{v}_{T,p}^{(\omega)T} \mathbf{w}_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} \right| &\leq \left| \left(\mathbf{v}_{T,p}^{(\omega+1)} - \mathbf{v}_{T,p}^{(\omega)} \right)^T \frac{\mathbf{w}_{T,p}^{(\omega+1)}}{w_{T,p}^{(\omega+1)}} \right| \\ &\quad + \left| \mathbf{v}_{T,p}^{(\omega)T} \mathbf{w}_{T,p}^{(\omega+1)} \left(\frac{1}{w_{T,p}^{(\omega+1)}} - \frac{1}{w_{T,p}^{(\omega)}} \right) \right| \\ &\quad + \left| \frac{1}{w_{T,p}^{(\omega)}} \mathbf{v}_{T,p}^{(\omega)T} \left(\mathbf{w}_{T,p}^{(\omega+1)} - \mathbf{w}_{T,p}^{(\omega)} \right) \right| \\ &\leq \Delta v_{T,p}^{(\omega)} + v_{T,p}^{(\omega)} w_{T,p}^{(\omega+1)} \frac{|w_{T,p}^{(\omega)} - w_{T,p}^{(\omega+1)}|}{w_{T,p}^{(\omega)} w_{T,p}^{(\omega+1)}} + v_{T,p}^{(\omega)} \frac{\Delta w_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} \\ &\leq \Delta v_{T,p}^{(\omega)} + 2v_{T,p}^{(\omega)} \frac{\Delta w_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}}, \end{aligned}$$

where we have set $\Delta v_{T,p}^{(\omega)} := \|\mathbf{v}_{T,p}^{(\omega+1)} - \mathbf{v}_{T,p}^{(\omega)}\|_2$ and we again used that $|w_{T,p}^{(\omega)} - w_{T,p}^{(\omega+1)}| \leq \Delta w_{T,p}^{(\omega)}$. Defining $\Delta v_{R,p}^{(\omega)} := \|\mathbf{v}_{R,p}^{(\omega+1)} - \mathbf{v}_{R,p}^{(\omega)}\|_2$ and using the exact same arguments for the receiver side yields

$$\left| \frac{\mathbf{v}_{R,p}^{(\omega+1)T} \mathbf{w}_{R,p}^{(\omega+1)}}{w_{R,p}^{(\omega+1)}} - \frac{\mathbf{v}_{R,p}^{(\omega)T} \mathbf{w}_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}} \right| \leq \Delta v_{R,p}^{(\omega)} + 2v_{R,p}^{(\omega)} \frac{\Delta w_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}}.$$

All in all, we thus have $\Delta\nu_p^{(\omega)} \leq \nu_{B,p}^{(\omega)}$ with

$$\nu_{B,p}^{(\omega)} := \frac{f_0}{c} \left(\Delta v_{T,p}^{(\omega)} + 2v_{T,p}^{(\omega)} \frac{\Delta w_{T,p}^{(\omega)}}{w_{T,p}^{(\omega)}} \right) + \frac{f_1}{c} \left(\Delta v_{R,p}^{(\omega)} + 2v_{R,p}^{(\omega)} \frac{\Delta w_{R,p}^{(\omega)}}{w_{R,p}^{(\omega)}} \right).$$

In order to better understand the bounds $\tau_{B,p}^{(\omega)}$ and $\nu_{B,p}^{(\omega)}$ we approximate $\mathbf{w}_{T,p}^{(\omega+1)} \approx \mathbf{w}_{T,p}^{(\omega)} + T_b \mathbf{v}_{T,p}^{(\omega)}$ and $\mathbf{w}_{R,p}^{(\omega+1)} \approx \mathbf{w}_{R,p}^{(\omega)} + T_b \mathbf{v}_{R,p}^{(\omega)}$. This would be exact if the velocity vectors

7 Compressive Channel Tracking

$\mathbf{v}_{T,p}^{(\omega)}$ and $\mathbf{v}_{R,p}^{(\omega)}$ did not change during the duration T_b of one symbol block, but since T_b is typically very small the approximation is very good. Thus, we have $\Delta w_{T,p}^{(\omega)} \approx T_b v_{T,p}^{(\omega)}$ and $\Delta w_{R,p}^{(\omega)} \approx T_b v_{R,p}^{(\omega)}$, which gives

$$\begin{aligned}\tau_{B,p}^{(\omega)} &\approx \frac{T_b \left(v_{T,p}^{(\omega)} + v_{R,p}^{(\omega)} \right)}{c}, \\ \nu_{B,p}^{(\omega)} &\approx \frac{f_0}{c} \left(\Delta v_{T,p}^{(\omega)} + 2T_b \frac{\left(v_{T,p}^{(\omega)} \right)^2}{w_{T,p}^{(\omega)}} \right) + \frac{f_1}{c} \left(\Delta v_{R,p}^{(\omega)} + 2T_b \frac{\left(v_{R,p}^{(\omega)} \right)^2}{w_{R,p}^{(\omega)}} \right).\end{aligned}$$

Now since T_b is very small in practice, $\tau_{B,p}^{(\omega)}$ is also be very small. Furthermore, since in this small period of time the velocities of the objects involved do not change much, $\Delta v_{T,p}^{(\omega)}$ and $\Delta v_{R,p}^{(\omega)}$ are also very small, which yields a very small bound $\nu_{B,p}^{(\omega)}$ as well.

To sum up, we have shown that in practical scenarios the time delays $\tau_p^{(\omega)}$ and the Doppler frequency shifts $\nu_p^{(\omega)}$ will change very little from one symbol block to the next. Therefore, the centerpoints $\xi_p^{(\omega)}$ and $\xi_p^{(\omega+1)}$ of the leakage kernels $\Lambda_p^{(\omega)}$ and $\Lambda_p^{(\omega+1)}$, respectively, will differ by at most $\Delta m_{B,p}^{(\omega)} := \left\lceil \tau_{B,p}^{(\omega)} / T_s \right\rceil$ in the m -direction and by $\Delta i_{B,p}^{(\omega)} := \left\lceil \nu_{B,p}^{(\omega)} T_s N_0 \right\rceil$ in the i -direction. Therefore, the supports of $\Lambda_p^{(\omega)}$ and $\Lambda_p^{(\omega+1)}$, and consequently of the spreading functions $S_h^{(\omega)}$ and $S_h^{(\omega+1)}$ (see (5.15)), and in turn of the 2D DFT coefficients $F_{m,i}^{(\omega)}$ and $F_{m,i}^{(\omega+1)}$ (see (3.16)), have a very large overlap, which is exactly what we wanted to show in this section.

Finally note that although there is no guarantee that the expansion coefficients $G_{m,i}^{(\omega)}$ emerging from the use of a basis that is constructed according to the optimization technique presented in Section 5.1.4 are also approximately sequentially sparse, the simulation results presented in Section 7.3 indicate that they in fact are. Therefore, using that basis optimization technique also yields a substantial performance gain for the compressive tracking method presented here.

7.2 Extension to the MIMO Case

The analysis carried out in Section 6.2.3 yields that in the MIMO case the 2D DFT coefficients $F_{m,i}^{(\theta)}$ are jointly group compressible. Since the analysis given in the previous section can be adopted for each of the cross-channels of the MIMO channel without any change, the compressive channel tracker presented in Section 7.1.1 can easily be

extended to the MIMO case, also incorporating the (joint) group compressibility of the coefficients. To do so, fix the pilot sets $\mathcal{P}^{(s,\omega)}$ for each transmit antenna $s = 1, \dots, N_T$ and each symbol block $\omega = 0, \dots, \Omega - 1$, as well as the partition $\mathcal{J} = \{I_b\}_{b=0}^{B-1}$ with the groups $I_b = \mathbf{S}(\mathcal{B}_b)$ of size $\Delta\tilde{m} \times \Delta\tilde{i}$ as defined in Section 6.2.1. For the first symbol block ($\omega = 0$) the receiver then uses the general multichannel compressive channel estimator utilizing group sparsity from Section 6.2.1. Denote the estimates obtained in Step 1 of the method by $\hat{\mathbf{x}}^{(\theta,\omega)}$, and their stacking into a matrix by $\hat{\mathbf{X}}^{(0)} = [\hat{\mathbf{x}}^{(\theta_1,0)}, \dots, \hat{\mathbf{x}}^{(\theta_{N_R N_T},0)}]$ for an arbitrary but fixed ordering $\{\theta_1, \dots, \theta_{N_R N_T}\}$ of Θ . Then, $\mathcal{S}^{(0)}$ is defined as the set of indices corresponding to the groups b yielding the largest Frobenius norms $\|\hat{\mathbf{X}}^{(0)}[b]\|_F$, where $\hat{\mathbf{X}}^{(0)}[b]$ denotes the $|I_b| \times N_R N_T$ submatrix of $\hat{\mathbf{X}}^{(0)}$ constructed from the rows corresponding to the indices in I_b . Again, either the number of groups to be chosen, i.e. $|\mathcal{S}^{(0)}|$, is fixed, or all groups for which $\|\hat{\mathbf{X}}^{(0)}[b]\|_F > \gamma$ for some threshold $\gamma > 0$ are chosen.

For the remaining blocks $\omega = 1, \dots, \Omega - 1$, we simply have to use a MOD-CS technique, extended to take joint group sparsity into account, instead of the MGSCS-technique in Step 1 of that estimator, and insert an additional step in between Step 1 and Step 2, where the set $\mathcal{S}^{(\omega)}$ is determined as explained above. The remaining steps can then be adopted without any changes.

As before, the computational complexity will be dominated by the complexity of the sparse recovery technique. The adaption of G-DCS-SOMP to this setting is straight forward. Just as in MOD-OMP an initialization step is inserted utilizing the known part of the support, first. The computational complexity of this *Modified G-DCS-SOMP* (MOD-G-DCS-SOMP) algorithm in this setting is clearly $\mathcal{O}(N_T J D (n'_{\text{MOD-GDS}})^2 + (n_{\text{MOD-GDS}} - \min_{\omega} |\mathcal{S}^{(\omega)}|) N_R \max_{\omega} \{ \sum_s \Phi^{(s,\omega)} \})$, where $n_{\text{MOD-GDS}}$ denotes the maximal number of iterations, $n'_{\text{MOD-GDS}}$ is the sum of the magnitudes of the chosen groups, and $\Phi^{(s,\omega)}$ denotes the measurement matrix corresponding to antenna s at symbol block ω . Adapting G-CoSaMP can be done exactly as adapting CoSaMP to MOD-CoSaMP as described in Section 4.6. The computational complexity of *Modified G-CoSaMP* (MOD-G-CoSaMP) with $n_{\text{MOD-GC}}$ iterations then is $\mathcal{O}(n_{\text{MOD-GC}} N_R \max_{\omega} \{ \sum_s \Phi^{(s,\omega)} \})$.

As for the quality of channel estimation once more note that it is hard to give an error bound since the coefficients are only sequentially compressible and not exactly sparse, but that the simulation results presented in Section 7.3 suggest an improved performance compared to the multichannel group sparse estimator from Section 6.2.1.

7.3 Simulation Results

We compared the proposed channel tracker to the conventional compressive channel estimator of Section 5.1.1 in the SISO case, as well as to the multichannel compressive estimator utilizing group sparsity of Section 6.2.1 for the MIMO case.

MC MIMO system parameters. As described in Section 6.1.5, we simulated a CP-MIMO-OFDM system with again $K = 512$ subcarriers, CP-length $L_{\text{cp}} = 128$, center frequency $f_0 = 5$ GHz and bandwidth $1/T_s = 5$ MHz. As before, we used 4-QAM symbols and root-raised cosine filters $f_1(t) = f_2(t)$ with roll-off factor $\alpha = 1/4$. The number of transmit and receive antennas was $N_T = N_R \in \{1, 2\}$ (note that $N_T = N_R = 1$ corresponds to the SISO case).

Channel. For 50 different randomly generated scenarios we generated doubly selective channels during $\Omega = 10$ blocks of $L = 32$ OFDM symbols each, again using `IlmProp` [143]. The geometric setting was exactly like the one described in Section 5.1.5.

Subsampling and pilot setup. As always we used a subsampled time-frequency grid \mathcal{G} with spacing $\Delta L = 1$ and $\Delta K = 4$. We fixed the number of pilots as $Q = |\mathcal{P}^{(s,\omega)}| = 1024$ ($s = 1, \dots, N_T$; $\omega = 0, \dots, \Omega - 1$), which means that $6.25 \cdot N_T\%$ of all the symbols were pilots. For the pilot sets $\mathcal{P}^{(s,\omega)}$ we chose two different settings. In the first setting we used the same pilot set for all the symbol blocks, i.e. $\mathcal{P}^{(s,\omega)} \equiv \mathcal{P}^{(s,0)}$ for all $\omega = 1, \dots, \Omega - 1$. As explained in Section 6.1.5, we therefore chose a set of size $N_T Q$ uniformly at random from \mathcal{G} and divided it into N_T sets of equal size Q to obtain the pilot sets $\mathcal{P}^{(s,0)}$ for $s = 1, \dots, N_T$. In the second setting we chose different pilot sets for each symbol block, so we repeated the construction of the pilot set $\mathcal{P}^{(s,0)}$ described above $\Omega - 1$ times in order to obtain the pilot sets $\mathcal{P}^{(s,\omega)}$ for the remaining symbol blocks $\omega = 1, \dots, \Omega - 1$.

Performance measure. The performance is measured by the MSE normalized by the mean energy of all the channel coefficients, i.e. $\sum_{\omega,\theta,l,k} |H_{l,k}^{(\theta,\omega)} - \hat{H}_{l,k}^{(\theta,\omega)}|^2 / \sum_{\omega,\theta,l,k} |H_{l,k}^{(\theta,\omega)}|^2$.

Channel estimation. For the SISO case we chose to only use OMP and MOD-OMP for sparse reconstruction, since OMP was seen to give the best results for the conventional compressive estimator. In the MIMO case we used G-DCS-SOMP and MOD-G-DCS-SOMP, where the groups were defined as described in Section 6.2.1 with blocks \mathcal{B}_b of size $\Delta \tilde{m} \times \Delta \tilde{i} = 2 \times 2$, only. For the modified algorithms we defined the sets $\mathcal{S}^{(\omega)}$ as described in Sections 7.1.1 and 7.2 with fixed magnitudes. In order to describe how these sets were

defined exactly we first consider the SISO case. For the first symbol block we used conventional OMP with n_{OMP} iterations. Therefore, the first estimate $\hat{\mathbf{x}}^{(0)}$ is n_{OMP} -sparse. Then, we defined $\mathcal{S}^{(0)}$ as the $|\mathcal{S}^{(0)}| = \lfloor \mu n_{\text{OMP}} \rfloor$ indices corresponding to the entries of $\hat{\mathbf{x}}^{(0)}$ with largest absolute value for some $\mu \in [0, 1]$. In other words, $100\mu\%$ of these indices were chosen as approximate support for the next symbol block. From then on, $\mathcal{S}^{(\omega)}$ was always chosen as described above, where $|\mathcal{S}^{(\omega)}| = \mu \lfloor n_{\text{MOD-OMP}}^{(\omega)} + |\mathcal{S}^{(\omega-1)}| \rfloor$, $\omega = 1, \dots, \Omega - 1$. In the MIMO case we proceeded analogously (see Section 7.2 for details). For all the simulations we used $\mu \in \{0, 0.1, \dots, 1\}$, Note that $\mu = 0$ corresponds to $\mathcal{S}^{(\omega)} = \emptyset$ and therefore the tracking method coincides with the corresponding conventional estimation method.

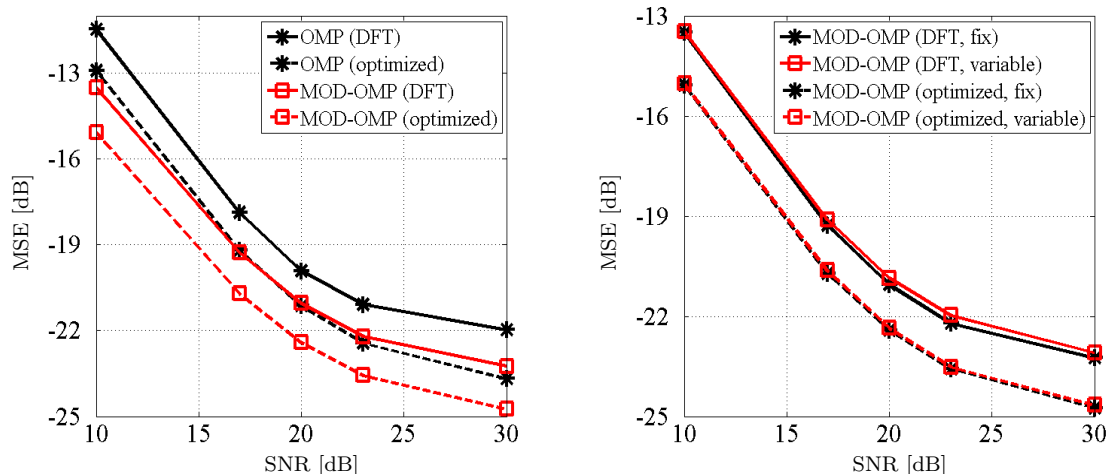
Furthermore, we also used the optimized bases that were constructed for the simulations in Section 5.1.5 for the SISO case and in Section 6.2.5 for the MIMO case. The numbers of iterations for MOD-OMP were chosen according to Table 7.1.

In the MIMO case, the number of MOD-G-DCS-SOMP iterations was chosen as $n_{\text{MOD-G-DCS-SOMP}} = 54$ for the 2D DFT basis, and $n_{\text{MOD-G-DCS-SOMP}}^{\text{opt}} = 30$ for the optimized basis. Again, all these parameters were found experimentally.

Results. First we consider the SISO case. In Fig. 7.2(a) we plot the MSE of channel estimation versus the SNR to compare the performance of the proposed channel tracker using MOD-OMP to the conventional compressive estimator of Section 5.1.1 using OMP, both for the use of the 2D DFT and an optimized basis. Regarding the definition of $\mathcal{S}^{(\omega)}$ we chose $\mu = 0.9$ in this setting, i.e. 90% of the support of each estimate $\hat{\mathbf{x}}^{(\omega)}$ were kept as estimated partial support of $\mathbf{x}^{(\omega+1)}$, and we used the first (fixed) pilot setup. It is obvious that the compressive channel tracker actually gives much better results than the conventional compressive estimator. Furthermore, since the sets $\mathcal{S}^{(\omega)}$ were chosen quite large, the computational complexity is reduced substantially. In addition, although it

Par. \ μ	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$n_{\text{MOD-OMP}}$	120	120	120	120	120	120	120	130	140	150	140
$n_{\text{MOD-OMP}}^{\text{opt}}$	100	100	100	100	100	100	100	100	110	120	120

Table 7.1: Simulation parameters $n_{\text{MOD-OMP}}$ and $n_{\text{MOD-OMP}}^{\text{opt}}$ for MOD-OMP for the various values of μ



(a) Compressive channel tracking compared to conventional compressive channel estimation for both the 2D DFT and an optimized basis.

(b) Comparison of the use of a fixed to that of a variable pilot set.

Figure 7.2: Performance of the compressive channel tracker: MSE versus SNR

is impossible to show that the expansion coefficients $G_{m,i}^{(\omega)}$ with regard to the optimized basis are again approximately sequentially sparse, these simulation results suggest that they in fact are, and a considerable performance gain can be achieved. This is probably due to the fact that the optimization algorithm was initialized with the DFT matrix, which already yields sequentially compressible coefficients.

In Fig. 7.2(b) we present the results for OMP and MOD-OMP for both the 2D DFT and the optimized basis where again $\mu = 0.9$, but now for the fixed pilot setup, i.e. $\mathcal{P}^{(\omega)} \equiv \mathcal{P}^{(0)}$ for all $\omega = 1, \dots, \Omega-1$, compared to the variable pilot setup where each pilot set $\mathcal{P}^{(\omega)}$ was different. We would have expected the variable setup to give better results, because here at each symbol block "different" information is gained about the signal $\mathbf{x}^{(\omega)}$, so that it is less likely to "miss" important entries. Nevertheless, the simulation results show no significant difference between the two setups. We think that this is due the fact that the most important entries are captured anyway because they are so dominant.

Furthermore, in Fig. 7.3 we present the MSE of channel tracking (using the fixed pilot setup) versus $\mu \in \{0, 0.1, \dots, 1\}$, which characterizes how much of the support is kept from one symbol block to the next. Here we only show the results for the 2D DFT basis. Obviously, $\mu = 0$ corresponds to the conventional compressive estimator, since

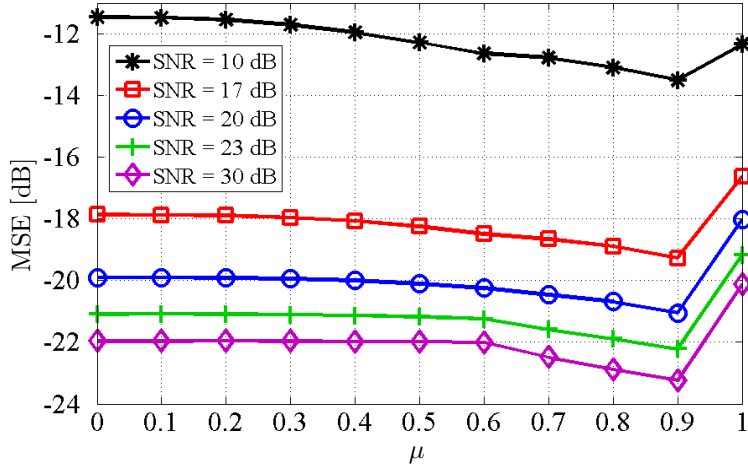


Figure 7.3: Performance of the compressive channel tracker using the 2D DFT basis: MSE versus μ .

in this case no support information is kept from one symbol block to the next. Setting $\mu = 1$ corresponds to keeping the entire support from the first signal estimate for all the following symbol blocks.

It can be seen that in the case of a low SNR tracking the channel improves the estimation quality even for rather small μ . This is due to the fact that in this case the noise contributions can easily be confused with relevant contributions of the signal. Nevertheless, the largest entries of $\mathbf{x}^{(\omega)}$ are most likely to correspond to true information about the channel, and therefore keeping their indices for the next symbol block $\omega + 1$, where they might already be smaller, is advantageous.

However, in the case of a high SNR the relevant contributions can be identified easier. Thus, even if the largest entries of $\mathbf{x}^{(\omega)}$ are smaller at the next symbol block $\omega + 1$, they will still be distinguishable from the noise contributions. Therefore the corresponding indices are chosen as part of the essential support for the estimation of $\mathbf{x}^{(\omega+1)}$ anyway, which is why the performance of the compressive channel tracker hardly changes for $\mu < 0.6$ in this case. Nevertheless, by choosing μ larger we make sure that entries corresponding to essential contributions whose magnitudes are close to the noise level will not be confused with noise at the next symbol block, and therefore the performance improves again. For $\mu = 1$ we obviously obtain worse results, since then the support of the $\mathbf{x}^{(\omega)}$ is kept equal although the channel changes from one symbol block to the next. Note that in the case of a very low SNR even $\mu = 1$ yields some improvement.

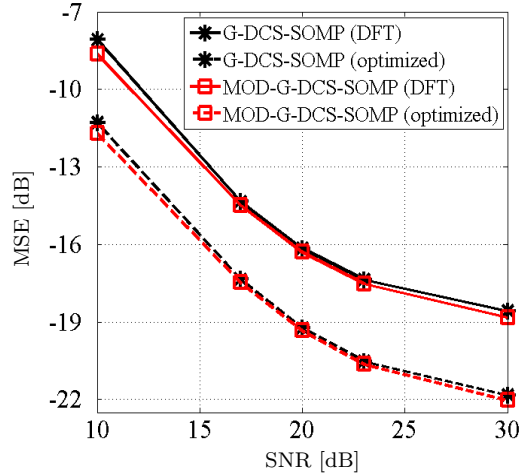


Figure 7.4: Performance of the multichannel compressive tracker for both the 2D DFT and an optimized basis with block size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$: MSE versus SNR

In Fig. 7.4 the results for a MIMO system with $N_T = N_R = 2$ transmit/receive antennas are shown, using blocks of size $\Delta\tilde{m} \times \Delta\tilde{i} = 2 \times 2$ as described above. Again, we chose $\mu = 0.9$ for the definition of $\mathcal{S}^{(\omega)}$, and we used the fixed pilot setup. In this setting, the performance gain obtained by the compressive tracker is only minimal. But note again that due to the large size of the sets $\mathcal{S}^{(\omega)}$ the computational complexity is reduced substantially. Furthermore, also the use of an optimized basis yields an additional performance gain, although, as mentioned above, there is no way of showing that the coefficient matrices $\mathbf{G}_{m,i}^{(\omega)}$ with regard to this basis are also approximately sequentially sparse.

Conclusion. It has been shown that the compressive channel tracker outperforms the conventional compressive channel estimator both in the SISO and MIMO case, as well as if the group compressibility is utilized or not. Actually, it never performs worse than the compressive channel estimation methods described in the previous sections for any $\mu < 1$. In fact, μ (and thus $|\mathcal{S}^{(\omega)}|$) can be chosen very large, which reduces the computational complexity of the method drastically. Furthermore, using an optimized basis once more gives better results, whereas the computational complexity is increased. All in all, the proposed compressive channel tracker is to be preferred in the SISO as well as in the MIMO case.

Conclusion

In this thesis we have investigated the field of compressive channel estimation that subsumes all channel estimation techniques utilizing the theory and methodology of CS and its variants. In Section 5.1 we reviewed the basic compressive channel estimator introduced in [1], and studied its performance and computational complexity. Furthermore, we analyzed the sparsity of typical doubly selective channels in the delay-Doppler region in more detail, and recalled the basis optimization technique presented in [2, 3]. Our simulation results demonstrated the superior performance of the compressive channel estimator compared to standard LS channel estimation, as well as the performance gain achieved by using an optimized basis.

Thereafter, we proposed several variations of the compressive channel estimator for various settings and situations. In Section 5.2 we adapted the compressive channel estimator so that it can also take group sparsity into account, and analyzed its performance and computational complexity. Furthermore, we showed that the leakage effect, which impairs the performance of compressive channel estimators in general, actually yields that typical channels can be considered group compressible, and thus the proposed estimator can be used. Moreover, we adapted the basis optimization technique mentioned above to this setting. As before, simulation results demonstrated the performance gain the proposed estimator can achieve compared to the conventional compressive channel estimator, as well as the improvement that can be achieved through the use of an optimized basis instead of the 2D DFT basis.

Furthermore, in Section 6.1 and Section 6.2 we extended the estimators described before to the MIMO case. We showed that the individual cross-channels in such a MIMO system approximately share a common (group) sparsity pattern, and therefore MCS (MGSCS) techniques can be utilized. As before, their performance and computational complexity were analyzed, and the basis optimization technique was adapted. Simulation results documented the superior performance of these multichannel estimators compared to the channel-per-channel use of their SISO-counterparts for the use of both the 2D DFT and an optimized basis.

In Chapter 7 we demonstrated that in a typical scenario the delay-Doppler (group) sparsity pattern of the channel does not change very much from one symbol block to the next, and that it can therefore be used to track the channel by utilizing the methodology of MOD-CS (and some variants). Here, the simulation results showed that not only the performance can be improved, but also that the computational complexity can be substantially reduced.

Index

- ℓ_p -norm, 6
- ℓ_p/ℓ_q -norm, 6

- Baseband signal, 14
- Basis expansion model, 26
- Basis pursuit, 35
- Basis pursuit denoising, 35
- Bernoulli random matrix, 36
- Big-O notation, 12

- Channel
 - doubly selective, 20
 - frequency selective, 20
- Channel coding, 14
- Channel estimation, 24
 - blind, 24
 - pilot-aided, 24
 - semilind, 24
- Compressible signal, 34
- Convolution, 8
- Convolution Theorem, 8
- CoSaMP, 38
- CP-OFDM, 15
- Cross-ambiguity function, 23

- DCS-SOMP, 44
- Demodulator, 15
- Dirac-delta, 9
- Discrete Fourier transform, 9
- Distribution
 - Bernoulli, 10
- distribution
 - Gaussian (normal), 11
 - Uniform (continuous), 11
 - Uniform (discrete), 10

- Equalization, 23
- Essential support, 34
- Expectation, 11

- Fading, 17
 - frequency selective, 20
 - large-scale, 18
 - small-scale, 17
- Fourier transform, 8
- Free space path loss, 17
- Frobenius norm, 7

- Gaussian random matrix, 36
- Greedy algorithms, 37

Index

- Group BPDN, 41
- Group CoSaMP, 42
- Group DCS-SOMP, 108
- Group OMP, 41
- Group RIC, 40
- Group RIP, 40
- Group sparse signal, 40

- Ideal filter, 10
- Impulse response, 19
- Inner product, 6
- Intercarrier interference (ICI), 20
- Intersymbol interference (ISI), 18
- Inverse DFT, 9
- Inverse Fourier transform, 8

- Jointly sparse vectors, 43

- Kronecker-delta, 9

- LASSO, 35
- Leakage effect, 61
- Line of Sight (LOS), 17
- Linear time-invariant system (LTI), 19
- Linear time-varying system (LTV), 20
- LS channel estimation, 24

- Mean square error (MSE), 68
- Measurement matrix, 34
- MIMO systems, 26
- MISO systems, 27
- MMSE channel estimation, 24
- Modified BPDN, 48
- Modified CoSaMP, 50
- Modified G-CoSaMP, 123
- Modified G-DCS-SOMP, 123

- Modified OMP, 49
- Modulator, 14
- Monte Carlo approximation, 12
- Multicarrier (MC), 14
- Multichannel BPDN, 44
- Multichannel CoSaMP, 44
- Multipath
 - components, 17
 - propagation, 17
- Multiplexing, 27

- Nyquist rate, 31

- OFDM, 15
- Orthogonal matching pursuit (OMP), 37

- Partial Fourier transform, 8
- Partial random Fourier matrix, 37
- Passband signal, 14
- Phase-shift keying (PSK), 14
- Pilot arrangement
 - block-type, 24
 - comb-type, 25
 - FDKD, 26
 - hybrid, 25
- Pilot pattern
 - non-zero, 30
 - zero, 30
- Pilot symbols, 24
- Pilot vector, 30
- Probability density function, 10
 - conditional, 11
 - joint, 11

- Restricted isometry constant (RIC), 35
- Restricted isometry property (RIP), 35

Row support, 7

Sampling, 31

Scatterer, 19

Sequential Sparsity, 116

Shadowing, 18

Shifted leakage kernel, 60

Signal-to-Noise Ratio (SNR), 19

SIMO systems, 27

Simultaneous OMP, 44

Sinc-function, 10

SISO systems, 27

Source coding, 14

Sparse approximation, 32

Sparse vector, 33

Spatial diversity, 27

Spreading function, 21

Structured sparsity, 39

Support, 7

System channel coefficient, 22

Training data, 24

Transfer function, 20

Transmit symbol, 14

Underspread channel, 21

Unitary matrix, 6

Variance, 11

Bibliography

- [1] G. Tauböck and F. Hlawatsch, “A compressed sensing technique for OFDM channel estimation in mobile environments: Exploiting channel sparsity for reducing pilots,” in *Proc. IEEE ICASSP-08*, (Las Vegas, NV), pp. 2885–2888, Apr. 2008.
- [2] G. Tauböck and F. Hlawatsch, “Compressed Sensing Based Estimation of Doubly Selective Channels Using a Sparsity-Optimized Basis Expansion,” in *Proc. EUSIPCO-08*, (Lausanne, Switzerland), Aug. 2008.
- [3] G. Tauböck, F. Hlawatsch, D. Eiwen, and H. Rauhut, “Compressive Estimation of Doubly Selective Channels in Multicarrier Systems: Leakage Effects and Sparsity-Enhancing Processing,” *IEEE J. Sel. Topics Sig. Process.*, vol. 4, pp. 255–271, Feb. 2010.
- [4] S. Coleri, M. Ergen, A. Puri, and A. Bahai, “Channel Estimation Techniques Based on Pilot Arrangement in OFDM Systems,” *IEEE Trans. Broadcasting*, vol. 48, pp. 223–229, Sep. 2002.
- [5] M.-H. Hsieh and C.-H. Wei, “Channel estimation for OFDM systems based on comb-type pilot arrangement in frequency selective fading channels,” *IEEE Trans. Consumer Electronics*, vol. 44, pp. 217–225, Feb. 1998.
- [6] O. Edfors, M. Sandell, J.-J. van de Beek, S. K. Wilson, and P. Borjesson, “OFDM channel estimation by singular value decomposition,” *IEEE Trans. Comm.*, vol. 46, pp. 931–939, Jul. 1998.
- [7] J. Xu and T. Strohmer, “Adaptive and Robust Channel Estimation for Pilot-aided OFDM Systems,” in *Proc. IEEE AusWireless*, (Sydney, Australia), Mar. 2006.

- [8] S. Das, *Mathematical methods for wireless channel estimation and equalization*. PhD thesis, University of Vienna, Vienna, Austria, Sep. 2009.
- [9] F.-C. Zheng, S. McLaughlin, and B. Mulgrew, "Blind equalization of nonminimum phase channels: higher order cumulant based algorithm," *IEEE Trans. Sig. Process.*, vol. 41, pp. 681–691, Feb. 1993.
- [10] S. Chen and S. McLaughlin, "Blind channel identification based on higher order cumulant fitting using genetic algorithms," in *Proc. IEEE Sig. Process. Workshop on Higher-Order Stat.*, (Banff, Canada), pp. 184–188, Jul. 1997.
- [11] H. Zheng and L. Tong, "Blind Channel Estimation Using the Second-Order Statistics: Asymptotic Performance and Limitations," *IEEE Trans. Sig. Process.*, vol. 45, pp. 2060–2071, Aug. 1997.
- [12] T. Petermann, S. Vogeler, K.-D. Kammeyer, and D. Boss, "Blind Turbo Channel Estimation in OFDM Receivers," in *Proc. 35th Asilomar Conf. Sig., Sys., Comput.*, Nov. 2001.
- [13] M. Necker and G. Stuber, "Totally blind channel estimation for OFDM on fast varying mobile radio channels," *IEEE Trans. Wireless Comm.*, vol. 3, pp. 1512–1525, Sep. 2004.
- [14] B. Muquet and M. de Courville, "Blind and Semi-Blind Channel Identification Methods using Second Order Statistics for OFDM Systems," in *Proc. IEEE ICASSP-99*, vol. 5, (Phoenix, AZ), pp. 2745–2748, Mar. 1999.
- [15] B. Muquet, M. de Courville, and P. Duhamel, "Subspace-based blind and semi-blind channel estimation for OFDM systems," *IEEE Trans. Sig. Process.*, vol. 50, pp. 1699–1712, Jul. 2002.
- [16] H. Cirpan and M. Tsatsanis, "Stochastic Maximum Likelihood Methods for Semi Blind Channel Estimation," *IEEE Sig. Process. Letters*, vol. 5, pp. 21–24, Feb. 1998.
- [17] W. Bajwa, J. Haupt, G. Raz, and R. Nowak, "Compressed Channel Sensing," in *Proc. IEEE CISS-08*, (Princeton, NJ), pp. 5–10, Mar. 2008.
- [18] W. Bajwa, A. Sayeed, and R. Nowak, "Learning sparse doubly-selective channels," in *Proc. 46th Annu. Allerton Conf. Comm., Contr., Comput.*, (Monticello, IL), pp. 575–582, Sep. 2008.

- [19] S. Cotter and B. Rao, "Sparse channel estimation via matching pursuit with application to equalization," *IEEE Trans. Comm.*, vol. 50, pp. 374–377, Mar. 2002.
- [20] W. Li and J. Preisig, "Estimation of rapidly time-varying sparse channels," *IEEE J. Oceanic Eng.*, vol. 32, pp. 927–939, Oct. 2007.
- [21] M. Sharp and A. Scaglione, "Application of sparse signal recovery to pilot-assisted channel estimation," in *Proc. IEEE ICASSP-08*, (Las Vegas, NV), pp. 3469 – 3472, Apr. 2008.
- [22] W. Bajwa, A. Sayeed, and R. Nowak, "Compressed sensing of wireless channels in time, frequency, and space," in *Proc. 42nd Asilomar Conf. Sig., Sys., Comput.*, (Pacific Grove, CA), pp. 2048–2052, Oct. 2008.
- [23] W. Bajwa, J. Haupt, A. M. Sayeed, and R. Nowak, "Compressed Channel Sensing: A New Approach to Estimating Sparse Multipath Channels," *Proc. IEEE*, vol. 98, pp. 1058–1076, Jun. 2010.
- [24] O. Simeone, Y. Bar Ness, and U. Spagnolini, "Pilot-based channel estimation for OFDM systems by tracking the delay-subspace," *IEEE Trans. Wireless Comm.*, vol. 3, pp. 315–325, Jan. 2004.
- [25] Y. Li, Z. Li, Y. Cai, and Y. Xu, "An improved channel estimation scheme for OFDM systems by tracking the subspace," in *Proc. IEEE PIMRC-03*, vol. 2, (Beijing, China), pp. 1109–1113, Sep. 2003.
- [26] K.-Y. Han, S.-W. Lee, J.-S. Lim, and K.-M. Sung, "Channel estimation for OFDM with fast fading channels by modified Kalman filter," *IEEE Trans. Consumer Electronics*, vol. 50, pp. 443–449, May 2004.
- [27] D. Schafhuber, G. Matz, and F. Hlawatsch, "Kalman tracking of time-varying channels in wireless MIMO-OFDM systems," in *Proc. 37th Asilomar Conf. Sig., Sys., Comput.*, vol. 2, (Pacific Grove, CA), pp. 1261–1265, Nov. 2003.
- [28] D. Eiwen, G. Tauböck, F. Hlawatsch, and H. G. Feichtinger, "Group Sparsity Methods for Compressive Channel Estimation in Doubly Dispersive Multicarrier Systems," in *Proc. IEEE SPAWC-10*, (Marrakech, Morocco), Jun. 2010.

- [29] D. Eiwen, G. Tauböck, F. Hlawatsch, H. Rauhut, and N. Czink, “Multichannel-Compressive Estimation of Doubly Selective Channels in MIMO-OFDM Systems: Exploiting and Enhancing Joint Sparsity,” in *Proc. IEEE ICASSP-10*, (Dallas, TX), pp. 3082–3085, Mar. 2010.
- [30] D. Eiwen, G. Tauböck, F. Hlawatsch, and H. G. Feichtinger, “Compressive Tracking Of Doubly Selective Channels In Multicarrier Systems Based On Sequential Delay-Doppler Sparsity,” in *Proc. IEEE ICASSP-11*, (Prague, Czech Republic), pp. 2928–2931, May 2011.
- [31] G. Golub and C. F. van Loan, *Matrix Computations*. Baltimore, MD: The Johns Hopkins University Press, 3rd ed., 1996.
- [32] K. Gröchenig, *Foundations of Time-Frequency Analysis*. Appl. Numer. Harmon. Anal., Boston, MA: Birkhäuser Boston, 2001.
- [33] A. V. Oppenheim and R. W. Schaffer, *Discrete-time Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.
- [34] J. Cooley and J. Tukey, “An algorithm for the machine calculation of complex Fourier series,” *Math. Comp.*, vol. 19, pp. 297–301, 1965.
- [35] F. Stenger, *Numerical methods based on Sinc and analytic functions*. Springer New York, 1993.
- [36] W. Feller, *An Introduction to Probability Theory and its Applications Vol II*. New York-London-Sydney: John Wiley and Sons, Inc. XVIII, 1966.
- [37] J. Hammersley and D. Handscomb, *Monte Carlo Methods*. London: Methuen & Co Ltd, 1964.
- [38] A. F. Molisch, ed., *Wireless Communications*. John Wiley and Sons, Ltd., 2nd ed., 2010.
- [39] B. Sklar, *Digital Communications: Fundamentals and Applications*. Prentice Hall PTR, 2nd ed., 2001.
- [40] W. Kozek and A. Molisch, “Nonorthogonal pulseshapes for multicarrier communications in doubly dispersive channels,” *IEEE J. Sel. Areas Comm.*, vol. 16, pp. 1579–1589, Oct. 1998.

- [41] G. Matz, D. Schafhuber, K. Gröchenig, M. Hartmann, and F. Hlawatsch, “Analysis, optimization, and implementation of low-interference wireless multicarrier systems,” *IEEE Trans. Wireless Comm.*, vol. 6, pp. 1921–1931, May 2007.
- [42] R. Chang, “Synthesis of band-limited orthogonal signals for multichannel data transmission,” *Bell System Tech. J.*, vol. 45, pp. 1775–1796, Dec. 1966.
- [43] A. Peled and A. Ruiz, “Frequency domain data transmission using reduced computational complexity algorithms,” in *Proc. IEEE ICASSP-80*, vol. 5, (Denver, CO), pp. 964–967, Apr. 1980.
- [44] G. Giannakis, “Filterbanks for blind channel identification and equalization,” *IEEE Sig. Process. Letters*, pp. 184–187, Jun. 1997.
- [45] A. Scaglione, G. Giannakis, and S. Barbarossa, “Redundant filterbank precoders and equalizers, Parts I and II,” *IEEE Trans. Sig. Process.*, pp. 1988–2006 and 2007–2022, Jul. 1999.
- [46] J. Eiwen, “Signalwege.jpg.” i2eye-grafix, 2012.
- [47] H. Bölcskei, R. Koetter, and S. Mallik, “Coding and modulation for underspread fading channels,” in *Proc. IEEE ISIT-02*, p. 358, Jun. 2002.
- [48] G. Durisi, U. Schuster, H. Bölcskei, and S. Shamai, “Noncoherent Capacity of Underspread Fading Channels,” *IEEE Trans. Inform. Theory*, vol. 56, pp. 367 – 395, Jan. 2010.
- [49] P. Flandrin, *Time-frequency/time-scale Analysis (Wavelet Analysis and Its Applications)*. San Diego, CA: Academic Press, 1999.
- [50] “IEEE Standard 802.11a-1999(2003): Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” 2003.
- [51] Y. Shen and E. Martinez, “Channel estimation in OFDM systems,” *Freescale Semiconductor, Inc.*, Feb. 2006.
- [52] “IEEE Standard 802.11g-2003: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” 2003.
- [53] “IEEE Standard 802.16-2009: Air Interface for Broadband Wireless Access Systems,” 2009.

- [54] A. Kannu and P. Schniter, "MSE-optimal training for linear time-varying channels," in *Proc. IEEE ICASSP-05*, vol. 3, (Philadelphia, PA), pp. 789–792, Mar. 2005.
- [55] A. Kannu and P. Schniter, "Design and analysis of MMSE pilot-aided cyclic-prefixed block transmission for doubly selective channels," *IEEE Trans. Sig. Process.*, vol. 56, pp. 1148–1160, Mar. 2008.
- [56] G. B. Giannakis and C. Tepedelenlioglu, "Basis Expansion Models and Diversity Techniques for Blind Identification and Equalization of TimeVarying Channels," in *Proc. IEEE*, vol. 86, pp. 1969–1986, Oct. 1998.
- [57] D. Borah and B. Hart, "Frequency-selective fading channel estimation with a polynomial time-varying channel model," *IEEE Trans. Comm.*, vol. 47, pp. 862–873, Jun. 1999.
- [58] G. Leus, "On the estimation of rapidly varying channels," in *Proc. EUSIPCO-04*, vol. 4, pp. 2227–2230, Sep. 2004.
- [59] T. Zemen and C. Mecklenbräuker, "Time-variant channel estimation using discrete prolate spheroidal sequences," *IEEE Trans. Sig. Process.*, vol. 53, pp. 3597–3607, Sep. 2005.
- [60] A. Paulraj and T. Kailath, "Increasing capacity in wireless broadcast Systems using distributed transmission/directional reception (DTDR)," *US Patent No. 5, 345, 599*, Feb. 1993.
- [61] "IEEE Standard 802.11n-2009: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 2009.
- [62] "3GPP Release 7 HSPA+ (Evolved HSPA) Network Migration Analysis," Jan. 2009.
- [63] "3GPP TS36.300: Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN): Overall Description," 2008.
- [64] "Overview of 3GPP Release 12 V0.0.3," 2012.
- [65] J. Jayakumari, "MIMO-OFDM for 4G wireless systems," *Int. J. Eng. Sc. Tech.*, vol. 2, pp. 2886–2889, Jul. 2010.

- [66] J. Ma, Y. Zhang, X. Su, and Y. Yao, "Maximal Ratio Combining in Cellular MIMO-CDMA Downlink Systems," in *Proc. ICC-07*, pp. 4243–4248, Jun. 2007.
- [67] D. Palomar, J. Cioffi, and M. Lagunas, "Joint Tx-Rx beamforming design for multicarrier MIMO channels: A unified framework for convex optimization," *IEEE Trans. Sig. Process.*, vol. 51, pp. 2381–2401, Sep. 2003.
- [68] V. Tarokh, N. Seshadri, and R. Calderbank, "Space-time Codes for High Data Rate Wireless Communications: Performance Criterion and Code Construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.
- [69] H. Bölcskei, D. Gesbert, C. B. Papadias, and A.-J. van der Veen, *Space-time wireless systems. From array processing to MIMO communications*. Cambridge Univ. Press, 2006.
- [70] C. Oestges and B. Clerckx, *MIMO Wireless Communications*. Academic Press, 1st ed., 2007.
- [71] D. Wang, G. Zhu, and Z. Hu, "Optimal Pilots in Frequency Domain for Channel," in *Proc. IEEE 59th VTC-04 Spring*, vol. 2, pp. 608–612, May 2004.
- [72] Z. Wu, J. He, and G. Gu, "Design of Optimal Pilot-tones for Channel Estimation in MIMO-OFDM Systems," vol. 1, pp. 12–17, Mar. 2005.
- [73] H. Zhang, J. Chen, Y. Tang, and S. Li, "Analysis of Pilot-Symbol Aided Channel Estimation for MIMO-OFDM Systems," vol. 1, pp. 299–303, Jun. 2004.
- [74] V. Kotelnikov, "On the carrying capacity of the "ether" and wire in telecommunications," Material for the First All-Union Conference on Questions of Communication, Izd. Red. Upr. Svyazi RKKA, Moscow, Jan. 1933.
- [75] H. Nyquist, "Certain Topics in Telegraph Transmission Theory," *Trans. Am. Inst. El. Eng. (AIEE)*, vol. 47, pp. 617–644, April 1928.
- [76] C. E. Shannon, "Communication in the presence of noise," in *Proc. IRE*, vol. 37, pp. 10–21, Jan. 1949.
- [77] M. Whittaker, "The "Fourier" theory of the cardinal function," *Proc. Edinburgh Math. Soc. (Ser. 2)*, pp. 169–176, Jul. 1928.
- [78] "Information Technology Digital Compression and Coding of Continuous-Tone Still Images Requirements and Guidelines, ISO/IEC 10918-1, ITU T.81," 1992.

Bibliography

- [79] “Information technology – Multimedia content description interface, ISO/IEC 15938-1:2002,” 2002.
- [80] J. A. Tropp, *Topics in sparse approximation*. PhD thesis, Univ. Texas at Austin, Austin, TX, Aug. 2004.
- [81] P.-A. Nitsche, “Sparse approximation of singularity functions,” *Constr. Approx.*, vol. 21, no. 1, pp. 63–81, 2005.
- [82] J. Quinonero Candela and C. Rasmussen, “A Unifying View of Sparse Approximate Gaussian Process Regression,” *J. Machine Learn. Res.*, vol. 6, pp. 1939–1959, Dec. 2005.
- [83] A. K. Fletcher, S. Rangan, and V. K. Goyal, “Rate-distortion bounds for sparse approximation,” in *Proc. IEEE/SP SSP-07*, pp. 254–258, Aug. 2007.
- [84] E. J. Candès, J. T. Tao, and J. K. Romberg, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.
- [85] E. J. Candès, J. K. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [86] D. L. Donoho, “For most large underdetermined systems of linear equations the minimal l^1 solution is also the sparsest solution,” *Commun. Pure Appl. Anal.*, vol. 59, no. 6, pp. 797–829, 2006.
- [87] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [88] E. J. Candès, “Compressive sampling,” in *Proc. Int. Congr. Math.*, (Madrid, Spain), Aug. 2006.
- [89] B. K. Natarajan, “Sparse approximate solutions to linear systems,” *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, 1995.
- [90] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

- [91] M. Fornasier, “Numerical methods for sparse recovery,” in *Theoretical Foundations and Numerical Methods for Sparse Recovery* (M. Fornasier, ed.), pp. 93–200, deGruyter, 2010.
- [92] E. J. Candès, “The restricted isometry property and its implications for compressed sensing,” *C. R. Acad. Sci. Paris S’er. I Math.*, vol. 346, pp. 589–592, May 2008.
- [93] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *J. Roy. Statist. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.
- [94] R. G. Baraniuk, M. Davenport, R. A. DeVore, and M. Wakin, “A simple proof of the restricted isometry property for random matrices,” *Constr. Approx.*, vol. 28, pp. 253–263, Dec. 2008.
- [95] H. Rauhut, “Compressive Sensing and Structured Random Matrices,” in *Theoretical Foundations and Numerical Methods for Sparse Recovery* (M. Fornasier, ed.), vol. 9 of *Radon Series Comp. Appl. Math.*, pp. 1–92, deGruyter, 2010.
- [96] M. Figueiredo, R. D. Nowak, and S. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE J. Sel. Top. Sig. Process.*, vol. 1, pp. 586–598, Dec. 2007.
- [97] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, “An Interior-Point Method for Large-Scale ℓ_1 -Regularized Least Squares,” *IEEE J. Sel. Top. Sig. Process.*, vol. 4, pp. 606–617, Dec. 2007.
- [98] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*. Philadelphia, PA: SIAM Studies Appl. Math., 1994.
- [99] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition,” in *1993 Conf. Rec. Asilomar Conf. Sig., Sys. Comp.*, vol. 1, pp. 40 – 44, Nov. 1993.
- [100] J. A. Tropp, “Greed is good: Algorithmic results for sparse approximation,” *IEEE Trans. Inform. Theory*, vol. 50, pp. 2231–2242, Oct. 2004.
- [101] Z. Xu, “A remark about orthogonal matching pursuit algorithm,” *Preprint, arXiv:1005.3093*, 2010.
- [102] T. Blumensath and M. Davies, “Gradient pursuits,” *IEEE Trans. Sig. Process.*, vol. 56, pp. 2370–2382, Jun. 2008.

- [103] J. A. Tropp and D. Needell, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. Comput. Harmon. Anal.*, vol. 26, pp. 301–321, Apr. 2008.
- [104] T. Blumensath and M. Davies, “Iterative hard thresholding for compressed sensing,” *Independent Component Analysis and Signal Separation*, vol. 27, pp. 265–274, Nov. 2009.
- [105] D. L. Donoho, I. Drori, J.-L. Starck, and Y. Tsaig, “Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit,” tech. rep., 2006.
- [106] D. Needell and R. Vershynin, “Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit,” *Found. Comput. Math.*, vol. 9, no. 3, pp. 317–334, 2009.
- [107] D. Needell and R. Vershynin, “Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit,” *IEEE J. Sel. Top. Sig. Process.*, vol. 4, pp. 310 – 316, Apr. 2010.
- [108] M. Mishali and Y. Eldar, “Blind multi-band signal reconstruction: Compressed sensing for analog signals,” *IEEE Trans. Sig. Process.*, vol. 57, pp. 993–1009, Mar. 2009.
- [109] M. Mishali and Y. C. Eldar, “From theory to practice: Sub-nyquist sampling of sparse wideband analog signals,” *IEEE J. Sel. Top. Sig. Process.*, vol. 4, pp. 375–391, Apr. 2010.
- [110] F. Parvaresh, H. Vikalo, S. Misra, and B. Hassibi, “Recovering sparse signals using sparse measurement matrices in compressed DNA microarrays,” *IEEE J. Sel. Top. Sign. Process.*, vol. 2, pp. 275–285, Jun. 2008.
- [111] A. Gramfort and M. Kowalski, “Improving M/EEG source localization with an inter-condition sparse prior,” in *Proc. IEEE ISBI-09*, (Paris, France), pp. 141–144, Jun. 2009.
- [112] Z. Xiang, Y. Xi, U. Hasson, and P. Ramadge, “Boosting with spatial regularization,” in *Proc. NIPS-09*, vol. 22, (British Columbia, Canada), pp. 2107–2115, Dec. 2009.

- [113] M. van den Schmidt, M. Friedlander, and K. Murphy, “Group sparsity via linear-time projection,” tech. rep., Univ. British Columbia, 2008.
- [114] Y. Eldar, P. Kuppinger, and H. Bölcskei, “Compressed sensing of Block-sparse signals: Uncertainty relations and efficient recovery,” *IEEE Trans. Sig. Process.*, vol. 58, pp. 3042–3054, Jun. 2010.
- [115] M. Stojnic, F. Parvaresh, and B. Hassibi, “On the reconstruction of block-sparse signals with an optimal number of measurements,” *IEEE Trans. Sig. Process.*, vol. 57, pp. 3075–3085, Aug. 2009.
- [116] R. G. Baraniuk, V. Cevher, M. Duarte, and C. Hedge, “Model-based compressive sensing,” *IEEE Trans. Inform. Theory*, vol. 56, pp. 1982–2001, Apr. 2010.
- [117] Y. Eldar and M. Mishali, “Robust recovery of signals from a structured union of subspaces,” *IEEE Trans. Inform. Theory*, vol. 55, pp. 5302–5316, Nov. 2009.
- [118] A. Lozano, G. Swirszcz, and N. Abe, “Group orthogonal matching pursuit for variable selection and prediction,” in *Proc. NIPS-09*, (Vancouver, Canada), Dec. 2009.
- [119] Z. Luo, M. Gaspar, J. Liu, and A. Swarni, “Distributed signal processing in sensor networks,” *IEEE Sign. Process. Mag.*, vol. 23, pp. 14–15, Jun. 2006.
- [120] M. Kowalski, “Sparse regression using mixed norms,” *Appl. Comput. Harmon. Anal.*, vol. 27, pp. 303 – 324, Nov. 2009.
- [121] M. Davies and Y. Eldar, “Rank Awareness in Joint Sparse Recovery,” *IEEE Trans. Inform. Theory*, vol. 58, pp. 1135 –1146, Feb. 2012.
- [122] J. Chen and X. Huo, “Theoretical results on sparse representations of multiple measurement vectors,” *IEEE Trans. Sig. Process.*, vol. 54, pp. 4634–4643, Dec. 2006.
- [123] J. A. Tropp, “Algorithms for simultaneous sparse approximation: part II: Convex relaxation,” *J. Sig. Process.*, vol. 86, pp. 589 – 602, Mar. 2006.
- [124] S. Cotter, B. Rao, K. Engan, and K. Kreutz Delgado, “Sparse solutions to linear inverse problems with multiple measurement vectors,” *IEEE Trans. Sig. Process.*, vol. 53, pp. 2477–2488, Jul. 2005.

- [125] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, “Algorithms for simultaneous sparse approximation: part I: Greedy pursuit,” *J. Sig. Process.*, vol. 86, pp. 572 – 588, Apr. 2006.
- [126] R. Gribonval, H. Rauhut, K. Schnass, and P. Vandergheynst, “Atoms of all channels, unite! Average case analysis of multi-channel sparse recovery using greedy algorithms,” *J. Fourier Anal. Appl.*, vol. 14, no. 5, pp. 655–687, 2008.
- [127] D. Baron, M. F. Duarte, M. B. Wakin, S. Sarvotham, and R. G. Baraniuk, “Distributed compressed sensing,” tech. rep., Oct. 2006.
- [128] M. Duarte, V. Cevher, and R. Baraniuk, “Model-based compressive sensing for signal ensembles,” in *Proc. Allerton Conf. Comm., Contr., Comp.*, (Monticello, IL), Sep. 2008.
- [129] Y. Eldar and H. Rauhut, “Average case analysis of multichannel sparse recovery using convex relaxation,” *IEEE Trans. Inform. Theory*, vol. 56, pp. 505–519, Jan. 2010.
- [130] N. Vaswani and W. Lu, “Modified-CS: Modifying compressive sensing for problems with partially known support,” *IEEE Trans. Sig. Process.*, vol. 58, pp. 4595–4607, Sep. 2010.
- [131] W. Lu and N. Vaswani, “Modified basis pursuit denoising (modified- BPDN) for noisy compressive sensing with partially known support,” in *Proc. IEEE ICASSP-10*, (Dallas, TX), pp. 3926–3929, Mar. 2010.
- [132] L. Jacques, “A Short Note on Compressed Sensing with Partially Known Signal Support,” tech. rep., 2009.
- [133] T. Zhang, “Sparse recovery with orthogonal matching pursuit under RIP,” *IEEE Trans. Inform. Theory*, vol. 57, pp. 6215–6221, Sep. 2011.
- [134] M. Davenport and M. Wakin, “Analysis of orthogonal matching pursuit using the restricted isometry property,” *IEEE Trans. Inform. Theory*, vol. 56, pp. 4395–4401, Sep. 2010.
- [135] J. Paredes, G. Arce, and Z. Wang, “Ultra-Wideband Compressed Sensing: Channel Estimation,” *IEEE J. Sel. Top. Sign. Process.*, vol. 1, pp. 383–395, Oct. 2007.

- [136] E. Lagunas and M. Najar, “Sparse Channel Estimation based on Compressed Sensing for Ultra WideBand Systems,” in *Proc. IEEE ICUWB-11*, (Cesena, Italy), pp. 365–369, Sep. 2011.
- [137] R. Baraniuk and P. Steeghs, “Compressive radar imaging,” in *Proc. IEEE Radar Conf. 07*, (Boston, MA), pp. 128–133, Apr. 2007.
- [138] M. Herman and T. Strohmer, “High-resolution radar via compressed sensing,” *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2275–2284, 2009.
- [139] C. Berger, S. Zhou, J. Preisig, and P. Willett, “Sparse Channel Estimation for Multicarrier Underwater Acoustic Communication: From Subspace Methods to Compressed Sensing,” *IEEE Trans. Sig. Process.*, vol. 58, pp. 1708–1721, Mar. 2010.
- [140] C. Berger, Z. Wang, J. Huang, and S. Zhou, “Application of Compressive Sensing to Sparse Channel Estimation,” *IEEE Comm. Mag.*, vol. 48, pp. 164–174, Nov. 2010.
- [141] A. F. Molisch, ed., *Wideband Wireless Digital Communications*. Englewood Cliffs (NJ): Prentice Hall, 2001.
- [142] G. Matz and F. Hlawatsch, “Time-varying communication channels: Fundamentals, recent developments, and open problems,” in *Proc. EUSIPCO-06*, (Florence, Italy), Sep. 2006.
- [143] G. Del Galdo and M. Haardt, “IlmProp: A flexible geometry-based simulation environment for multiuser MIMO communications,” *COST 273 Temporary Document, No. TD(03)188*, Sep. 2003.
- [144] E. van den Berg and M. Friedlander, “SPGL1: A solver for large-scale sparse reconstruction,” <http://www.cs.ubc.ca/labs/scl/spgl1>, Jun. 2007.
- [145] D. Mary, “cosamp.m.”
- [146] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. The MIT Press, 3rd ed., 2009.

Curriculum Vitae

Mag. Daniel Eiwien

Personal Information

Date of birth: February 28th, 1983, Oberpullendorf, Austria
Marital status: Married, to Mag. Julie Eiwien since 2004
Nationality: Austria

Education

1989–1990 Volksschule Neunkirchen
1990–1993 Volksschule Würflach
1993–1997 Gymnasium Neunkirchen
1997–2001 Bundesoberstufenrealgymnasium Wiener Neustadt,
with special focus on musical education
2001 Matura with distinction
2001–2003 Diploma studies in Mathematics, first part, University of Vienna
2003–2004 Civil service, Wiener Neustadt
2004–2008 Diploma studies in Mathematics, second part, University of Vienna,
focus topic: *Applied Mathematics*
2008 Magister (Mag. rer. nat.) with distinction,
master thesis: *Die Golay Codes*
2008–present PhD studies in Mathematics, University of Vienna

Current Position

2008–present Graduate research student at Numerical Harmonic Analysis Group (NuHAG), Faculty of Mathematics, University of Vienna, funded by the WWTF project SPORTS and the FWF project SISE.

The information furnished in this thesis is true to my knowledge as of May 1st, 2012.

Mag. Daniel Eiwien