Fakultät für Elektrotechnik und Informationstechnik
Technische Universität München

Transceiver Design for Large Scale Communication Systems

David Neumann

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:
Prof. Dr. Wolfgang Kramer

Prüfende der Dissertation:

1. Prof. Dr.-Ing. Wolfgang Utschick
2. Assoc. Prof. Emil Björnson, Ph.D.

Die Dissertation wurde am 08.01.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 21.08.2020 angenommen.

Abstract

In large-scale wireless networks, effective transceiver design based on imperfect knowledge of the wireless channel is crucial. In this work, we introduce methods for transceiver design and resource allocation that make use of asymptotic results to reduce computational complexity. We focus on linear transceivers, which are close to optimal for the system dimensions we have in mind. Since the introduced methods rely heavily on second order statistics of the wireless channel, we further discuss efficient methods for the acquisition of these statistics in practical systems.

Acknowledgments

My time at the institute was both fun and educational. I greatly appreciate my supervisor Wolfgang Utschick for providing such an open, friendly and collaborative environment, which was the perfect foundation for working on challenging research topics. With some guidance at the right time, he and our postdoc Michael Joham managed to steer my focus in the right direction, helping me to find the right topic for this dissertation.

I thank all my colleagues for being great friends and co-workers and for always being open for technical and non-technical discussions. I especially thank Thomas Wiese for helping with proof-reading this document (and for sharing the blame for any remaining errors).

Last but not least, I also thank my wife Lu for her continuous love and support.

Contents

Chapter 1

Introduction

In the quest for higher spectral efficiency of wireless communication systems, multiple-input-multiple-output (MIMO) technology plays a crucial role. Advocates of MIMO technology promise huge gains which are achieved by spatial multiplexing of several data streams. However, while MIMO is slowly adopted in real-world applications, the gains are significantly lower than those promised by theory due to a myriad of practical problems. Early work on simple MIMO systems, but also recent work on more complicated setups, assumes perfect knowledge of the channel state. Acquiring this channel state information is one of the major challenges in practical wireless MIMO systems.

Starting a few years back, there has been a growing interest in large-scale wireless systems. The vision is to have large antenna arrays at the base stations, which serve a large number of mobile users simultaneously. If the number of base-station antennas is several times the number of served users, simple linear signal processing methods are close to optimal. The large systems, sometimes called massive MIMO systems, again promise large multiplexing gains through multi-user MIMO, but with simpler, more practical signal processing methods than those necessary for smaller MIMO systems.

While single-user MIMO is widely used in todays communication systems, effective multi-user MIMO operation is still challenging.

The question which a curious reader might ask now is, what is the difference between the large-scale system and the small system from a theoretical perspective? While the basic models and signal processing approaches are the same, new challenges become apparent when scaling a wireless system. Not surprisingly, these challenges mostly concern the acquisition of channel state information. Since imperfect channel state information is one of the main challenges for MIMO, and even more so for massive MIMO communications, most work on massive MIMO uses stochastic models to describe the uncertainty of the available side-information with respect to the channel state.

One central issue concerning acquisition of channel state information in large-scale communication systems is the limited coherence interval of the fast-fading channels. In fact, for certain channel models, the limited coherence interval acts as a dimensionality bottleneck of the system: We know from information theory that for a block-fading MIMO channel with the usual assumptions, the maximal number of interference-free transmission streams – interpreted as the dimension of the system – is half the length of the coherence interval (in channel accesses). Because it is not reasonable to serve more users than interference-free transmission streams can be generated, the number of simultaneously served users is essentially limited by the length of the coherence interval.

After first encountering those results, one might have a grim outlook on the potential of a large-scale wireless communication system. After all, the coherence intervals are quite short and thus scaling the system beyond a certain point appears futile. We are saved, however, by the fact that this theoretic result only holds for channels without correlation between the channel coefficients. Since in a typical wireless MIMO channel, we have different spacial correlations for different users, these correlations can be exploited to actually break out of the dimensionality bottleneck. While work on transceiver design with imperfect channel state information (CSI) existed before massive MIMO, we will see that for large-scale systems it is essential to use an explicit model for the imperfect CSI.

It will come as no surprise to the seasoned researcher that a lot of work in massive MIMO rediscovers results from the last thirty years.

In this sense, research on signal processing for massive MIMO is an evolution from previous work that focuses on the imperfect CSI, but also on the imperfect knowledge of the channel statistics. We will see that robust methods, which take imperfect CSI into account, rely on the channel covariance matrices. We will also see that in large-scale systems the influence of the fast fading vanishes and accurate approximations of system performance can be calculated using the channel statistics only.[1] Such approximations can be used for system analysis but more importantly for low-complexity resource allocation. Since all of these methods are based on covariance matrices of the

[1] An effect which is known as channel hardening [1].

channels, we will additionally discuss how to acquire second-order statistics in a practical manner.

This work is divided into three major chapters. We go from perfect knowledge of the channel in Chapter 2 over imperfect knowledge of the channel in Chapter 3 to imperfect knowledge of the channel covariance matrices in Chapter 4. Each reduction in available information leads to additional challenges that we need to overcome with efficient signal processing methods.

Chapter 2, where we discuss optimal linear transceiver design with perfect CSI, introduces the system model and also some basic methods for asymptotic analysis. These methods are extended in Chapter 3, where we augment the system model by different models for the available CSI. We will see that knowledge of the covariance matrices is important to achieve good performance with a large number of antennas. As usual in engineering, we can design algorithms with different trade-offs between computational complexity and performance. Since for a large-scale system it is essential to keep the computational complexity in check, we introduce a low-complexity bilinear transceiver design, which is asymptotically optimal.

Since covariance matrix information plays such an important role, we discuss methods for covariance matrix information in Chapter 4. We introduce established maximum likelihood estimators, but also novel learning-based methods that make use of prior information of the covariance matrices.

The work is concluded in Chapter 5 where we summarize the key takeaways of this work and give an outlook on future research directions.

Chapters 2 to 4 all have a Summary section that summarizes important theoretical results of the chapters.

1.1 Notation

Boldface lowercase letters ($\boldsymbol{a}, \boldsymbol{x}, \ldots$) are used for vectors and boldface upper case letters ($\boldsymbol{A}, \boldsymbol{X}, \ldots$) for matrices. We use the following special matrices and vectors.

$\mathbf{0}$	vector or matrix of all zeros
$\mathbf{1}$	vector or matrix of all ones
$\mathbf{I}$	identity matrix

$\mathrm{diag}(a_1, \ldots, a_n)$	diagonal matrix with diagonal elements $a_i, i = 1, \ldots, n$
$\mathrm{blkdiag}(\boldsymbol{A}_1, \ldots, \boldsymbol{A}_n)$	block-diagonal matrix with diagonal blocks $\boldsymbol{A}_i, i = 1, \ldots, n$
$\boldsymbol{C}_{\boldsymbol{x}}$	covariance matrix of a random vector $\boldsymbol{x}$

We use the following operators.

$\mathrm{diag}(\boldsymbol{a})$	diagonal matrix with the elements in $\boldsymbol{a}$ on the diagonal
$\mathrm{diag}^*(\boldsymbol{A})$	adjoint of the $\mathrm{diag}(\boldsymbol{x})$ operator, i.e., the result is a vector containing the diagonal elements of $\boldsymbol{A}$
$\boldsymbol{A}^{\mathrm{T}}$	transpose of a matrix of vector
$\boldsymbol{A}^{\mathrm{H}}$	conjugate transpose of a matrix or vector
$\boldsymbol{A}^{-1}$	inverse of a matrix
$\mathrm{tr}(\boldsymbol{A})$	trace of a matrix
$\det(\boldsymbol{A})$	determinant of a matrix
$\mathrm{rank}(\boldsymbol{A})$	rank of a matrix
$\mathrm{E}[\cdot]$	expectation
$a \asymp b$	asymptotic equivalence
$\boldsymbol{A} \succ \mathbf{0}$	positive definite
$\mathrm{var}(a)$	variance

Chapter 2

Transceiver Design with Perfect CSI

In this chapter, we introduce the system model for wireless communication in a cellular network that is used throughout this work. We continue to discuss the design of linear receive filters and linear precoders based on this system model. For now, we assume that the wireless channel between receiver and transmitter is perfectly known on both sides. An assumption that will be removed in the subsequent chapters.

The results in the first few sections of this chapter are well-known and are covered in textbooks and fundamental publications on MIMO systems. A detailed introduction to wireless communication is given in [2] and additional information on MIMO systems can be found in [3].

2.1 System Model for the Cellular Network

In most of this work, we will consider a single base station which serves several users simultaneously. The base station makes use of an array of antennas to transmit and receive signals, while each user has only a single antenna. The antennas could also be distributed in a larger area in a "cell-free" architecture. The antenna arrangement determines the structure of the channel and will be quite important in the next chapter when we discuss imperfect CSI.

For more details on system models used in wireless communications we refer again to the excellent book by Viswanath and Tse [2].

We assume flat-fading channels between the different users and the base station, which in a practical cellular system is the result of orthogonal frequency division multiplexing (OFDM). That is, in one channel access in the uplink each user k transmits a unit-variance data symbol s_k, amplified by the normalized transmit power coefficients p_k, which is linearly scaled by the channel vector $\boldsymbol{h}_k$. The base station receives the superposition of the scaled signals

$$\boldsymbol{y}^{\text{ul}} = \sum_k \sqrt{p_k}\boldsymbol{h}_k s_k + \boldsymbol{v}^{\text{ul}} \quad \in \mathbb{C}^M \tag{2.1}$$

with additive white Gaussian noise $\boldsymbol{v}^{\text{ul}} \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{C}_{\boldsymbol{v}})$.

In the downlink, we have the reverse model. That is, the base-station transmits a vector of signals $\boldsymbol{x}$ from the antenna array leading to the scalar received signal

$$y_k^{\text{dl}} = \boldsymbol{h}_k^{\text{T}}\boldsymbol{x} + v_k^{\text{dl}} \tag{2.2}$$

at user k. Without loss of generality, we assume that the system is normalized such that the noise has unit variance, i.e., $v_k^{\text{dl}} \sim \mathcal{N}_{\mathbb{C}}(0,1)$. Thus, if we impose an average sum-power constraint on the transmit vector $\boldsymbol{x}$ of the form

$$\mathrm{E}[\boldsymbol{x}^{\text{H}}\boldsymbol{x}] \leq \rho_{\text{dl}}$$

then ρ_{dl} denotes the signal to noise ratio (SNR) in the downlink, i.e., the ratio of the maximum transmit power at the base station to the noise power at the receiver. In the discussion of imperfect CSI in Chapter 3, we assume that the channels in the uplink and downlink are identical. This assumption of channel reciprocity is not actually necessary in the case of perfect CSI at both the transmitter and the receiver.

In practice, we need proper calibration to ensure channel reciprocity (see e.g. [4]). We assume that such a calibration is in place.

The results for perfect CSI, which we derive in the following, serve as an upper bound for the case of imperfect CSI, which we discuss later on. We will also introduce several techniques and results that will appear in similar forms in subsequent chapters.

2.2 Uplink Filter Design

The amount of data that can be transmitted depends on the processing of the received signals $\boldsymbol{y}^{\text{ul}}$ in the uplink and the design of the transmit signals $\boldsymbol{x}$ in the downlink. We focus the discussion on linear signal processing since, especially for larger antenna arrays, non-linear methods need a significant overhead in computational complexity for limited gains. In the uplink, we apply linear filters to the received signals to get scalars

$$\hat{s}_k = \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{y}^{\mathrm{ul}} = \sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k s_k + \sum_{n \neq k} \sqrt{p_n} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n s_n + \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{v}^{\mathrm{ul}} \qquad (2.3)$$

which can be interpreted as estimations of the data symbols s_k transmitted by the users. Now only the estimates $\hat{s}_k$ are used to reconstruct the information transmitted by user k. The challenge from a signal processing perspective is to optimize the linear filters $\boldsymbol{g}_k$ and the transmit powers p_k with respect to some performance measure and subject to constraints on the p_k. For the estimation of s_k we consider all interfering users as noise. Consequently, it does not matter whether an interfering user is served by the same base station or whether the interferer is located in a neighboring cell.

In a communication system, the performance measure of choice is the achievable rate, i.e., the maximum amount of information that can be received with arbitrarily small probability of error for a given transceiver design. A fundamental result from information theory states that the achievable rate for our system is given by the mutual information $\mathcal{I}(s_k; \hat{s}_k)$ which depends on the probability distribution of the transmit symbols s_k. In the following, we assume Gaussian signaling $s_k \sim \mathcal{N}_{\mathbb{C}}(0, 1)$, for which closed form expressions of the achievable rates are known.

Gaussian signalling is optimal for a single user but also for optimal non-linear processing at the base station. It is not optimal for the per-stream linear processing we are doing here due to the inter-stream interference [2]. For most interference scenarios, the optimal distribution of the transmit symbols is unknown.

If we consider again the system model for linear processing for user k,

$$\hat{s}_k = \underbrace{\sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k}_{h_{\text{eff}}} s_k + \underbrace{\sum_{n \neq k} \sqrt{p_n} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n s_n + \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{v}^{\mathrm{ul}}}_{v_{\text{eff}}} \qquad (2.4)$$

we note that it can be interpreted as a single input single output (SISO) system with effective channel $h_{\text{eff}} = \sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k$ and effective noise v_{eff}, which contains bot the filtered additive noise and the interference. Thus, we can apply the results from Appendix A to calculate the mutual information

$$r_k^{\mathrm{ul}} = \mathcal{I}(s_k; \hat{s}_k) = \log_2(1 + \gamma_k^{\mathrm{ul}}) \qquad (2.5)$$

where γ_k^{ul} is the signal to interference and noise ratio (SINR)

$$\begin{aligned}\gamma_k^{\mathrm{ul}} &= \frac{|h_{\mathrm{eff}}|^2}{\mathrm{var}(v_{\mathrm{eff}})} \\ &= \frac{\left|\sqrt{p_k}\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_k\right|^2}{\mathrm{var}(\boldsymbol{g}_k\boldsymbol{v}^{\mathrm{ul}} + \sum_{n\neq k}\sqrt{p_n}\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_n s_n)} \\ &= \frac{p_k\left|\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_k\right|^2}{\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}\boldsymbol{g}_k + \sum_{n\neq k}p_n\left|\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_n\right|^2} \\ &= \frac{p_k\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{g}_k}{\boldsymbol{g}_k^{\mathrm{H}}\left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n\neq k}p_n\boldsymbol{h}_n\boldsymbol{h}_n^{\mathrm{H}}\right)\boldsymbol{g}_k}.\end{aligned} \tag{2.6}$$

The rate is maximized with respect to the filter by solving

$$\max_{\boldsymbol{g}_k}\gamma_k^{\mathrm{ul}}. \tag{2.7}$$

For $\boldsymbol{A}\succ\boldsymbol{0}$, we know that

$$\underset{\boldsymbol{x}}{\mathrm{argmax}}\,\frac{\boldsymbol{x}^{\mathrm{H}}\boldsymbol{c}\boldsymbol{c}^{\mathrm{H}}\boldsymbol{x}}{\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{x}} = \alpha\boldsymbol{A}^{-1}\boldsymbol{c} \tag{2.8}$$

for arbitrary $\alpha\neq 0$. Consequently, one optimal filter is given by

$$\boldsymbol{g}_k^{\star} = \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n\neq k}p_n\boldsymbol{h}_n\boldsymbol{h}_n^{\mathrm{H}}\right)^{-1}\boldsymbol{h}_k \tag{2.9}$$

This filter is proportional to the linear minimum mean square error (LMMSE) filter, i.e., the linear filter that minimizes the mean square error (MSE) $\mathrm{E}[|s_k - \hat{s}_k|^2]$. It is well known that an LMMSE receive filter is optimal in our setup.

leading to the optimal SINR

$$\begin{aligned}\gamma_k^{\star} &= p_k\boldsymbol{h}_k^{\mathrm{H}}\left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n\neq k}p_n\boldsymbol{h}_n\boldsymbol{h}_n^{\mathrm{H}}\right)^{-1}\boldsymbol{h}_k \qquad (2.10)\\ &= p_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{g}_k^{\star}. \qquad (2.11)\end{aligned}$$

The following reformulation of the optimal SINRs will be useful for asymptotic analysis later on.

Lemma 2.1. *For a non-trivial power allocation $\boldsymbol{p} > \boldsymbol{0}$, the optimal SINR of user k is given by*

$$\gamma_k^{\star} = p_k\frac{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\left(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\right)^{-1}\boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}}\left(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\right)^{-1}\boldsymbol{e}_k} \tag{2.12}$$

where $\boldsymbol{P} = \operatorname{diag}(\boldsymbol{p})$ *and* $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_K]$*. The optimal LMMSE filter for user* k *can be expressed as*

$$\tilde{\boldsymbol{g}}_k^\star = \frac{1}{\sqrt{p_k}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{H} (\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{H})^{-1} \boldsymbol{e}_k \tag{2.13}$$

which is a scaled versions of the (also) optimal filter $\boldsymbol{g}_k^\star$ *in* (2.9)*.*

Proof. With the matrices introduced in Lemma 2.1 the optimal filter (2.9) can be rewritten to

$$\boldsymbol{g}_k^\star = \left(\boldsymbol{C}_{\boldsymbol{v}} + \boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^{\mathrm{H}} - p_k \boldsymbol{h}_k \boldsymbol{h}_k^{\mathrm{H}}\right)^{-1} \boldsymbol{h}_k. \tag{2.14}$$

Substituting $\boldsymbol{A} = \boldsymbol{C}_{\boldsymbol{v}} + \boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^{\mathrm{H}}$ and applying the matrix inversion lemma we get

$$\begin{aligned}
\boldsymbol{g}_k^\star &= \left(\boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}\boldsymbol{h}_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}}{\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k - \frac{1}{p_k}}\right)\boldsymbol{h}_k &(2.15)\\
&= \boldsymbol{A}^{-1}\boldsymbol{h}_k \left(1 - \frac{\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k}{\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k - \frac{1}{p_k}}\right) &(2.16)\\
&= \boldsymbol{A}^{-1}\boldsymbol{h}_k \left(\frac{-\frac{1}{p_k}}{\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k - \frac{1}{p_k}}\right) &(2.17)\\
&= \boldsymbol{A}^{-1}\boldsymbol{h}_k \left(\frac{1}{1 - p_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k}\right). &(2.18)
\end{aligned}$$

If we apply the matrix inversion lemma again, we get

$$\begin{aligned}
\boldsymbol{A}^{-1}\boldsymbol{h}_k &= \left(\boldsymbol{C}_{\boldsymbol{v}} + \boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^{\mathrm{H}}\right)^{-1}\boldsymbol{H}\boldsymbol{e}_k &(2.19)\\
&= \frac{1}{p_k}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}\left(\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H} + \boldsymbol{P}^{-1}\right)^{-1}\boldsymbol{e}_k &(2.20)
\end{aligned}$$

and thus

$$\begin{aligned}
\gamma_k^\star = p_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{g}_k^\star &= \frac{p_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k}{1 - p_k\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{A}^{-1}\boldsymbol{h}_k} &(2.21)\\
&= \frac{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\boldsymbol{e}_k}{1 - \boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\boldsymbol{e}_k}\\
&= \frac{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}}\left(\mathbf{I} - \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\right)\boldsymbol{e}_k}\\
&= p_k\frac{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}}(\boldsymbol{P}^{-1} + \boldsymbol{H}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{v}}^{-1}\boldsymbol{H})^{-1}\boldsymbol{e}_k}. &(2.22)
\end{aligned}$$

From (2.20), we see that (2.13) is optimal, since scaling of the receive filter does not affect the SINR. □

The optimal SINRs $\gamma_k^\star$ of the different users are coupled via the power coefficients p_k, increasing p_k for one user k increases $\gamma_k^\star$ but decreases the SINRs of other users n for which $\boldsymbol{g}_n^{\mathrm{H}}\boldsymbol{h}_k \neq 0$. In other words, we have a multi-objective problem, since the rates of different users cannot be maximized separately. Each set of feasible power allocations $\boldsymbol{p} = [p_1, \ldots, p_K]^{\mathrm{T}}$ leads to a vector of achievable rates $\boldsymbol{r} = [r_1, \ldots, r_K]^{\mathrm{T}}$. The set of all possible rate vectors, the rate region, depends on the constraints that we impose on the power allocation. If we require $\boldsymbol{p}$ to be fixed and impose an average power constraint on each user we get

For notational convenience we assume that all users have the same maximum transmit power ρ_{ul}.

$$\mathcal{R} = \{\boldsymbol{r}(\boldsymbol{p}) : \mathbf{0} \leq \boldsymbol{p} \leq \rho_{\mathrm{ul}}\mathbf{1}\} \tag{2.23}$$

which is non-convex in general. Typically, the multiple objectives are combined in a network utility maximization (NUM) problem. That is, we want to solve

$$\max_{\boldsymbol{r}\in\mathcal{R}} U(\boldsymbol{r}) \tag{2.24}$$

for a concave utility function $U(\boldsymbol{r})$. Important examples include the proportional fair utility

$$U^{\mathrm{PF}}(\boldsymbol{r}) = \sum_k \log(r_k) \tag{2.25}$$

the weighted sum-rate (WSR) maximization

$$U^{\mathrm{WSR}}(\boldsymbol{r};\boldsymbol{\xi}) = \boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{r} \tag{2.26}$$

which is parameterized by the weights $\boldsymbol{\xi}$, and the max-min utility

$$U^{\text{max-min}}(\boldsymbol{r}) = \min_k r_k. \tag{2.27}$$

In general, the rate region $\mathcal{R}$ is non-convex and there is no practical algorithm that solves (2.24) globally optimally. The exception is the max-min problem that results from using the max-min utility (2.27). The max-min problem can be shown to have a unique optimizer [5, 6], which can be found efficiently via a fixed-point method or other convex programming approaches [7, 8, 6].

Locally optimal solutions for other objectives can be obtained in a straightforward manner by applying a projected gradient algorithm to the problem.

In a multi-cell network, we only control the power coefficients of the local users. There are various approaches for distributed optimization in such a setup (e.g. [7]). There are some heuristics we could use that do not require communication between the base stations. For example, we could try to estimate the power coefficients of the interfering users during the data phase and then apply a gradient step on the local coefficients, assuming the transmit power of the interferers stays constant. In this work, we will focus on the single-cell setup when we discuss resource allocation and leave the extension to the multi-cell case for future work.

2.2.1 Ergodic Rate

In a wireless communication scenario with mobile users, the channel vectors are time-varying in relatively short time intervals, so called fast fading. In this case, the achievable rate has to be averaged over the different channel realizations. This leads us to the ergodic rate

$$\bar{r}_k = \mathrm{E}[r_k] = \mathrm{E}[\log_2(1 + \gamma_k^\star)]. \tag{2.28}$$

The size of the ergodic rate region depends on how fast we are able to adapt the power coefficients $\boldsymbol{p}$.

If the power coefficients vary slowly compared to the channel coefficients, we can consider the power coefficients as deterministic and define the ergodic rate region

$$\bar{\mathcal{R}} = \{\mathrm{E}[\boldsymbol{r}(\boldsymbol{p})] : \mathbf{0} \leq \boldsymbol{p} \leq \rho_{\mathrm{ul}}\mathbf{1}\} \tag{2.29}$$

very similarly to the instantaneous rate region. Instead of a gradient-based approach we could use a stochastic gradient method to find the optimal power allocation for some NUM problem.

On the other hand, if we are able to choose a different $\boldsymbol{p}$ for each channel realization $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_K]$ we have to consider all mappings $\boldsymbol{p}(\boldsymbol{H})$ that return a feasible power allocation. That is, the region of achievable ergodic rates is given by

$$\widetilde{\mathcal{R}} = \{\mathrm{E}[\boldsymbol{r}(\boldsymbol{p}(\boldsymbol{H}))] : \mathbf{0} \leq \mathrm{E}[\boldsymbol{p}(\boldsymbol{H})] \leq \rho_{\mathrm{ul}}\mathbf{1}, \boldsymbol{p} : \mathbb{C}^{M\times K} \mapsto \mathbb{R}^K\}. \tag{2.30}$$

Here we restrict the average power $\mathrm{E}[\boldsymbol{r}(\boldsymbol{p}(\boldsymbol{H}))]$ instead of the deterministic coefficients we had in (2.29). Clearly, we have $\bar{\mathcal{R}} \subseteq \widetilde{\mathcal{R}}$, since $\boldsymbol{p}(\boldsymbol{H})$ is no longer restricted to be constant.

Analogously to the case of constant channels, the goal is to solve the NUM problem

$$\max_{\bar{\boldsymbol{r}} \in \bar{\mathcal{R}}} U(\bar{\boldsymbol{r}}) \quad \text{or} \quad \max_{\bar{\boldsymbol{r}} \in \widetilde{\mathcal{R}}} U(\bar{\boldsymbol{r}}) \tag{2.31}$$

with concave utility $U(\bar{\boldsymbol{r}})$. There is no easy method to solve the NUM problem with either the rate region $\bar{\mathcal{R}}$ in (2.29) or the rate region $\widetilde{\mathcal{R}}$ in (2.30). In principle, optimizing a constant power allocation needs lower complexity, since the optimization has to be done at the time scale of the channel statistics. To (approximately) optimize the adaptive power allocation in (2.30), we have to solve an optimization problem for each channel realization (cf. [2]). Also, as we will see later on, the large-scale systems we are interested in are governed by the channel statistics, i.e., a constant power allocation is close to optimal for large numbers of antennas.

For these reasons, we focus on achievable rate regions with constant power allocation throughout this work. As we will demonstrate in the next section, we can use asymptotically accurate approximations of the ergodic rate to find a close to optimal constant power allocation. An analogous approach for imperfect CSI is discussed in the next chapter.

2.3 Asymptotic Analysis

Since we are interested in scenarios where the number of users K is significantly smaller than number of antennas M, we are interested in the behavior of the SINRs $\gamma_k^\star$ for growing M. Additionally to the insights on the system performance, the resulting asymptotically equivalent rate expressions are also useful to optimize the resource allocation.

Remember: Two sequences a_M and b_M are asymptotically equivalent (denoted as $a_M \asymp b_M$) if $\lim_{M\to\infty} a_M/b_M = 1$.

For the asymptotic analysis for growing numbers of antennas, the distribution of the channel vectors $\boldsymbol{h}_k$ plays a fundamental role. Throughout this work, we assume the channel vectors of different users to be independent. Each channel vector is circularly symmetric complex Gaussian, i.e., $\boldsymbol{h}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{h}_k})$.

If we analyze the asymptotic behaviour for M going to infinity, we implicitly consider growing sequences of channel vectors $\boldsymbol{h}_k$ with growing covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k}$. The following results cannot be proven for all possible sequences of channel and noise covariance matrices.

In the following, we will discuss the conditions on the covariance matrices that are required for our asymptotic analysis. The first condition concerns the additive noise.

Condition 2.1.

$$\limsup_{M\to\infty} \|\boldsymbol{C}_{\boldsymbol{v}}\| < \infty \textit{ and } \limsup_{M\to\infty} \left\|\boldsymbol{C}_{\boldsymbol{v}}^{-1}\right\| < \infty. \tag{2.32}$$

This simply means that each spatial direction has a finite amount of noise power. This is a reasonable condition, since directions with infinite amount of noise are not useful and, on the other hand, modelling a communication system without noise (in some directions) is “like playing tennis without a net”. For some results in this paper we assume the noise covariance matrix to be a scaled identity matrix, which is also the typical assumption in massive MIMO literature. This special case assumes each receive antenna and the corresponding processing chain introduces an independent noise source. A scaled identity matrix clearly fulfills Condition 2.1.

The channel covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k}$ need to fulfill

Condition 2.2.

$$\liminf_{M\to\infty} \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})/M > 0 \textit{ and } \limsup_{M\to\infty} \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})/M < \infty. \tag{2.33}$$

With this condition, we restrict ourselves to models for which the captured energy grows linearly with the number of antennas. Most channel models that ignore antenna coupling have this property. If we take antenna coupling into account, Condition 2.2 requires a growing aperture of the array with the number of antennas.

Conditions 2.1 and 2.2 are used in all of the asymptotic results in this work. We need at least one additional condition that makes sure that the energy of the channel vector is spread in many directions.

For our first result, the following channel hardening condition is sufficient.

Condition 2.3.

$$\lim_{M\to\infty} \operatorname{var}(\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_k/M) = \lim_{M\to\infty} \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k})/M^2 = 0 \tag{2.34}$$

This condition is violated, e.g., by line of sight models where a single eigenvalue of $\boldsymbol{C}_{\boldsymbol{h}_k}$ grows linearly with M, but holds for typical non-line-of-sight models.

In other work, the channel hardening condition is often replaced by more restrictive conditions on the spectral norm of the covariance matrices [9]. It is easy to see that Condition 2.3 holds when the spectral norm is asymptotically bounded. However, requiring a bounded spectral norm excludes common channel models, e.g., the ubiquitous far-field model with a uniform linear array at the base station (cf. Appendix B.1).

We get the following, less restrictive result.

Lemma 2.2. *Given Condition 2.2, the channel hardening condition (Condition 2.3) is equivalent to*

$$\lim_{M\to\infty} \left\| \boldsymbol{C}_{\boldsymbol{h}_k} \right\| / M = 0. \tag{2.35}$$

Proof. If we have (2.35) and Condition 2.2, then

$$\limsup_{M\to\infty} \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k})/M^2 \leq \limsup_{M\to\infty} \frac{\left\|\boldsymbol{C}_{\boldsymbol{h}_k}\right\|}{M}\frac{\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})}{M} = 0. \tag{2.36}$$

For the converse, consider the case where $\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k})/M^2$ is fixed and we want to find the covariance matrix with maximum spectral norm. Clearly, the optimal choice is a rank-one matrix, such that $\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k})/M^2 = \left\|\boldsymbol{C}_{\boldsymbol{h}_k}\right\|^2/M^2$. Thus, we get (2.35) directly from Condition 2.3. □

Conditions 2.2 and 2.3 together tell us that the normalized inner product $\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_k/M$ converges towards a deterministic non-zero value. Due to this property, we can derive an asymptotically equivalent SINR

$$\gamma_k^{\text{asy}} = p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{v}}^{-1}). \tag{2.37}$$

Theorem 2.1. *Suppose we have independent channel vectors $\boldsymbol{h}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{h}_k})$. If the channel and noise covariance matrices fulfill Conditions 2.1, 2.2, and 2.3 we have*

$$\gamma_k^\star \asymp \gamma_k^{asy} \tag{2.38}$$

and

$$\liminf_{M\to\infty} \gamma_k^\star/M > 0. \tag{2.39}$$

Proof. Due to Conditions 2.1 and 2.2 we have

$$\liminf_{M\to\infty} \gamma_k^{\text{asy}}/M = \liminf_{M\to\infty} p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{v}}^{-1})/M > 0. \tag{2.40}$$

Thus, if we can show that

$$\lim_{M\to\infty} \gamma_k^\star/M - \gamma_k^{\text{asy}}/M = 0 \tag{2.41}$$

we proof the theorem.

We define a general function to express the normalized SINR

$$\tilde{\gamma}_k(\boldsymbol{B}, \alpha) = p_k \frac{\boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{B} (\alpha \boldsymbol{P}^{-1} + \boldsymbol{B})^{-1} \boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}} (\alpha \boldsymbol{P}^{-1} + \boldsymbol{B})^{-1} \boldsymbol{e}_k}. \tag{2.42}$$

With $\boldsymbol{B} = \boldsymbol{H}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{H} / M$ and $\alpha = 1/M$ we get $\tilde{\gamma}_k(\boldsymbol{B}, \alpha) = \gamma_k^{\star}/M$.

Due to Conditions 2.1 and 2.3, the variance of the elements of $\boldsymbol{B}$ vanishes and thus the entries are asymptotically equivalent to their expectations. We have

$$\lim_{M \to \infty} \frac{1}{M} \boldsymbol{h}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{h}_n = 0 \tag{2.43}$$

for $n \neq k$ and for $n = k$ we get

$$\lim_{M \to \infty} \frac{1}{M} \boldsymbol{h}_k^{\mathrm{H}} \boldsymbol{h}_k - \frac{\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})}{M} = 0. \tag{2.44}$$

With $\widetilde{\boldsymbol{B}} = \operatorname{diag}(\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_1} \boldsymbol{C}_{\boldsymbol{v}}^{-1})/M, \ldots, \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_K} \boldsymbol{C}_{\boldsymbol{v}}^{-1})/M)$, we have thus

$$\lim_{M \to \infty} \boldsymbol{B} - \widetilde{\boldsymbol{B}} = \mathbf{0}. \tag{2.45}$$

We see that $\tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0) = p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{C}_{\boldsymbol{v}}^{-1})/M$ is exactly the normalized asymptotic SINR. From $\lim_{n \to \infty} a_n - b_n = 0$ does not always follow $\lim_{n \to \infty} g(a_n) - g(b_n) = 0$. But it does follow if g is uniformly continuous.

Due to the conditions on the covariance matrices we know that in the limit, the smallest and largest singular values of $\tilde{\boldsymbol{B}}$ are bounded below and above respectively. Thus, for large M, the $\tilde{\boldsymbol{B}}$ (and consequently also the $\boldsymbol{B}$ with high probability) are in a compact subset of the positive definite matrices. Since $\tilde{\gamma}_k$ is continuously differentiable we even have Lipschitz continuity and thus

$$\lim_{\alpha \to 0, \left\| \boldsymbol{B} - \widetilde{\boldsymbol{B}} \right\| \to 0} \tilde{\gamma}_k(\boldsymbol{B}, \alpha) - \tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0) = 0 \tag{2.46}$$

which is the desired result. □

That is, we have an asymptotically equivalent SINR where the interference from other users is completely suppressed by the linear receive filter. Thus, for a large number of antennas it is clearly optimal to use the full power budget for each user irrespective of the utility function. For imperfect CSI, the asymptotically equivalent SINR has a fundamentally different structure as we will see in the next chapter.

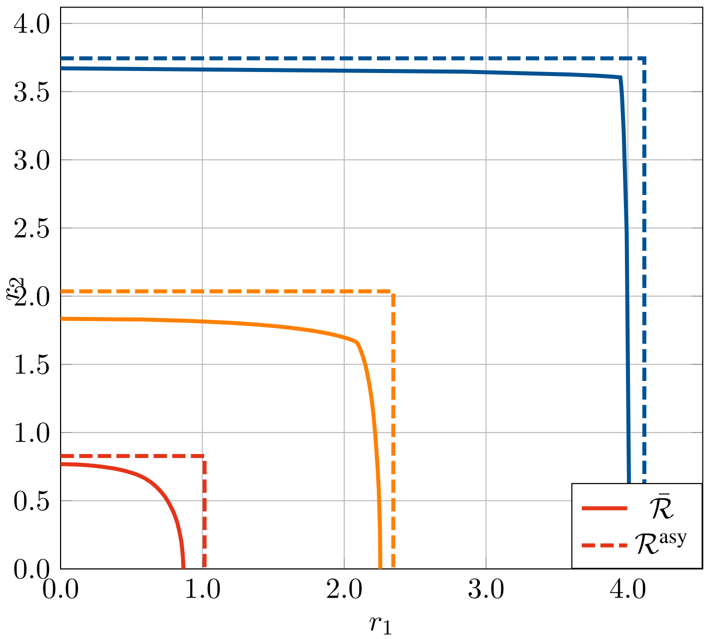


Figure 2.1: Achievable ergodic rate regions for a base-station with $M = 4, 16, 64$ antennas that serves $K = 2$ users. We also depict the approximate rate regions that result from replacing the SINRs with the asymptotically equivalent expressions. That is, we replace γ_k^{ul} with γ_k^{asy} to calculate the achievable rate of user k.

The asymptotically equivalent rate region

$$\mathcal{R}^{\text{asy}} = \{\boldsymbol{r} : [\boldsymbol{r}]_k = \log_2(1 + \gamma_k^{\text{asy}}),\ \mathbf{0} \leq \boldsymbol{p} \leq \rho_{\text{ul}}\mathbf{1}\} \tag{2.47}$$

is rectangular and only depends on the statistics of the channel. In Fig. 2.1 we depict the ergodic rate regions and the corresponding asymptotic rate regions for different numbers of antennas. We can see clearly that the rate region in (2.29) approaches the rectangular shape of the asymptotic region.

Since the asymptotically equivalent SINR in (2.37) does not depend on the instantaneous channel realizations, the asymptotic approximation of the instantaneous rates is also the asymptotic approximation of the ergodic rates. In fact, we can show that the asymptotic equivalent SINR yields an asymptotically tight outer bound of the ergodic rates.

Theorem 2.2. *The asymptotic rate region is an outer approximation of the ergodic rate region in* (2.30)

Proof. For a single user and constant channel $\boldsymbol{h}_k$ with maximum average transmit power ρ_{ul}, the capacity, i.e., the maximum of the mutual information $\mathcal{I}(s_k; \boldsymbol{y}^{\text{ul}})$ with respect to all distributions of s_k with unit variance, is given by

$$C_k = \log_2(1 + \rho_{\text{ul}} \boldsymbol{h}_k^{\text{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{h}_k). \tag{2.48}$$

We use Jensen's inequality to bound the ergodic capacity

$$\begin{aligned} \bar{C}_k &= \text{E}[C_k] = \text{E}[\log_2(1 + \rho_{\text{ul}} \boldsymbol{h}_k^{\text{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{h}_k)] \\ &\leq \log_2(1 + \rho_{\text{ul}} \, \text{E}[\boldsymbol{h}_k^{\text{H}} \boldsymbol{C}_{\boldsymbol{v}}^{-1} \boldsymbol{h}_k]) = \log_2(1 + \rho_{\text{ul}} \, \text{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{C}_{\boldsymbol{v}}^{-1})). \end{aligned} \tag{2.49}$$

This completes the proof, since the single user capacity is an upper bound for achievable multi-user rates, and since the upper bound in (2.49) is exactly the asymptotic rate for maximum transmit power $p_k = \rho_{\text{ul}}$. □

Corollary 2.1. *Linear precoding achieves capacity in the asymptotic limit.*

Proof. The proof follows directly from Theorem 2.2. □

These types of results prompted the rise in popularity of large-scale "massive MIMO" systems [10]. Since linear processing is asymptotically optimal, we no longer have to discuss complicated non-linear transceiver designs. Instead we can focus on more practical problems concerning resource allocation and the acquisition of channel state information.

2.3.1 Degrees of Massiveness

In interference limited networks, one popular performance measure are the degrees of freedom (DoF) [11]. If the network achieves a sum rate $r(\rho)$ with respect to the SNR ρ, the number of DoFs is given by

$$\#\text{DoF} = \lim_{\rho \to \infty} \frac{r(\rho)}{\log(\rho)}. \tag{2.50}$$

Basically, the DoFs are the number of interference-free data streams that can be transmitted in the network. In our scenario, for $\rho_{\text{ul}} \to \infty$, the optimal filters $\boldsymbol{g}_k^\star$ converge to zero-forcing filters, which achieve #DoF $= K$ as long as the channel vectors are linearly independent.

Similarly, for a sum-rate $r(M)$ that depends on the number of antennas M, we can define the degrees of massiveness as

$$\#\text{DoM} = \lim_{M \to \infty} \frac{r(M)}{\log(M)}. \tag{2.51}$$

Analogously to the DoF, the DoM tell us something about the number of interference-free data streams, but for an infinite number of antennas at the base station. For perfect CSI #DoM $=$ #DoF $= K$ since for large M the SINRs grow linearly with M. For suboptimal low-complexity transceiver designs or for systems with imperfect CSI, there are significant differences between the DoF and the DoM.

2.3.2 Is the Matched Filter Asymptotically Optimal?

Interestingly, due to the asymptotic orthogonality of the channel vectors (cf. (2.43)), the SINR for simple matched filters $\boldsymbol{g}_k = \boldsymbol{h}_k$ at the receiver also scales linearly with the number of antennas. Thus, the relative difference of the achievable rates between using matched filters and optimal linear filters vanishes for large numbers of antennas, i.e., the matched filter achieves the full K DoM. This is in contrast to the DoF for the matched filter, which are zero for non-orthogonal channel vectors since the SINRs saturate for large SNRs.

To show this, we first introduce a lower bound on the ergodic rate that is used in numerous works on massive MIMO [12, 7, 9, 13] and is based on the results in [14, 15]. The assumption behind the bound is that the decoder has no instantaneous CSI and can rely only on statistics to decode the desired signals.

Theorem 2.3. *For deterministic power allocations p_k, the achievable ergodic rates are lower bounded by*

$$\bar{r}_k \geq \log_2(1 + \gamma_k^{LB}) \tag{2.52}$$

with

$$\gamma_k^{LB} = \frac{p_k \left|\mathrm{E}[\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k]\right|^2}{\mathrm{E}[\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}} \boldsymbol{g}_k] + p_k \operatorname{var}(\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k) + \sum_{n \neq k} p_n \mathrm{E}[\left|\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n\right|^2]}. \tag{2.53}$$

Proof. The result follows from the considerations in Appendix A for a SISO system. Due to the per-user linear processing

$$\hat{s}_k = \underbrace{\sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k}_{g} s_k + \underbrace{\sum_{n \neq k} \sqrt{p_n} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n s_n + \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{v}^{\mathrm{ul}}}_{v} \tag{2.54}$$

we have the effective scalar channel $g = \sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k$, which we now assume to be unknown at the decoder. That is, the decoder only knows the statistics of the channel and thus the MMSE estimate of the effective channel is simply the expectation $\bar{g} = \mathrm{E}[\sqrt{p_k}\boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{h}_k]$. From Appendix A, Eq. (A.19) we know that for imperfect CSI we get an effective SINR of

$$\gamma_k^{\mathrm{LB}} = \frac{|\bar{g}|^2}{\mathrm{var}(v) + \mathrm{var}(g)} \tag{2.55}$$

which is the SINR given in (2.53). □

We can now evaluate the expectations in the bound in (2.53) when we use a matched filter at the base station.

Corollary 2.2. *For complex Gaussian channel vectors $\boldsymbol{h}_k \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{C}_{\boldsymbol{h}_k})$ and matched filters $\boldsymbol{g}_k = \boldsymbol{h}_k$ at the receiver, the achievable ergodic rates are lower bounded by*

$$\bar{r}_k \geq \log_2(1 + \gamma_k^{MF}) \tag{2.56}$$

with

$$\gamma_k^{MF} = \frac{p_k \,\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})^2}{\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{v}}) + \sum_n p_n \,\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_n})} \tag{2.57}$$

Proof. To derive the lower bound for the matched filter we need to evaluate the expectations in (2.52) for $\boldsymbol{g}_k = \boldsymbol{h}_k$. We use Lemma C.2 to get

$$\mathrm{E}[|\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_k|^2] = \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k}) + \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})^2 \tag{2.58}$$

and thus

$$\mathrm{var}(\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_k) = \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_k}). \tag{2.59}$$

Due to the independence of the different channel vectors we have

$$\mathrm{E}[|\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_n|^2] = \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_n}) \tag{2.60}$$

for $k \neq n$. Incorporating these results into (2.52) leads to the SINR given in (2.57). □

For the following asymptotic result, the channel hardening condition (Condition 2.3) is no longer sufficient. We need the more restrictive condition on the Frobenius norm

Condition 2.3 tells us that $\|\boldsymbol{C}_{\boldsymbol{h}_k}\|_F = o(M)$ while Condition 2.4 requires $\|\boldsymbol{C}_{\boldsymbol{h}_k}\|_F = O(\sqrt{M})$.

Condition 2.4.

$$\limsup_{M\to\infty} \|\boldsymbol{C}_{\boldsymbol{h}_k}\|_F^2 / M < \infty \tag{2.61}$$

to get the result

Corollary 2.3. *The rates achievable with matched filters are asymptotically equivalent to the capacity under Conditions 2.1, 2.2, and 2.4. That is, the matched filter achieves the full #DoM* $= K$.

Proof. For the given conditions on the covariance matrices, the normalized lower bound for matched filters in (2.57) is further bounded below by

$$\gamma_k^{\mathrm{MF}}/M \geq \frac{p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})^2/M^2}{\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{v}})/M + \sum_n p_n \left\|\boldsymbol{C}_{\boldsymbol{h}_k}\right\|_F \left\|\boldsymbol{C}_{\boldsymbol{h}_n}\right\|_F/M}. \tag{2.62}$$

Since $\liminf_{M\to\infty} \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})/M > 0$ due to Condition 2.2 the numerator is strictly greater than zero. Due to Conditions 2.1 and 2.2 the noise term is bounded and since $\limsup_{M\to\infty} \left\|\boldsymbol{C}_{\boldsymbol{h}_k}\right\|_F/\sqrt{M} < \infty$ due to Condition 2.4, the summands in the denominator are also bounded. Thus, we have

The noise term is bounded since $\operatorname{tr}(\boldsymbol{A}\boldsymbol{B}) \leq \|\boldsymbol{B}\| \operatorname{tr}(\boldsymbol{A})$ for positive semi-definite $\boldsymbol{A}$.

$$\lim_{M\to\infty} \gamma_k^{\mathrm{MF}}/M > 0 \tag{2.63}$$

and consequently

$$\lim_{M\to\infty} \log_2(\gamma_k^{\mathrm{MF}}/M) > -\infty. \tag{2.64}$$

We get the limit

The limits of both, γ_k^{MF}/M and $\gamma_k^{\mathrm{asy}}/M$, are bounded above and below due to the properties for the covariance matrices described earlier.

$$\lim_{M\to\infty} \frac{\log_2(1+\gamma_k^{\mathrm{MF}})}{\log_2(1+\gamma_k^{\mathrm{asy}})} = \lim_{M\to\infty} \frac{\log_2(M) + \log_2(1/M + \gamma_k^{\mathrm{MF}}/M)}{\log_2(M) + \log_2(1/M + \gamma_k^{\mathrm{asy}}/M)} \tag{2.65}$$

$$= \lim_{M\to\infty} \frac{\log_2(M)}{\log_2(M)} = 1. \tag{2.66}$$

Thus, we have one DoM per user which gives the full K DoMs in total. □

Just a quick comment regarding Condition 2.4. As we mentioned before, for different results in this work we need different levels of restrictions on the channel covariance matrices. We could just assume a bounded spectral norm of the channel covariance matrices and be done with it (because this would be the most severe restriction), but we like to use the most general conditions that still lead to the desired result.

In fact, Condition 2.4 is already too restrictive in that it does not include the uniform linear array (ULA) and uniform rectangular array (URA) channel models in Appendix B.1. The reason is that the underlying spectrum from which the eigenvalues of the channel covariance matrices are sampled is (barely) not quadratically integrable in some cases. That is, the terms $\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_n})$ might grow as $M\log(M)$ instead of M. However, since $M\log(M)$ is almost the same as M, this effect is not really visible in our simulations. In Section 3.8 of the next Chapter, we discuss a robust matched filter that does not have this limitation.

The result in Corollary 2.3 indicates that for a setup with $M \gg K$ we get close to optimal results by transmitting with full power and applying matched filters at the receiver. However, one important thing to note is that there can be a significant difference in the number of antennas that achieve a certain performance, even in the asymptotic regime. For example, if the covariance matrices are scaled identities with $\boldsymbol{C}_{\boldsymbol{h}_k} = \beta_k \mathbf{I}$ and we transmit with full power ($p_k = \rho_{\text{ul}}\ \forall k$), we have

$$\gamma_k^{\text{MF}} = M\frac{\beta_k}{1/\rho_{\text{ul}} + \sum_n \beta_n} = \frac{M\rho_{\text{ul}}\beta_k}{1 + \sum_n \rho_{\text{ul}}\beta_n} \tag{2.67}$$

and

$$\gamma_k^{\text{asy}} = M\rho_{\text{ul}}\beta_k. \tag{2.68}$$

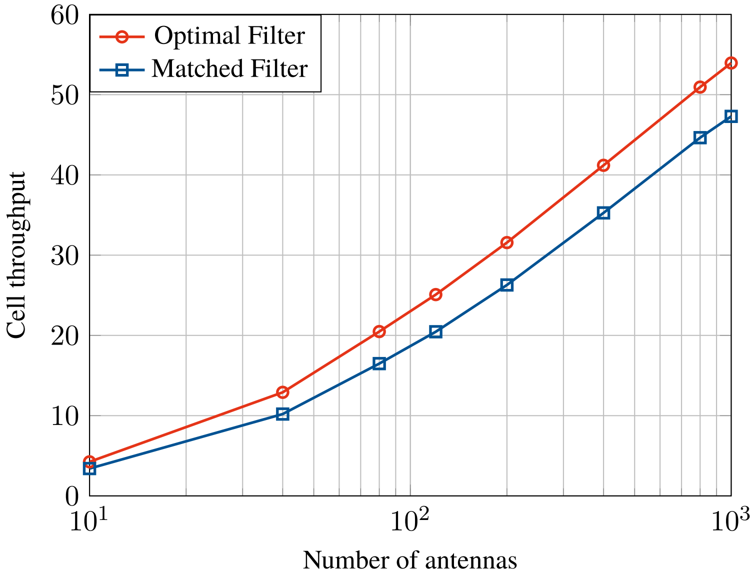
To achieve the same SINRs in the asymptotic regime, we need a factor of

$$1 + \rho_{\text{ul}}\sum_n \beta_n \tag{2.69}$$

more antennas for the matched filter. Since γ_k^{MF} yields a lower bound on the performance, the factor might be smaller in practice, but nevertheless, we can expect such a constant factor in required number of antennas even for high spectral efficiencies.

In other words, we have a choice between a more sophisticated filter design with a lower number of antennas and a simple filter with a higher number of antennas. Which choice is preferable depends on many practical considerations. One advantage of the matched filter is that it enables decentralized processing at each antenna, i.e., the channel coefficients do not have to be collected at a central processor. This is especially important for cell-free systems where the antennas are distributed in a large area.

Figure 2.2: Comparison of a matched filter to the optimal filter in a simple example scenario with $K = 6$ users.



In Fig. 2.2 the achievable uplink rates of a matched filter are compared to the optimal filter. As described above, the lower scaling of the SINR when using a matched filter leads to a constant gap in achievable rate. For a large number of antennas, the relative difference vanishes, but the relative amount of antennas that has to be added to achieve identical performance with the MF stays constant. For this example, we need to increase the number of antennas about 50% to achieve the same performance with the matched filter as with the optimal one.

Another point is that for strong variation in the channel gains of the different users, we need significantly more antennas to reach the asymptotic regime. The large-scale fading coefficients β_k can vary by several orders of magnitude depending on the position of the user in the cell. This issue is often referred to as the near-far-effect. If β_n/β_k is of the order of 30 dB, we need thousands of base station antennas to reach the asymptotic regime. Thus, even for a large number of antennas at the base station, say in the hundreds, there might still be a benefit of proper power allocation to the different users.

A popular heuristic for power allocation in setups with a lot of antennas is to simply choose the power coefficient inversely proportional to the large-scale fading β_k such that

$$p_k \beta_k = \min_n \beta_n = \bar{\beta} \ \ \forall k. \tag{2.70}$$

For this allocation the lower bound on the SINR simplifies to

$$\gamma_k^{\mathrm{MF}} = M \frac{\bar{\beta}}{1/\rho_{\mathrm{ul}} + K\bar{\beta}} = M \frac{1}{\frac{1}{\rho_{\mathrm{ul}}\bar{\beta}} + K}. \tag{2.71}$$

Clearly, if the effective SNR, $\rho_{\mathrm{ul}}\bar{\beta}$, is much smaller than one, this approach might not offer any benefits. When possible, it is preferable to serve users simultaneously if they have similar channel gains.

In summary, saying that the matched filter is asymptotically optimal is not wrong, but it is also not really helpful for practical considerations.

2.4 Downlink Precoder Design with Uplink-Downlink Duality

For the precoder design in the downlink we focus on a single-cell scenario. Same as for the uplink power allocation problem, the extension to a multi-cell scenario is possible but not straightforward. Since we need several power constraints for a multi-cell setup, the simple SINR duality introduced in Appendix C.1, which can only handle a single sum-power constraint, does not apply. Results based on Lagrange-duality can be used to deal with multiple constraints [16, 17]. A detailed discussion of the multi-cell case is out of scope for this work.

The expressions for instantaneous and ergodic rates in the downlink are similar to the expressions in the uplink. We use linear precoding to design the transmit signal

$$\boldsymbol{x} = \sum_k \boldsymbol{t}_k^* s_k \tag{2.72}$$

with the data symbols $s_k \sim \mathcal{N}_{\mathbb{C}}(0,1)$ and the beamforming vectors $\boldsymbol{t}_k$. With the linear precoding in (2.72), user k receives the signal

$$y_k^{\mathrm{dl}} = \boldsymbol{h}_k^{\mathrm{T}} \boldsymbol{t}_k^* s_k + \sum_{n \neq k} \boldsymbol{h}_k^{\mathrm{T}} \boldsymbol{t}_n^* s_n + v_k^{\mathrm{dl}}. \tag{2.73}$$

The achievable rate for constant channels $\boldsymbol{h}_k$ is thus given by

$$r_k^{\mathrm{dl}} = \log_2(1 + \gamma_k^{\mathrm{dl}}) \tag{2.74}$$

where

$$\gamma_k^{\mathrm{dl}} = \frac{\left|\boldsymbol{h}_k^{\mathrm{H}} \boldsymbol{t}_k\right|^2}{1 + \sum_{n \neq k} \left|\boldsymbol{h}_k^{\mathrm{H}} \boldsymbol{t}_k\right|^2} \tag{2.75}$$

Remember that we assume that the beamforming vectors $\boldsymbol{t}_k$ are normalized such that the noise has unit variance.

Analogously to the uplink case, ρ_{dl} denotes the ratio of the maximum total transmit power at the base station to the noise power at receiver of the mobile user, which we assume is the same for all users.

is the downlink SINR of user k. At the base station we typically have a sum-power constraint

$$\mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{x}] = \sum_k \boldsymbol{t}_k^{\mathrm{H}}\boldsymbol{t}_k \le \rho_{\text{dl}} \tag{2.76}$$

leading to the non-convex rate region

$$\mathcal{R}^{\text{dl}} = \{\boldsymbol{r}^{\text{dl}} : \sum_k \boldsymbol{t}_k^{\mathrm{H}}\boldsymbol{t}_k \le \rho_{\text{dl}}\}. \tag{2.77}$$

To solve a NUM maximization in the downlink we can optimize with respect to the beamforming vectors $\boldsymbol{t}_k$ by applying a projected gradient algorithm. Compared to the uplink we have MK optimization variables instead of the K uplink power coefficients.

Fortunately, there is a connection between the feasible uplink and downlink SINRs [18, 19, 20]. In fact, for our system model we can express the downlink rate region in terms of uplink rates. That is,

$$\mathcal{R}^{\text{dl}} = \{\boldsymbol{r}^{\text{ul}} : \mathbf{1}^{\mathrm{T}}\boldsymbol{p} \le \rho_{\text{dl}}, \boldsymbol{p} \ge \mathbf{0}\} \tag{2.78}$$

where the difference to the uplink rate region in (2.23) is that we replaced the per-user constraints by a sum-power constraint and the uplink noise covariance is set to $\boldsymbol{C}_{\boldsymbol{v}} = \mathbf{I}$. The beamforming vectors that achieve the dual uplink rates can be expressed in terms of the optimal uplink filters (cf. (2.9))

$$\boldsymbol{t}_k = \sqrt{q_k}\boldsymbol{g}_k^{\star}(\boldsymbol{p}). \tag{2.79}$$

The downlink power allocation q_k is determined by equating

$$\gamma_k^{\text{dl}} = \gamma_k^{\star} \; \forall k \tag{2.80}$$

and solving the resulting linear system of equations. For a detailed derivation of the uplink-downlink duality, we refer to Appendix C.1.

As a result of the uplink-downlink duality, the analysis of the downlink rates is analogous to the analysis of the uplink rates. Again, the only difference is the constraint set with the sum-power constraint $\mathbf{1}^{\mathrm{T}}\boldsymbol{p} \le \rho_{\text{dl}}$ for the downlink rates instead of the component-wise constraints $\boldsymbol{p} \le \rho_{\text{ul}}\mathbf{1}$. We remember the asymptotically equivalent SINR (now with $\boldsymbol{C}_{\boldsymbol{v}} = \mathbf{I}$)

$$\gamma_k^{\text{Asy}} = p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}). \tag{2.81}$$

For the uplink it is clearly optimal to choose $\boldsymbol{p} = \rho_{\text{ul}}\mathbf{1}$ for any NUM problem based on the asymptotic rates, i.e., to let each user transmit

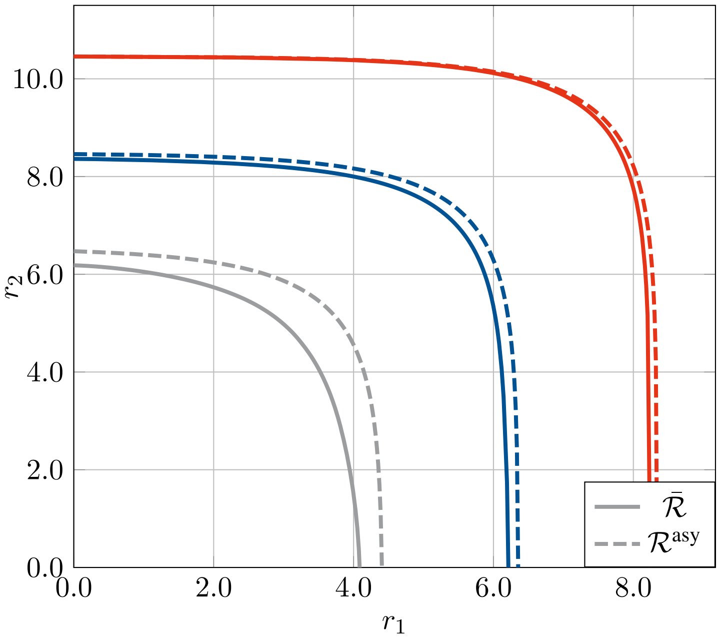


Figure 2.3: Achievable ergodic rate regions in the downlink for a base-station with $M = 4, 16, 64$ antennas that serves $K = 2$ users. We also depict the approximate rate regions that result from replacing the SINRs with the asymptotically equivalent expressions. That is, we replace the dual uplink SINR γ_k^{ul} with γ_k^{asy} to calculate the achievable rate of user k.

with maximal power. For the downlink, the asymptotic rates $r_k^{\text{asy}} = \log_2(1 + \gamma_k^{\text{asy}})$ are still coupled due to the sum-power constraint, but, since the asymptotic SINRs are linear in the power coefficients, NUM maximization based on the asymptotic rates is a convex problem as long as we have a concave utility function. That is, we can solve the NUM problem

This holds as long as the utility function is non-decreasing in the individual achievable rates, which is typically the case.

$$\boldsymbol{p}^\star = \underset{\boldsymbol{p}\geq 0, \mathbf{1}^{\mathrm{T}}\boldsymbol{p}=\rho_{\text{dl}}}{\text{argmax}} \quad U(\boldsymbol{r}^{\text{asy}}(\boldsymbol{p})) \tag{2.82}$$

globally optimal with a projected gradient method if the utility function is concave.

Fig. 2.3 shows ergodic downlink rate regions and the corresponding region of asymptotically equivalent rates. Due to the uplink-downlink duality, NUM problems can be treated similarly for uplink and downlink scenarios. However, due to the different nature of the power constraints, the resulting rate regions differ significantly.

2.5 Summary

We saw that, for linear, per-user processing in the uplink with perfect CSI, we get achievable data rates of

$$r_k^{\mathrm{ul}} = \log_2(1 + \gamma_k^{\mathrm{ul}}) \tag{2.83}$$

with SINRs

$$\gamma_k^{\mathrm{ul}} = \frac{p_k \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k \boldsymbol{h}_k^{\mathrm{H}} \boldsymbol{g}_k}{\boldsymbol{g}_k^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n \neq k} p_n \boldsymbol{h}_n \boldsymbol{h}_n^{\mathrm{H}} \right) \boldsymbol{g}_k} \tag{2.84}$$

where the $\boldsymbol{h}_k$ are the channel vectors, the $\boldsymbol{g}_k$ the linear receive filters, and the p_k the transmit powers of the different users.

The optimal filters are given by

$$\boldsymbol{g}_k^{\star} = \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n \neq k} p_n \boldsymbol{h}_n \boldsymbol{h}_n^{\mathrm{H}} \right)^{-1} \boldsymbol{h}_k \tag{2.85}$$

and lead to the optimal SINRs

$$\gamma_k^{\star} = p_k \boldsymbol{h}_k^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_{n \neq k} p_n \boldsymbol{h}_n \boldsymbol{h}_n^{\mathrm{H}} \right)^{-1} \boldsymbol{h}_k. \tag{2.86}$$

Since we have a fast-fading channel, we were interested in the ergodic rates

$$\bar{r}_k = \mathrm{E}[r_k] = \mathrm{E}[\log_2(1 + \gamma_k^{\star})]. \tag{2.87}$$

We assumed that the power coefficients $\boldsymbol{p}$ are deterministic to reduce the complexity of the resource allocation. Each user has an individual power constraint. For notational convenience we assumed it is the same for all users, i.e., we have $\boldsymbol{p} \leq \rho_{\mathrm{ul}} \mathbf{1}$. Thus the ergodic rate region, i.e., the region of all achievable ergodic rates in the uplink, is given by

$$\bar{\mathcal{R}} = \{ \mathrm{E}[\boldsymbol{r}(\boldsymbol{p})] : \mathbf{0} \leq \boldsymbol{p} \leq \rho_{\mathrm{ul}} \mathbf{1} \}. \tag{2.88}$$

Optimal power control maximizes a network utility function. That is, we solve the NUM problem

$$\max_{\bar{\boldsymbol{r}} \in \bar{\mathcal{R}}} U(\bar{\boldsymbol{r}}). \tag{2.89}$$

Due to the expectation in (2.87) this problem is not easy to solve.

Instead, we introduced the deterministic SINRs

$$\gamma_k^{\text{asy}} = p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{C}_{\boldsymbol{v}}^{-1}) \tag{2.90}$$

which are asymptotically equivalent to $\gamma_k^\star$. Since γ_k^{asy} is deterministic, we were able to simplify the NUM maximization significantly by replacing the ergodic rates $\bar{r}_k$ with the asymptotic rates $r_k^{\text{asy}} = \log_2(1+\gamma_k^{\text{asy}})$. Due to the asymptotic equivalence, the resulting power allocation is close to optimal for systems with many antennas. In fact, since γ_k^{asy} only depends on p_k, the optimal solution for large numbers of antennas is that all users transmit with full power.

For linear beamforming in the downlink, there is no analytical solution for the beamforming vectors comparable to the solution we have for the optimal uplink filters in (2.85). Thus, we have M times more optimization variables in the NUM problem. However, thanks to uplink-downlink duality, we could formulate an equivalent dual uplink problem were we then only have the dual uplink powers as optimization variables. The difference compared to the NUM for the actual uplink is that we have a sum-power constraint $\mathbf{1}^{\mathrm{T}}\boldsymbol{p} \leq \rho_{\text{dl}}$ in the dual uplink. Thus we find the optimal dual uplink power allocation by solving

$$\boldsymbol{p}^\star = \underset{\boldsymbol{p}\geq 0, \mathbf{1}^{\mathrm{T}}\boldsymbol{p}=\rho_{\text{dl}}}{\operatorname{argmax}} \; U(\boldsymbol{r}^{\text{asy}}(\boldsymbol{p})) \tag{2.91}$$

with a projected gradient method. If the utility function U is concave, we find the globally optimal solution.

The uplink-downlink duality allowed us to calculate optimal downlink beamforming vectors from the optimal uplink power allocation $\boldsymbol{p}^\star$ and the corresponding optimal uplink filters $\boldsymbol{g}_k^\star$.

Chapter 3

Transceiver Design with Imperfect CSI

For imperfect CSI matters get even more exciting. We will shortly see an example with “naive” signal processing to demonstrate that it is important to approach the issue of imperfect CSI in a principled manner. One major challenge is that there is no closed-form expression for the mutual information if the receiver has imperfect knowledge of the channel. It is thus common practice to use a lower bound instead of the actual mutual information to optimize the system. We use the bound introduced in [14, 15], which lets us incorporate the imperfect CSI in a straightforward manner into our objective function.

For this we need a model for the available knowledge, i.e., a stochastic model for the distribution of the channel given the information that is available at the receiver. We will see that, if we apply the lower bound from [14, 15] to our system model, the achievable rate only depends on the first and second order moment of this conditional distribution.

The conditional distribution is usually expressed in terms of a prior distribution of the channels and a joint distribution of the channels with the available side-information. We assumed Gaussian distributed channel vectors for some asymptotic results in the previous chapter. We will use the same assumption in this chapter.

We will discuss scenarios with uplink and downlink training, where the conditional distribution is Gaussian, but also semi-blind estimation where we have to deal with a non-Gaussian conditional distribution. For this non-Gaussian distribution it is more challenging to acquire the first and second order moments that appear in the mutual information lower bound and are required for the transceiver design. We will see that a variational inference approach can be used to get good approximations.

As mentioned before, computational complexity is a major concern when we develop algorithms for large-scale wireless systems. For practically relevant antenna-array geometries at the base station, the channel covariance matrices have structure which can be exploited to reduce the complexity of the transceiver designs we discuss in this chapter.

We can still use uplink-downlink duality for downlink precoder design, but the resulting rate expressions may actually no longer yield achievable rates due to side-information asymmetry between transmitter and receiver, as we will discuss later on. In other words, precoder design based on uplink-downlink duality is a heuristic method in this scenario and does not yield guarantees on the achievable rates.

3.1 Extensions to the System Model

We assume block-fading channels. That is, the channel is assumed constant for a coherence interval of T channel accesses. The channels in different coherence intervals are assumed to be independent. However, the covariance matrices are static over several channel coherence intervals. For now we assume that the channel covariance matrices are known. Imperfect knowledge of the channel statistics is discussed in the next chapter.

The number of channel accesses T depends on the mobility of the user and the physical environment. In a typical LTE setting T ranges from tens to hundreds of channel accesses.

Out of the coherence interval of length T, T_{tr} channel accesses are used for training, the remaining channel accesses are for data transmission. For the data transmission we have the same model as before, i.e., in the uplink

$$\boldsymbol{y}^{\text{ul}} = \sum_k \sqrt{p_k}\boldsymbol{h}_k s_k + \boldsymbol{v}^{\text{ul}} \quad \in \mathbb{C}^M \tag{3.1}$$

and in the downlink

$$y_k^{\text{dl}} = \sum_n \boldsymbol{h}_k^{\text{T}} \boldsymbol{t}_n^* s_n + v_k^{\text{dl}}. \tag{3.2}$$

However, the Gaussian-distributed channel vectors $\boldsymbol{h}_k \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{C}_{\boldsymbol{h}_k})$ are unknown.

Remember that we are interested in asymmetric large-scale systems, i.e., the number of antennas M is significantly larger than the number of simultaneously served users K. Thus, time-division duplexing (TDD), which allows us to exploit the reciprocity of the channel, has an advantage over frequency-division duplexing (FDD) in terms of necessary training and feedback resources. To estimate the channel in a TDD system, we ideally send K orthogonal pilot sequences from the different users. The resulting estimate is then used to design both, the uplink filters and the downlink precoders. In an FDD system we have to resort to downlink pilots and feedback to be able to design suitable precoders at the base station. We will discuss both approaches in this chapter.

3.2 Training as Dimensionality Bottleneck

To illustrate the fundamental difference between perfect and imperfect CSI, we first consider the case of matched filters, which was discussed in Section 2.3.2 for perfect CSI. We consider a TDD system with uplink training and assume that each user transmits one of T_{tr} predefined orthogonal pilot sequences. After correlating with the pilot sequence at the base station, we get T_{tr} observations of the form

$$\varphi_p = \sum_{k \in \Omega_p} \boldsymbol{h}_k + \frac{1}{\sqrt{\rho_{\mathrm{tr}}}} \boldsymbol{v}^{\mathrm{tr}} \tag{3.3}$$

where Ω_p is the set of users that transmit pilot sequence p, with $p = 1$, …, T_{tr} and we have additive noise $\boldsymbol{v}^{\mathrm{tr}} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{I})$.

We do not only model noise in the observation, but also distortions due to interference. In the uplink, it does not matter whether the interferers in Ω_p are from the same cell as user k or whether the interference originates from neighboring cells. The effect on uplink performance is the same. In a large network with many users, this interference during the training phase, so called pilot-contamination, is difficult to avoid due to scarcity of training resources.

Now, we directly use the observation corresponding to the pilot sequence transmitted by a user to filter the received signals. That is, we set $\boldsymbol{g}_k = \varphi_p$ if user k transmitted pilot sequence p.

The observation φ_p is the least-squares estimate of the channel vector $\boldsymbol{h}_k$ and the matched filter based on the least-squares estimate is a common choice when complexity is a concern.

If we assume that none of the observations are interference free, we get the following result for the asymptotic rates.

Theorem 3.1. *Suppose Conditions 2.1 and 2.2 are fulfilled (the noise is covariance matrix is well-conditioned and the channel power grows linearly with M) and we have the observations in* (3.3)*, with* $|\Omega_p| \geq$

2 $\forall p$. *An upper bound on the asymptotic SINR with the matched filter* $\boldsymbol{g}_k = \boldsymbol{\varphi}_p$ *for a user* k *that transmits pilot sequence* p *is given by*

$$\gamma_k^{asy} = \frac{p_k \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_k})^2}{\sum_{n\in\Omega_p\setminus\{k\}} p_n \operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_n})^2}. \tag{3.4}$$

Proof. We use the uplink SINR definition from (2.6) in Chapter 2. Since we do not actually have perfect CSI at the receiver, this SINR leads to an upper bound on the achievable rate. Incorporating the receive filters $\boldsymbol{g}_k = \boldsymbol{\varphi}_p$ into the SINR for perfect CSI leads to

$$\gamma_k = \frac{p_k \left|\boldsymbol{\varphi}_p^{\mathrm{H}} \boldsymbol{h}_k\right|^2}{\boldsymbol{\varphi}_p^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}^{\mathrm{ul}}} \boldsymbol{\varphi}_p + \sum_{n\neq k} p_n \left|\boldsymbol{\varphi}_p^{\mathrm{H}} \boldsymbol{h}_n\right|^2}. \tag{3.5}$$

Using Lemma C.3 we get

$$\lim_{M\to\infty} \frac{\boldsymbol{\varphi}_p^{\mathrm{H}} \boldsymbol{h}_n}{M} - \frac{\operatorname{tr}(\boldsymbol{C}_{\boldsymbol{h}_n})}{M} = 0 \tag{3.6}$$

if user n transmitted pilot sequence p. On the other hand, if user n transmitted a pilot sequence different from pilot sequence p we get

$$\lim_{M\to\infty} \frac{\boldsymbol{\varphi}_p^{\mathrm{H}} \boldsymbol{h}_n}{M} = 0. \tag{3.7}$$

Extending numerator and denominator of (3.5) with $1/M^2$, we can replace both with their asymptotically equivalent expressions. This works since we know that the denominator is non-zero in the limit due to Condition 2.2. The noise part vanishes and we end up with $\gamma_k^{\mathrm{asy}} \asymp \gamma_k$. □

The important thing to note here is that the SINRs do not grow linearly with M, in contrast to the SINRs for perfect CSI (cf. Corollary 2.3). Instead, if the observation for a user k is subject to interference, the SINR saturates to a deterministic value, i.e., we get zero DoMs for users that suffer from pilot contamination. The SINR of users that do not suffer pilot-contamination, increases linearly, i.e., we get one DoM.

The proof for the linear growth without pilot-contamination is almost identical to the one in Corollary 2.3 and is omitted here.

This pilot-contamination problem prompted a lot of research in recent years. The impact of pilot-contamination can, e.g., be reduced by a smart allocation of pilot sequences to the mobile users. One simple approach is to divide the available pilot sequences in G groups and employ a reuse pattern for the pilot sequences as depicted in Fig. 3.1

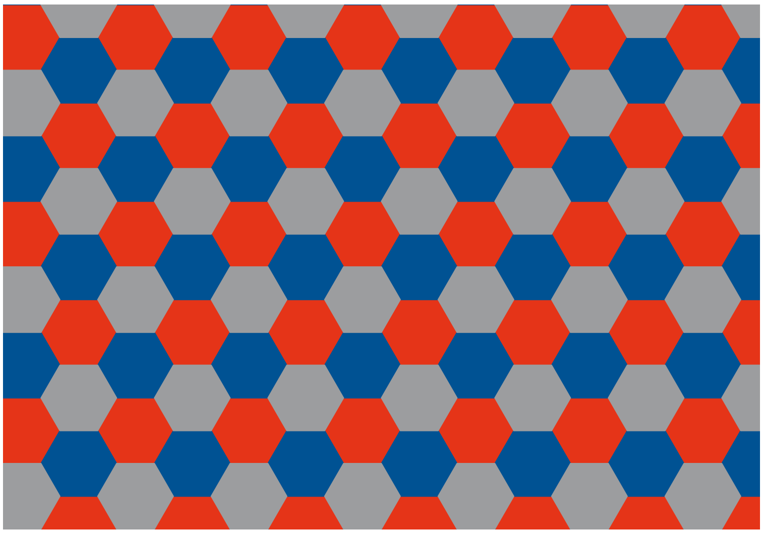


Figure 3.1: Depiction of a factor three reuse pattern in a cellular network of regular hexagonal cells. Differently colored cells use different pilots. That is, the pilot sequences are split into three disjoint sets, one of which is used by each color.

for $G = 3$ (cf. [13]). This way, neighboring cells never use the same pilot sequences. However, the amount of training sequences per cell is reduced to T_{tr}/G and thus if we want to avoid pilot-contamination within the cell we are only able to serve $K = T_{\text{tr}}/G$ users simultaneously. On the other hand, the amount of training cannot exceed the coherence interval T and in practice should be significantly lower.

Other work exploits structure in the covariance matrices of the channel vectors in some way. Be it to suppress the interference during the training [21] or to design two-stage linear transceivers, where one stage only depends on the channel statistics [22, 23]. As we will see in the next section, if we use a reasonable objective function that captures our model for the imperfect CSI, the impact of pilot-contamination is already significantly reduced for commonly used channel models.

It is also possible to design a “robust” matched filter that can deal with pilot-contamination and exhibits the desired linear growth of the SINR with the number of antennas. Discussion of this robust matched filter approach for imperfect CSI follows in Section 3.8.

3.3 Exploiting Statistical Information

In practice, the channels to different users have different spatial structure, i.e., different channel covariance matrices. In the following we will derive a well-known lower bound on the achievable rates which depends on the channel statistics. Asymptotic analysis will show that under mild assumptions, we can exceed the DoM that are achievable by the simple matched filter discussed in the previous section.

Suppose that in the uplink, the side information φ on the channels is available at the base station (e.g., after training). This side information is used to design the filters $\boldsymbol{g}_k$. The signal model is the same as before

$$\hat{s}_k = \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{y}^{\mathrm{ul}} = \sqrt{p_k} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k s_k + \sum_{n \neq k} \sqrt{p_n} \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n s_n + \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{v}^{\mathrm{ul}}. \tag{3.8}$$

Due to the uncertainty in the channels $\boldsymbol{h}_k$ we can no longer find a closed-form expression for the mutual information $\mathcal{I}(s_k; \hat{s}_k | \varphi)$. However, using the same approach as in Theorem 2.3, we can derive a lower bound.

Theorem 3.2. *The conditional mutual information $\mathcal{I}(s_k; \hat{s}_k | \varphi)$ is lower bounded by*

$$r_k^{\varphi} = \log_2(1 + \gamma_k^{\varphi}) \tag{3.9}$$

with

$$\gamma_k^{\varphi} = \frac{p_k \left| \mathrm{E}_{|\varphi}[\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k] \right|^2}{\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}} \boldsymbol{g}_k + p_k \operatorname{var}_{|\varphi}(\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k) + \sum_{n \neq k} p_n \mathrm{E}_{|\varphi}[\left| \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_n \right|^2]} \tag{3.10}$$

which simplifies to

$$\gamma_k^{\varphi} = \frac{p_k \left| \boldsymbol{g}_k^{\mathrm{H}} \hat{\boldsymbol{h}}_k \right|^2}{\boldsymbol{g}_k^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{v}} + p_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \sum_{n \neq k} p_n (\boldsymbol{C}_{\boldsymbol{h}_n|\varphi} + \hat{\boldsymbol{h}}_n \hat{\boldsymbol{h}}_n^{\mathrm{H}}) \right) \boldsymbol{g}_k} \tag{3.11}$$

where $\boldsymbol{h}_k | \varphi \sim \mathcal{N}_{\mathbb{C}}(\hat{\boldsymbol{h}}_k, \boldsymbol{C}_{\boldsymbol{h}_k|\varphi})$.

Proof. The derivation of (3.10) is equivalent to the one for Theorem 2.3 and also makes use of the results in Appendix A. The only difference is that the expectations are now conditioned on the available observation φ.

Since the filters $\boldsymbol{g}_k$ are deterministic given the observation φ, we can express the expectations in terms of the conditional means

$$\hat{\boldsymbol{h}}_k = \mathrm{E}_{|\varphi}[\boldsymbol{h}_k] \tag{3.12}$$

which are the MMSE estimates of the channel, and the conditional covariance matrices

$$\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} = \mathrm{E}_{|\varphi}[\boldsymbol{h}_k \boldsymbol{h}_k^{\mathrm{H}}] - \hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\mathrm{H}} \tag{3.13}$$

which describe the estimation error. □

The difference to the uplink SINRs for perfect CSI in (2.6) is that the actual channels are replaced by the MMSE estimates and we have additional deterministic interference terms, which depend on the covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi}$ of the channel estimation errors and scale with the transmit powers p_k.

Analogously to the perfect CSI case, the optimal uplink filters (with respect to the lower bound on the achievable rate) are given by

$$\boldsymbol{g}_k^\star = \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_n p_k \boldsymbol{C}_{\boldsymbol{h}_n|\varphi} + \sum_{n\neq k} p_n \hat{\boldsymbol{h}}_n \hat{\boldsymbol{h}}_n^{\mathrm{H}} \right)^{-1} \hat{\boldsymbol{h}}_k \tag{3.14}$$

with resulting SINRs

$$\begin{aligned} \gamma_k^\star &= p_k \hat{\boldsymbol{h}}_k^{\mathrm{H}} \boldsymbol{g}_k^\star && (3.15) \\ &= p_k \hat{\boldsymbol{h}}_k^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{v}} + \sum_n p_n \boldsymbol{C}_{\boldsymbol{h}_n|\varphi} + \sum_{n\neq k} p_n \hat{\boldsymbol{h}}_n \hat{\boldsymbol{h}}_n^{\mathrm{H}} \right)^{-1} \hat{\boldsymbol{h}}_k. && (3.16) \end{aligned}$$

Since the rates r_k^φ are a lower bound on the instantaneous achievable rate for a given observation φ, we get the lower bound on the ergodic rates

$$\bar{r}_k^{\mathrm{ul}} = \mathrm{E}[r_k^\varphi]. \tag{3.17}$$

The discussion regarding NUM with respect to the ergodic rates is identical to the one for perfect CSI. We only have to replace “for each channel realization” by “for each observation φ”. Same as for perfect CSI, we prefer a static power allocation $\boldsymbol{p}$ that does not have to be recalculated for each observation.

Since the structure of the equivalent SINRs in (3.11) is similar to the SINRs for perfect CSI in (2.6), we might expect the rate regions to take on similar shapes in both cases. In fact, the shape of the rate region depends on the available observation φ. If we have independent observations for different users, the behavior is similar to the case of perfect CSI. However, if the observations are coupled, e.g., by interference during the training phase, the behavior is fundamentally different. We will see this shortly when we discuss the different types of channel observations encountered in practice.

To facilitate the asymptotic analysis we use the same kind of reformulation of the SINRs that we used for perfect CSI.

Lemma 3.1. *For a non-trivial power allocation $\boldsymbol{p} > 0$, the optimal SINR of user k is given by*

$$\gamma_k^\star = p_k \frac{\boldsymbol{e}_k^{\mathrm{T}} \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} \left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}}\right)^{-1} \boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}} \left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}}\right)^{-1} \boldsymbol{e}_k} \tag{3.18}$$

where $\boldsymbol{P} = \mathrm{diag}(\boldsymbol{p})$ and $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{h}}_1, \ldots, \hat{\boldsymbol{h}}_K]$ and

$$\boldsymbol{C}_{\boldsymbol{y}|\varphi} = \boldsymbol{C}_{\boldsymbol{v}} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi}. \tag{3.19}$$

Optimal filters can be calculated as

$$\tilde{\boldsymbol{g}}_k^\star = \frac{1}{\sqrt{p_k}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} \left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}}\right)^{-1} \boldsymbol{e}_k. \tag{3.20}$$

Proof. The proof is equivalent to the one for perfect CSI in Lemma 2.1. □

The filters in (3.20) not only maximize the SINR in (3.11) but also minimize the MSE $\mathrm{E}[|s_k - \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{y}|^2 | \varphi]$. That is, the filter in (3.20) is the linear minimum mean squared error (LMMSE) filter for the estimation of the transmit symbols s_k from the signals $\boldsymbol{y}$ given observations φ [24].

3.4 Uplink-downlink Duality

Basically, uplink-downlink duality works the same as for perfect CSI. The system model for the downlink is the same as before

$$y_k^{\mathrm{dl}} = \boldsymbol{h}_k^{\mathrm{T}} \boldsymbol{t}_k^* s_k + \sum_{n \neq k} \boldsymbol{h}_k^{\mathrm{T}} \boldsymbol{t}_n^* s_n + v_k^{\mathrm{dl}}. \tag{3.21}$$

In the downlink, if we have the observation φ at the receiver, we can derive a lower bound on the conditional mutual information.

Theorem 3.3. *The conditional mutual information $I(y_k; s_k|\varphi)$ is lower bounded by $r_k^{dl} = \log_2(1 + \gamma_k^{dl})$ with*

$$\gamma_k^\varphi = \frac{\left|\boldsymbol{t}_k^{\mathrm{H}} \hat{\boldsymbol{h}}_k\right|^2}{1 + \boldsymbol{t}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{h}_k|\varphi} \boldsymbol{t}_k + \sum_{n \neq k} \boldsymbol{t}_n^{\mathrm{H}} (\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\mathrm{H}}) \boldsymbol{t}_n} \tag{3.22}$$

Proof. The proof is analogously to the one in Theorem 3.2 based on the results in Appendix A. □

If we refer to the notes in Appendix C.1, we realize that the uplink SINR in (3.11) with $\boldsymbol{C}_{\boldsymbol{v}} = \mathbf{I}$ and the downlink SINR in (3.22) fit into our uplink-downlink duality framework. Thus, uplink-downlink duality can be applied the same way as in Chapter 2.

The issue with designing precoding vectors for the downlink is that the result in Theorem 3.3 assumes that the same observation φ is available at transmitter and receiver, which is not the case in general.

We will still use the SINR expression in (3.22) with the MMSE estimates at the base station to design the precoders. In this case, the rate expressions that we optimize are not lower bounds on the mutual information but just approximations. The accuracy of the approximation depends on the system design. We could, e.g., send additional downlink pilots along the precoding vectors or use a more sophisticated detector at the receivers [25] to make sure that the calculated rates are actually achievable.

In practice, we would change the channel code rate using an outer loop link adaptation at the base station based on decoding success and decoding failure reports from the mobile user. Thus, it is not necessary to know the achievable rate exactly.

3.5 Uplink Pilots

In this section we analyze the achievable rates with the SINRs in (3.15) when we have observations from uplink training. That is, we consider the same system model as in Section 3.2, where each user transmits one of T_{tr} predefined orthogonal pilot sequences. After correlating with the pilot sequence at the base station, we get T_{tr} observations of the form

$$\boldsymbol{\varphi}_p = \sum_{k\in\Omega_p} \boldsymbol{h}_k + \frac{1}{\rho_{\text{tr}}}\boldsymbol{v}^{\text{tr}} \tag{3.23}$$

where Ω_p is the set of users that transmits pilot sequence p, with $p = 1$, …, T_{tr} and $\boldsymbol{v}^{\text{tr}} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{I})$ is additive noise.

Because everything is Gaussian, the MMSE channel estimate of a user k that transmits pilot sequence p is given by

$$\hat{\boldsymbol{h}}_k = \mathrm{E}[\boldsymbol{h}_k\boldsymbol{\varphi}_k^{\mathrm{H}}]\,\mathrm{E}[\boldsymbol{\varphi}_k\boldsymbol{\varphi}_k^{\mathrm{H}}]\boldsymbol{\varphi}_k = \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}\boldsymbol{\varphi}_k \tag{3.24}$$

with the covariance matrix of the observations

$$\boldsymbol{C}_{\boldsymbol{\varphi}_p} = \frac{1}{\rho_{\text{tr}}}\mathbf{I} + \sum_{k\in\Omega_p} \boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.25}$$

The covariance matrices of the estimation errors are given by

$$\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} = \boldsymbol{C}_{\boldsymbol{h}_k} - \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{\varphi}_k}^{-1}\boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.26}$$

3.5.1 Asymptotic Analysis

Asymptotic analysis of this scenario for optimal receive filters has also been done in [9] for both the uplink and the downlink. We show the same result, namely that the SINRs grow linearly with the number of antennas, even in the presence of pilot-contamination. Our focus is on the asymptotically equivalent SINRs, since, analogously to Chapter 2, we want to use the deterministic asymptotic SINRs for power allocation. In the scenario with imperfect CSI the asymptotic results can also be used for other resource allocation tasks such as pilot-sequence allocation.

If we have observations from an uplink training phase as given in (3.23), we can perform the analysis for large numbers of antennas similarly to the analysis for perfect CSI. Analogously to the analysis in Chapter 2, the main idea is that the matrix $\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}/M$ in the optimal SINR $\gamma_k^\star$ in (3.15) converges to a deterministic matrix. Again we need the channel hardening condition (Condition 2.3) to show that the variance of the products $\hat{\boldsymbol{h}}_k^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{h}}_n/M$ vanishes.

Lemma 3.2. *Under Conditions 2.1 and 2.3, we have*

$$\lim_{M\to\infty} \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}/M - \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}/M] = 0. \tag{3.27}$$

Proof. We use Lemma C.2 to calculate the following variances for users k and n, which use pilots p and q respectively:

$$\mathrm{var}(\hat{\boldsymbol{h}}_k^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{h}}_n/M) = \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\boldsymbol{C}_{\boldsymbol{h}_n}\boldsymbol{C}_{\boldsymbol{\varphi}_q}^{-1}\boldsymbol{C}_{\boldsymbol{h}_n}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1})/M^2 \tag{3.28}$$

where $\boldsymbol{C}_{\boldsymbol{y}|\varphi}$ is defined in (3.19).

Since $\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}\boldsymbol{C}_{\boldsymbol{h}_k} \prec \boldsymbol{C}_{\boldsymbol{h}_k}$ and $\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \prec \boldsymbol{C}_{\boldsymbol{v}}^{-1} \prec \left\|\boldsymbol{C}_{\boldsymbol{v}}^{-1}\right\|_2 \mathbf{I}$ we get

$$\begin{aligned}\mathrm{var}(\hat{\boldsymbol{h}}_k^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{h}}_n/M) &\leq \left\|\boldsymbol{C}_{\boldsymbol{v}}^{-1}\right\|_2^2 \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{h}_n})/M^2 &(3.29)\\ &\leq \left\|\boldsymbol{C}_{\boldsymbol{v}}^{-1}\right\|_2^2 \frac{\|\boldsymbol{C}_{\boldsymbol{h}_k}\|_F}{M}\frac{\|\boldsymbol{C}_{\boldsymbol{h}_n}\|_F}{M} &(3.30)\end{aligned}$$

which goes to zero for $M \to \infty$ due to Conditions 2.1 and 2.3. □

We define $\boldsymbol{\Gamma} = \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}/M] \in \mathbb{C}^{K\times K}$ which has the elements

$$[\boldsymbol{\Gamma}]_{kn} = \begin{cases} \frac{1}{M}\,\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\boldsymbol{C}_{\boldsymbol{h}_n}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}) & \text{if both } k \text{ and } n \text{ use pilot } p,\\ 0 & \text{otherwise.}\end{cases} \tag{3.31}$$

Incorporating Lemma 3.2 into the optimal SINR we get

Theorem 3.4. *If additionally to Conditions 2.1 and 2.3 we have*

$$\limsup_{M\to\infty} \left\|\boldsymbol{\Gamma}^{-1}\right\|_2 < \infty \tag{3.32}$$

then $\gamma_k^\star \asymp \gamma_k^{asy}$ *where*

$$\gamma_k^{asy} = \frac{Mp_k}{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{\Gamma}^{-1}\boldsymbol{e}_k}. \tag{3.33}$$

Additionally,

$$\lim_{M\to\infty} \gamma_k^\star/M > 0. \tag{3.34}$$

Proof. The proof is almost identical to the one for Theorem 2.1. We can express the involved SINR expressions using the same function

$$\tilde{\gamma}_k(\boldsymbol{B},\alpha) = p_k\frac{\boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{B}(\alpha\boldsymbol{P}^{-1}+\boldsymbol{B})^{-1}\boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}}(\alpha\boldsymbol{P}^{-1}+\boldsymbol{B})^{-1}\boldsymbol{e}_k}. \tag{3.35}$$

With $\boldsymbol{B} = \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}\hat{\boldsymbol{H}}/M$ and $\widetilde{\boldsymbol{B}} = \boldsymbol{\Gamma}$ we have $\gamma_k^\star/M = \tilde{\gamma}_k(\boldsymbol{B}, 1/M)$ and $\gamma_k^{\mathrm{asy}}/M = \tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0)$.

We already know from Lemma 3.2 that $\boldsymbol{B} \asymp \widetilde{\boldsymbol{B}}$. Due to the condition that $\|\boldsymbol{\Gamma}^{-1}\|_2$ is asymptotically bounded

$$\lim_{M\to\infty} \tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0) > 0. \tag{3.36}$$

For the same reason, $\tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0)$ is well-behaved for large M and asymptotically equivalence $\tilde{\gamma}_k(\boldsymbol{B}, 1/M) \asymp \tilde{\gamma}_k(\widetilde{\boldsymbol{B}}, 0)$ follows from $\boldsymbol{B} \asymp \widetilde{\boldsymbol{B}}$ and $1/M \to 0$. Since $\gamma_k^{\mathrm{asy}}/M$ is non-zero in the limit, the same holds for $\gamma_k^\star/M$ due to the asymptotic equivalence. □

We still have to clarify under which conditions on the channel covariance matrices, we have the full-rank property of $\boldsymbol{\Gamma}$ in (3.32). To answer this question we take a closer look at $\boldsymbol{\Gamma}$. If the users' indices are arranged according to pilot association, $\boldsymbol{\Gamma} = \mathrm{blkdiag}(\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_{T_{\mathrm{tr}}})$ has a block-diagonal structure with one block $\boldsymbol{\Gamma}_p \in \mathbb{C}^{K_p\times K_p}$ for each pilot sequence. The number of users that transmit pilot sequence p is denoted as K_p.

We consider the set of users $\Omega_p = \{1, \ldots, K_p\}$ which use the pilot sequence p, where without loss of generality we assume that the users are indexed from 1 to K_p. We collect the vectorized covariance matrices of the users in

$$\boldsymbol{\Xi} = [\mathrm{vec}(\boldsymbol{C}_{\boldsymbol{h}_1}), \ldots, \mathrm{vec}(\boldsymbol{C}_{\boldsymbol{h}_{K_p}})] \tag{3.37}$$

We use $\mathrm{tr}(\boldsymbol{A}^{\mathrm{H}}\boldsymbol{B}) = \mathrm{vec}(\boldsymbol{A})^{\mathrm{H}}\mathrm{vec}(\boldsymbol{B})$ and $\mathrm{vec}(\boldsymbol{AXB}) = (\boldsymbol{B}^{\mathrm{T}} \otimes \boldsymbol{A})\mathrm{vec}(\boldsymbol{X})$.

to be able to express $\boldsymbol{\Gamma}_p$ in matrix notation

$$\boldsymbol{\Gamma}_p = \frac{1}{M}\boldsymbol{\Xi}^{\mathrm{H}}(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1})\boldsymbol{\Xi} \tag{3.38}$$

where $\otimes$ denotes the Kronecker product.

If we have bounded spectral norms of the covariance matrices

Condition 3.1.

$$\limsup_{M\to\infty} \|\boldsymbol{C}_{\boldsymbol{h}_k}\| < \infty, \quad \forall k \tag{3.39}$$

and asymptotically linearly independent covariance matrices

Condition 3.2.

$$\liminf_{M\to\infty} \min_{\boldsymbol{\lambda}:\|\boldsymbol{\lambda}\|_2=1} \frac{1}{M}\left\| \sum_{k\in\Omega_p} \lambda_k \boldsymbol{C}_k \right\|_F^2 > 0 \tag{3.40}$$

or equivalently

$$\limsup_{M\to\infty} \|(\boldsymbol{\Xi}^{\mathrm{H}}\boldsymbol{\Xi}/M)^{-1}\|_2 < \infty \tag{3.41}$$

we get

Theorem 3.5. *If the covariance matrices fulfill Conditions 3.2 and 3.1, we have* $\limsup \|\boldsymbol{\Gamma}^{-1}\| < \infty$.

Proof. Since the covariance matrices have bounded spectral norm by Condition 3.1, we have $\limsup_M \|\boldsymbol{C}_{\boldsymbol{y}|\varphi}\|_2 < \infty$ and $\limsup_M \|\boldsymbol{C}_{\boldsymbol{\varphi}_p}\|_2 < \infty$ and thus

For the spectral norm of a Kronecker product we have $\|A \otimes B\| = \|A\|\|B\|$.

$$\limsup_{M} \|\boldsymbol{\Gamma}^{-1}\|_2 \leq \limsup_{M} \|\boldsymbol{C}_{\boldsymbol{y}|\varphi}\|_2 \|\boldsymbol{C}_{\boldsymbol{\varphi}_p}\|_2 \|(\boldsymbol{\Xi}^{\mathrm{H}}\boldsymbol{\Xi}/M)^{-1}\|_2 < \infty \tag{3.42}$$

due to Condition 3.2. □

Consequently, if all covariance matrices are asymptotically linearly independent, the SINRs grow linearly with the number of antennas even in the presence of pilot-contamination. We could thus train all users with a single channel access – using the remaining $T-1$ channel accesses for data transmission – and achieve #DoM $= K(T-1)/T \approx K$. That is, compared to the matched filter example in Section 3.2, the DoM are not limited by the number of available orthogonal training sequences but grow with the number of users K.

In practice, the covariance matrices of different users are linearly independent as long as the users are at different positions in the cell. However, as mentioned earlier, the scaling factor of the asymptotic SINR has a significant impact on the number of antennas needed for a certain prformance. Thus, for a finite number of antennas, it is still preferable to use several channel accesses for training to avoid unfavorable situations where users with similar covariance matrix structure use the same pilot sequence. This opens the door to various resource allocation methods that assign pilot sequences to users.

As we already noted in the last chapter, Condition 3.1 is too restrictive for many practical channel models. Fortunately, the bounded norm is not necessary. In Appendix B we discuss the conditions that are required for some relevant channel models.

3.5.2 Resource Allocation

For imperfect CSI, same as for perfect CSI, we want to solve a NUM problem with respect to (a lower bound of) the ergodic rates. Additionally to the (dual) uplink power allocation, we can optimize over the allocation of pilot sequences to users. Since it is not possible to change the pilot-sequence allocation for a given observation after the fact, the allocation should be quasi-static. That is, the pilot sequence allocation should change in the same time frame as the channel statistics which for our purposes means the allocation is static.

Working with the lower bound on the ergodic rate in (3.17) to optimize the pilot allocation is difficult due to the expectation over the observations. A less complex approach is to use the asymptotic SINRs in (3.33) which do not actually depend on the instantaneous observations but only on the channel statistics. That is, we solve

$$\max_{\Omega_p} \max_{\boldsymbol{p}} U(\boldsymbol{r}^{\text{asy}}). \tag{3.43}$$

Clearly, the outer maximization with respect to the pilot sequence allocation is a combinatorial optimization problem. In practice, greedy approaches are often used to find suboptimal solutions to combinatorial problems [26].

3.5.3 Computational Complexity

When discussing the computational complexity of the introduced signal processing approaches, it is important to consider the time frame in which the calculations have to be done. Since the channel statistics vary slowly compared to the fast-fading channel coefficients, we will in the following only consider the complexity of operations that rely on the instantaneous observation φ of the channel vectors. This excludes for example the allocation of pilot sequences discussed in the previous section.

For given beamforming vectors $\boldsymbol{t}_k$ or receive filters $\boldsymbol{g}_k$ the required computational complexity in processing the transmit and receive signals, respectively, is the same, since we rely on linear processing. For one channel access we need one matrix-vector multiplication which requires $\mathrm{O}(MK)$ floating-point operations. If the same filters and beamforming vectors are used for T_{data} channel accesses in one coherence interval we need $\mathrm{O}(MKT_{\text{data}})$ operations in total.

The filters and beamforming vectors have to be calculated once per coherence interval. In terms of computational complexity, the LMMSE filters in (3.20) are preferable over the optimal filters in (3.14), due to smaller dimensionality of the matrix that needs to be inverted.

The calculation of all LMMSE filters $\tilde{\boldsymbol{g}}_k^\star$ requires $\mathrm{O}(M^2K + MK^2 + K^3)$ complex operations. This assumes that the deterministic covariance matrices $\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}$ are pre-calculated. The calculations necessary for the uplink-downlink transformation are $\mathrm{O}(K^3)$ and thus do not affect the total order of complexity.

Overall, the computational complexities for the filter calculations might be too high for practical application in a massive MIMO system. In contrast to the complexity for the application of the filters, which is linear in M, the calculation of the filters is at least quadratic in M.

The issue with respect to the computational complexity stems from the $M \times M$ covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k}$. To reduce the complexity, we want to find a lower-dimensional parameterization of the covariance matrices. Incidentally, such a parameterization is also beneficial for the covariance matrix estimation discussed in Chapter 4.

The following assumption on the covariance matrices will be exploited throughout this work to reduce the computational complexity of the introduced methods.

Assumption 3.1. *The covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k}$ are diagonal, i.e.,*

$$\boldsymbol{C}_{\boldsymbol{h}_k} = \operatorname{diag}(\boldsymbol{c}_{\boldsymbol{h}_k}) \tag{3.44}$$

or equivalently, they can be decomposed as

$$\boldsymbol{C}_{\boldsymbol{h}_k} = \boldsymbol{Q} \operatorname{diag}(\boldsymbol{c}_{\boldsymbol{h}_k}) \boldsymbol{Q}^{\mathrm{H}} \tag{3.45}$$

with a common unitary matrix $\boldsymbol{Q} \in \mathbb{C}^{M \times M}$, for which the matrix product $\boldsymbol{Q}\boldsymbol{x}$ can be calculated in $O(M \log M)$ floating-point operations.

If the all covariance matrices have the same eigenbasis $\boldsymbol{Q}$ we can transform all incoming signals in the uplink once by $\boldsymbol{Q}^{\mathrm{H}}$. That is, if we receive $\boldsymbol{y}^{ul}$, we use $\widetilde{\boldsymbol{y}}^{ul} = \boldsymbol{Q}^{\mathrm{H}}\boldsymbol{y}^{ul}$ instead and then only work with the diagonal matrices $\operatorname{diag}(\boldsymbol{c}_{\boldsymbol{h}_k})$ that contain the eigenvalues.

Examples. The following examples of array geometries are discussed in more detail in Appendix B. For a ULA, the channel covariance matrices, which have Toeplitz structure, are asymptotically equivalent to corresponding circulant matrices [27]. That is, if $\boldsymbol{F}$ denotes the unitary DFT matrix, we have the asymptotic equivalence

$$\boldsymbol{C}_{\boldsymbol{h}_k} \asymp \boldsymbol{F}^{\mathrm{H}} \operatorname{diag}(\boldsymbol{c}_{\boldsymbol{h}_k}) \boldsymbol{F} \ \ \forall k \tag{3.46}$$

where $\boldsymbol{c}_{\boldsymbol{h}_k}$ contains the diagonal elements of $\boldsymbol{F}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{F}^{\mathrm{H}}$.

An analogous result can be derived for uniform rectangular arrays. In this case the transformation $\boldsymbol{Q}^{\mathrm{H}} = \boldsymbol{F}^{\mathrm{T}} \otimes \boldsymbol{F}$ is the Kronecker product of two DFT matrices. The sizes of the DFT matrices correspond to the number of antennas in both directions of the array.

A third example with a decomposition as in Assumption 3.1 are distributed antennas [28, 29]. For distributed antennas, the covariance matrices are typically modelled as diagonal matrices. That is, for distributed antennas we simply have $\boldsymbol{Q} = \mathbf{I}$.

If we use Assumption 3.1, namely that the covariance matrices are diagonal (or jointly diagonalizable) the complexity reduces significantly. For the examples following Assumption 3.1, the transformation into array space needs at most $\mathrm{O}(M \log(M))$ operations. Calculating the LMMSE filter only needs $\mathrm{O}(MK^2 + K^3)$ operations, which is linear in the number of antennas. Thus, per channel coherence interval we need in total $\mathrm{O}(MKT_{\text{data}} + M\log(M)T_{\text{data}} + MK^2 + K^3)$ operations.

By transformation to array space we mean the multiplication with $\boldsymbol{Q}$. The columns of $\boldsymbol{Q}$ can be seen as the “natural” basis for the given array geometry.

3.6 Downlink Pilots

In an FDD system we cannot exploit channel reciprocity and thus have to use downlink training and feedback to get side information on the channel state. To reduce the training and feedback overhead, the number of channel accesses used for training should be significantly smaller than the number of antennas. Authors of earlier work realized that low-rank structure of the channel covariance matrices can be exploited to reduce the amount of training without sacrificing performance. These considerations lead to multi-stage precoder design [23, 22]. With the insights from Section 3.3, we already know how to design the precoders based on the LMMSE filter. The only remaining issue is how to design the downlink pilot signals.

In Section 3.5 we saw that for uplink training, the asymptotic SINRs are useful to optimize the power allocation. Similarly, we would like to use asymptotic SINRs that only depend on the channel statistics to design the downlink pilot signals. However, as we will see in the following, if we fix the amount of training we do not get the linear scaling of the SINRs that we could show with uplink training and we also do not get a deterministic asymptotically equivalent SINR.

Suppose the base station transmits downlink pilot vectors $\boldsymbol{b}_p$, $p =$ 1, …, T_{tr}. If we assume perfect feedback, we get from user k the observation

$$\boldsymbol{\varphi}_k = \boldsymbol{B}^{\text{H}}\boldsymbol{h}_k + \boldsymbol{v}_k \tag{3.47}$$

where $\boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_{T_{\text{tr}}}] \in \mathbb{C}^{M \times T_{\text{tr}}}$ and $\boldsymbol{v}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{I})$. The pilot-matrix has to fulfill at least the average sum-power constraint $\text{tr}(\boldsymbol{B}^{\text{H}}\boldsymbol{B}) \leq \rho_{\text{tr}}$ with $\rho_{\text{tr}} = T_{\text{tr}}\rho_{\text{dl}}$.

We denote those observations collectively by $\varphi^{\text{dl}} = \{\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_K\}$. We calculate the MMSE estimates

$$\hat{\boldsymbol{h}}_k = \text{E}[\boldsymbol{h}_k|\varphi^{\text{dl}}] = \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{B}\left(\boldsymbol{B}^{\text{H}}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{B} + \mathbf{I}\right)^{-1}\boldsymbol{\varphi}_k \tag{3.48}$$

and conditional covariance matrices

$$\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{dl}}} = \boldsymbol{C}_{\boldsymbol{h}_k} - \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{B}\left(\boldsymbol{B}^{\text{H}}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{B} + \mathbf{I}\right)^{-1}\boldsymbol{B}^{\text{H}}\boldsymbol{C}_{\boldsymbol{h}_k} \tag{3.49}$$

that are required to evaluate the dual uplink SINR given in (3.18). With the conditions on the covariance matrices we have used previously, we can show that the SINR does not scale with M if T_{tr} is kept constant.

Theorem 3.6. *With Conditions 2.1 to 2.3 we get*

$$\lim_{M \to \infty} \gamma_k^{\star}/M = 0. \tag{3.50}$$

Proof. The optimal SINR $\gamma_k^\star$ is bounded above by the single-user SINR of user k

Clearly $\gamma_k^{\text{su}} \geq \gamma_k^\star$ since we remove the interference caused by all other users.

$$\gamma_k^{\text{su}} = \hat{\boldsymbol{h}}_k^{\text{H}} \left(\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \frac{1}{\rho_{\text{dl}}} \mathbf{I} \right)^{-1} \hat{\boldsymbol{h}}_k \tag{3.51}$$

that we get if we evaluate the SINR in (3.15) for a single user.

We have

$$\begin{aligned} \mathrm{E}[\gamma_k^{\text{su}}] &= \mathrm{E}\left[\hat{\boldsymbol{h}}_k^{\text{H}} \left(\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \frac{1}{\rho_{\text{dl}}} \mathbf{I} \right)^{-1} \hat{\boldsymbol{h}}_k \right] && (3.52) \\ &= \operatorname{tr}\left(\left(\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \frac{1}{\rho_{\text{dl}}} \mathbf{I} \right)^{-1} \mathrm{E}[\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\text{H}}] \right) && (3.53) \\ &\leq \rho_{\text{dl}} \operatorname{tr}\left(\mathrm{E}[\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\text{H}}] \right). && (3.54) \end{aligned}$$

We use the identity $\operatorname{tr}(\boldsymbol{AB}) = \operatorname{tr}(\boldsymbol{BA})$ and $(\boldsymbol{A} + \mathbf{I})^{-1} \preceq \mathbf{I}$.

We know that $\mathrm{E}[\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\text{H}}] \prec \boldsymbol{C}_{\boldsymbol{h}_k}$ and $\operatorname{rank}(\mathrm{E}[\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\text{H}}]) \leq T_{\text{tr}}$. Thus, $\operatorname{tr}(\mathrm{E}[\hat{\boldsymbol{h}}_k \hat{\boldsymbol{h}}_k^{\text{H}}]) \leq T_{\text{tr}} \|\boldsymbol{C}_{\boldsymbol{h}_k}\|$. Since $\lim_{M\to\infty} \|\boldsymbol{C}_{\boldsymbol{h}_k}\| / M = 0$ due to Condition 2.3, the bound on the SINR goes to zero for M going to infinity. □

The result follows intuitively from the channel-hardening condition, which requires that for large M the energy of the channel vectors is spread in many spatial directions. With a small, fixed T_{tr} our signal processing methods are limited to a small subspace and thus lose the array gain and asymptotic orthogonality that make massive MIMO so compelling. In the uplink, on the other hand, we saw that the SINR scales linearly with M irrespective of T_{tr}. For this reason, TDD operation is preferable if we want to scale the system to massive numbers.

If, for whatever reason, we still need to do downlink training, we can nevertheless achieve significant gains by increasing the number of antennas, as long as the covariance matrices adhere to certain low-rank assumptions. The first assumption that we will use in the following is Assumption 3.1, namely that all covariance matrices are diagonal ($\boldsymbol{C}_{\boldsymbol{h}_k} = \operatorname{diag}(\boldsymbol{c}_{\boldsymbol{h}_k})$).

Now suppose we have $T_{\text{tr}} = M$, i.e., we can scan the whole M-dimensional space with our training signals. Since we assume that all channel covariance matrices are diagonal, the only reasonable choice for the pilot covariance matrix and thus for $\boldsymbol{B}$ seems to be a diagonal matrix. Intuitively, this should be the optimal choice. We do not yet have a proof that a diagonal $\boldsymbol{B}$ is optimal for diagonal covariance matrices, but we will still use it as a working assumption.

After all, we are engineers not mathematicians.

Since a complex phase-shift of the pilot vectors does not affect the estimation, we choose a real-valued matrix without loss of generality. That is, we have $\boldsymbol{B} = \boldsymbol{D}^{1/2}$ where $\boldsymbol{D} = \text{diag}(\boldsymbol{d})$ and $\boldsymbol{d} \in \mathbb{R}^M$. The power constraint is then simply $\mathbf{1}^{\mathrm{T}}\boldsymbol{d} = \rho_{\text{tr}}$.

With this pilot matrix, we get the MMSE estimate (cf. (3.48))

$$\begin{aligned}\hat{\boldsymbol{h}}_k &= \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}^{1/2}\left(\boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k} + \mathbf{I}\right)^{-1}\boldsymbol{\varphi}_k && (3.55)\\ &= \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}^{1/2}\left(\boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k} + \mathbf{I}\right)^{-1}(\boldsymbol{D}^{1/2}\boldsymbol{h}_k + \boldsymbol{v}_k). && (3.56)\end{aligned}$$

Now we investigate for which assumptions we get an equivalent MMSE estimate but with a smaller amount of training. Consider the following low-rank assumption.

Assumption 3.2. *There exists a constant $S > 1$ such that for M antennas, the spectra $\boldsymbol{c}_{\boldsymbol{h}_k}$ are non-zero in at most M/S consecutive entries. For notational convenience, we additionally assume that both S and M/S are integers.*

This assumption holds (asymptotically), e.g., for the one-ring model, which has been used to justify various approaches to FDD signal processing with large numbers of antennas and users [23].

Theorem 3.7. *For a fixed total training power ρ_{tr}, and for covariance matrices following Assumption 3.2 and $T_{tr} = M/S$, the pilot design*

$$\boldsymbol{B} = \boldsymbol{D}^{1/2}\boldsymbol{S} \in \mathbb{R}^{M \times T_{tr}} \tag{3.57}$$

with $\boldsymbol{S} = [\mathbf{I}, \ldots, \mathbf{I}]^{\mathrm{T}}$ achieves the same performance as the full-dimensional design $\boldsymbol{B} = \boldsymbol{D}^{1/2}$.

Proof. Firstly, note that $\text{tr}(\boldsymbol{D}\boldsymbol{S}\boldsymbol{S}^{\mathrm{T}}) = \text{tr}(\boldsymbol{D})$ and thus the power constraint is still $\mathbf{1}^{\mathrm{T}}\boldsymbol{d} \le \rho_{\text{tr}}$.

Since $\boldsymbol{D}$ and the $\boldsymbol{C}_{\boldsymbol{h}_k}$ are diagonal the matrix multiplications commute.

If we incorporate (3.57) into (3.48), we get

$$\begin{aligned}\hat{\boldsymbol{h}}_k &= \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}^{1/2}\boldsymbol{S}\left(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}\boldsymbol{S} + \mathbf{I}\right)^{-1}\boldsymbol{\varphi}_k && (3.58)\\ &= \boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}^{1/2}\boldsymbol{S}\left(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{D}\boldsymbol{S} + \mathbf{I}\right)^{-1}(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{D}^{1/2}\boldsymbol{C}_{\boldsymbol{h}_k}^{1/2}\tilde{\boldsymbol{h}}_k + \boldsymbol{v}_k) && (3.59)\end{aligned}$$

where both $\tilde{\boldsymbol{h}}_k$ and $\boldsymbol{v}_k$ have i.i.d. Gaussian distributed entries with zero mean and unit variance (we substituted $\boldsymbol{h}_k = \boldsymbol{C}_{\boldsymbol{h}_k}^{1/2}\tilde{\boldsymbol{h}}_k$). What we need to show is that, with Assumption 3.2, we can eliminate all occurrences of the matrix $\boldsymbol{S}$. Because then, the estimate (3.59) has exactly the same stochastic distribution as the estimate from a full-dimensional pilot-matrix in (3.55), which leads to identical ergodic rates.

We have several diagonal matrices in (3.59) that all have the same sparsity structure

$$\boldsymbol{D}_1 = \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{D}^{1/2}, \quad \boldsymbol{D}_2 = \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{D}, \quad \boldsymbol{D}_3 = \boldsymbol{C}_{\boldsymbol{h}_k}^{1/2} \boldsymbol{D}^{1/2}. \tag{3.60}$$

The matrices can thus all be represented by a diagonalization $\boldsymbol{D}_i = \boldsymbol{U}_k \boldsymbol{\Lambda}_i \boldsymbol{U}_k^{\mathrm{T}}$, where $\boldsymbol{U} \in \{0,1\}^{M \times T_{\mathrm{tr}}}$ is a subunitary matrix. More specifically, due to Assumption 3.2, $\boldsymbol{U}_k$ contains a shifted identity matrix, i.e. it has the form

$$\boldsymbol{U}_k = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix}. \tag{3.61}$$

It is easy to see that $\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S}$ is a unitary permutation matrix. Consequently, it can be moved out of the inverse and eliminated altogether:

We replace $\boldsymbol{v}_k$ by $\tilde{\boldsymbol{v}}_k = \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S} \boldsymbol{v}_k$ which has the identical distribution since $\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S}$ is unitary.

$$\hat{\boldsymbol{h}}_k = \boldsymbol{U}_k \boldsymbol{\Lambda}_1 \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S} \left(\boldsymbol{S}^{\mathrm{T}} \boldsymbol{U}_k \boldsymbol{\Lambda}_2 \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S} + \mathbf{I} \right)^{-1} \boldsymbol{S}^{\mathrm{T}} \boldsymbol{U}_k (\boldsymbol{\Lambda}_3 \boldsymbol{U}_k^{\mathrm{T}} \tilde{\boldsymbol{h}}_k + \tilde{\boldsymbol{v}}_k) \tag{3.62}$$

$$= \boldsymbol{U}_k \boldsymbol{\Lambda}_1 \left(\boldsymbol{\Lambda}_2 + \mathbf{I} \right)^{-1} (\boldsymbol{\Lambda}_3 \boldsymbol{U}_k^{\mathrm{T}} \tilde{\boldsymbol{h}}_k + \tilde{\boldsymbol{v}}_k) \tag{3.63}$$

$$= \boldsymbol{U}_k \boldsymbol{\Lambda}_1 \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{U}_k \left(\boldsymbol{\Lambda}_2 + \mathbf{I} \right)^{-1} \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{U}_k (\boldsymbol{\Lambda}_3 \boldsymbol{U}_k^{\mathrm{T}} \tilde{\boldsymbol{h}}_k + \tilde{\boldsymbol{v}}_k) \tag{3.64}$$

$$= \boldsymbol{U}_k \boldsymbol{\Lambda}_1 \boldsymbol{U}_k^{\mathrm{T}} \left(\boldsymbol{U}_k \boldsymbol{\Lambda}_2 \boldsymbol{U}_k^{\mathrm{T}} + \mathbf{I} \right)^{-1} (\boldsymbol{U}_k \boldsymbol{\Lambda}_3 \boldsymbol{U}_k^{\mathrm{T}} \tilde{\boldsymbol{h}}_k + \boldsymbol{U}_k \tilde{\boldsymbol{v}}_k), \tag{3.65}$$

where after eliminating the $\boldsymbol{S}$, we move the $\boldsymbol{U}_k$ and $\boldsymbol{U}_k^{\mathrm{T}}$ back into the inverse. Now, since the null-space of $\boldsymbol{U}_k^{\mathrm{T}} \left(\boldsymbol{U}_k \boldsymbol{\Lambda}_2 \boldsymbol{U}_k^{\mathrm{T}} + \mathbf{I} \right)^{-1}$ is the orthogonal complement of range($\boldsymbol{U}_k$), we can simply replace $\boldsymbol{U}_k \tilde{\boldsymbol{v}}_k$ by a full vector with i.i.d. entries, which completes the proof. □

The key requirements for the derivation are that $\boldsymbol{S}$ has norm-one rows (for the power constraint) and that $\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{S}$ is unitary for all $\boldsymbol{U}_k$. In our case, both, $\boldsymbol{U}_k$ and $\boldsymbol{S}$, only contain zeros and ones, thus, each column of $\boldsymbol{S}$ has to be orthogonal to all but one column of each $\boldsymbol{U}_k$. In other words, each pilot vector should only pick one non-zero dimension of each covariance matrix.

If the spectra $\boldsymbol{c}_{\boldsymbol{h}_k}$ are sparse but the non-zero entries are not consecutive, we can in principle try to design the non-zero entries of $\boldsymbol{S}$ with a greedy method such that each column of $\boldsymbol{S}$ picks at most one of the non-zero entries of the spectra. If the covariance matrices are orthogonal, i.e., we have non-overlapping supports of the spectra $\boldsymbol{c}_{\boldsymbol{h}_k}$, it is trivial to find an optimal selection matrix. However, in general we can make no guarantee that $T_{\text{tr}} = M/S$ pilot signals will be sufficient for a greedy assignment.

If we find a selection matrix $\boldsymbol{S}$ such that we have an estimate as in (3.55), the remaining question is how to choose the power allocation $\boldsymbol{D}$. To this end, we now perform the asymptotic analysis given the channel estimates in (3.55). The analysis requires that the amount of training scales with the number of antennas but only at a factor $T_{\text{tr}} = M/S$.

Theorem 3.8. *For Conditions 2.1 to 2.3 and with Assumptions 3.1 and 3.2 we can choose* $\boldsymbol{D}$ *such that*

$$\lim_{M\to\infty} \gamma_k/M - \gamma_k^{asy}/M = 0 \tag{3.66}$$

and

$$\lim_{M\to\infty} \gamma_k/M > 0 \quad \forall k. \tag{3.67}$$

The matrix $\boldsymbol{C}_{\boldsymbol{y}|\varphi^{dl}}$ *is the matrix* $\boldsymbol{C}_{\boldsymbol{y}|\varphi}$ *with observations from downlink training and an identity matrix for the noise covariance, i.e.* $\boldsymbol{C}_{\boldsymbol{y}|\varphi^{dl}} = \mathbf{I} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{dl}}$.

where

$$\gamma_k^{asy} = p_k \operatorname{tr}\left(\left(\boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k} + \mathbf{I}\right)^{-1} \boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k}^2 \boldsymbol{C}_{\boldsymbol{y}|\varphi^{dl}}^{-1} \right) \tag{3.68}$$

$$= \sum_m \frac{p_k [\boldsymbol{c}_{\boldsymbol{h}_k}]_m^2 d_m}{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m d_m + 1} \times \frac{1}{1 + \sum_n p_n \frac{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m}{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m d_m + 1}}. \tag{3.69}$$

Proof. We can use steps equivalent to the ones in Lemma 3.2 to show that under Conditions 2.1 to 2.3

$$\lim_{M\to\infty} \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi^{\text{dl}}}^{-1} \hat{\boldsymbol{H}}/M - \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi^{\text{dl}}}^{-1} \hat{\boldsymbol{H}}/M] = 0. \tag{3.70}$$

Since there is no pilot-contamination, $\boldsymbol{\Gamma}^{\text{dl}} = \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi^{\text{dl}}}^{-1} \hat{\boldsymbol{H}}/M]$ is a diagonal matrix. The diagonal elements are given by

$$[\boldsymbol{\Gamma}^{\text{dl}}]_{kk} = \mathrm{E}[\hat{\boldsymbol{h}}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi^{\text{dl}}}^{-1} \hat{\boldsymbol{h}}_k/M] \tag{3.71}$$

$$= \frac{1}{M} \operatorname{tr}\left(\left(\boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k} + \mathbf{I}\right)^{-1} \boldsymbol{D}\boldsymbol{C}_{\boldsymbol{h}_k}^2 \boldsymbol{C}_{\boldsymbol{y}|\varphi^{\text{dl}}}^{-1} \right). \tag{3.72}$$

If we compare to Theorem 3.4, we see that as long as

$$\lim_{M\to\infty} [\boldsymbol{\Gamma}^{\text{dl}}]_{kk} > 0 \tag{3.73}$$

then

$$\gamma_k^{\text{asy}} = \frac{p_k M}{\boldsymbol{e}_k^{\mathrm{T}} (\boldsymbol{\Gamma}^{\text{dl}})^{-1} \boldsymbol{e}_k} = p_k M [\boldsymbol{\Gamma}^{\text{dl}}]_{kk} \tag{3.74}$$

which is exactly what we have in (3.68).

To show (3.73) for some $\boldsymbol{D}$, we set $\boldsymbol{D} = \frac{\rho_{\text{tr}}}{M}\mathbf{I}$. Since $\rho_{\text{tr}} = T_{\text{tr}}\rho_{\text{dl}}$ and $T_{\text{tr}} = M/S$ this leads to $\boldsymbol{D} = \frac{\rho_{\text{dl}}}{S}\mathbf{I}$. We get

$$\gamma_k^{\text{asy}}/M = \frac{1}{M}\sum_m \frac{p_k[\boldsymbol{c}_{\boldsymbol{h}_k}]_m^2}{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m + S/\rho_{\text{dl}}} \times \frac{1}{1+\sum_n p_n \frac{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m}{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m \rho_{\text{dl}}/S+1}} \tag{3.75}$$

We replace $\sum_n p_n \frac{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m}{[\boldsymbol{c}_{\boldsymbol{h}_n}]_m \rho_{\text{dl}}/S+1}$ by its upper bound $\frac{S}{\rho_{\text{dl}}}\sum_n p_n \leq S$

$$\geq \frac{1}{M}\frac{p_k}{1+S}\sum_m \frac{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m^2}{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m + S/\rho_{\text{dl}}}. \tag{3.76}$$

We know from Condition 2.2 that

$$\lim_{M\to\infty} \frac{1}{M}\sum_m [\boldsymbol{c}_{\boldsymbol{h}_k}]_m > 0. \tag{3.77}$$

We partition the elements of the vector $\boldsymbol{c}_{\boldsymbol{h}_k}$ into two vectors. The vector $\boldsymbol{c}_1$, which contains elements larger than S/ρ_{dl}, and the vector $\boldsymbol{c}_2$, which contains all other elements. We have

$$\lim_{M\to\infty} \frac{1}{M}\sum_m [\boldsymbol{c}_1]_m + \frac{1}{M}\sum_m [\boldsymbol{c}_2]_m > 0 \tag{3.78}$$

and thus at least one of the normalized summations does not vanish.

We use the same partition for the bound in (3.75) to get

$$\begin{aligned}
\gamma_k^{\text{asy}}/M &\geq \frac{\alpha}{M}\sum_m \frac{[\boldsymbol{c}_1]_m^2}{[\boldsymbol{c}_1]_m + S/\rho_{\text{dl}}} + \frac{\alpha}{M}\sum_m \frac{[\boldsymbol{c}_2]_m^2}{[\boldsymbol{c}_2]_m + S/\rho_{\text{dl}}} &(3.79)\\
&\geq \frac{\alpha}{M}\sum_m \frac{[\boldsymbol{c}_1]_m^2}{[\boldsymbol{c}_1]_m + [\boldsymbol{c}_1]_m} + \frac{\alpha}{M}\sum_m \frac{[\boldsymbol{c}_2]_m^2}{S/\rho_{\text{dl}} + S/\rho_{\text{dl}}} &(3.80)\\
&= \frac{\alpha}{M}\sum_m \frac{1}{2}[\boldsymbol{c}_1]_m + \frac{\alpha}{M}\frac{1}{2S/\rho_{\text{dl}}}\boldsymbol{c}_2^{\mathrm{T}}\boldsymbol{c}_2 &(3.81)
\end{aligned}$$

with $\alpha = \frac{p_k}{1+S}$. Due to the Cauchy-Schwartz inequality we have

$$\boldsymbol{c}_2^{\mathrm{T}}\boldsymbol{c}_2 \geq \frac{1}{M}\left(\sum_m [\boldsymbol{c}_2]_m\right)^2 \tag{3.82}$$

and thus we can combine (3.78) and (3.81) to complete the proof. □

Note that the asymptotic SINR is an upper bound on the achievable rates. The inter-user interference is eliminated and the resulting SINR is replaced by its expectation, which leads to an upper bound due to Jensen's inequality (cf. Theorem 3.6). However, it is an upper bound which is asymptotically tight as $M \to \infty$.

Since the asymptotic SINR only depends on the channel statistics, we can use it to optimize the power allocation $\boldsymbol{D}$. If we only have a single user the asymptotic SINR in (3.69) simplifies resulting in the following optimization problem

$$\max_{\boldsymbol{d}} \sum_m \frac{p_k [\boldsymbol{c}_{\boldsymbol{h}_k}]_m^2 d_m}{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m d_m + 1} \frac{1}{1 + p_k \frac{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m}{[\boldsymbol{c}_{\boldsymbol{h}_k}]_m d_m + 1}} \quad \text{s.t.} \quad \boldsymbol{d} \geq \boldsymbol{0}, \boldsymbol{1}^{\mathrm{T}} \boldsymbol{d} = \rho_{\mathrm{tr}}. \tag{3.83}$$

If we analyze the KKT conditions, we find that the solution to (3.83) can be found with a typical waterfilling procedure. A similar approach was developed in [30] in a slightly different context. Since the single-user case is not really of interest for our large-scale systems, we will not give further details.

In a multi-user scenario where we optimize a general network utility function, we can no longer find a semi-analytical waterfilling solution. In fact, the SINRs are no longer guaranteed to be concave in the power allocation $\boldsymbol{d}$. Nevertheless, we can as always find a locally optimal power allocation with our favorite projected gradient method.

Since we also need to optimize the power allocation for the data transmission, we should perform a joint optimization of the power allocation of both, the training and the data phase, with a projected gradient method. In the end we get a close to optimal pilot allocation and precoder design.

If we desire a simpler power allocation that does not require an adaptive update, we can simply use a normalized selection vector for the power allocation. That is, we choose $\boldsymbol{d} = \alpha \boldsymbol{s}$ where $\boldsymbol{s} \in \{0, 1\}^M$ and α is a normalization factor. We can for example set the non-zero dimensions in $\boldsymbol{s}$ such that for each user at least 80% of the energy in the spectrum $\boldsymbol{c}_{\boldsymbol{h}_k}$ is covered, i.e., $\boldsymbol{s}$ selects the larges elements of each $\boldsymbol{c}_{\boldsymbol{h}_k}$ such that $\boldsymbol{s}^{\mathrm{T}} \boldsymbol{c}_{\boldsymbol{h}_k} \geq \frac{80}{100} \boldsymbol{1}^{\mathrm{T}} \boldsymbol{c}_{\boldsymbol{h}_k}$.

While the projected gradient approaches work well for a continuously differentiable utility function, we need to take special care when we are interested in maximizing the minimum rate. Since we consider the max-min objective in the simulation results at the end of this chapter, we will briefly describe a method that maximizes the minimum of the asymptotic SINRs of all users. The optimization problem is given by

$$\max_{\boldsymbol{d}:\mathbf{1}^{\mathrm{T}}\boldsymbol{d}=\rho_{\mathrm{tr}},\boldsymbol{d}\geq\mathbf{0}} \max_{\boldsymbol{p}:\mathbf{1}^{\mathrm{T}}\boldsymbol{p}=\rho_{\mathrm{dl}},\boldsymbol{p}\geq\mathbf{0}} \min_{k} \gamma_k^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d}) \tag{3.84}$$

and can be rewritten to

$$\max_{\boldsymbol{d}:\mathbf{1}^{\mathrm{T}}\boldsymbol{d}=\rho_{\mathrm{tr}},\boldsymbol{d}\geq\mathbf{0}} \max_{\boldsymbol{p}:\mathbf{1}^{\mathrm{T}}\boldsymbol{p}=\rho_{\mathrm{dl}},\boldsymbol{p}\geq\mathbf{0}} \alpha \quad \text{s.t.} \quad \gamma_k^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d}) \geq \alpha \; \forall k. \tag{3.85}$$

For a fixed $\boldsymbol{d}$ we can solve the inner problem with respect to $\boldsymbol{p}$ efficiently. The reason is that $p_k/\gamma_k^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d})$ fulfills the conditions of a standard interference function [31]. Consequently, all inequality constraints will be fulfilled with equality at the optimum and there is a unique solution $\boldsymbol{p}^\star$, $\alpha^\star$ to the system of equations

$$\begin{aligned} \boldsymbol{\gamma}^{\mathrm{asy}}(\boldsymbol{p}^\star,\boldsymbol{d}) &= \alpha^\star\mathbf{1} \\ \mathbf{1}^{\mathrm{T}}\boldsymbol{p}^\star &= \rho_{\mathrm{dl}} \end{aligned} \tag{3.86}$$

where $\boldsymbol{\gamma}^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d}) = [\gamma_1^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d}),\ldots,\gamma_K^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d})]^{\mathrm{T}}$.

We can, e.g., use the Yates fixed point method [31] to find an optimal $\boldsymbol{p}^\star$ given some $\boldsymbol{d}$. Specifically, we iterate the following steps until convergence:

$$[p_1,\ldots,p_k]^{\mathrm{T}} \leftarrow [p_1/\gamma_1^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d}),\ldots,p_K/\gamma_K^{\mathrm{asy}}(\boldsymbol{p},\boldsymbol{d})]^{\mathrm{T}} \tag{3.87}$$

$$\boldsymbol{p} \leftarrow \frac{\boldsymbol{p}}{\mathbf{1}^{\mathrm{T}}\boldsymbol{p}}. \tag{3.88}$$

Once we found an optimal power allocation $\boldsymbol{p}^\star(\boldsymbol{d})$ and objective $\alpha^\star(\boldsymbol{d})$ for some training power $\boldsymbol{d}$, we optimize the training power with a projected gradient method. That is, we solve

$$\max_{\boldsymbol{d}:\mathbf{1}^{\mathrm{T}}\boldsymbol{d}=\rho_{\mathrm{tr}},\boldsymbol{d}\geq\mathbf{0}} \alpha^\star(\boldsymbol{d}). \tag{3.89}$$

To this end, we need to calculate the derivative of $\alpha^\star(\boldsymbol{d})$ with respect to $\boldsymbol{d}$. The function $\alpha^\star(\boldsymbol{d})$ is only defined implicitly in the non-linear equation system (3.86). We can use the chain rule for multivariate differentiation to get

$$\left.\frac{\partial\boldsymbol{\gamma}^{\mathrm{asy}}}{\partial\boldsymbol{p}^{\mathrm{T}}}\right|_{\boldsymbol{p}=\boldsymbol{p}^\star}\frac{\partial\boldsymbol{p}^\star}{\partial\boldsymbol{d}^{\mathrm{T}}} + \left.\frac{\partial\boldsymbol{\gamma}^{\mathrm{asy}}}{\partial\boldsymbol{d}^{\mathrm{T}}}\right|_{\boldsymbol{p}=\boldsymbol{p}^\star} = \mathbf{1}\frac{\partial\alpha^\star}{\partial\boldsymbol{d}^{\mathrm{T}}} \tag{3.90}$$

$$\mathbf{1}^{\mathrm{T}}\frac{\partial\boldsymbol{p}^\star}{\partial\boldsymbol{d}^{\mathrm{T}}} = 0. \tag{3.91}$$

The Jacobi matrix of a vector valued function $\boldsymbol{f} : \mathbb{R}^m \mapsto \mathbb{R}^n$ is here denoted by $\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}^{\mathrm{T}}}$. Thus $\frac{\partial\boldsymbol{\gamma}^{\mathrm{asy}}}{\partial\boldsymbol{p}^{\mathrm{T}}}$, $\frac{\partial\boldsymbol{\gamma}^{\mathrm{asy}}}{\partial\boldsymbol{d}^{\mathrm{T}}}$, and $\frac{\partial\boldsymbol{p}^\star}{\partial\boldsymbol{d}^{\mathrm{T}}}$ are $M \times M$ matrices and $\frac{\partial\alpha^\star}{\partial\boldsymbol{d}^{\mathrm{T}}}$ is an M-dimensional row-vector.

We solve the first equation for

$$\frac{\partial \boldsymbol{p}^\star}{\partial \boldsymbol{d}^\mathrm{T}} = \left(\frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{p}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right)^{-1} \left(\mathbf{1} \frac{\partial \alpha^\star}{\partial \boldsymbol{d}^\mathrm{T}} - \frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{d}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right) \tag{3.92}$$

which we plug into the second equation to get

$$\mathbf{1}^\mathrm{T} \left(\frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{p}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right)^{-1} \left(\mathbf{1} \frac{\partial \alpha^\star}{\partial \boldsymbol{d}^\mathrm{T}} - \frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{d}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right) = 0 \tag{3.93}$$

which we finally solve for

$$\frac{\partial \alpha^\star}{\partial \boldsymbol{d}^\mathrm{T}} = \frac{\mathbf{1}^\mathrm{T} \left(\frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{p}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right)^{-1} \frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{d}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star}}{\mathbf{1}^\mathrm{T} \left(\frac{\partial \boldsymbol{\gamma}^\mathrm{asy}}{\partial \boldsymbol{p}^\mathrm{T}} \Big|_{\boldsymbol{p}=\boldsymbol{p}^\star} \right)^{-1} \mathbf{1}}. \tag{3.94}$$

Equipped with the derivative, we can apply the methods from Appendix D to solve the problem (3.89).

For a given power allocation, the instantaneous complexity of calculating the receive filters is equivalent to the complexity for uplink training (cf. Section 3.5.3). With our approach to pilot design, the matrices $\boldsymbol{C}_{\boldsymbol{y}|\varphi^\mathrm{dl}}$ are diagonal and the MMSE estimates can be calculated in $\mathrm{O}(M)$ operations. Note that state-of-the-art methods such as JSDM [23] rely on equivalent assumptions for practical implementation.

JSDM makes additional assumptions on the user covariance matrices and relies on a grouping of the users which is not necessary with our approach. Thus, a direct comparison in our simulation setup is difficult. Since JSDM does not optimize the power allocation, we expect the performance to be similar to our approach for a fixed power allocation as long as those grouping assumptions are fulfilled.

3.7 Semi-blind Channel Estimation

In addition to observations from an uplink training phase we can also incorporate all signals received during the uplink data phase to improve the channel estimates. We have the uplink data signals

$$\boldsymbol{y}_t^\mathrm{ul} = \sum_k \sqrt{p_k} \boldsymbol{h}_k s_{kt} + \boldsymbol{v}_t^\mathrm{ul} \quad t = 1, \ldots, T_\mathrm{ul}. \tag{3.95}$$

or in matrix-vector notation

$$\boldsymbol{Y} = \boldsymbol{H} \boldsymbol{P}^{1/2} \boldsymbol{S} + \boldsymbol{V}^\mathrm{ul} \in \mathbb{C}^{M \times T_\mathrm{ul}} \tag{3.96}$$

where the entries of the data matrix $\boldsymbol{S}$ and the noise $\boldsymbol{V}^{\text{ul}}$ are i.i.d. complex Gaussian with zero-mean and unit variance.

In the following, we consider semi-blind filter design using the set of observations containing uplink training

$$\varphi^{\text{tr}} = \{\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{T_{\text{tr}}}\} \tag{3.97}$$

and uplink data signals $\boldsymbol{Y}$, i.e., we use

$$\varphi^{\text{sb}} = \{\varphi^{\text{tr}}, \boldsymbol{Y}\}. \tag{3.98}$$

To calculate the LMMSE filters and the corresponding achievable uplink SINRs we need the MMSE estimates $\hat{\boldsymbol{h}}_k^{\text{sb}} = \mathrm{E}[\boldsymbol{h}_k|\varphi^{\text{sb}}]$ and the corresponding covariance matrices of the estimation errors $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{sb}}}$. Since the posterior $f_{h|\varphi^{\text{sb}}}(\boldsymbol{H}|\varphi^{\text{sb}})$ cannot be given in closed form in this scenario, it is impractical to calculate the conditional means exactly.

There are various methods to get channel estimates based on semi-blind observations, some blind methods even work with no training signals at all. We have methods that make use of the eigenvalue decomposition of the uplink signals $\boldsymbol{Y}$ [32, 33, 34], which can be connected to maximum-likelihood estimation [34]. Other approaches exist for semi-blind estimation [35].

Since we want to use the LMMSE filter, we focus on the approach used in [24] to approximate the joint posterior of the channels and data symbols as the product

$$f_{h,s|\varphi^{\text{sb}}}(\boldsymbol{H}, \boldsymbol{S}|\varphi^{\text{sb}}) \approx q_{\boldsymbol{S}}(\boldsymbol{S}) \prod_k q_{\boldsymbol{h}_k}(\boldsymbol{h}_k). \tag{3.99}$$

Specifically, we look for the best approximation in terms of Kullback-Leibler divergence that factorizes in a function of $\boldsymbol{S}$ and functions of $\boldsymbol{h}_k$. This approach is known as variational Bayesian inference [36] and allows us to get estimates of the conditional mean and the conditional covariance matrix of the channel vectors.

In general, we approximate a posterior of a vector of random variables $\boldsymbol{x}$ by a function that factorizes with respect to the entries of $\boldsymbol{x}$

$$f_{\boldsymbol{x}|\varphi}(\boldsymbol{x}|\varphi) \approx \prod_{i=1}^{n} q_{x_i}(x_i). \tag{3.100}$$

That is, we want to solve

$$\min_{q_{x_1},\dots,q_{x_n}} D\left(\prod_{i=1}^{n} q_{x_i} \middle\| f_{\boldsymbol{x}|\varphi}\right). \tag{3.101}$$

This problem is typically solved using coordinate descent, i.e., we iteratively optimize one q_{x_i} while keeping all other q_{x_j} with $j \neq i$ fixed. The optimal q_{x_i} for fixed q_{x_j}, $j \neq i$ is given by

$$\log q_{x_i} = \mathrm{E}_{q_{x_j}, j \neq i}[\log f_{\boldsymbol{x}|\varphi}(\boldsymbol{x})] + c_i \tag{3.102}$$

where the expectation is with respect to all x_j with $j \neq i$ and is calculated using the approximate densities q_{x_j}. The constant c_i scales q_{x_i} such that it integrates to one.

This alternating optimization approach can be applied to approximate the posterior $f_{h,s|\varphi^{\mathrm{sb}}}$ by the factorization in (3.99). That is, we alternatingly calculate

$$\log q_{\boldsymbol{S}}(\boldsymbol{S}) = \mathrm{E}_{q_{\boldsymbol{h}_n}, n=1,\ldots,K}[\log f_{h,s|\varphi^{\mathrm{sb}}}(\boldsymbol{H}, \boldsymbol{S}|\varphi^{\mathrm{sb}})] + c_s \tag{3.103}$$

and

$$\log q_{\boldsymbol{h}_k}(\boldsymbol{h}_k) = \mathrm{E}_{q_s, q_{\boldsymbol{h}_n}, n \neq k}[\log f_{h,s|\varphi^{\mathrm{sb}}}(\boldsymbol{H}, \boldsymbol{S}|\varphi^{\mathrm{sb}})] + c_k. \tag{3.104}$$

The posterior is given by

$$\begin{aligned}\log f_{h,s|\varphi^{\mathrm{sb}}}(\boldsymbol{H}, \boldsymbol{S}|\varphi^{\mathrm{sb}}) = &- \rho_{\mathrm{ul}} \|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{S}\|_F^2 + \|\boldsymbol{S}\|_F^2 \\ &- \sum_k (\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^{\mathrm{tr}})^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\mathrm{tr}}}^{-1} (\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^{\mathrm{tr}}) + c_f\end{aligned}$$

with the *training based* MMSE estimate $\hat{\boldsymbol{h}}_k^{\mathrm{tr}}$ in (3.24) and the corresponding error covariance matrix $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\mathrm{tr}}}$ in (3.26).

Calculating the optimal $q_{\boldsymbol{S}}$ and $q_{\boldsymbol{h}_k}$ as in (3.103) and (3.104), we realize that the results are multivariate Gaussian densities and the update of one of the factors $q_{\boldsymbol{S}}$ or $q_{\boldsymbol{h}_k}$ only depends on the first and second order moments of the other factors. If we have the following (estimated) moments for $q_{\boldsymbol{S}}$ and $q_{\boldsymbol{h}_k}$

$$\begin{aligned}\hat{\boldsymbol{H}} &= \mathrm{E}_{q_{\boldsymbol{h}_1},\ldots,q_{\boldsymbol{h}_K}}[\boldsymbol{H}], & \hat{\boldsymbol{S}} &= \mathrm{E}_{q_{\boldsymbol{S}}}[\boldsymbol{S}] \\ \boldsymbol{B}_{\boldsymbol{H}} &= \mathrm{E}_{q_{\boldsymbol{h}_1},\ldots,q_{\boldsymbol{h}_K}}[\boldsymbol{H}^{\mathrm{H}}\boldsymbol{H}], & \boldsymbol{B}_{\boldsymbol{S}} &= \mathrm{E}_{q_{\boldsymbol{S}}}[\boldsymbol{S}\boldsymbol{S}^{\mathrm{H}}]\end{aligned}$$

then we get

$$\begin{aligned}\log q_{\boldsymbol{S}}(\boldsymbol{S}) &= -\rho_{\mathrm{ul}} \operatorname{tr}(\boldsymbol{S}^{\mathrm{H}}\boldsymbol{B}_{\boldsymbol{H}}\boldsymbol{S} - \boldsymbol{Y}^{\mathrm{H}}\hat{\boldsymbol{H}}\boldsymbol{S} - \boldsymbol{S}^{\mathrm{H}}\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{Y}) \\ &\quad - \operatorname{tr}(\boldsymbol{S}\boldsymbol{S}^{\mathrm{H}}) + c \qquad (3.105) \\ \log q_{\boldsymbol{h}_k}(\boldsymbol{h}_k) &= -\rho_{\mathrm{ul}}(-\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{Y}\boldsymbol{S}^{\mathrm{H}}\boldsymbol{e}_k - \boldsymbol{e}_k^{\mathrm{T}}\boldsymbol{S}\boldsymbol{Y}^{\mathrm{H}}\boldsymbol{h}_k \\ &\quad + [\boldsymbol{B}_{\boldsymbol{S}}]_{kk}\boldsymbol{h}_k^{\mathrm{H}}\boldsymbol{h}_k + \sum_{n \neq k}[\boldsymbol{B}_{\boldsymbol{S}}]_{nk}\boldsymbol{h}_k^{\mathrm{H}}\hat{\boldsymbol{h}}_n + [\boldsymbol{B}_{\boldsymbol{S}}]_{kn}\hat{\boldsymbol{h}}_n^{\mathrm{H}}\boldsymbol{h}_k) \\ &\quad - (\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^{\mathrm{tr}})^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\mathrm{tr}}}^{-1}(\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k^{\mathrm{tr}}) + c_k. \qquad (3.106)\end{aligned}$$

We identify the moments of $q_{\boldsymbol{S}}$ and $q_{\boldsymbol{h}_k}$ to formulate an algorithm that iteratively updates the moments of one density based on the moments of the other densities. The algorithm is given in detail in Alg. 1. After convergence of the updates we get approximate MMSE estimates $\hat{\boldsymbol{h}}_k^{\text{sb}}$ and corresponding error covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{sb}}}$ which we can use to calculate uplink filters and corresponding achievable rates.

Algorithm 1 Variational Bayesian inference applied to joint channel estimation and detection

```
Calculate the training based MMSE estimates $\hat{\boldsymbol{h}}_k^{\text{tr}}$ and the corresponding error covariance matrices $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{tr}}}$
Initialize $\hat{\boldsymbol{S}}$ with its linear MMSE estimate (cf. [24])
    $\hat{\boldsymbol{S}} \leftarrow \hat{\boldsymbol{H}}^{\text{tr,H}}\Big(\frac{1}{\rho_{\text{tr}}}\mathbf{I}+\sum_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{tr}}} + \hat{\boldsymbol{H}}^{\text{tr}}(\hat{\boldsymbol{H}}^{\text{tr}})^{\text{H}}\Big)^{-1}\boldsymbol{Y}$
The second order statistics are given by
    $\boldsymbol{B}_{\boldsymbol{S}} \leftarrow \hat{\boldsymbol{S}}\hat{\boldsymbol{S}}^{\text{H}} + T_{\text{ul}}\left(\mathbf{I}+\hat{\boldsymbol{H}}^{\text{tr,H}}\Big(\frac{1}{\rho_{\text{tr}}}\mathbf{I}+\sum_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{tr}}}\Big)^{-1}\hat{\boldsymbol{H}}^{\text{tr}}\right)^{-1}$
for $i = 1, \dots$ do
    for $k = 1, \dots, K$ do
        Update moments of $\boldsymbol{h}_k$
            $\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{sb}}} \leftarrow ([\boldsymbol{B}_{\boldsymbol{S}}]_{kk}\rho_{\text{ul}}\mathbf{I}+\boldsymbol{C}^{-1}_{\boldsymbol{h}_k|\varphi^{\text{tr}}})^{-1}$
            $\hat{\boldsymbol{h}}_k \leftarrow \rho_{\text{ul}}\boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{sb}}}\Big(\boldsymbol{Y}\hat{\boldsymbol{S}}^{\text{H}}\boldsymbol{e}_k - \sum_{n\neq k}\hat{\boldsymbol{h}}_n[\boldsymbol{B}_{\boldsymbol{S}}]_{nk}\Big) + \boldsymbol{C}_{\boldsymbol{h}_k|\varphi^{\text{sb}}}\boldsymbol{C}^{-1}_{\boldsymbol{h}_k|\varphi^{\text{tr}}}\hat{\boldsymbol{h}}_k^{\text{tr}}$
            $\boldsymbol{B}_{\boldsymbol{H}} \leftarrow \hat{\boldsymbol{H}}^{\text{H}}\hat{\boldsymbol{H}} + \text{diag}(\text{tr}(\boldsymbol{C}_{\boldsymbol{h}_1|\varphi^{\text{sb}}}), \dots, \text{tr}(\boldsymbol{C}_{\boldsymbol{h}_K|\varphi^{\text{sb}}})).$
    end for
    Update moments of $\boldsymbol{S}$
        $\hat{\boldsymbol{S}} \leftarrow \big(\rho_{\text{ul}}\boldsymbol{B}_{\boldsymbol{H}} + \mathbf{I}\big)^{-1}\rho_{\text{ul}}\hat{\boldsymbol{H}}^{\text{H}}\boldsymbol{Y}$
        $\boldsymbol{B}_{\boldsymbol{S}} \leftarrow \hat{\boldsymbol{S}}\hat{\boldsymbol{S}}^{\text{H}} + T_{\text{ul}}(\rho_{\text{ul}}\boldsymbol{B}_{\boldsymbol{H}} + \mathbf{I})^{-1}$
end for
```

Since φ^{sb} contains more information than φ^{tr}, we expect better performance compared to the methods that only use training based estimation. Of course, the computational complexity is also considerably larger. Unfortunately, asymptotic analysis seems difficult for the variational approach. We can still use a fixed uplink power allocation, e.g., the same as for uplink training, and only use the improved channel estimates to calculate the instantaneous filter vectors.

The computational complexity per iteration of the variational inference method is comparable to the computational complexity of the LMMSE filter. However, for the variational inference approach we have to do several iterations per coherence interval to get significant performance gains.

The variational method also works if the data symbols are from a discrete modulation alphabet [24]. The algorithm can be adapted such that the complexity is linear in the size of the alphabet. Thus, if we use the correct, discrete distribution of the data symbols we get better channel estimates at the cost of a slightly higher computational complexity.

Using iterative variational inference might be too complex for practical consideration, but it demonstrates that there are indeed gains available if we do not restrict ourselves to training based estimation. There surely is room for a middle ground between the simple training based estimation and the iterative variational method. This is room that we leave to future researchers.

3.8 Robust matched Filter

We saw that for perfect CSI, a simple matched filter achieves the full DoMs. For imperfect CSI with pilot-contamination, however, the SINRs saturate and we have zero DoMs. In the following we present a robust matched filter design, that keeps the simplicity of the matched filter but enables us to exploit channel structure to reach the full DoMs. Basically, we combine the classical matched filter with a pre-filter that exploits statistical information. We focus on the scenario with uplink pilots as described in Section 3.5.

Consider again the optimal LMMSE filter (3.20)

$$\tilde{\boldsymbol{G}} = [\tilde{\boldsymbol{g}}_1, \ldots, \tilde{\boldsymbol{g}}_K] = \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} \left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} \right)^{-1} \boldsymbol{P}^{-1/2}. \tag{3.107}$$

The filtered uplink signals are given by

The uplink receive signals were modeled as $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{P}^{1/2}\boldsymbol{s} + \boldsymbol{v}$.

$$\hat{\boldsymbol{s}} = \tilde{\boldsymbol{G}}^{\mathrm{H}}\boldsymbol{y} = \boldsymbol{P}^{-1/2}\left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}\right)^{-1}\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\boldsymbol{y} \quad (3.108)$$

$$= \boldsymbol{P}^{-1/2}\left(\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}\right)^{-1}\left(\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\boldsymbol{H}\boldsymbol{P}^{1/2}\boldsymbol{s} + \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\boldsymbol{v}\right). \quad (3.109)$$

Following the analysis in Section 3.5, we get the asymptotically equivalent

Remember: $\boldsymbol{\Gamma} = \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}/M] \asymp \hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}$.

$$\hat{\boldsymbol{s}} \asymp \boldsymbol{P}^{-1/2}\boldsymbol{\Gamma}^{-1}\left(\boldsymbol{\Gamma}\boldsymbol{P}^{1/2}\boldsymbol{s} + \boldsymbol{0}\right) = \boldsymbol{s}. \quad (3.110)$$

That is, the estimation error goes to zero, which is consistent with our result that the SINR goes to infinity.

If we replace the filter in (3.107) by

$$\boldsymbol{G} = \frac{1}{M}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}\boldsymbol{\Gamma}^{-1}\boldsymbol{P}^{-1/2} \quad (3.111)$$

we get the same result. The estimation error still goes to zero since we replaced a part of the filter with an asymptotically equivalent part.

The instantaneous observations $\boldsymbol{\varphi}_p$ appear in $\boldsymbol{G}$ only in form of the MMSE estimates $\hat{\boldsymbol{H}}$. Thus, the filters $\boldsymbol{G}$ are linear in the observations since the MMSE estimates are linear in the observations. Furthermore, since $\boldsymbol{\Gamma}$ is block-diagonal the filter for user k that transmits pilot sequence p only depends on $\boldsymbol{\varphi}_p$. That is,

$$\boldsymbol{g}_k = \boldsymbol{B}_k\boldsymbol{\varphi}_p \quad (3.112)$$

where the linear transformation $\boldsymbol{B}_k \in \mathbb{C}^{M\times M}$ is deterministic, i.e., depends only on the channel statistics.

We call this structure for a general deterministic $\boldsymbol{B}_k$ a generalized matched filter (GMF). We can interpret $\boldsymbol{B}_k$ as a generalized spatial weighting of the received signals, however, note that $\boldsymbol{B}_k$ does not have to be positive semi-definite. For a GMF receiver, the estimate

$$\hat{s}_k = \boldsymbol{\varphi}_p^{\mathrm{H}}\boldsymbol{B}_k^{\mathrm{H}}\boldsymbol{y}^{\mathrm{ul}} \quad (3.113)$$

is not only linear in the data vector $\boldsymbol{y}^{\mathrm{ul}}$, which is a common restriction, but also linear in the observations $\boldsymbol{\varphi}_k$. We already know from the analysis above that for a specific choice of $\boldsymbol{B}_k$, the estimation error goes to zero. We will study the optimal choice for $\boldsymbol{B}_k$ in the following. This leads to a robust matched filter (RMF) that is able to suppress the interference caused by pilot contamination as the number of antennas grows large and thus exhibits the desired linear growth of the effective SINR.

To characterize the achievable rates we use the lower bound in (2.52) which assumes that the decoder only knows the channel statistics. We have in the uplink SINRs

$$\gamma_k^{\mathrm{ul}} = \frac{p_k \left|\mathrm{E}[\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{h}_k]\right|^2}{\mathrm{E}[\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{v}^{\mathrm{ul}}} \boldsymbol{g}_k] + p_k \operatorname{var}[\boldsymbol{g}_k \boldsymbol{h}_k] + \sum_{n \neq k} p_n \mathrm{E}[|\boldsymbol{g}_k \boldsymbol{h}_n|^2]}. \tag{3.114}$$

With Lemma C.2 we are able to evaluate the expectations for the filter design in (3.112) to get

$$\gamma_k^{\mathrm{ul}} = \frac{p_k \left|\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{B}_k)\right|^2}{\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{y}} \boldsymbol{B}_k \boldsymbol{C}_{\boldsymbol{\varphi}_p} \boldsymbol{B}_k^{\mathrm{H}}) + \sum_{n \in \Omega_p \backslash \{k\}} p_n \left|\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_n} \boldsymbol{B}_k)\right|^2} \tag{3.115}$$

where

$$\boldsymbol{C}_{\boldsymbol{y}} = \boldsymbol{C}_{\boldsymbol{v}} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.116}$$

With vectorized linear transformations $\boldsymbol{b}_k = \mathrm{vec}(\boldsymbol{B}_k)$ and covariance matrices $\boldsymbol{c}_{\boldsymbol{h}_k} = \mathrm{vec}(\boldsymbol{C}_{\boldsymbol{h}_k})$ the notation simplifies to

$$\gamma_k^{\mathrm{ul}} = \frac{p_k \left|\boldsymbol{c}_{\boldsymbol{h}_k}^{\mathrm{H}} \boldsymbol{b}_k\right|^2}{\boldsymbol{b}_k^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}} + \sum_{n \in \Omega_p \backslash \{k\}} p_n \boldsymbol{c}_{\boldsymbol{h}_n} \boldsymbol{c}_{\boldsymbol{h}_n}^{\mathrm{H}}\right) \boldsymbol{b}_k} \tag{3.117}$$

which has exactly the same Rayleigh-quotient structure as, e.g., the SINRs in (3.11). The optimal vectorized transformation is thus given by

$$\boldsymbol{b}_k^{\star} = \left(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}} + \sum_{n \in \Omega_p \backslash \{k\}} p_n \boldsymbol{c}_{\boldsymbol{h}_n} \boldsymbol{c}_{\boldsymbol{h}_n}^{\mathrm{H}}\right)^{-1} \boldsymbol{c}_{\boldsymbol{h}_k} \tag{3.118}$$

with optimal SINRs

$$\begin{aligned} \gamma_k^{\mathrm{ul}} &= p_k \boldsymbol{c}_{\boldsymbol{h}_k}^{\mathrm{H}} \boldsymbol{b}_k^{\star} && (3.119) \\ &= p_k \boldsymbol{c}_{\boldsymbol{h}_k}^{\mathrm{H}} \left(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}} + \sum_{n \in \Omega_p \backslash \{k\}} p_n \boldsymbol{c}_{\boldsymbol{h}_n} \boldsymbol{c}_{\boldsymbol{h}_n}^{\mathrm{H}}\right)^{-1} \boldsymbol{c}_{\boldsymbol{h}_k}. && (3.120) \end{aligned}$$

The filter $\boldsymbol{g}_k = \boldsymbol{B}_k^{\star} \boldsymbol{\varphi}_p$ (where $\boldsymbol{B}_k^{\star}$ is obtained from $\boldsymbol{b}_k^{\star}$ by writing the vector into a matrix) is the robust matched filter (RMF). As we will see later on, with this filter we achieve the maximum DoM.

In the downlink we can use equivalent beamforming vectors

$$\boldsymbol{t}_k = \boldsymbol{A}_k \boldsymbol{\varphi}_p \tag{3.121}$$

with the deterministic linear transformation $\boldsymbol{A}_k$. We use the same lower bound as in the uplink, i.e., we assume that the users only know the channel statistics. This leaves us with

$$\gamma_k^{\text{dl}} = \frac{\left|\mathrm{E}[\boldsymbol{t}_k^{\mathrm{H}} \boldsymbol{h}_k]\right|^2}{1 + \mathrm{var}[\boldsymbol{t}_k^{\mathrm{H}} \boldsymbol{h}_k] + \sum_{n \neq k} \mathrm{E}[|\boldsymbol{t}_n^{\mathrm{H}} \boldsymbol{h}_k|^2]}. \tag{3.122}$$

Again we evaluate the expectations yielding

$$\gamma_k^{\text{dl}} = \frac{|\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{A}_k)|^2}{1 + \sum_n \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{A}_n \boldsymbol{Q}_{p(n)} \boldsymbol{A}_n^{\mathrm{H}}) + \sum_{n \in \Omega_p \setminus \{k\}} |\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{A}_n)|^2} \tag{3.123}$$

which we rewrite in vectorized form with $\boldsymbol{a}_k = \mathrm{vec}(\boldsymbol{A}_k)$

$$\gamma_k^{\text{dl}} = \frac{\left|\boldsymbol{c}_{\boldsymbol{h}_k}^{\mathrm{H}} \boldsymbol{a}_k\right|^2}{1 + \sum_n \boldsymbol{a}_n^{\mathrm{H}} \boldsymbol{Q}_{p(n)} \otimes \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{a}_n + \sum_{n \in \Omega_p \setminus \{k\}} \boldsymbol{a}_n^{\mathrm{H}} \boldsymbol{c}_{\boldsymbol{h}_k} \boldsymbol{c}_{\boldsymbol{h}_k}^{\mathrm{H}} \boldsymbol{a}_n}. \tag{3.124}$$

An average sum-power constraint $\mathrm{E}[\boldsymbol{x}^{\mathrm{H}} \boldsymbol{x}] \leq \rho_{\text{dl}}$ on the transmit signal expands to

$$\mathrm{E}[\boldsymbol{x}^{\mathrm{H}} \boldsymbol{x}] = \sum_k \mathrm{tr}(\mathrm{E}[\boldsymbol{t}_k \boldsymbol{t}_k^{\mathrm{H}}]) = \sum_k \mathrm{tr}(\boldsymbol{A}_k \boldsymbol{C}_{\boldsymbol{\varphi}_p} \boldsymbol{A}_k^{\mathrm{H}}) \tag{3.125}$$

$$= \sum_k \boldsymbol{a}_k^{\mathrm{T}} (\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{\mathrm{T}} \otimes \mathbf{I}) \boldsymbol{a}_k \leq \rho_{\text{dl}}. \tag{3.126}$$

It follows from the uplink-downlink duality presented in Appendix C.1 that, for each feasible downlink SINR, we can find an uplink SINR of the form (3.119) with the noise covariance matrix $\boldsymbol{C}_{\boldsymbol{v}^{\text{ul}}} = \mathbf{I}$ and the sum-power constraint $\mathbf{1}^{\mathrm{T}} \boldsymbol{p} \leq \rho_{\text{dl}}$. That is, uplink-downlink duality also works for the GMF design.

In fact, we do not have the issue with information asymmetry that we had for the conditional mutual information in Section 3.4. Since here we use a bound on the mutual information that assumes no CSI at the decoder, it is applicable in the uplink and the downlink.

As before, we reformulate the SINR to a form which is more convenient for asymptotic analysis. Applying steps equivalent to those in Lemma 2.1 we get

$$\gamma_k^{\star} = M p_k \frac{\boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{\Gamma}^{\text{RMF}} (\frac{1}{M} \boldsymbol{P}^{-1} + \boldsymbol{\Gamma}^{\text{RMF}})^{-1} \boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}} (\frac{1}{M} \boldsymbol{P}^{-1} + \boldsymbol{\Gamma}^{\text{RMF}})^{-1} \boldsymbol{e}_k} \tag{3.127}$$

Since the SINR in (3.127) already only depends on the channel statistics, the goal of the asymptotic analysis is not to find a deterministic asymptotic equivalent SINR that we can use to optimize the power allocation. Instead, we are interested in how the performance of the RMF compares to that of the LMMSE filter in (3.20) for a large number of antennas.

with

$$[\mathbf{\Gamma}^{\mathrm{RMF}}]_{kn} = \begin{cases} \frac{1}{M}\,\mathrm{tr}(\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{y}}^{-1}\boldsymbol{C}_{\boldsymbol{h}_n}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}) & \text{if both } k \text{ and } n \text{ use pilot sequence } p \\ 0 & \text{otherwise.} \end{cases} \tag{3.128}$$

This matrix is very similar to $\mathbf{\Gamma}$ in (3.31). Instead of the matrix $\boldsymbol{C}_{\boldsymbol{y}|\varphi}$ with the conditional covariance matrices, we have

$$\boldsymbol{C}_{\boldsymbol{y}} = \boldsymbol{C}_{\boldsymbol{v}} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.129}$$

Remember that the p-th diagonal block of $\mathbf{\Gamma}$, $\mathbf{\Gamma}_p$, is the same as $\mathbf{\Gamma}_p^{\mathrm{RMF}}$ only with $\boldsymbol{C}_{\boldsymbol{y}}$ replaced by $\boldsymbol{C}_{\boldsymbol{y}|\varphi}$.

Same as for $\mathbf{\Gamma}$ we also have a block-diagonal structure $\mathbf{\Gamma}^{\mathrm{RMF}} = \mathrm{blkdiag}(\mathbf{\Gamma}_1^{\mathrm{RMF}}, \ldots, \mathbf{\Gamma}_{T_{\mathrm{tr}}}^{\mathrm{RMF}})$. We can use the matrix that contains the vectorized covariance matrices of users $\Omega_p = \{1, \ldots, K_p\}$ that transmit pilot sequence p, namely

$$\mathbf{\Xi}_p = [\boldsymbol{c}_{\boldsymbol{h}_1}, \ldots, \boldsymbol{c}_{\boldsymbol{h}_{K_p}}] \tag{3.130}$$

to write

$$\mathbf{\Gamma}_p^{\mathrm{RMF}} = \mathbf{\Xi}_p^{\mathrm{H}}(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}}^{-1})\mathbf{\Xi}_p. \tag{3.131}$$

Analogously to the LMMSE filter (3.20), we have optimal transformations

$$\tilde{\boldsymbol{b}}_k^\star = \frac{1}{M\sqrt{p_k}}(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}}^{-1})\mathbf{\Xi}\big(\boldsymbol{P}^{-1}/M + \mathbf{\Gamma}^{\mathrm{RMF}}\big)^{-1}\boldsymbol{e}_k \tag{3.132}$$

which are scaled versions of $\boldsymbol{b}_k^\star$ in (3.118).

For the RMF, the asymptotic SINR is similar to the one in (3.33). Note that since $\boldsymbol{C}_{\boldsymbol{h}_k} \succeq \boldsymbol{C}_{\boldsymbol{h}_k|\varphi}$ also $\boldsymbol{C}_{\boldsymbol{y}} \succeq \boldsymbol{C}_{\boldsymbol{y}|\varphi}$ and thus $\mathbf{\Gamma}^{\mathrm{RMF}} \preceq \mathbf{\Gamma}$. We have

$$\gamma_k^{\mathrm{asy}} = \frac{Mp_k}{\boldsymbol{e}_k^{\mathrm{T}}\big(\mathbf{\Gamma}^{\mathrm{RMF}}\big)^{-1}\boldsymbol{e}_k}. \tag{3.133}$$

Because $\mathbf{\Gamma}^{\mathrm{RMF}} \preceq \mathbf{\Gamma}$, the asymptotic scaling factor for the RMF could be significantly lower than for the LMMSE filter. However, we have the same conditions for linear scaling with M.

Theorem 3.9. *For observations $\boldsymbol{\varphi}_k$ as in* (3.23) *and channel covariance matrices that fulfill Conditions 3.2 and 3.1 we have $\gamma_k^\star \asymp \gamma_k^{asy}$ and*

$$\liminf_{M\to\infty} \gamma_k^\star/M > 0. \tag{3.134}$$

Proof. As for Theorem 3.4, we need

$$\limsup_{M\to\infty} \left\| \left(\mathbf{\Gamma}^{\mathrm{RMF}}\right)^{-1} \right\|_2 < \infty \tag{3.135}$$

which follows from the same steps as in Theorem 3.5. □

Since the SINRs scale linearly with M, RMF achieves the full DoMs. For a finite number of antennas we expect the LMMSE filter to perform better. As so often, we have a trade-off between performance and complexity.

The optimal transformations $\boldsymbol{B}_k^\star$ only depend on the channel statistics. Therefore, they do not have to be recalculated in each channel coherence interval, but in each coherence interval of the covariance matrices. The complexity of calculating the transformations $\boldsymbol{B}_k^\star$ is dominated by the calculation of the matrix $\mathbf{\Gamma}^{\mathrm{RMF}}$. An efficient way to calculate $\mathbf{\Gamma}_p^{\mathrm{RMF}}$ is to first calculate $\boldsymbol{C}_{\boldsymbol{y}}^{-1}\boldsymbol{C}_{\boldsymbol{h}_k}\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1}$ for all $k \in \Omega_p$ and then calculate the inner products with all $\boldsymbol{C}_{\boldsymbol{h}_n}$ where $n \in \Omega_p$. This procedure leads to a complexity of $\mathrm{O}(M^3K)$ floating point operations.

Given the $\boldsymbol{B}_k^\star$ the calculation of the RMFs $\boldsymbol{g}_k = \boldsymbol{B}_k^\star\boldsymbol{\varphi}_p$ is one matrix-vector multiplication per user. For full covariance matrices, the complexity of the filter calculations is thus $\mathrm{O}(M^2K)$.

We can use the assumptions in Section 3.3 to reduce the complexity. We notice that the vectorized optimal transformations $\boldsymbol{b}_k^\star$ in (3.132) are linear combinations of vectors $(\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-\mathrm{T}} \otimes \boldsymbol{C}_{\boldsymbol{y}}^{-1})\boldsymbol{c}_{\boldsymbol{h}_n}$. Reverting the vectorization we see that

$$\boldsymbol{B}_k^\star = \boldsymbol{C}_{\boldsymbol{y}}^{-1}\left(\sum_{\ell\in\Omega_k}\sigma_{k\ell}\boldsymbol{C}_{\boldsymbol{h}_\ell}\right)\boldsymbol{C}_{\boldsymbol{\varphi}_p}^{-1} \tag{3.136}$$

for some σ_ℓ. For the following discussion, we assume $\boldsymbol{C}_{\boldsymbol{v}} = \mathbf{I}$ and thus, we have

$$\boldsymbol{C}_{\boldsymbol{y}} = \mathbf{I} + \sum_n p_k\boldsymbol{C}_{\boldsymbol{h}_k}, \tag{3.137}$$

and

$$\boldsymbol{C}_{\boldsymbol{\varphi}_p} = \frac{1}{\rho_{\mathrm{tr}}}\mathbf{I} + \sum_{k\in\Omega_p}\boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.138}$$

Taking a close look at (3.136), we realize that certain structure of the covariance matrices carries over to the transformations $\boldsymbol{B}_k^\star$. An important example are covariance matrices that fulfill Assumption 3.1, i.e., covariance matrices that share the same eigenbasis:

$$C_{h_k} = Q \operatorname{diag}(c_{h_k}) Q^{\mathrm{H}} \;\; \forall k \tag{3.139}$$

for some unitary Q. We can easily verify that the matrices C_y and C_{φ_p} and thus the optimal transformations have the same eigenbasis as well, i.e. we have

$$B_k^\star = Q \operatorname{diag}(b_k^\star) Q^{\mathrm{H}}. \tag{3.140}$$

As mentioned before, if the channel covariance matrices have the desired structure, we can simply transform the incoming signals y and φ_k by Q^{H}. Then we no longer have to think about the eigenbasis Q and can work with the diagonal covariance matrices, which is exactly what we will do in the following.

If the covariance matrices are diagonal, the matrices $C_y = \operatorname{diag}(\hat{c}_y)$ and $C_{\varphi_p} = \operatorname{diag}(\hat{c}_{\varphi_p})$ are diagonal as well. The operation which dominates the complexity of calculating the transformations $B_k^\star$ is still the calculation of the matrix Γ^{RMF}. For full matrices, all entries could be calculated in $\mathrm{O}(M^3K)$ floating point operations. For diagonal covariance matrices, the computational complexity reduces to $\mathrm{O}(MK^2)$.

The complexity to calculate all linear filters $g_k = B_k^\star \varphi_p$ reduces to $\mathrm{O}(MK)$ which is significantly lower than the complexity of the LMMSE filter in (3.14) which is $\mathrm{O}(MK^2)$ for diagonal covariance matrices.

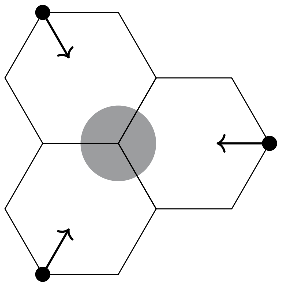


Figure 3.2: Small network with three hexagonal cells. The base stations are positioned at the corners and the users are uniformly distributed in the shaded circular area in the center.

3.9 Simulation Results

We want to present some simulation results of the methods discussed in this chapter. We will show simulations for two setups. One multi-cell setup to demonstrate the asymptotic behaviour of the methods that we analyzed in this chapter and one single-cell scenario to illustrate our approach for resource allocation based on asymptotic equivalent SINRs.

First we consider the multi-cell uplink scenario where we have three base stations as depicted in Fig. 3.2. Each base station employs a large ULA to receceive signals from the users. The users are uniformly distributed in the shaded circular area in the center of the network, whose radius is half of the cell radius. We analyze the performance in the uplink, when all users transmit with the same power, $p_k = \rho_{\mathrm{ul}}$

The smallest possible distance of a user to the serving base station is thus three quarters of the distance to the center of the network.

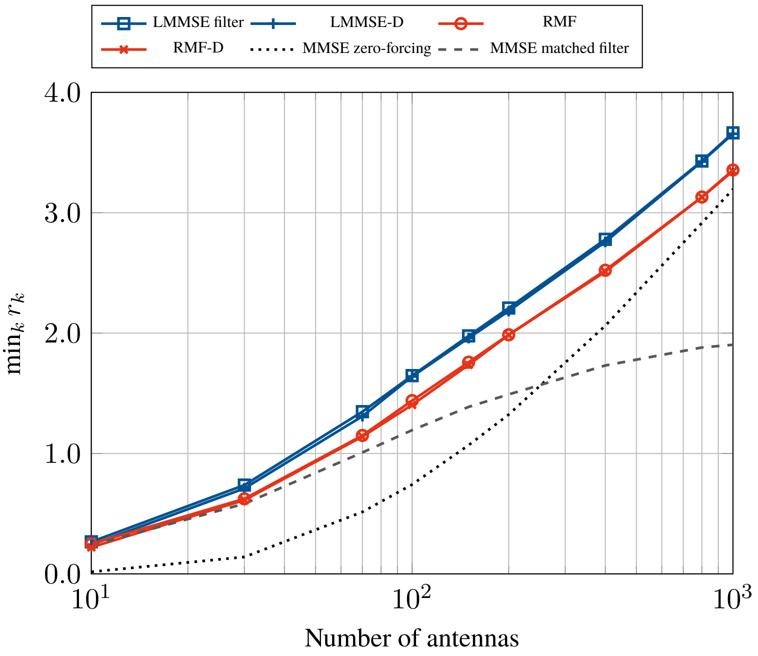


Figure 3.3: Average achievable rate of the worst user in the cell. Results are for the multi-cell scenario as depicted in Fig. 3.2, with 5 users per cell and the same number of orthogonal training sequences. The cell-edge SNR is -6dB.

for all k. The number of simultaneously served users in each cell is exactly the number of available pilot sequences, i.e., we have a full pilot-reuse.

We use the spatial channel model from the 3GPP report in [37] to generate the covariance matrices, but without any shadow-fading. The channel covariance matrices are normalized such that $\mathrm{tr}(\boldsymbol{C}_k)/M = 1$ for a user exactly in the center of the network (the base stations are positioned at the edges). Thus, the value ρ_{ul} is the worst possible SNR for a user to the serving base station. The pathloss coefficient is set to 3.76.

The largest possible SNR of a user in the network is consequently $(4/3)^{3.76}\rho_{\mathrm{ul}} \approx 2.95\rho_{\mathrm{ul}}$.

Our analysis shows that for the LMMSE filter as well as for the RMF, the SINRs of all users go to infinity as long as the covariance matrices are linearly independent. To demonstrate this result in our setup, we depict the achievable rate with respect to the number of antennas in Fig. 3.3. Since the result has to hold for all users, we depict the achievable rate of the worst user in the network. As a baseline we show results for the matched filter and the zero-forcing filter based on MMSE channel estimates, i.e., we use filters $\boldsymbol{G} = \hat{\boldsymbol{H}}$ and $\boldsymbol{G} = \hat{\boldsymbol{H}}(\hat{\boldsymbol{H}}^{\mathrm{H}}\hat{\boldsymbol{H}})^{-1}$ respectively.

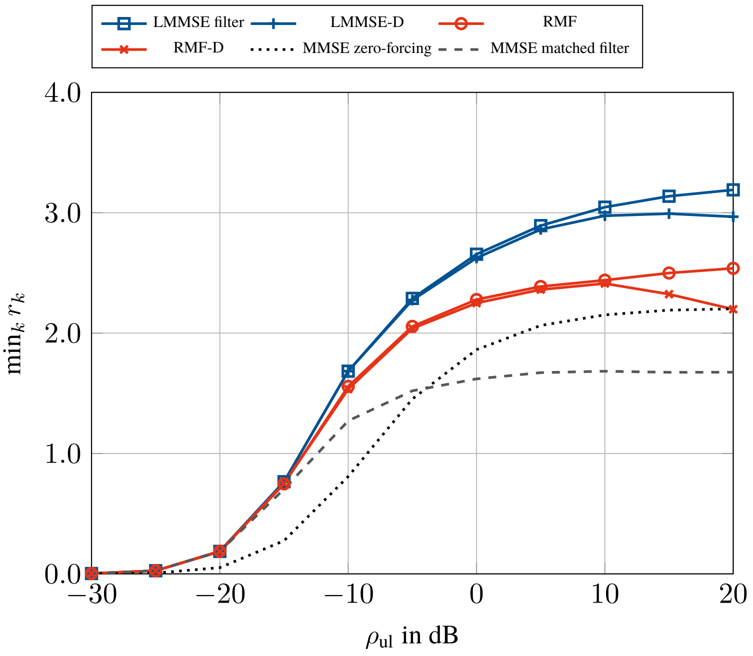


Figure 3.4: Average achievable rate of the worst user in the cell. Results are for the multi-cell scenario as depicted in Fig. 3.2, with 5 users per cell and the same number of orthogonal training sequences. The number of antennas at the base stations is $M = 200$.

We also consider the case where only the diagonals of the covariance matrices are known.[2] We use the diagonal covariance matrices to calculate (subotimal) LMMSE and RMF filters, denoted as LMMSE-D and RMF-D. As we can see, the effect on the achievable rate is negligible. Thus, there is no reason to use full covariance matrices in this setup.

In Fig. 3.4 we show results with respect to the cell-edge SNR ρ_{ul}. We see that the achievable rates saturate for high SNR. That is – at least for our bound on the mutual information – pilot-contamination limits the DoFs even though it does not limit the DoMs. Interestingly, the saturation point for zero-forcing is lower than that of our RMF approach.

Now let us switch to the second scenario to analyze the effectiveness of our resource allocation methods. We consider a single hexagonal cell with the base station in one corner. The users are uniformly distributed in the cell with a minimum distance from the base station. All other parameters are the same as for the multi-cell scenario.

We first focus on the different methods that rely on observations from an uplink training phase. In Fig. 3.5 we see the average rate of

[2] Here we mean the diagonals of the covariance matrices after transformation in a suitable space. Since we assume ULAs at the base stations, all incoming signals are transformed into the array space via FFTs.

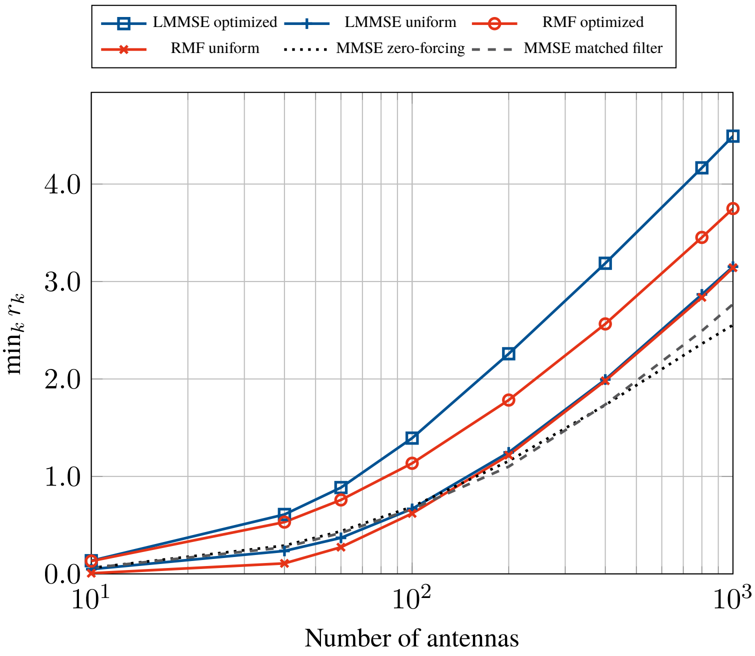


Figure 3.5: Average achievable rate of the worst user in the cell. Results are for a single-cell scenario with $K = 20$ users per cell and $T_{\text{tr}} = 10$ orthogonal training sequences. The cell-edge SNR is at -6dB. For the LMMSE and the RMF filter we depict results with uniform power allocation and with optimized power allocation.

the worst user in the cell with respect to the number of base-station antennas. We show results for the LMMSE and the RMF filter both with an optimized dual-uplink power allocation and with a fixed uniform allocation of the uplink powers. For the LMMSE we use the asymptotically equivalent SINR expression (3.33) to optimize the static power allocation. Clearly, the results with optimized allocation for maximum minimal rate are far superior to the approaches without optimized allocation.

In Fig. 3.6 we analyze the performance with respect to the number of users in the cell. We keep the number of orthogonal training sequences fixed, thus, adding more users adds more interference during the training. We have to normalize the per-user rates to get a fair comparison of the achievable rate of the worst user for different numbers of users. Serving more users simultaneously does not increase the achievable rate per user, but it allows to schedule the same user more frequently. Say, for example, we have 20 active users in the cell. If we serve all 20 users simultaneously instead of only 10 users, we can schedule each user on twice as many channel accesses. Consequently, we normalize the achievable rate of the worst user by a factor which

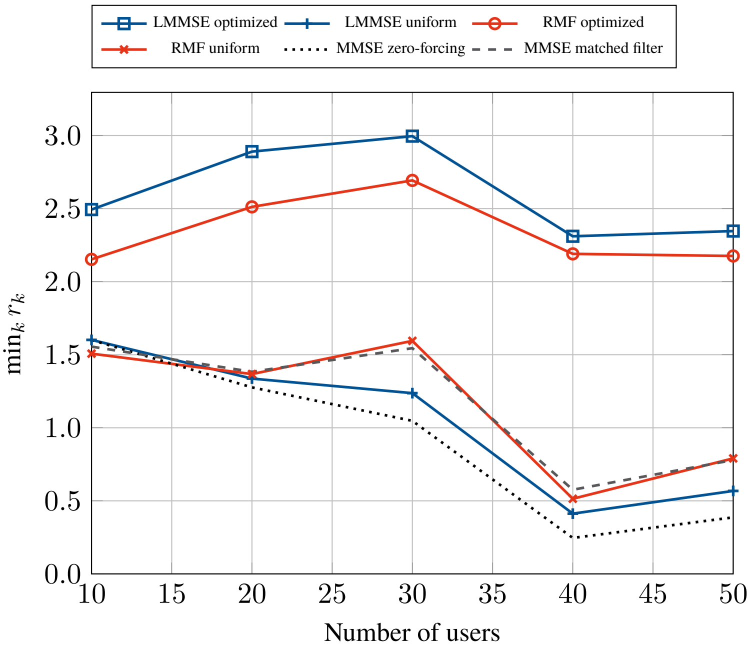


Figure 3.6: Average achievable rate of the worst user in the cell. Results are for a single-cell scenario with $M = 200$ antennas at the base-station and $T_{\text{tr}} = 10$ orthogonal training sequences. The cell-edge SNR is at -6dB. For the LMMSE and the RMF filter we depict results with uniform power allocation and with optimized power allocation.

is proportional to the number of simultaneously served users. Specifically, we multiply the achievable rate by K/T_{tr}, since T_{tr} is the minimal number of users that we use in the simulations.

Since some of the LMMSE and RMF methods are in principle able to deal with pilot contamination, the normalized achievable rate does increase with growing numbers of users for the optimized methods. The gains also depend on the pilot allocation and could, e.g., be improved by approximately solving the NUM based on the asymptotic results with a greedy algorithm (cf. [38, 21]). Methods that serve a flexible number of users per channel access could lead to even larger gains.

For the system with downlink training we use the same setup as for the uplink training. We first want to verify the effectiveness of the pilot design proposed in Theorem 3.7. To this end we fix the training power ρ_{tr} and vary the number of channel accesses T_{tr} used for training. We compare different methods for the power allocation. We denote the method which distributes power uniformly and only on certain dimensions as the "simple training power" method. Namely, we chose the dimensions where the channel covariance matrices have large eigenvalues. For this simulation, we choose the dimensions such that for each user at least 90% of the channel power is captured.

The results are shown in Fig. 3.7. We note that the achievable minimal rates saturate for growing T_{tr} at a value much smaller than the number of antennas at the base station. We observe a significant gain for optimized user power allocation and another similar gain if the training power allocation is also optimized.

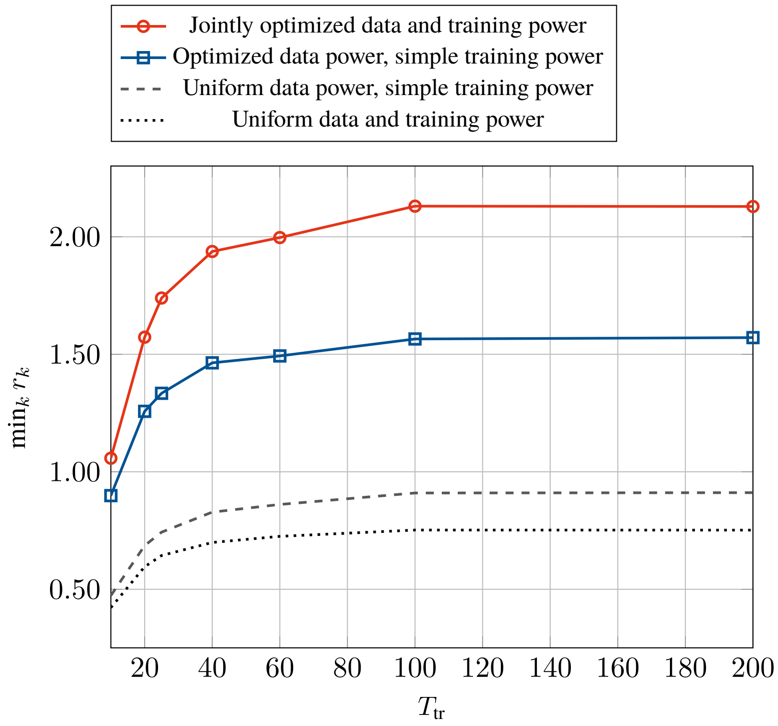


Figure 3.7: Average achievable rate of the worst user in the cell with respect to T_{tr} for different training power allocations. We depict a scenario with $M = 200$ antennas and $K = 20$ users. Due to the sparsity of the covariance matrices, the sum-rates saturate at a T_{tr} much smaller than the number of antennas.

The performance with respect to the number of users is shown in Fig. 3.8. As for the uplink training, we show the normalized achievable rate of the worst user. Note that there is no interference during the downlink training and thus increasing the number of users does not lead to pilot-contamination. However, for a higher number of users, the training power has to be spread in more spatial directions and the general inter-user interference increases due to the linear beamforming.Thus, the performance peaks at a number of users much lower than the number of transmit antennas.

3.10 Summary

We modelled imperfect channel state information by some side information φ which could, e.g., be observations from a training phase.

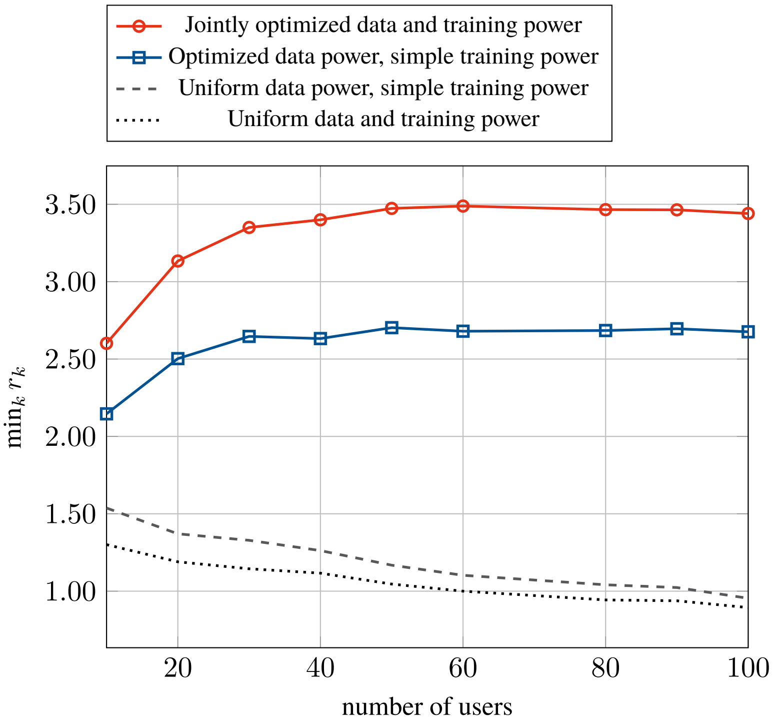


Figure 3.8: Average achievable rate of the worst user in the cell with respect to the number of users. We depict a scenario with $M = 200$ antennas and $T_{\text{tr}} = 20$ channel accesses for pilot transmission.

From the side information φ we calculated the MMSE estimates

$$\hat{\boldsymbol{h}}_k = \mathrm{E}[\boldsymbol{h}_k|\varphi] \tag{3.141}$$

and the corresponding estimation error covariance matrices

$$\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} = \mathrm{E}[\boldsymbol{h}_k\boldsymbol{h}_k^{\mathrm{H}}|\varphi] - \hat{\boldsymbol{h}}_k\hat{\boldsymbol{h}}_k^{\mathrm{H}}. \tag{3.142}$$

The achievable uplink rate given this side information is lower bounded by

$$r_k^{\varphi} = \log_2(1+\gamma_k^{\varphi}) \tag{3.143}$$

with

$$\gamma_k^{\varphi} = \frac{p_k\left|\boldsymbol{g}_k^{\mathrm{H}}\hat{\boldsymbol{h}}_k\right|^2}{\boldsymbol{g}_k^{\mathrm{H}}\left(\boldsymbol{C}_{\boldsymbol{v}} + p_k\boldsymbol{C}_{\boldsymbol{h}_k|\varphi} + \sum_{n\neq k} p_n(\boldsymbol{C}_{\boldsymbol{h}_n|\varphi} + \hat{\boldsymbol{h}}_n\hat{\boldsymbol{h}}_n^{\mathrm{H}})\right)\boldsymbol{g}_k}. \tag{3.144}$$

The ergodic rates are thus lower bounded by $\bar{r}_k \geq \mathrm{E}[r_k^{\varphi}]$.

For a non-trivial power allocation $\boldsymbol{p} > 0$, the optimal uplink SINR of user k is given by

$$\gamma_k^{\star} = p_k\frac{\boldsymbol{e}_k^{\mathrm{T}}\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}}(\boldsymbol{P}^{-1}+\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}})^{-1}\boldsymbol{e}_k}{\boldsymbol{e}_k^{\mathrm{T}}(\boldsymbol{P}^{-1}+\hat{\boldsymbol{H}}^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1}\hat{\boldsymbol{H}})^{-1}\boldsymbol{e}_k} \tag{3.145}$$

where $\boldsymbol{P} = \operatorname{diag}(\boldsymbol{p})$ and $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{h}}_1, \dots, \hat{\boldsymbol{h}}_K]$ and

$$\boldsymbol{C}_{\boldsymbol{y}|\varphi} = \boldsymbol{C}_{\boldsymbol{v}} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k|\varphi}. \tag{3.146}$$

Optimal LMMSE filters can be calculated as

$$\tilde{\boldsymbol{g}}_k^\star = \frac{1}{\sqrt{p_k}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} (\boldsymbol{P}^{-1} + \hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}})^{-1} \boldsymbol{e}_k. \tag{3.147}$$

We are still able to use uplink-downlink duality to design downlink beamforming vectors. However, the resulting downlink rates are only guaranteed to be achievable if the receiver has the same side information as the transmitter.

A common model for massive MIMO systems is that the side information is acquired in an uplink training phase. That is, we have a training phase where each user transmits one of T_{tr} orthogonal pilot sequences, resulting in observations

$$\boldsymbol{\varphi}_p = \sum_{k \in \Omega_p} \boldsymbol{h}_k + \frac{1}{\rho_{\text{tr}}} \boldsymbol{v}^{\text{tr}} \tag{3.148}$$

where Ω_p is the set of users that transmit pilot sequence p, with $p = 1$, …, T_{tr} and $\boldsymbol{v}^{\text{tr}} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{I})$ is additive noise. For some conditions on the covariance matrices (Conditions 2.2, 3.1 and 3.2), we get an asymptotically equivalent SINR

$$\gamma_k^{\text{asy}} = \frac{M p_k}{\boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{\Gamma}^{-1} \boldsymbol{e}_k} \tag{3.149}$$

where $\boldsymbol{\Gamma} = \mathrm{E}[\hat{\boldsymbol{H}}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{y}|\varphi}^{-1} \hat{\boldsymbol{H}} / M] \in \mathbb{C}^{K \times K}$. We have

$$\lim_{M \to \infty} \gamma_k^{\text{asy}} / M > 0 \tag{3.150}$$

that is, the SINR grows linearly with the number of antennas.

Since the asymptotic SINR only depends on channel statistics, it is useful to find close to optimal power and pilot allocations that do not depend on the instantaneous side information.

We saw that for several common channel models, we can work with diagonal covariance matrices. For diagonal covariance matrices, the LMMSE filters can be calculated in $\mathrm{O}(MK^2 + K^3)$ floating-point operations.

We were able to improve the performance by considering both, the observations from the training phase and the received uplink data, as side information. In this case, the MMSE estimates for the channel vectors could no longer be given in closed form. With the iterative variational inference approach we could obtain approximations. The complexity is $\mathrm{O}(MK^2 + K^3)$ per iteration.

Since low complexity is important in large-scale systems, we introduced a robust matched filter (RMF)

$$\boldsymbol{g}_k = \boldsymbol{B}_k^\star \boldsymbol{\varphi}_p \tag{3.151}$$

with a deterministic matrix $\boldsymbol{B}_k^\star$ that only depends on the channel statistics. For diagonal covariance matrices, the $\boldsymbol{B}_k^\star$ are also diagonal and thus all filters can be calculated with $\mathrm{O}(MK)$ operations. We showed that the SINR of the RMF also grows linearly with the number of antennas and for low SNR, the performance is similar to the LMMSE filter.

In an FDD system we cannot rely on uplink training to estimate the downlink channels. We send downlink pilot vectors $\boldsymbol{b}_p$ and get observations

$$\boldsymbol{\varphi}_k = \boldsymbol{B}^{\mathrm{H}} \boldsymbol{h}_k + \boldsymbol{v}_k \tag{3.152}$$

where $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{T_{\mathrm{tr}}}] \in \mathbb{C}^{M \times T_{\mathrm{tr}}}$ and $\boldsymbol{v}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{I})$. We assume perfect feedback of the observations to the base station.

With the downlink observations φ^{dl} we obtained MMSE estimates

$$\hat{\boldsymbol{h}}_k = \mathrm{E}[\boldsymbol{h}_k | \varphi^{\mathrm{dl}}] = \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{B} \left(\boldsymbol{B}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{B} + \mathbf{I} \right)^{-1} \boldsymbol{\varphi}_k \tag{3.153}$$

and conditional covariance matrices

$$\boldsymbol{C}_{\boldsymbol{h}_k | \varphi^{\mathrm{dl}}} = \boldsymbol{C}_{\boldsymbol{h}_k} - \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{B} \left(\boldsymbol{B}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{h}_k} \boldsymbol{B} + \mathbf{I} \right)^{-1} \boldsymbol{B}^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{h}_k}. \tag{3.154}$$

For low-rank diagonal covariance matrices, with consecutive non-zero entries on the diagonal, the pilot design

$$\boldsymbol{B} = \boldsymbol{D}^{1/2} \boldsymbol{S} \in \mathbb{R}^{M \times T_{\mathrm{tr}}} \tag{3.155}$$

with $\boldsymbol{S} = [\mathbf{I}, \ldots, \mathbf{I}]^{\mathrm{T}}$ and $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{d})$, $\boldsymbol{d} \in \mathbb{R}^M$, is optimal.

The power allocations for pilot transmission $\boldsymbol{d}$ and for data transmission $\boldsymbol{p}$ can be jointly optimized using the asymptotically equivalent SINR

$$\gamma_k^{\mathrm{asy}} = p_k \,\mathrm{tr} \left(\left(\boldsymbol{D} \boldsymbol{C}_{\boldsymbol{h}_k} + \mathbf{I} \right)^{-1} \boldsymbol{D} \boldsymbol{C}_{\boldsymbol{h}_k}^2 \boldsymbol{C}_{\boldsymbol{y} | \varphi^{\mathrm{dl}}}^{-1} \right) \tag{3.156}$$

with $\boldsymbol{C}_{\boldsymbol{y}|\varphi^{\mathrm{dl}}} = \mathbf{I} + \sum_k p_k \boldsymbol{C}_{\boldsymbol{h}_k | \varphi^{\mathrm{dl}}}$.

For diagonal covariance matrices and our pilot design, the complexity of calculating the LMMSE filter is the same as for uplink observations.

Chapter 4

Imperfect Covariance Matrix Information

In the previous chapter we saw that with the help of covariance matrix information, we are able to achieve asymptotically optimal rates even in the presence of pilot-contamination. So naturally, the acquisition of the channel covariance matrices is an important topic for the considered large-scale communication systems. In principle, we exploit that the coherence interval of the covariance matrices is much longer than that of the fast-fading channel.

The classical approach would be a two step procedure. First, estimate the covariance matrices based on uplink observations from multiple channel coherence intervals. Then, assuming the estimated covariance matrices are exact, calculate the MMSE channel estimates and error covariance matrices required for the transceiver design.

Alternatively, we can directly formulate the MMSE estimator for the channel with unknown covariance matrices. In this case, we need a model for the prior distribution of the covariance matrices.

In the following we first discuss the classical maximum likelihood (ML) estimation for interference-free observations. We continue with the MMSE estimation for a known prior of the covariance matrices. As we will see, the resulting estimator is too complex for practical application. Thus, we exploit structure of the channel model to reduce the complexity and we use a learning approach to close the gap to the complex MMSE estimator.

We go on to discuss the more difficult case with pilot-contamination. We introduce a novel approximate ML estimator that is able to deal with interference in the observations. As of now, the extension of the learning-based approximate MMSE estimator to the case with interference is still open.

4.1 Interference Free Observations

Since the covariance matrices are quasi-static over many channel coherence intervals, we can use additional pilots to get interference-free observations. These observations can then be used to estimate the channel covariance matrices. Such a scheme is, for example, discussed in [39].

For example, if we spend one additional channel access per channel coherence interval for covariance matrix estimation and the covariance matrices are quasi-static for 700 channel coherence intervals, we get 700 channel accesses to estimate the covariance matrices of all relevant users. If we serve 10 users in our cell and 60 additional users from the neighboring cells create significant interference ($K = 70$ relevant users), then we could use 10 channel accesses to gather interference free channel estimates for each of those users, which we then use for covariance matrix estimation.

We need nK extra pilots in the coherence interval of the covariance matrix to get n interference-free observations per user. These observations are of the form

$$\boldsymbol{\varphi}_{ki} = \boldsymbol{h}_{ki} + \boldsymbol{v}_{ki}^{\mathrm{tr}} \quad i = 1, \dots, n. \tag{4.1}$$

If we have independent observations for each user we can focus on a single user and drop the user index for notational convenience to get

$$\boldsymbol{\varphi}_{i} = \boldsymbol{h}_{i} + \boldsymbol{v}_{i}^{\mathrm{tr}} \quad i = 1, \dots, n. \tag{4.2}$$

The observations for different i are independent and identically distributed with covariance matrices

$$\boldsymbol{C}_{\boldsymbol{\varphi}} = \boldsymbol{C}_{\boldsymbol{h}} + \boldsymbol{C}_{\boldsymbol{v}}. \tag{4.3}$$

A popular approach for parameter estimation with more complex models is the maximum likelihood (ML) estimator. Given n different independent observations $\boldsymbol{\varphi}_i$ and known noise covariance matrix $\boldsymbol{C}_{\boldsymbol{v}}$, the log-likelihood function of the covariance matrix $\boldsymbol{C}_{\boldsymbol{\varphi}}$ is given by

$$\log(L(\boldsymbol{C}_{\boldsymbol{\varphi}}|\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_n)) = -\sum_i \boldsymbol{\varphi}_i^{\mathrm{H}} \boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} \boldsymbol{\varphi}_i - n \log\det(\boldsymbol{C}_{\boldsymbol{\varphi}}). \tag{4.4}$$

The well-known solution to the ML problem

$$\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} = \underset{\boldsymbol{C}_{\boldsymbol{\varphi}}}{\operatorname{argmax}} \log\left(L(\boldsymbol{C}_{\boldsymbol{\varphi}}|\boldsymbol{\varphi}_i, \dots, \boldsymbol{\varphi}_n)\right) \tag{4.5}$$

is given by

$$\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} = \frac{1}{n} \sum_i \boldsymbol{\varphi}_i \boldsymbol{\varphi}_i^{\mathrm{H}} \tag{4.6}$$

which is called the sample covariance matrix.

ML estimation of the channel covariance matrix is slightly more involved. The log-likelihood is given by

$$\begin{aligned}&\log\left(L(\boldsymbol{C_h}|\boldsymbol{\varphi}_1,\ldots,\boldsymbol{\varphi}_n)\right)\\&\quad=-\sum_i \boldsymbol{\varphi}_i^{\mathrm{H}}\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)^{-1}\boldsymbol{\varphi}_i-n\log\det\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)\\&\qquad\propto-\operatorname{tr}\left(\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)^{-1}\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}\right)-\log\det\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)\end{aligned}\tag{4.7}$$

where $f\propto g$ denotes $f=\alpha g$ with a constant α. The derivative is given by

$$\frac{dL}{d\boldsymbol{C_h}}=\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)^{-1}\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)^{-1}-\left(\boldsymbol{C_h}+\boldsymbol{C_v}\right)^{-1}.\tag{4.8}$$

Setting the derivative to zero yields the intuitive result

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}}=\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}-\boldsymbol{C_v}.\tag{4.9}$$

The issue is that the resulting estimate $\widehat{\boldsymbol{C}}_{\boldsymbol{h}_k}$ might not be positive semi-definite. For a correct ML estimate we need to restrict the set of feasible variables to the cone of positive semi-definite matrices. With the constraint $\boldsymbol{C_h}\succeq\mathbf{0}$ we get [40]

Theorem 4.1. *The solution to*

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}}=\underset{\boldsymbol{C_h}\succeq\mathbf{0}}{\operatorname{argmax}}\log\left(L(\boldsymbol{C_h}|\boldsymbol{\varphi}_i,\ldots,\boldsymbol{\varphi}_n)\right)\tag{4.10}$$

is given by

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}}=\boldsymbol{C}_{\boldsymbol{v}}^{1/2}P_{\mathbb{S}}\Big(\boldsymbol{C}_{\boldsymbol{v}}^{-1/2}\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}\boldsymbol{C}_{\boldsymbol{v}}^{-1/2}-\mathbf{I}\Big)\boldsymbol{C}_{\boldsymbol{v}}^{1/2}\tag{4.11}$$

where $P_{\mathbb{S}}(\cdot)$ denotes the orthogonal projection onto the cone of positive semi-definite matrices.

Proof. Given the decomposition $\boldsymbol{C_v}=\boldsymbol{A}\boldsymbol{A}^{\mathrm{H}}$ and the transformations $\boldsymbol{R_h}=\boldsymbol{A}^{-1}\boldsymbol{C_h}(\boldsymbol{A}^{-1})^{\mathrm{H}}$ and $\hat{\boldsymbol{R}}_{\boldsymbol{\varphi}}=\boldsymbol{A}^{-1}\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}(\boldsymbol{A}^{-1})^{\mathrm{H}}$, we formulate the equivalent optimization problem

$$\boldsymbol{R}_{\boldsymbol{h}}^{\star}=\underset{\boldsymbol{R_h}\succeq\mathbf{0}}{\operatorname{argmin}}\operatorname{tr}(\hat{\boldsymbol{R}}_{\boldsymbol{\varphi}}(\boldsymbol{R_h}+\mathbf{I})^{-1})+\log\det(\boldsymbol{R_h}+\mathbf{I}).\tag{4.12}$$

After substituting $\boldsymbol{T}=\boldsymbol{R_h}+\mathbf{I}\in\mathbb{C}^{M\times M}$ we get

$$\boldsymbol{T}^{\star}=\underset{\boldsymbol{T}\succeq\mathbf{I}}{\operatorname{argmin}}\operatorname{tr}(\hat{\boldsymbol{R}}_{\boldsymbol{\varphi}}\boldsymbol{T}^{-1})+\log\det\boldsymbol{T}.\tag{4.13}$$

Incorporating the eigenvalue decomposition $\boldsymbol{T} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\mathrm{H}}$, with $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_M)$, yields

$$\boldsymbol{T}^{\star} = \underset{\substack{\lambda_i \geq 1, \forall i=1,\ldots,M \\ \boldsymbol{V} \text{ with } \boldsymbol{V}\boldsymbol{V}^{\mathrm{H}}=\mathbf{I}}}{\operatorname{argmin}} \ \mathrm{tr}(\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1}) + \sum_{i=1}^{M} \log \lambda_i. \tag{4.14}$$

Note that the constraints for the eigenvectors $\boldsymbol{V}$ and the eigenvalues λ_i are decoupled. We use the following lemma to determine an optimal choice for $\boldsymbol{V}$.

Lemma 4.1. *The matrix of eigenvectors $\boldsymbol{W}$ of the eigendecomposition $\hat{\boldsymbol{R}}_{\varphi} = \boldsymbol{W}\boldsymbol{\Xi}\boldsymbol{W}^{\mathrm{H}}$ is an optimizer of*

$$\min_{\boldsymbol{V}} \mathrm{tr}(\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1}) \text{ s.t. } \boldsymbol{V}^{\mathrm{H}}\boldsymbol{V} = \mathbf{I} \tag{4.15}$$

Proof. The Lagrangian of (4.15) is given by

$$L(\boldsymbol{V}, \boldsymbol{\Phi}) = \mathrm{tr}(\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1}) + \mathrm{tr}(\boldsymbol{\Phi}(\boldsymbol{V}^{\mathrm{H}}\boldsymbol{V} - \mathbf{I})) \tag{4.16}$$

where $\boldsymbol{\Phi}$ is the Hermitian Lagrangian multiplier. Differentiation with respect to $\boldsymbol{V}^*$ leads to

$$\frac{\partial L(\boldsymbol{V}, \boldsymbol{\Phi})}{\partial \boldsymbol{V}^*} = \hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1} + \boldsymbol{V}\boldsymbol{\Phi} = \mathbf{0} \tag{4.17}$$

from which follows that

$$\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1} + \boldsymbol{\Phi} = \mathbf{0}. \tag{4.18}$$

Consequently,

$$\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V}\boldsymbol{\Lambda}^{-1} = \boldsymbol{\Lambda}^{-1}\boldsymbol{V}^{\mathrm{H}}\hat{\boldsymbol{R}}_{\varphi}\boldsymbol{V} \tag{4.19}$$

since $\boldsymbol{\Phi}$ has to be Hermitian. It can be inferred that the optimal eigenvectors $\boldsymbol{V}$ diagonalize $\hat{\boldsymbol{R}}_{\varphi}$ as long as all eigenvalues λ_i are distinct. If some of eigenvalues λ_i are identical, $\boldsymbol{V}^{\star} = \boldsymbol{W}$ is one possible optimizer. □

Using the eigenvalue decomposition $\hat{\boldsymbol{R}}_{\varphi} = \boldsymbol{W}\boldsymbol{\Xi}\boldsymbol{W}^{\mathrm{H}}$ and incorporating the optimizer $\boldsymbol{V}^{\star} = \boldsymbol{W}$ into (4.14) yields

$$\boldsymbol{\Lambda}^{\star} = \underset{\lambda_i \geq 1, \forall i=1,\ldots,M}{\operatorname{argmin}} \ \mathrm{tr}(\boldsymbol{\Sigma}\boldsymbol{\Lambda}^{-1}) + \sum_{i=1}^{M} \log \lambda_i. \tag{4.20}$$

With the eigenvalues ξ_i of $\hat{\boldsymbol{R}}_{\boldsymbol{\varphi}}$ we can also write the optimization as

$$\{\lambda_i^\star\}_{i=1}^M = \operatorname*{argmin}_{\lambda_i \geq 1, \forall i=1,\dots,M} \sum_{i=1}^M \frac{\xi_i}{\lambda_i} + \sum_{i=1}^M \log \lambda_i, \tag{4.21}$$

which is clearly decoupled in the eigenvalues λ_i. The unconstrained problem for each eigenvalue

$$\min_{\lambda_i} \frac{\xi_i}{\lambda_i} + \log \lambda_i \tag{4.22}$$

has a single stationary point at ξ_i and the optimal eigenvalues can be readily identified as

$$\lambda_i^\star = \max(\xi_i, 1). \tag{4.23}$$

Consequently, the eigenvalues of $\boldsymbol{R}_{\boldsymbol{h}}^\star = \boldsymbol{T}^\star - \mathbf{I}$ are given by

$$\max(\xi_i - 1, 0) \tag{4.24}$$

which can be expressed with the projection onto the cone of positive semidefinite matrices

$$\boldsymbol{R}_{\boldsymbol{h}}^\star = \mathrm{P}_{\mathbb{S}}\left(\hat{\boldsymbol{R}}_{\boldsymbol{\varphi}} - \mathbf{I}\right) = \mathrm{P}_{\mathbb{S}}\left(\boldsymbol{A}^{-1}\hat{\boldsymbol{C}}_{\boldsymbol{\varphi}}(\boldsymbol{A}^{\mathrm{H}})^{-1} - \mathbf{I}\right). \tag{4.25}$$

Finally, we have for the ML estimate of the covariance matrix of $\boldsymbol{h}$

$$\boldsymbol{C}_{\boldsymbol{h}}^{\mathrm{ML}} = \boldsymbol{A}\,\mathrm{P}_{\mathbb{S}}\left(\boldsymbol{A}^{-1}\hat{\boldsymbol{C}}_{\boldsymbol{\varphi}}(\boldsymbol{A}^{\mathrm{H}})^{-1} - \mathbf{I}\right)\boldsymbol{A}^{\mathrm{H}}. \tag{4.26}$$

□

If the noise covariance matrix is a scaled identity $\boldsymbol{C}_{\boldsymbol{v}} = \frac{1}{\rho_{\mathrm{tr}}}\mathbf{I}$, the solution simplifies to

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}} = \mathrm{P}_{\mathbb{S}}\left(\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} - \frac{1}{\rho_{\mathrm{tr}}}\mathbf{I}\right). \tag{4.27}$$

That is, we simply have to project the possibly indefinite estimate (4.9) onto the cone of positive semi-definite matrices. This projection is quite costly, since it requires an eigenvalue decomposition of the estimated covariance matrices $\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}}$.

Additional knowledge of structure of the covariance matrices helps to increase the estimation accuracy and reduce complexity. If we use Assumption 3.1, namely that the channel covariance matrices are diagonal, we get the following results (cf. also [41]).

Corollary 4.1. *For $\boldsymbol{C_v} = \frac{1}{\rho_{tr}}\mathbf{I}$ and with Assumption 3.1, the ML estimate for the covariance matrix is given by*

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}} = \operatorname{diag}(\hat{\boldsymbol{c}}_{\boldsymbol{h}}) \tag{4.28}$$

where

$$\hat{\boldsymbol{c}}_{\boldsymbol{h}} = \left[\hat{\boldsymbol{c}}_{\boldsymbol{\varphi}} - \frac{1}{\rho_{tr}}\mathbf{1}\right]_{+} \tag{4.29}$$

and

$$\hat{\boldsymbol{c}}_{\boldsymbol{\varphi}} = \frac{1}{n}\sum_i |\boldsymbol{\varphi}_i|^2 \,. \tag{4.30}$$

Proof. By Assumption 3.1 the covariance matrices are diagonal and thus the covariance matrices of the observations are also diagonal. In this case, the estimation problem decouples in separate variance estimation problems for each dimension. Corollary 4.1 follows directly from Theorem 4.1 applied to the M scalar problems. □

To reiterate yet another time, one important example of matrices which can be jointly diagonalized are circulant matrices. For the physical channel model that we consider for the channel covariance matrices, the covariance matrices are Toeplitz and not circulant. As mentioned before, the Toeplitz matrices can be approximated as circulant matrices with vanishing approximation error for a large number of antennas. Thus, performing all signal processing under the assumption of circulant matrices is reasonable from a practical stand-point. All calculations simplify significantly and for a large enough number of antennas the performance penalty is negligible as indicated by the numerical simulations in the previous chapter.

To quantify the performance penalty of the circulant assumption with estimated covariance matrices, we need to find ML estimates for general positive semi-definite Toeplitz matrices as reference. Unfortunately, there is no closed-form solution to this problem. We could apply a general optimization method to the ML problem to find a locally optimal solution. The complexity of the optimization depends on the parameterization of the Toeplitz matrices.

There is plenty of literature on the estimation of structured covariance matrices. There is well established work in the field [42, 43], which can be extended to the case of noisy observations. Some recent advances [44, 45] aim directly at ML estimation at low SNR and reveal an interesting connection between ML covariance matrix estimation and compressed sensing methods.

A common form of structure is a linear parameterization of the covariance matrices. That is, we know that the covariance matrices are of the form

$$\boldsymbol{C_h} = \sum_i \boldsymbol{S}_i \alpha_i \tag{4.31}$$

where the $\alpha_i \in \mathbb{R}$ are variables and the $\boldsymbol{S}_i$ are Hermitian but not necessarily positive-semidefinite matrices. Clearly, the $\boldsymbol{S}_i$ span a subspace of the real-valued vector space of Hermitian matrices. Of course, we can represent the full space of Hermitian matrices with M^2 basis vectors $\boldsymbol{S}_i$. Matrices with special structure, such as Toeplitz matrices, lie in a subspace with $O(M)$ dimensions.

If we incorporate the linear parameterization into the ML problem, the most challenging part is the constraint $\boldsymbol{C_h} \succeq \boldsymbol{0}$. Preferably, we would want a constraint directly on the variables α_i, which is not possible for most parameterizations. If we simply drop the positive-semidefiniteness constraint, we get the optimality conditions

$$\operatorname{tr}\left((\boldsymbol{C_h} + \boldsymbol{C_v})^{-1}(\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} - \boldsymbol{C_v} - \boldsymbol{C_h})(\boldsymbol{C_h} + \boldsymbol{C_v})^{-1}\boldsymbol{S}_i\right) = 0 \ \forall i. \tag{4.32}$$

We have $\partial L/\partial \alpha_i = \operatorname{tr}((\partial L/\partial \boldsymbol{C_h})(\partial \boldsymbol{C_h}/\partial \alpha_i))$. The derivative $\partial L/\partial \boldsymbol{C_h}$ is given in (4.8) and $\partial \boldsymbol{C_h}/\partial \alpha_i = \boldsymbol{S}_i$.

which we want to solve for the variables α_i. If we replace the inverses by fixed positive definite matrices $\boldsymbol{T}$, we get the linear system of equations (since $\boldsymbol{C_h}$ is linear in the α_i)

$$\sum_j \operatorname{tr}(\boldsymbol{S}_j \boldsymbol{T} \boldsymbol{S}_i \boldsymbol{T})\alpha_j = \operatorname{tr}(\boldsymbol{T}(\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} - \boldsymbol{C_v})\boldsymbol{T}\boldsymbol{S}_i) \ \forall i. \tag{4.33}$$

This suggests an iterative procedure where we use the current estimate of the covariance matrix to calculate $\boldsymbol{T} = (\widehat{\boldsymbol{C}}_{\boldsymbol{h}} + \boldsymbol{C_v})^{-1}$ which we then use to get a more accurate estimate. In fact, such a procedure has been shown to yield consistent, asymptotically efficient estimates of the covariance matrices even if only one iteration is performed [42].

For a small number of noisy observations, we have to force the estimate to be positive definite to get good performance. If all $\boldsymbol{S}_i$ are positive definite we can restrict the constraint set and only allow $\alpha_i \geq 0$, which enables us to use a projected gradient method.

Example. Circulant matrices can be represented as a linear combination of DFT basis vectors. For circulant matrices we have $\boldsymbol{S}_i = \boldsymbol{f}_i \boldsymbol{f}_i^{\mathrm{H}}$ with $i = 1, \ldots, M$, where $\boldsymbol{f}_i$ is the i-th DFT basis vector. In this case, clearly all α_i have to be non-negative to get a positive semi-definite estimate.

Toeplitz matrices can be represented similarly as combinations of $\boldsymbol{S}_i = \tilde{\boldsymbol{f}}_i \tilde{\boldsymbol{f}}_i^{\mathrm{H}}$, $i = 1, \ldots, 2M$, where $\tilde{\boldsymbol{f}}_i$ is now the i-th vector of a two times oversampled DFT matrix. In this case it is possible to have negative α_i and still have a positive-semidefinite combination. If we restrict the optimization to positive α_i we exclude some Toeplitz matrices. However, the constraint set is clearly larger than if we restrict the optimization to circulant matrices.

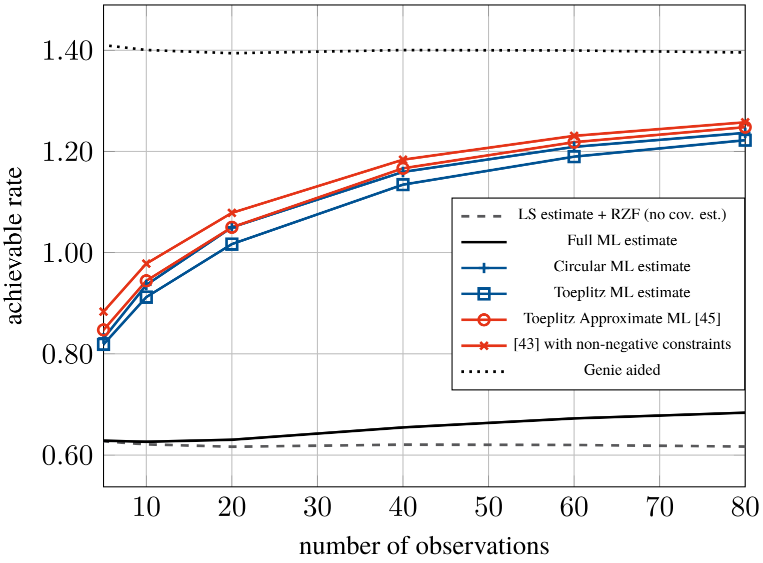


Figure 4.1: Achievable uplink rate of a single user, when the estimated channel covariance matrix is used to calculate the LMMSE filter. We assume perfect knowledge of the statistics at the decoder. We have $M = 64$ antennas at the base station and the effective training SNR is $\rho_{\mathrm{tr}} = -10$dB.

It is not obvious that restricting the optimization to circulant matrices will yield worse results than using a more general constraint set. In the simulations in Fig. 4.1 we see that in fact, for certain system parameters, the circulant approximation outperforms the projected gradient method based on Toeplitz structure. The low-complexity, approximate ML method from [45] actually outperforms both, even though it uses an approximate version of the likelihood function.

We also adapt the method from [43], which iteratively solves the equation system in (4.33). Instead of solving the equation system directly, we use a projected quasi-newton method (cf. Appendix D) to solve the equivalent least-squares problem with a non-negativity constraint on the coefficients α_i. Interestingly, this approach yields the best performance of the methods in Fig. 4.1. But none of the approaches is significantly better than the simple circulant ML estimate.

In the end, what we actually want is MMSE estimates of the channel vectors. As the results in Fig. 4.1 demonstrate, ML estimation of the covariance matrices is clearly a suboptimal approach and leads to inconsistent results regarding the desired performance criterion. In the following we discuss a more sophisticated Bayesian channel estimator that uses techniques from machine learning and outperforms the ML methods depicted in Fig. 4.1.

4.1.1 MMSE Channel Estimation

The techniques for covariance matrix estimation introduced in this chapter only partially exploit the underlying spatial channel model. For example, we can restrict the feasible set of the estimated covariance matrices to the set of positive semi-definite Toeplitz matrices if we know that the antennas at the base station form a uniform linear array. However, the ML based techniques do not allow us to incorporate general prior information in the estimators for the channel vectors and covariance matrices. In this chapter we explore a Bayesian setting that includes the spatial channel model in the channel estimation. In particular, we derive a MMSE estimator for a hierarchical channel model, where the channel covariance matrices depend on a set of hyperparameters which are themselves random. We derive a low-complexity estimator with free parameters, that can be trained to achieve close to optimal performance in terms of estimation error.

To derive the MMSE estimator we use the simple system model without pilot contamination. Extension to the case with interference during the training should be investigated in future work.

Remember the system model with n independent observations

$$\boldsymbol{\varphi}_i = \boldsymbol{h}_i + \boldsymbol{v}_i, \quad i = 1, \dots, n. \tag{4.34}$$

The channel vectors are Gaussian distributed given the hyperparameters $\boldsymbol{\delta}$. That is $\boldsymbol{h}_i | \boldsymbol{\delta} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\delta}})$ with the same $\boldsymbol{\delta}$ for all n observations. The hyperparameters are distributed with the probability density function $p(\boldsymbol{\delta})$ and there is a deterministic mapping from hyperparameters to covariance matrices. The channel model is explained in detail in Appendix B.

We know from earlier discussion, that the MMSE estimate of $\boldsymbol{h}_i$ from the observations given $\boldsymbol{\delta}$ evaluates to

$$\mathrm{E}[\boldsymbol{h}_i | \boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_n, \boldsymbol{\delta}] = \boldsymbol{W}_{\boldsymbol{\delta}} \boldsymbol{\varphi}_i \tag{4.35}$$

where

$$\boldsymbol{W}_{\boldsymbol{\delta}} = \boldsymbol{C}_{\boldsymbol{\delta}} \boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} = \boldsymbol{C}_{\boldsymbol{\delta}} \left(\boldsymbol{C}_{\boldsymbol{\delta}} + \boldsymbol{C}_{\boldsymbol{v}} \right)^{-1}. \tag{4.36}$$

In the following we will always assume i.i.d. noise coefficients, i.e., $\boldsymbol{C_v} = \sigma^2 \mathbf{I}$.

For unknown hyperparameters we can exploit the hierarchical channel model to rewrite

$$\hat{\boldsymbol{h}}_i = \mathrm{E}[\boldsymbol{h}_i|\varphi] = \mathrm{E}[\mathrm{E}[\boldsymbol{h}_i|\varphi,\boldsymbol{\delta}]] = \mathrm{E}[\boldsymbol{W_\delta}|\varphi]\boldsymbol{\varphi}_i = \widehat{\boldsymbol{W}}\boldsymbol{\varphi}_i. \tag{4.37}$$

That is, to calculate the MMSE estimate of the channel vector we need the MMSE estimate of the filter $\boldsymbol{W_\delta}$.

We use Bayes' theorem to express the posterior distribution of $\boldsymbol{\delta}$ as

$$p(\boldsymbol{\delta}|\varphi) = \frac{p(\varphi|\boldsymbol{\delta})p(\boldsymbol{\delta})}{\int p(\varphi|\boldsymbol{\delta})p(\boldsymbol{\delta})d\boldsymbol{\delta}} \tag{4.38}$$

We can write the MMSE estimate of $\boldsymbol{W_\delta}$ as

$$\widehat{\boldsymbol{W}} = \int \boldsymbol{W_\delta} p(\boldsymbol{\delta}|\varphi)d\boldsymbol{\delta} = \frac{\int \boldsymbol{W_\delta} p(\varphi|\boldsymbol{\delta})\, p(\boldsymbol{\delta})d\boldsymbol{\delta}}{\int p(\varphi|\boldsymbol{\delta})\, p(\boldsymbol{\delta})d\boldsymbol{\delta}}. \tag{4.39}$$

For our system model we get the explicit expression

Lemma 4.2. *With*

$$b_{\boldsymbol{\delta}} = T\log|\mathbf{I} - \boldsymbol{W_\delta}| \tag{4.40}$$

and the scaled sample covariance matrix

$$\boldsymbol{S} = \frac{1}{\sigma^2}\sum_{i=1}^{n} \boldsymbol{\varphi}_i\boldsymbol{\varphi}_i^{\mathrm{H}} \tag{4.41}$$

we get the MMSE estimate of $\boldsymbol{W_\delta}$

$$\widehat{\boldsymbol{W}} = \frac{\int \boldsymbol{W_\delta} p(\boldsymbol{\delta}) \exp\big(\operatorname{tr}(\boldsymbol{W_\delta S}) + b_{\boldsymbol{\delta}}\big)d\boldsymbol{\delta}}{\int p(\boldsymbol{\delta}) \exp\big(\operatorname{tr}(\boldsymbol{W_\delta S}) + b_{\boldsymbol{\delta}}\big)d\boldsymbol{\delta}}. \tag{4.42}$$

Proof. Common factors of $p(\varphi|\boldsymbol{\delta})$ that do not depend on $\boldsymbol{\delta}$ can be dropped, since $p(\varphi|\boldsymbol{\delta})$ appears in both the numerator and denominator of the MMSE filter in (4.39). From our system model we get

$$\begin{aligned} p(\varphi|\boldsymbol{\delta}) &\propto \exp\Big(-\sum_i \boldsymbol{\varphi}_i^{\mathrm{H}}\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1}\boldsymbol{\varphi}_i + T\log|\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1}|\Big) \\ &\propto \exp\Big(-\operatorname{tr}(\sigma^2\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1}\boldsymbol{S}) + T\log|\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1}|\Big). \end{aligned}$$

We use the fact that

$$\mathbf{I} - \boldsymbol{W_\delta} = \boldsymbol{C_\varphi}\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} - \boldsymbol{C_\delta}\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} = \sigma^2\boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} \tag{4.43}$$

to get the expression in (4.42). For the final expression, we remove all summands in the exponent that do not depend on $\boldsymbol{\delta}$. □

Note that the scaled sample covariance matrix $\boldsymbol{S}$ is a sufficient statistic to calculate the MMSE filter $\widehat{\boldsymbol{W}}$. We have the basic structure

$$\hat{\boldsymbol{h}}_i = \widehat{\boldsymbol{W}}(\boldsymbol{S})\boldsymbol{\varphi}_i. \tag{4.44}$$

With $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{h}}_1, \dots, \hat{\boldsymbol{h}}_n]$ and $\boldsymbol{\Phi} = [\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_n]$, we get

$$\hat{\boldsymbol{H}} = \widehat{\boldsymbol{W}}(\boldsymbol{S})\ \boldsymbol{\Phi}. \tag{4.45}$$

That is, we only need to estimate $\boldsymbol{W}_{\boldsymbol{\delta}}$ once and apply the resulting filter to all observations to calculate all channel estimates simultaneously.

This structure is also beneficial for applications where we are only interested in the estimate of the most recent channel vector. In this case, we can apply an adaptive method to track the scaled sample covariance matrix. That is, given the most recent observation $\boldsymbol{\varphi}$, we apply the update

$$\boldsymbol{S} \leftarrow \alpha \boldsymbol{S} + \beta \boldsymbol{\varphi}\boldsymbol{\varphi}^{\mathrm{H}} \tag{4.46}$$

with suitable $\alpha, \beta > 0$ and then calculate the channel estimate

$$\hat{\boldsymbol{h}} = \widehat{\boldsymbol{W}}(\boldsymbol{S})\ \boldsymbol{\varphi}. \tag{4.47}$$

4.2 MMSE Estimation and Neural Networks

For arbitrary prior distributions $p(\boldsymbol{\delta})$, the MMSE filter as given by Lemma 4.2 cannot be evaluated in closed form. To make the filter computable, we need the following assumption.

Assumption 4.1. *The prior $p(\boldsymbol{\delta})$ is discrete and uniform, i.e., we have a grid $\{\boldsymbol{\delta}_i : i = 1, \dots, N\}$ of possible values for $\boldsymbol{\delta}$ and*

$$p(\boldsymbol{\delta}_i) = \frac{1}{N}\ \forall i = 1, \dots, N. \tag{4.48}$$

Under this assumption, we can evaluate the MMSE estimator of $\boldsymbol{W}_{\boldsymbol{\delta}}$ as

$$\boldsymbol{W}_{\mathrm{GE}}(\boldsymbol{S}) = \frac{\frac{1}{N}\sum_{i=1}^{N} \exp\big(\operatorname{tr}(\boldsymbol{W}_{\boldsymbol{\delta}_i}\boldsymbol{S}) + b_i\big)\boldsymbol{W}_{\boldsymbol{\delta}_i}}{\frac{1}{N}\sum_{i=1}^{N} \exp\big(\operatorname{tr}(\boldsymbol{W}_{\boldsymbol{\delta}_i}\boldsymbol{S}) + b_i\big)} \tag{4.49}$$

where $\boldsymbol{W}_{\boldsymbol{\delta}_i}$ is obtained by evaluating (4.36) for $\boldsymbol{\delta} = \boldsymbol{\delta}_i$ and

$$b_i = T \log|\mathbf{I} - \boldsymbol{W}_{\boldsymbol{\delta}_i}|. \tag{4.50}$$

If Assumption 4.1 does not hold, e.g., if $p(\boldsymbol{\delta})$ describes a continuous distribution, expression (4.49) is approximately true if the grid points $\boldsymbol{\delta}_i$ are chosen as random samples from $p(\boldsymbol{\delta})$. By the law-of-large-numbers, the approximation error vanishes as N is increased. Of course, by using more samples N, we increase the complexity of the channel estimation.

We can improve the performance of the estimator – for a fixed N – by interpreting $\boldsymbol{W}_{\boldsymbol{\delta}_i}$ and b_i as variables that can be optimized instead of using the values in (4.36) and (4.50). This is the idea underlying the learning-based approach we present in the following.

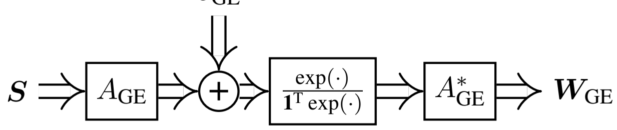

$\boldsymbol{b}_{\text{GE}}$
$\boldsymbol{S}$
A_{GE}
+
$\frac{\exp(\cdot)}{\mathbf{1}^{\text{T}}\exp(\cdot)}$
A^*_{GE}
$\boldsymbol{W}_{\text{GE}}$

Figure 4.2: Block diagram of the gridded estimator $\boldsymbol{W}_{\text{GE}}$

Let us analyze the structure of the gridded estimator. The input $\boldsymbol{S}$ is element of the real-valued vector space of Hermitian matrices $\mathbb{S}$. The first step in $\boldsymbol{W}_{\text{GE}}(\cdot)$ is to calculate inner products of the input $\boldsymbol{S}$ with matrices $\boldsymbol{W}_{\boldsymbol{\delta}_i} \in \mathbb{S}$. That is, we have a linear operator $A_{\text{GE}} : \mathbb{S} \mapsto \mathbb{R}^N$, parameterized by the $\boldsymbol{W}_{\boldsymbol{\delta}_i}$, $i = 1, \dots, n$.

Then, after adding the offset $\boldsymbol{b}_{\text{GE}} = [b_1, \dots, b_N]^{\text{T}}$ we apply a nonlinear transformation, namely the softmax function

$$\phi_{\text{SM}}(\boldsymbol{x}) = \frac{\exp(\boldsymbol{x})}{\mathbf{1}^{\text{T}}\exp(\boldsymbol{x})} \tag{4.51}$$

and finally the adjoint of the linear operator A_{GE}. The block-diagram of $\boldsymbol{W}_{\text{GE}}(\cdot)$ is given in Fig. 4.2. Note that, since $\mathbb{S}$ is a M^2 dimensional space over the reals, the linear operator can be represented by a $N \times M^2$ real valued matrix.

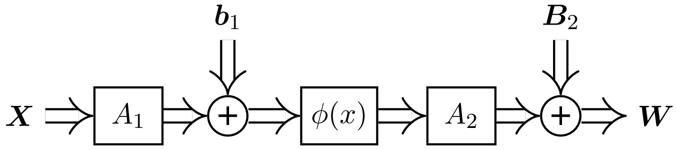


Figure 4.3: Neural network with two layers and activation function $\phi(x)$

A slightly more general structure is depicted in Fig. 4.3, which is readily identified as a common structure of a feed-forward neural network (NN) with two *linear layers*, which are connected by a nonlinear *activation function*. The gridded estimator $\boldsymbol{W}_{\text{GE}}$ is a special case of the neural network in Fig. 4.3, which uses the *softmax* function in (4.51) as activation function and the specific choices $A_1 = A_{\text{GE}}$, $A_2 = A^*_{\text{GE}}$, $\boldsymbol{b}_1 = \boldsymbol{b}$ and $\boldsymbol{B}_2 = \boldsymbol{0}$ for the variables.

To formulate the learning problem mathematically, we define the set of all functions that can be represented by the NN in Fig. 4.3 as

$$\begin{aligned}\mathcal{W}_{\text{NN}} = \Big\{ & f : \mathbb{S} \mapsto \mathbb{S}, f(\boldsymbol{X}) = A_2 \phi(A_1 \boldsymbol{X} + \boldsymbol{b}_1) + \boldsymbol{B}_2 \\ & \qquad A_1 : \mathbb{S} \mapsto \mathbb{R}^N, A_2 : \mathbb{R}^N \mapsto \mathbb{S}, \boldsymbol{b}_1 \in \mathbb{R}^N, \boldsymbol{B}_2 \in \mathbb{S} \Big\}. \end{aligned} \tag{4.52}$$

Since we are interested in minimizing the channel MSE, the cost function with respect to the estimator $\boldsymbol{W}(\cdot)$ that we use to estimate $\boldsymbol{W}_{\boldsymbol{\delta}}$ from $\boldsymbol{S}$ is given by

$$\varepsilon(\boldsymbol{W}(\cdot)) = \mathrm{E}[\|\boldsymbol{H} - \boldsymbol{W}(\boldsymbol{S})\ \boldsymbol{\Phi}\|_F^2]. \tag{4.53}$$

The optimal neural network, i.e., the NN-MMSE estimator, is given by

$$\boldsymbol{W}^\star_{\text{NN}}(\cdot) = \underset{\boldsymbol{W}(\cdot) \in \mathcal{W}_{\text{NN}}}{\operatorname{argmin}}\ \varepsilon(\boldsymbol{W}(\cdot)). \tag{4.54}$$

Since we assume that the dimension N and the activation function $\phi(\cdot)$ are fixed, the variational problem in (4.54) is simply an optimization over the linear transformations A_ℓ and the biases $\boldsymbol{b}_1$ and $\boldsymbol{B}_2$.

If we choose the softmax function as activation function, and if Assumption 4.1 is fulfilled, we have

$$\varepsilon(\boldsymbol{W}_{\text{GE}}(\cdot)) = \varepsilon(\boldsymbol{W}^\star_{\text{NN}}(\cdot)) = \varepsilon(\widehat{\boldsymbol{W}}(\cdot)) \tag{4.55}$$

since the gridded estimator is the MMSE estimator in this case. In general, we have

$$\varepsilon(\boldsymbol{W}_{\text{GE}}(\cdot)) \geq \varepsilon(\boldsymbol{W}^\star_{\text{NN}}(\cdot)) \geq \varepsilon(\widehat{\boldsymbol{W}}(\cdot)) \tag{4.56}$$

since $\boldsymbol{W}_{\text{GE}}(\cdot) \in \mathcal{W}_{\text{NN}}$ and $\boldsymbol{W}^\star_{\text{NN}}(\cdot)$ is the best estimator in this set.

The problem in (4.54) is a typical learning problem for a NN, with a slightly unusual cost function. Due to the expectation in the objective function, we have to revert to stochastic gradient methods to find (local) optima for the variables of the NN. Unlike the *gridded estimator* (4.49), which relies on analytic expressions for the covariance matrix $\boldsymbol{C}_{\boldsymbol{\delta}}$, the *neural network estimator* merely needs a large data set of channel realizations and observations to optimize the variables. In fact, we could also take samples of channel vectors and observations from a

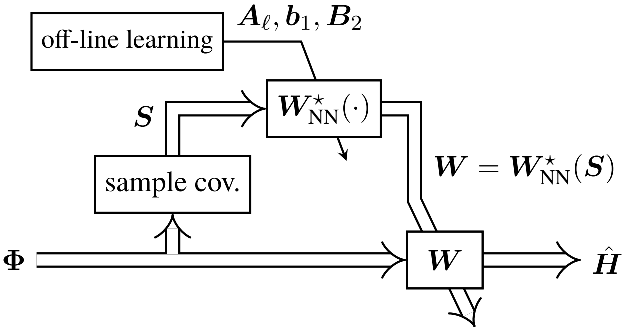

off-line learning
$\boldsymbol{A}_\ell, \boldsymbol{b}_1, \boldsymbol{B}_2$
$\boldsymbol{S}$
$\boldsymbol{W}_{\text{NN}}^\star(\cdot)$
sample cov.
$\boldsymbol{W} = \boldsymbol{W}_{\text{NN}}^\star(\boldsymbol{S})$
$\boldsymbol{\Phi}$
$\boldsymbol{W}$
$\hat{\boldsymbol{H}}$

Figure 4.4: Channel estimator with embedded neural network

measurement campaign to learn the NN-MMSE estimator for the "true" channel model. This requires that the SNR during the measurement campaign is significantly larger than the SNR in operation. If, as assumed, the noise covariance matrix is known, the observations can then be generated by adding noise to the channel measurements.

Note that the learning of the optimal variables is performed off-line. That is, the training data that is used to learn the optimal variables of the neural network is not related to the training phase in operation. The basic structure of the channel estimator is depicted in Fig. 4.4. From the observations $\boldsymbol{\Phi}$ we form the scaled sample covariance matrix $\boldsymbol{S}$, which is fed into the neural network $\boldsymbol{W}_{\text{NN}}^\star(\cdot)$. The output $\boldsymbol{W}$ of the neural network is then applied as a linear filter to the observations $\boldsymbol{\Phi}$ to get the channel estimates $\hat{\boldsymbol{H}}$. Before this estimator is put into operation, a learning procedure is used to find the optimal linear operators A_ℓ and bias variables $\boldsymbol{b}_1$, $\boldsymbol{B}_2$ for the neural network.

With proper initialization and sufficient quality of the training data, the neural network estimator is guaranteed to outperform the gridded estimator with the same complexity. However, there are two problems with this learning approach, which we address in the following sections. First, finding the optimal neural network $\boldsymbol{W}_{\text{NN}}^\star$ is too difficult, because the number of variables is huge and the optimization problem is not convex. Second, even if the optimal variables were known, the computation of the channel estimate is too complex: Evaluating the output of the neural network needs $\mathrm{O}(M^2N)$ floating point operations due to the matrix vector products. For example, if the grid size N needs to scale linearly with the number of antennas M to obtain accurate estimates, the computational complexity scales as $\mathrm{O}(M^3)$, which is too high for practical applications.

To reduce the computational complexity and the number of variables that have to be learned, we impose restrictions onto the linear operators A_1 and A_2. These restrictions are motivated by properties of the linear operator A_{GE} in $\boldsymbol{W}_{\text{GE}}$, which emerge for certain array geometries and channel models. Specifically, under certain assumptions, the operator A_{GE} lies in a low-dimensional subspace $\mathcal{A}$ of all linear operators $\mathcal{L}(\mathbb{S}, \mathbb{R}^N)$ and the adjoint operator A^*_{GE} in the corresponding subspace $\mathcal{A}^*$. Thus, for the NN, we add the constraints $A_1 \in \mathcal{A}$ and $A_2 \in \mathcal{A}^*$ to the set $\mathcal{W}_{\text{NN}}$ resulting in a smaller set

$$\begin{aligned} \mathcal{W}_{\mathcal{A}} = \Big\{ & f : \mathbb{S} \mapsto \mathbb{S}, f(\boldsymbol{X}) = A_2 \phi(A_1 \boldsymbol{X} + \boldsymbol{b}_1) + \boldsymbol{B}_2 \\ & A_1 \in \mathcal{A},\ A_2 \in \mathcal{A}^*,\ \boldsymbol{b}_1 \in \mathbb{R}^N,\ \boldsymbol{B}_2 \in \mathcal{U} \Big\} \qquad (4.57) \\ \subset\ & \mathcal{W}_{\text{NN}}. \qquad (4.58) \end{aligned}$$

As we will see in examples later on, the range space of the operators in $\mathcal{A}^*$ is typically a subspace $\mathcal{U} \subset \mathbb{S}$. Thus, we also restrict the bias variable $\boldsymbol{B}_2$ to this subspace $\mathcal{U}$ for consistency.

Analogously to (4.54), we can search for the optimal estimator in $\mathcal{W}_{\mathcal{A}}$, i.e.,

$$\boldsymbol{W}^\star_{\mathcal{A}}(\cdot) = \underset{\boldsymbol{W}(\cdot) \in \mathcal{W}_{\mathcal{A}}}{\operatorname{argmin}}\ \varepsilon(\boldsymbol{W}(\cdot)). \qquad (4.59)$$

If we have $A_{\text{GE}} \in \mathcal{A}$ for the linear operator in the gridded estimator, we have $\boldsymbol{W}_{\text{GE}} \in \mathcal{W}_{\mathcal{A}}$. Thus, we have

$$\varepsilon(\boldsymbol{W}_{\text{GE}}(\cdot)) \geq \varepsilon(\boldsymbol{W}^\star_{\mathcal{A}}(\cdot)) \geq \varepsilon(\widehat{\boldsymbol{W}}(\cdot)). \qquad (4.60)$$

If the operator A_{GE} is not exactly in $\mathcal{A}$, we can approximate the gridded estimator by projecting A_{GE} onto $\mathcal{A}$. We get the approximate gridded estimator

$$\boldsymbol{W}_{\text{GE},\mathcal{A}} = P_{\mathcal{A}}(A_{\text{GE}})^* \phi(P_{\mathcal{A}}(A_{\text{GE}}) \boldsymbol{S} + \boldsymbol{b}) \qquad (4.61)$$

where we could think of the projection as something like

$$P_{\mathcal{A}}(A) = \underset{A' \in \mathcal{A}, \boldsymbol{\Pi}}{\operatorname{argmin}} \left\| A' - \boldsymbol{\Pi} A \right\|_F^2, \boldsymbol{\Pi} \in \{0,1\}^{N \times N}, \boldsymbol{\Pi}^{\mathrm{T}} \boldsymbol{\Pi} = \mathbf{I}\,. \qquad (4.62)$$

We included a permutation matrix $\boldsymbol{\Pi}$ in the projection, since such a permutation has no effect on the estimator, but it might be relevant for the projection onto the subspace $\mathcal{A}$.

Similar to before, we get

$$\varepsilon(\boldsymbol{W}_{\mathrm{GE},\mathcal{A}}(\cdot)) \geq \varepsilon(\boldsymbol{W}^{\star}_{\mathcal{A}}(\cdot)) \geq \varepsilon(\widehat{\boldsymbol{W}}(\cdot)). \tag{4.63}$$

We cannot make any statements about the performance of $\boldsymbol{W}_{\mathrm{GE}}(\cdot)$ compared to $\boldsymbol{W}^{\star}_{\mathcal{A}}(\cdot)$ in this case, but it is important to remember that $\boldsymbol{W}^{\star}_{\mathcal{A}}(\cdot)$ potentially has a much lower complexity due to the restriction of the linear operators to the subspaces $\mathcal{A}$ and $\mathcal{A}^*$.

Two operator subspaces emerge from our channel model. First, the subspace

The operator $\operatorname{diag}^* : \mathbb{S} \mapsto \mathbb{R}^M$ is the adjoint of the $\operatorname{diag}(\cdot)$ operator, i.e., it is the operator that returns the vector of diagonal elements of a symmetric matrix.

$$\mathcal{A}_{\boldsymbol{Q}} = \{A : A(\boldsymbol{S}) = \boldsymbol{A}\operatorname{diag}^*(\boldsymbol{Q}^{\mathrm{H}}\boldsymbol{S}\boldsymbol{Q}), \boldsymbol{A} \in \mathbb{R}^{N \times S}\} \tag{4.64}$$

with a fixed matrix $\boldsymbol{Q} \in \mathbb{C}^{M \times S}$. The corresponding set of adjoint operators is given by

$$\mathcal{A}^*_{\boldsymbol{Q}} = \{A^* : A^*(\boldsymbol{x}) = \boldsymbol{Q}\operatorname{diag}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{x})\boldsymbol{Q}^{\mathrm{H}}, \boldsymbol{A} \in \mathbb{R}^{N \times S}\}. \tag{4.65}$$

This is exactly the set of operators A^* with $\operatorname{range}(A^*) \subseteq \mathcal{U}_{\boldsymbol{Q}}$ where

$$\mathcal{U}_{\boldsymbol{Q}} = \{\boldsymbol{X} : \boldsymbol{Q}\operatorname{diag}(\boldsymbol{x})\boldsymbol{Q}^{\mathrm{H}}, \boldsymbol{x} \in \mathbb{R}^S\}. \tag{4.66}$$

We have $A_{\mathrm{GE}} \in \mathcal{A}_{\boldsymbol{Q}}$ if the following assumption holds.

Assumption 4.2. *For all grid points $\boldsymbol{\delta}_i$, we have $\boldsymbol{W}_{\boldsymbol{\delta}_i} = \boldsymbol{Q}\operatorname{diag}(\boldsymbol{w}_i)\boldsymbol{Q}^{\mathrm{H}}$ for some $\boldsymbol{w}_i \in \mathbb{R}^S$.*

Examples follow shortly. To get $\boldsymbol{W}_{\mathrm{GE},\mathcal{A}_{\boldsymbol{Q}}}$ for an operator $A_{\mathrm{GE}} \notin \mathcal{A}_{\boldsymbol{Q}}$, we replace the $\boldsymbol{W}_{\boldsymbol{\delta}_i}$ that parameterize A_{GE} by matrices $\boldsymbol{Q}\operatorname{diag}(\boldsymbol{w}_i)\boldsymbol{Q}^{\mathrm{H}}$, where

$$\boldsymbol{w}_i = \operatorname*{argmin}_{\boldsymbol{w}} \left\|\boldsymbol{W}_{\boldsymbol{\delta}_i} - \boldsymbol{Q}\operatorname{diag}(\boldsymbol{w})\boldsymbol{Q}^{\mathrm{H}}\right\|_F^2. \tag{4.67}$$

The second subspace we encounter is

$$\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}} = \{A : A(\boldsymbol{S}) = \boldsymbol{A}\operatorname{diag}^*(\boldsymbol{Q}^{\mathrm{H}}\boldsymbol{S}\boldsymbol{Q}), \boldsymbol{A} \in \mathcal{C}_P\} \tag{4.68}$$

where

The set $\mathcal{C}_{\mathbb{R}}$ denotes the set of real-valued circulant matrices.

$$\mathcal{C}_P = \{\boldsymbol{A} : \boldsymbol{A} = [\boldsymbol{A}_1^{\mathrm{T}}, \ldots, \boldsymbol{A}_P^{\mathrm{T}}]^{\mathrm{T}}, \boldsymbol{A}_p \in \mathcal{C}_{\mathbb{R}}\ \forall p = 1, \ldots, P\}. \tag{4.69}$$

For the dimensions to match we need $N = PS$ for some $P \in \mathbb{N}$. So, the difference between $\mathcal{A}_{\boldsymbol{Q}}$ and $\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}$ is that operators in $\mathcal{A}_{\boldsymbol{Q}}$ are parameterized by a general matrix $\boldsymbol{A} \in \mathbb{R}^{N \times S}$, while for operators in $\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}$ this matrix consists of circular convolutions.

To get $A_{\mathrm{GE}} \in \mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}$ we need

Assumption 4.3. *Assumption 4.2 holds and for each $\boldsymbol{\delta}_i$ there is a $\boldsymbol{\delta}_j$ such that*

$$\boldsymbol{w}_j = \begin{bmatrix} [\boldsymbol{w}_i]_N \\ [\boldsymbol{w}_i]_{1:N-1} \end{bmatrix} \tag{4.70}$$

that is, $\boldsymbol{w}_j$ is a circularly shifted version of $\boldsymbol{w}_i$. Additionally, we have $p(\boldsymbol{w}_i) = p(\boldsymbol{w}_j)$.

With the set of convolutional neural networks (CNNs)

$$\begin{aligned} \mathcal{W}_{\text{CNN}} = \Big\{ & \boldsymbol{x} \mapsto \boldsymbol{A}_2 \phi\Big(\boldsymbol{A}_1 \boldsymbol{x} + \boldsymbol{b}_1\Big) + \boldsymbol{b}_2 \,, \\ & \boldsymbol{A}_1 \in \mathcal{C}_P, \boldsymbol{A}_2^{\mathrm{T}} \in \mathcal{C}_P, \boldsymbol{b}_1 \in \mathbb{R}^N, \boldsymbol{b}_2 \in \mathbb{R}^K \Big\} \end{aligned} \tag{4.71}$$

we can simplify NNs in $\mathcal{W}_{\mathcal{A}_{\boldsymbol{Q}}^{\mathcal{C}}}$ as follows.

Theorem 4.2. *For any $\boldsymbol{W}_{\mathcal{A}_{\boldsymbol{Q}}^{\mathcal{C}}} \in \mathcal{W}_{\mathcal{A}_{\boldsymbol{Q}}^{\mathcal{C}}}$ with $\boldsymbol{Q} \in \mathbb{C}^{M \times S}$, we have*

$$\boldsymbol{W}_{\mathcal{A}_{\boldsymbol{Q}}^{\mathcal{C}}}(\boldsymbol{S}) = \boldsymbol{Q} \operatorname{diag}(\boldsymbol{w}_{\mathit{CNN}}(\boldsymbol{s}))\boldsymbol{Q}^{\mathrm{H}} \tag{4.72}$$

for some $\boldsymbol{w}_{\mathit{CNN}}(\cdot) \in \mathcal{W}_{\mathit{CNN}}$ and

$$\boldsymbol{s} = \frac{1}{\sigma_2} \sum_i \left|\boldsymbol{Q}^{\mathrm{H}} \boldsymbol{\varphi}_i\right|^2 \tag{4.73}$$

where $|\cdot|^2$ is applied element-wise.

Proof. Follows directly from the parameterization in (4.68) and the fact that

$$[\boldsymbol{Q}^{\mathrm{H}} \boldsymbol{S} \boldsymbol{Q}]_{kk} = \frac{1}{\sigma^2} \sum_i \boldsymbol{q}_k^{\mathrm{H}} \boldsymbol{\varphi}_i \boldsymbol{\varphi}_i{}^{\mathrm{H}} \boldsymbol{q}_k = \frac{1}{\sigma^2} \sum_i \left|\boldsymbol{q}_k^{\mathrm{H}} \boldsymbol{\varphi}_i\right|^2 \tag{4.74}$$

with the k-th column $\boldsymbol{q}_k$ of $\boldsymbol{Q}$. □

The block-circulant matrices

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{A}_{11} \\ \vdots \\ \boldsymbol{A}_{1P} \end{bmatrix} \tag{4.75}$$

and

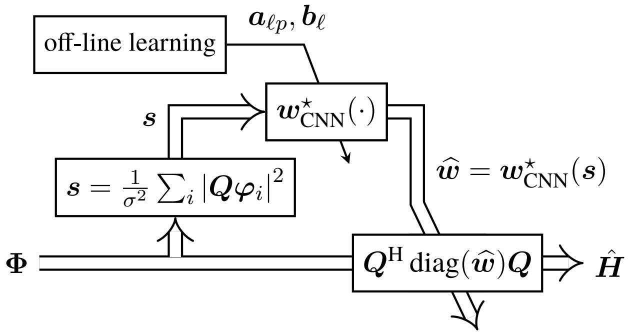
$$\boldsymbol{A}_2 = \begin{bmatrix} \boldsymbol{A}_{21}, & \cdots, & \boldsymbol{A}_{2P} \end{bmatrix} \tag{4.76}$$

in $\mathcal{W}_{\text{CNN}}$ can be specified with KP parameters each. Given the relationship between circulant matrices and circular convolution, we rewrite a multiplication with $\boldsymbol{A}_{\ell p} \in \mathcal{C}_{\mathbb{R}}$ as

$$\boldsymbol{A}_{\ell p}\boldsymbol{x} = \boldsymbol{F}^{\text{H}} \operatorname{diag}(\boldsymbol{F}\boldsymbol{a}_{\ell p})\boldsymbol{F}\boldsymbol{x} = \boldsymbol{a}_{\ell p} * \boldsymbol{x} \tag{4.77}$$

with $\boldsymbol{a}_{\ell p} \in \mathbb{R}^K$. For a fixed, small P, the complexity of evaluating the CNN is only $\text{O}(S \log S)$ thanks to the FFT. The complexity is thus dominated by matrix-vector multiplications with $\boldsymbol{Q}$ and $\boldsymbol{Q}^{\text{H}}$. Thus, it would be desirable if the transformation $\boldsymbol{Q}$ had a special structure that allowed for fast matrix-vector products.

Figure 4.5: CNN-estimator


off-line learning
$\boldsymbol{a}_{\ell p}, \boldsymbol{b}_\ell$
$\boldsymbol{w}^\star_{\text{CNN}}(\cdot)$
$\boldsymbol{s}$
$\widehat{\boldsymbol{w}} = \boldsymbol{w}^\star_{\text{CNN}}(\boldsymbol{s})$
$\boldsymbol{s} = \frac{1}{\sigma^2}\sum_i |\boldsymbol{Q}\boldsymbol{\varphi}_i|^2$
$\boldsymbol{\Phi}$
$\boldsymbol{Q}^{\text{H}} \operatorname{diag}(\widehat{\boldsymbol{w}})\boldsymbol{Q}$
$\hat{\boldsymbol{H}}$

For the learning procedure of the *CNN estimator*

$$\boldsymbol{w}^\star_{\text{CNN}}(\cdot) = \operatorname*{argmin}_{\boldsymbol{w}(\cdot)\in\mathcal{W}_{\text{CNN}}} \varepsilon(\boldsymbol{Q}^{\text{H}}\boldsymbol{w}(\operatorname{diag}^*(\boldsymbol{Q}^{\text{H}} \cdot \boldsymbol{Q}))\boldsymbol{Q}) \tag{4.78}$$

we assume as before that the activation function $\phi(\cdot)$ is fixed. Thus, the optimization is only with respect to the convolution kernels $\boldsymbol{a}_{\ell p}$ and the bias vectors $\boldsymbol{b}_\ell$. The structure of the low-complexity CNN-estimator is depicted in Fig. 4.5.

Of course, the inequalities we had for general subspaces $\mathcal{A}$ also hold for the specific subspace $\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}$. That is, if Assumptions 4.1 and 4.3 are fulfilled and we choose the softmax function as activation function we have $\boldsymbol{W}_{\text{GE}} \in \mathcal{W}_{\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}}$ and thus

$$\varepsilon(\boldsymbol{W}_{\text{GE}}) = \varepsilon(\boldsymbol{W}^\star_{\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}}) = \varepsilon(\widehat{\boldsymbol{W}}(\cdot)) \tag{4.79}$$

and in general, we have

$$\varepsilon(\boldsymbol{W}_{\text{GE},\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}}) \geq \varepsilon(\boldsymbol{W}^\star_{\mathcal{A}^{\mathcal{C}}_{\boldsymbol{Q}}}) \geq \varepsilon(\widehat{\boldsymbol{W}}(\cdot)). \tag{4.80}$$

In the following, we will look at examples of channel models and array geometries that motivate the subspaces $\mathcal{A}_{\boldsymbol{Q}}$ and $\mathcal{A}_{\boldsymbol{Q}}^{\mathcal{C}}$.

Examples.

Assumption 3.1 is quite similar to Assumption 4.2. In Chapter 3 we saw examples which motivate diagonal structure of the covariance matrices (Assumption 3.1) for ULAs, URAs, and distributed antennas. This corresponds to unitary transformations $\boldsymbol{Q}$ in Assumption 4.2. For a ULA we use the DFT matrix, for a URA we use a Kronecker product of two DFT matrices, and for distributed antennas we have $\boldsymbol{Q} = \mathbf{I}$. For ULAs and URAs the assumption holds approximately with vanishing error for large M (cf. also Appendices B.1 and B.2).

Specifically, the assumption of circulant covariance matrices for a large ULA corresponds to the restriction to the subset $\mathcal{A}_{\boldsymbol{F}}$ for the linear operators in the NN, where $\boldsymbol{F}$ is the discrete DFT matrix.

However, Assumption 4.2 is more general than Assumption 3.1 since we allow arbitrary, non-quadratic transformations $\boldsymbol{Q}$. If we consider the low SNR regime where $\rho = 1/\sigma^2 \ll 1$, we notice that

$$\boldsymbol{W}_{\boldsymbol{\delta}} = \boldsymbol{C}_{\boldsymbol{\delta}}(\boldsymbol{C}_{\boldsymbol{\delta}} + \sigma^2\,\mathbf{I})^{-1} = \rho\boldsymbol{C}_{\boldsymbol{\delta}}(\rho\boldsymbol{C}_{\boldsymbol{\delta}} + \mathbf{I})^{-1} \tag{4.81}$$

$$= \rho\boldsymbol{C}_{\boldsymbol{\delta}} + o(\rho). \tag{4.82}$$

That is, for low SNR, the filters $\boldsymbol{W}_{\boldsymbol{\delta}}$ have the same structure as the covariance matrices $\boldsymbol{C}_{\boldsymbol{\delta}}$.

For the ULA, the covariance matrices have Toeplitz structure and thus we can use $\boldsymbol{Q} = \boldsymbol{F}_2$, where $\boldsymbol{F}_2 \in \mathbb{C}^{M\times 2M}$ contains the first M rows of a $2M \times 2M$ DFT matrix.[3] Since $\mathcal{A}_{\boldsymbol{F}} \subset \mathcal{A}_{\boldsymbol{F}_2}$ we can also expect $\boldsymbol{W}^{\star}_{\mathcal{A}_{\boldsymbol{F}_2}}$ to perform well for a large number of antennas.

Analogous results can be derived for uniform rectangular arrays (cf. Appendix B.2). In this case, the transformation $\boldsymbol{Q}$ is the Kronecker product of two oversampled DFT matrices.

The CNN structure naturally arises in typical spatial channel models for a ULA at the base station. Given parameters $\boldsymbol{\delta}$, the covariance matrices are determined by the power density function $g(\theta; \boldsymbol{\delta})$ with respect to the angle of arrival θ. If the line of sight angle of a mobile user is uniformly distributed, there exist $\boldsymbol{\delta}_i$ and $\boldsymbol{\delta}_j$ such that $g(\theta; \boldsymbol{\delta}_j) = g(\theta - \phi; \boldsymbol{\delta}_i)$ for any angle ϕ and also $p(\boldsymbol{\delta}_i) = p(\boldsymbol{\delta}_j)$. This is due to the fact, that the 3GPP channel model has a shift invariance with respect to the line of sight angle of the users. In other words, if the model generates a power density $g(\theta)$ for a user, a shifted version $g(\theta - \phi)$ would have been equally likely.

How do we get from the shift invariance of the power densities to the shift invariance of the vectors $\boldsymbol{w}_{\boldsymbol{\delta}}$? If we have a ULA at the base station and a large number of antennas we know that the values in $\boldsymbol{w}_{\boldsymbol{\delta}}$

[3] All M-dimensional Toeplitz matrices can be parameterized by $\boldsymbol{F}_2\,\mathrm{diag}(\boldsymbol{x})\boldsymbol{F}_2^{\mathrm{H}}$ for some $\boldsymbol{x} \in \mathbb{C}^{2M}$.

are approximately sampled from a continuous function $w(\cdot;\boldsymbol{\delta})$ which depends point-wise on the spectrum $f(\cdot;\boldsymbol{\delta})$ (see Appendix B.1). Now for small angles we have $f(\theta;\boldsymbol{\delta}) \approx g(\theta;\boldsymbol{\delta})$ and thus the shift invariance holds approximately for $w(\theta;\boldsymbol{\delta})$ as long as $w(\theta;\boldsymbol{\delta})$ is non-zero only for small values of θ.

As another illustration, consider a toy example where we have an array of antennas along a long corridor, say in an airplane. Then we could reasonably assume diagonal covariance matrices, i.e., $\boldsymbol{Q} = \mathbf{I}$, but at the same time we have a shift-invariance for different positions of the users in the corridor, i.e., Assumption 4.3 also holds.

4.2.1 Practical Considerations

The stochastic-gradient method that learns the CNN is described in detail in Alg. 2. We want to stress again that the learning procedure is performed off-line and does not add to the complexity of the channel estimation. During operation, the channel estimation is performed by evaluating $\boldsymbol{w}^{\star}_{\text{CNN}}(\boldsymbol{s})$ and the transformations involving the $\boldsymbol{Q}$ matrix for given observations. If the variables are learned from simulated samples according to the 3GPP or any other channel model, this algorithm suffers from the same model-reality mismatch as does any other model-based algorithm. The fact that the proposed algorithm can also be trained on true channel realizations puts it into a significant advantage compared to other non-learning based algorithms, which have to rely on models only.

In the simulations, we compare two variants of the CNN estimator. First, we use the softmax activation function $\phi = \frac{\exp(\cdot)}{\mathbf{1}^{\mathrm{T}}\exp(\cdot)}$. The resulting softmax CNN estimator is a direct improvement over $\boldsymbol{W}_{\text{GE},\mathcal{A}^{C}_{\boldsymbol{Q}}}$.

In the second variant, we use a rectified linear unit (ReLU) $\phi(x) = [x]_{+}$ as activation function since ReLUs were found to be easier to train than other activation functions [47]. The order of complexity of estimating a channel vector is the same for both estimators, but clearly the ReLU function is easier to evaluate.

Local optima are a major issue when learning the neural networks, i.e., when calculating a solution of the nonlinear optimization problem (4.78). During our experiments, we observed that especially for a large number of antennas, the learning often gets stuck in local optimal. To deal with this problem, we devise a hierarchical learning procedure that starts the learning with a small number of antennas and then increases the number of antennas step-by-step.

Algorithm 2 Learning the CNN estimator

```
1: Initialize variables $a_\ell$ and $b_\ell$ randomly
2: Generate/select a mini-batch of $B$ channel vectors $H_b$ and corresponding observations $\Phi_b$ (and corresponding$s_b$) for $b = 1, \dots, B$
3: Calculate the stochastic gradients w.r.t. the variables of the CNN:
   $$g_{a_{\ell p}} = \frac{1}{B} \sum_{b=1}^{B} \frac{\partial}{\partial a_{\ell p}} \left\| H_b Q^{\mathrm{H}} \operatorname{diag}(w_{\mathrm{CNN}}(s_b)) Q \Phi_b \right\|_F^2$$
   and
   $$g_{b_\ell} = \frac{1}{B} \sum_{b=1}^{B} \frac{\partial}{\partial b_\ell} \left\| H_b Q^{\mathrm{H}} \operatorname{diag}(w_{\mathrm{CNN}}(s_b)) Q \Phi_b \right\|_F^2$$
   with $w_{\mathrm{CNN}}(\cdot) \in \mathcal{W}_{\mathrm{CNN}}$.
4: Update variables with a gradient algorithm (e.g., [46])
5: Repeat steps 2–4 until a convergence criterion is satisfied
```

If Assumptions 4.2 and 4.3 hold, we have $A_{\mathrm{GE}} \in \mathcal{A}_{Q}^{\mathcal{C}}$ and for a large uniform linear array, the convolution kernel w_i contains samples of the continuous function $w(u; \delta_i)$, i.e., $[w_i]_s = w(2\pi(s-1)/S; \delta_i)$. If we assume that $w(u; \delta_i)$ is a smooth function, we can quite accurately calculate the vector w_i for a system with M antennas from the corresponding vector of a system with less antennas by commonly used interpolation methods.

This observation inspires the following heuristic for initializing the variables a_ℓ and b_ℓ of a K-dimensional CNN. We first learn the variables of a smaller CNN, say we choose a CNN with dimension $S/2$. We use the resulting variables to initialize every second entry of the vectors a_ℓ and b_ℓ. The remaining entries can be obtained by numerical interpolation.

For the filter $w^{\star}_{\mathrm{CNN}}(\cdot)$ it is desirable to have outputs of similar magnitude, irrespective of the dimension S. By doubling the number of entries of the convolution kernels via interpolation, we approximately double the largest absolute value of $a_{\ell p} * x$. This does not matter if we use the softmax activation function, but for other activation functions a normalization is useful. Thus, we we normalize the kernels of the convolution after the interpolation such that we get approximately similar values at the outputs of each layer.

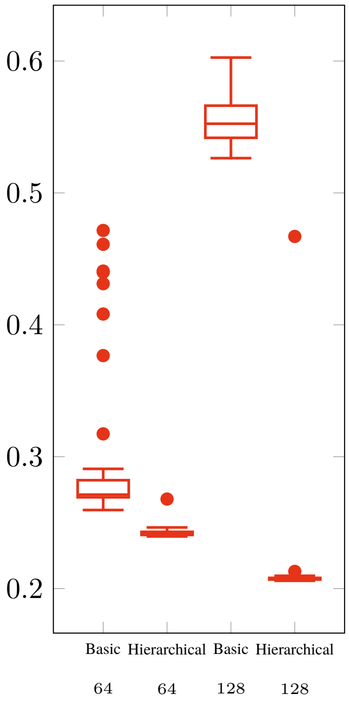


Figure 4.6: Box plot with outliers (marked as dots) of the MSE after learning for 10 000 iterations for hierarchical and non-hierarchical learning. We show results for $M = 64$ and $M = 128$ antennas for 50 data points per plot and with the DFT matrix $Q = F$ for the transformation. Scenario with three propagation paths, $\sigma^2 = 1, T = 1$. We see that without hierarchical learning, local optima are a severe issue.

Algorithm 3 Hierarchical Training

```
Choose upsampling factor $\beta > 1$ and number of stages $K$
Set $M^{(0)} = \lceil M/\beta^n \rceil$, $S^{(0)} = \lceil S/\beta^n \rceil$
Learn optimal $\boldsymbol{a}_{\ell p}^{(0)}, \boldsymbol{b}_\ell^{(0)} \in \mathbb{R}^{S_0}$ using Alg. 2 assuming $M_0$ antennas with random initializations
**for** $i$ from $1$ to $n$ **do**
    Set $M^{(i)} = \lceil M/\beta^{n-i} \rceil$ and $S^{(i)} = \lceil S/\beta^{n-i} \rceil$
    Interpolate $\boldsymbol{a}_{\ell p}^{(i)}, \boldsymbol{b}_\ell^{(i)} \in \mathbb{R}^{S^{(i)}}$ from $\boldsymbol{a}_{\ell p}^{(i-1)}, \boldsymbol{b}_\ell^{(i-1)} \in \mathbb{R}^{S^{(i-1)}}$
    Normalize $\boldsymbol{a}_{\ell p}^{(i)}$ by dividing by $\beta$
    Learn optimal $\boldsymbol{a}_{\ell p}^{(i)}, \boldsymbol{b}_\ell^{(i)}$ using Alg. 2 assuming $M_i$ antennas and using $\boldsymbol{a}_{\ell p}^{(i)}, \boldsymbol{b}_\ell^{(i)}$ as initializations
**end for**
```

This heuristic leads to the hierarchical learning described in Alg. 3. We start with a small number of antennas which is increased iteratively until we reach the desired number. After each increase of M, the parameters $\boldsymbol{a}_{\ell p}$ and $\boldsymbol{b}_\ell$ are initialized by interpolating from the previously trained smaller network.

The hierarchical learning significantly improves convergence speed while also lowering computational complexity per iteration due to the reduced number of antennas in most learning steps. In fact, for a large number of antennas the hierarchical learning is essential to obtain good performance. In Fig. 4.6 we show a standard box plot [48] of the MSE after learning with 10 000 iterations. The box plot depicts a summary of the resulting distribution, showing the median and the quartiles in a box and outliers outside of the "fences" as additional dots. As we can see, without the hierarchical learning, the learning procedure gets stuck in local optima and only the occasional outlier converges to an estimator with close to optimal performance. With the hierarchical approach, we are less likely to be caught in local optima during the learning process.

In Fig. 4.7, we see performance results of the neural network based estimators compared to the circular ML covaraince matrix estimator from Fig. 4.1. Another baseline is an iterative method from the area of compressed sensing called orthogonal matching pursuit (OMP). Finding an accurate stopping criterion for the OMP algorithm is not straightforward, thus we compare to a genie-aided variant, which uses the optimal number of iterations. For the CNN estimators we use $N = K$, i.e., we only have a single convolution per layer. The CNN

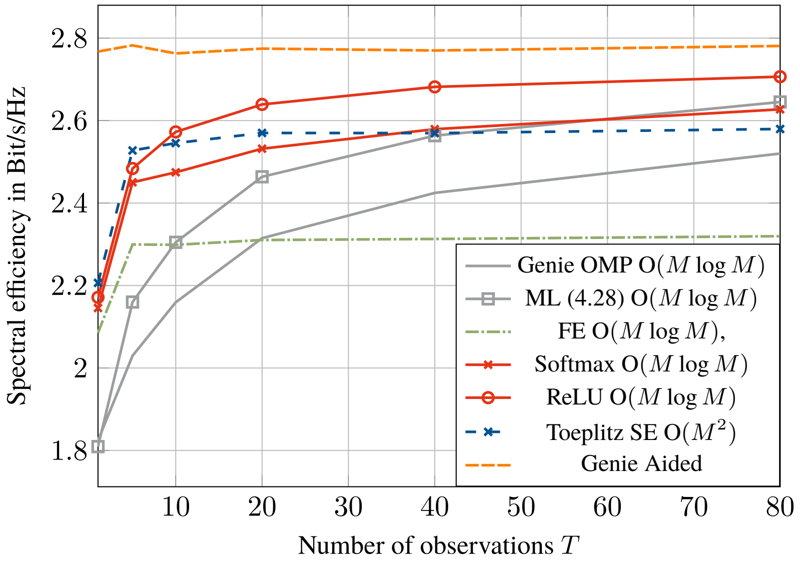


Figure 4.7: Single-user achievable rate for $M = 64$ antennas and an SNR of $-10\,$dB at the cell edge. The urban macro channel model specified in [37] is used to generate the channels. We show the result with respect to the number of observations, i.e., the coherence interval of the covariance matrices.

estimator with the ReLU activation function outperforms all other methods significantly.

4.3 Dealing with Pilot-Contamination

In Section 4.1 we assumed that we get interference free channel estimates for separate covariance matrix estimation by employing additional pilots which are only used for the covariance matrix estimation. This requires that the covariance matrices are static over many channel coherence intervals. The learning-based method in Section 4.2 does not require separate covariance matrix estimation, however the observations for channel estimation need to be interference free. The extension of this method to scenarios with pilot-contamination is still open and thus, for now, this approach cannot be applied in scenarios where pilot-contamination has a major impact on performance. In this section, we introduce a method that estimates the covariance matrices separately, but which uses the same observations that are also used for channel estimation. This poses additional challenges since, in general, the observations used for channel estimation are subject to interference. However, as we will demonstrate in the following, our method generates accurate estimates of the channel covariance matrices, even in the presence of interference and when the covariance matrices are constant only for a few channel coherence intervals.

If we want to use all available observations to estimate the covariance matrices we need a more complicated signal model. In general we want to consider the case where we have a different allocation of pilot sequences to users in different channel coherence intervals. We use the matrix $\mathbf{\Pi}_t \in \{0,1\}^{K\times T_{\text{tr}}}$ to define the allocation in coherence interval t. Specifically,

$$[\mathbf{\Pi}_t]_{kp} = \begin{cases} 1 & \text{if user } k \text{ uses pilot sequence } p \text{ in coherence interval } t \\ 0 & \text{otherwise.} \end{cases} \tag{4.83}$$

With $\mathbf{\Pi}_t = [\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_{T_{\text{tr}}}]$, the observations in channel coherence interval t can then be expressed as

$$\boldsymbol{\varphi}_{pt} = \boldsymbol{H}_t \boldsymbol{\pi}_{pt} + \boldsymbol{v}_{pt}, \quad p = 1, \ldots, T_{\text{tr}} \tag{4.84}$$

where $\boldsymbol{H}_t = [\boldsymbol{h}_{1t}, \ldots, \boldsymbol{h}_{Kt}]$ is the channel in coherence interval t.

For the covariance matrices we get

$$\boldsymbol{C}_{\boldsymbol{\varphi}_{pt}} = \sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_{pt}]_k + \boldsymbol{C}_{\boldsymbol{v}}. \tag{4.85}$$

With the vectorized covariance matrices $\boldsymbol{c}_{\boldsymbol{\varphi}_{pt}} = \text{vec}(\boldsymbol{C}_{\boldsymbol{\varphi}_{pt}})$ and $\boldsymbol{c}_{\boldsymbol{v}} = \text{vec}(\boldsymbol{C}_{\boldsymbol{v}})$ and

$$\boldsymbol{C}_{\boldsymbol{H}} = [\text{vec}(\boldsymbol{C}_{\boldsymbol{h}_1}), \ldots, \text{vec}(\boldsymbol{C}_{\boldsymbol{h}_K})] \tag{4.86}$$

we rewrite (4.85) to

$$\boldsymbol{c}_{\boldsymbol{\varphi}_{pt}} = \boldsymbol{C}_{\boldsymbol{H}} \boldsymbol{\pi}_{pt} + \boldsymbol{c}_{\boldsymbol{v}} \tag{4.87}$$

or for all observations in coherence interval t

$$[\boldsymbol{c}_{\boldsymbol{\varphi}_{1t}}, \ldots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}t}}] - \boldsymbol{c}_{\boldsymbol{v}} \mathbf{1}^{\text{T}} = \boldsymbol{C}_{\boldsymbol{H}} \mathbf{\Pi}_t. \tag{4.88}$$

If we use the same pilot allocation $\mathbf{\Pi}$ in all coherence intervals, the covariance matrices of the observations are the same for different t. We can thus reconstruct the channel covariance matrices

$$\boldsymbol{C}_{\boldsymbol{H}} = \left([\boldsymbol{c}_{\boldsymbol{\varphi}_1}, \ldots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}}}] - \boldsymbol{c}_{\boldsymbol{v}} \mathbf{1}^{\text{T}}\right) \mathbf{\Pi}^{+} \tag{4.89}$$

from the covariance matrices of the observations, as long as $K \leq T_{\text{tr}}$, i.e., only when we could have interference free observations anyway. For $K > T_{\text{tr}}$, when we have pilot-contamination, it is impossible to uniquely reconstruct the channel covariance matrices from the covariance matrices of the observations.

If we use S different allocations instead, i.e., we iterate through allocations $\widetilde{\boldsymbol{\Pi}}_i$, $i = 1$, …, S, we get ST_{tr} observations with different covariance matrices $\boldsymbol{C}_{\boldsymbol{\varphi}_{pi}}$. The equation system expands to

$$\boldsymbol{C}_{\boldsymbol{H}}[\widetilde{\boldsymbol{\Pi}}_1, \dots, \widetilde{\boldsymbol{\Pi}}_S] = [\boldsymbol{c}_{\boldsymbol{\varphi}_{11}}, \dots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}1}}, \dots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}S}}] - \boldsymbol{c}_{\boldsymbol{v}}\mathbf{1}^{\text{T}}. \quad (4.90)$$

We have $\boldsymbol{\Pi}_t = \boldsymbol{\Pi}_{t+S} = \widetilde{\boldsymbol{\Pi}}_{\text{mod}(t,S)}$. Thus, the covariance matrix of $\boldsymbol{\varphi}_{pt}$ is the same as that of $\boldsymbol{\varphi}_{p(t+S)}$ and is denoted by $\boldsymbol{C}_{\boldsymbol{\varphi}_{pi}}$ where $i = \text{mod}(t, S)$ is the remainder of t divided by S.

As long as the concatenation of the allocation matrices has full row rank, it is now possible to reconstruct the covariance matrices of the channel vectors from the covariance matrices of the observations. A simple two-step approach to estimate the covariance matrices is thus to first estimate the covariance matrices of the observations using the methods discussed in Section 4.1 and then solving (4.90) for the channel covariance matrices.

We see that to estimate the channel covariance matrices in the presence of pilot-contamination, we need a time-varying pilot allocation. The “additional pilots” which we used in Section 4.1 can bee seen as a special case of such a time-varying allocation.

Example 1. Suppose we have $T_{\text{tr}} = 2$ orthogonal training sequences and $K = 4$ users. This setup allows $S = 3$ distinct allocations of pilots to users

$$\widetilde{\boldsymbol{\Pi}}_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}, \quad \widetilde{\boldsymbol{\Pi}}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \widetilde{\boldsymbol{\Pi}}_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

The compound matrix $[\widetilde{\boldsymbol{\Pi}}_1, \widetilde{\boldsymbol{\Pi}}_2, \widetilde{\boldsymbol{\Pi}}_3]$ is well-conditioned with a condition number of $\sqrt{3}$.

Example 2. Suppose we have $T_{\text{tr}} = 3$ orthogonal training sequences and $K = 4$ users. We can use a fixed allocation for the first two pilot sequences for channel estimation and use the “additional” third pilot to facilitate the covariance matrix estimation. That is, we use the $S = 4$ different allocations

$$\widetilde{\boldsymbol{\Pi}}_1 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \widetilde{\boldsymbol{\Pi}}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \widetilde{\boldsymbol{\Pi}}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \widetilde{\boldsymbol{\Pi}}_4 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

In this case, the compound matrix $[\widetilde{\boldsymbol{\Pi}}_1, \widetilde{\boldsymbol{\Pi}}_2, \widetilde{\boldsymbol{\Pi}}_3, \widetilde{\boldsymbol{\Pi}}_4]$ also has full row-rank, but we waste one pilot-sequence per coherence interval for the covariance matrix estimation.

4.3.1 ML Estimation

Alternatively to the two-step approach, we can analyze the ML problem for general time-varying pilot allocations. We assume that we have a sensible allocation where each user only transmits one pilot sequence per coherence interval, i.e.,

$$\boldsymbol{\pi}_{pt}^{\mathrm{T}} \boldsymbol{\pi}_{st} = 0. \tag{4.91}$$

Thus, all observations $\boldsymbol{\varphi}_{pt}$ are mutually stochastically independent. For notational convenience, we merge the double index pt into one index $i = 1, \dots, n$ where $n = T_{\text{tr}} S$.

For one observation $\boldsymbol{\varphi}_i$ with corresponding allocation $\boldsymbol{\pi}_i$ we have the density

$$f_{\boldsymbol{\varphi}_i}(\boldsymbol{\varphi}_i) \propto \frac{1}{\det\left(\sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_i]_k + \boldsymbol{C}_{\boldsymbol{v}}\right)} e^{-\boldsymbol{\varphi}_i^{\mathrm{H}} \left(\sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_i]_k + \boldsymbol{C}_{\boldsymbol{v}}\right)^{-1} \boldsymbol{\varphi}_i}. \tag{4.92}$$

Thus, for several observations $\boldsymbol{\varphi}_i$ with $i = 1, \dots, n$, the log-likelihood is given by

$$L(\dots) = \sum_i \log\det\left(\sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_i]_k + \boldsymbol{C}_{\boldsymbol{v}}\right) + \boldsymbol{\varphi}_i^{\mathrm{H}} \left(\sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_i]_k + \boldsymbol{C}_{\boldsymbol{v}}\right)^{-1} \boldsymbol{\varphi}_i. \tag{4.93}$$

In the following we assume diagonal or, equivalently, jointly diagonalizable, covariance matrices (Assumption 3.1). The analysis also works for full covariance matrices, however, it requires ugly Kronecker product and vectorization hacks and the resulting estimator is too complex to use in practice anyway. For diagonal covariance matrices the estimation simplifies significantly, since the likelihood can be separately optimized for each antenna element.

Using Assumption 3.1, let us consider the likelihood for the variances of the mth spatial dimension. We define the vector of the mth entries of the channel vectors

$$\boldsymbol{h} = \left[[\boldsymbol{h}_1]_m, \dots, [\boldsymbol{h}_K]_m\right]^{\mathrm{T}}. \tag{4.94}$$

and the vector of the corresponding variances

$$\boldsymbol{c} = \left[[\boldsymbol{C}_{\boldsymbol{h}_1}]_{mm}, \dots, [\boldsymbol{C}_{\boldsymbol{h}_K}]_{mm}\right]^{\mathrm{T}}. \tag{4.95}$$

Our model for the observation $\varphi_i = [\boldsymbol{\varphi}_i]_m$ simplifies to

$$\varphi_i = \boldsymbol{\pi}_i^{\mathrm{T}} \boldsymbol{h} + v_i \tag{4.96}$$

with the noise $v_i = [\boldsymbol{v}_i]_m$ with variance $c_{\boldsymbol{v}} = [\boldsymbol{C}_{\boldsymbol{v}}]_{mm}$. If we plug this into Eq. (4.93), we get the likelihood for the variances $\boldsymbol{c}$

$$L(\boldsymbol{c}|\varphi) = \sum_i \log\left(\sum_k \boldsymbol{c}^{\mathrm{T}}\boldsymbol{\pi}_i + c_{\boldsymbol{v}}\right) + \frac{b_i}{\sum_k \boldsymbol{c}^{\mathrm{T}}\boldsymbol{\pi}_i + c_{\boldsymbol{v}}} \tag{4.97}$$

where $b_i = |\varphi_i|^2$. The derivative with respect to $\boldsymbol{c}$ is given by

$$\frac{\partial L}{\partial \boldsymbol{c}} = \sum_i \frac{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{\pi}_i + c_{\boldsymbol{v}} - b_i}{\left(\sum_k \boldsymbol{c}^{\mathrm{T}}\boldsymbol{\pi}_i + c_{\boldsymbol{v}}\right)^2}\boldsymbol{\pi}_i. \tag{4.98}$$

If we define

$$d_i = \frac{1}{\left(\sum_k \boldsymbol{c}^{\mathrm{T}}\boldsymbol{\pi}_i + c_{\boldsymbol{v}}\right)^2} \tag{4.99}$$

we can formulate the optimality condition

$$\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\pi}_i d_i \boldsymbol{\pi}_i^{\mathrm{T}} \boldsymbol{c} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\pi}_i d_i (b_i - c_{\boldsymbol{v}}). \tag{4.100}$$

To emphasize the interpretation of the left- and righ-hand-sides as averages we added the normalization with n.

We can also write the optimality condition in matrix-vector notation if we define the matrix

$$\boldsymbol{\Pi} = [\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_n] \tag{4.101}$$

and the vector of all observations

$$\boldsymbol{b} = [b_1, \ldots, b_n]^{\mathrm{T}} \tag{4.102}$$

and the diagonal matrix $\boldsymbol{D} = \operatorname{diag}(d_1, \ldots, d_n)$. The optimality condition is then given by

$$\frac{1}{n}\boldsymbol{\Pi}\boldsymbol{D}\boldsymbol{\Pi}^{\mathrm{T}}\boldsymbol{c} = \frac{1}{n}\boldsymbol{\Pi}\boldsymbol{D}(\boldsymbol{b} - c_{\boldsymbol{v}}\mathbf{1}). \tag{4.103}$$

Recall that our goal is to estimate the covariance matrix, that is, its diagonal elements $\boldsymbol{c}$ from the observations that we wrote into $\boldsymbol{b}$. If $\boldsymbol{D}$ was independent of $\boldsymbol{c}$, we could readily solve this linear system and obtain $\boldsymbol{c}$. Unfortunately, $\boldsymbol{D}$ depends on $\boldsymbol{c}$. However, as $\boldsymbol{D}$ is relatively insensitive to changes of $\boldsymbol{c}$, the condition in (4.103) suggests a fixed point iteration where we update

$$\hat{\boldsymbol{c}} \leftarrow \boldsymbol{f}(\hat{\boldsymbol{c}}) = (\boldsymbol{\Pi}\boldsymbol{D}(\hat{\boldsymbol{c}})\,\boldsymbol{\Pi}^{\mathrm{T}})^{-1}\boldsymbol{\Pi}\boldsymbol{D}(\hat{\boldsymbol{c}})\,(\boldsymbol{b} - c_{\boldsymbol{v}}\mathbf{1}) \tag{4.104}$$

based on a previous estimate of $\boldsymbol{c}$ that is used to calculate $\boldsymbol{D}$.

Let us analyze this fixed-point iteration. Since the entries of the vector of observations $\boldsymbol{b}$ are independent we know from the law of large numbers that

$$\boldsymbol{f}(\boldsymbol{c}) = \left(\frac{1}{n}\boldsymbol{\Pi D \Pi}^{\mathrm{T}}\right)^{-1} \frac{1}{n}\sum_i \boldsymbol{\pi}_i d_i (b_i - c_{\boldsymbol{v}}) \tag{4.105}$$

$$\asymp (\frac{1}{n}\boldsymbol{\Pi D \Pi}^{\mathrm{T}})^{-1} \frac{1}{n}\sum_i \boldsymbol{\pi}_i d_i (\mathrm{E}[b_i] - c_{\boldsymbol{v}}) \tag{4.106}$$

$$= \left(\frac{1}{n}\boldsymbol{\Pi D \Pi}^{\mathrm{T}}\right)^{-1} \frac{1}{n}\sum_i \boldsymbol{\pi}_i d_i \boldsymbol{\pi}_i^{\mathrm{T}} \boldsymbol{c}^{\star} \tag{4.107}$$

$$= \left(\frac{1}{n}\boldsymbol{\Pi D \Pi}^{\mathrm{T}}\right)^{-1} \frac{1}{n}\boldsymbol{\Pi D \Pi}^{\mathrm{T}} \boldsymbol{c}^{\star} = \boldsymbol{c}^{\star}. \tag{4.108}$$

That is, for an increasing number of observations n, a single iteration of the fixed point iteration converges to the desired variances $\boldsymbol{c}$ irrespective of the choice of $\boldsymbol{D}$ (and, thus, independent of the initialization $\hat{\boldsymbol{c}}$). This indicates that the fixed point iteration converges for a sufficient number of observations since the norm of the derivative $\frac{\partial \boldsymbol{f}(\boldsymbol{c})}{\partial \boldsymbol{c}^{\mathrm{T}}}$ is small in a large neighborhood of $\boldsymbol{c}^{\star}$ ($\boldsymbol{f}(\boldsymbol{c})$ is close to a constant function).

As the summations in (4.100) are over the n last coherence intervals, an adaptive algorithm would be desirable, which updates the left- and the right-hand-sides of (4.100) for new observations. To this end, we simply replace the averages by moving avarages, which leads to the adaptive algorithm given in Alg. 4. Note that, depending on the ratio of training T_{tr} to number of users K, it might be advantageous to directly update the Cholesky factorization of the left-hand-side in (4.103) to reduce the computational complexity.

The ML estimation is simplified if we use the same scaling matrix $\boldsymbol{D}$ for each spatial dimension m. In this case we can estimate all elements of the diagonal channel covariance matrices at once. Specifically, we have

$$\widehat{\boldsymbol{C}}_{\boldsymbol{H}} = (\boldsymbol{B} - \sigma_v^2 \mathbf{1})\boldsymbol{D \Pi}^{\mathrm{T}}(\boldsymbol{\Pi D \Pi}^{\mathrm{T}})^{-1} \tag{4.109}$$

where

$$\boldsymbol{B} = \left[\left|\boldsymbol{\varphi}_1\right|^2, \ldots, \left|\boldsymbol{\varphi}_n\right|^2\right].$$

If the same sequence $\widetilde{\boldsymbol{\Pi}} = [\widetilde{\boldsymbol{\Pi}}_1, \ldots, \widetilde{\boldsymbol{\Pi}}_S]$ of allocations (cf. Section 4.3) is repeated in N blocks, such that $n = SN$, we group the

Algorithm 4 Adaptive Variance Estimation

```
Similarly to many adaptive algorithms we use a constant forgetting factor $0 < \lambda < 1$.
Initialize the estimate of the matrix $\boldsymbol{\Xi} = \boldsymbol{\Pi}\boldsymbol{D}\boldsymbol{\Pi}^{\mathrm{T}}$
    $$\boldsymbol{\Xi} \leftarrow \mathbf{I}$$
Initialize the variance estimates and accumulated observations of the considered spatial dimension
    $$\hat{\boldsymbol{c}} \leftarrow \mathbf{1} \quad \boldsymbol{\psi} \leftarrow \mathbf{0}$$
for $t = 1, \ldots$ do
    Acquire the current allocation $\boldsymbol{\Pi}_t$ and resulting observations
        $$\boldsymbol{b}_t = \left[\left|[\boldsymbol{\varphi}_{1t}]_m\right|^2, \ldots, \left|[\boldsymbol{\varphi}_{T_{\mathrm{tr}}t}]_m\right|^2\right]^{\mathrm{T}}$$
    in the considered spatial dimension $m$
    Calculate the approximate scaling matrix using the current variance estimates
        $$[\boldsymbol{d}]_p \leftarrow 1/(\hat{\boldsymbol{c}}^{\mathrm{T}}\boldsymbol{\pi}_{pt} + \sigma_v^2)^2, \forall p = 1, \ldots, T_{\mathrm{tr}}$$
    Update the accumulated observations
        $$\boldsymbol{\psi} \leftarrow \lambda\boldsymbol{\psi} + \boldsymbol{\Pi}_t \operatorname{diag}(\boldsymbol{d})(\boldsymbol{b}_t - \sigma_v^2\mathbf{1})$$
    Update the matrix $\boldsymbol{\Xi}$
        $$\boldsymbol{\Xi} \leftarrow \lambda\boldsymbol{\Xi} + \boldsymbol{\Pi}_t \operatorname{diag}(\boldsymbol{d})\boldsymbol{\Pi}_t^{\mathrm{T}}$$
    Calculate the new variance estimates
        $$\hat{\boldsymbol{c}} \leftarrow \boldsymbol{\Xi}^{-1}\boldsymbol{\psi}$$
end for
```

observations into equally sized blocks $\boldsymbol{B} = [\boldsymbol{B}_1, \ldots, \boldsymbol{B}_N]$ to get

$$\widehat{\boldsymbol{C}} = \Big(\frac{\sum_{i=1}^{N} \boldsymbol{B}_i}{N} - \sigma_v^2 \mathbf{1}\Big) \widetilde{\boldsymbol{D}} \widetilde{\boldsymbol{\Pi}}^{\mathrm{T}} (\widetilde{\boldsymbol{\Pi}} \widetilde{\boldsymbol{D}} \widetilde{\boldsymbol{\Pi}}^{\mathrm{T}})^{-1}.$$

Since

$$\frac{\sum_{i=1}^{N} \boldsymbol{B}_s}{N} = [\hat{\boldsymbol{c}}_{\boldsymbol{\varphi}_1}, \ldots, \hat{\boldsymbol{c}}_{\boldsymbol{\varphi}_{NT_{\mathrm{tr}}}}]$$

is exactly the ML estimate of the covariance matrices of the observations, the suboptimal estimation in (4.109) has the two-step reconstruction method from Section 4.3 as special case, namely when we chose $\widetilde{\boldsymbol{D}} = \mathbf{I}$ (see (4.90)).

4.3.2 Pilot Allocation

If we implement the covariance matrix estimation with a repeated schedule $\widetilde{\boldsymbol{\Pi}} = [\widetilde{\boldsymbol{\Pi}}_1, \ldots, \widetilde{\boldsymbol{\Pi}}_S]$ of pilot allocations, $\widetilde{\boldsymbol{\Pi}}$ must have full row-rank for unique identifiability of the channel covariance matrices. If we want to serve all K users within one coherence interval, each user has to be assigned a pilot sequence, i.e., $\boldsymbol{\Pi}_t \mathbf{1} = \mathbf{1}$ for all t. Consequently, by adding one coherence interval to the schedule $\widetilde{\boldsymbol{\Pi}}$, the rank increases at most by $T_{\mathrm{tr}} - 1$. We have

$$\operatorname{rank}(\widetilde{\boldsymbol{\Pi}}) \leq T_{\mathrm{tr}} + (S-1)(T_{\mathrm{tr}} - 1) \tag{4.110}$$

and we need $\operatorname{rank}(\widetilde{\boldsymbol{\Pi}}) = K$, leading to the necessary condition for the minimal schedule interval

$$S \geq \frac{K-1}{T_{\mathrm{tr}} - 1}. \tag{4.111}$$

Thus, when we serve all users in each time-slot, we need at least two training sequences to ensure full row-rank of $\widetilde{\boldsymbol{\Pi}}$.

Since the pilot-allocation schedules can be generated off-line for given K, T_{tr}, and the resulting schedule interval S, we could theoretically do an exhaustive search over all feasible schedules to find the schedule with best condition number of $\widetilde{\boldsymbol{\Pi}}$. However, the design of the allocation is not actually an issue in practice. Typically the coherence interval of the covariance matrices is much larger than the bound in (4.111). Thus, if we use a slightly larger S and random allocations, we get a full-rank matrix with high probability.

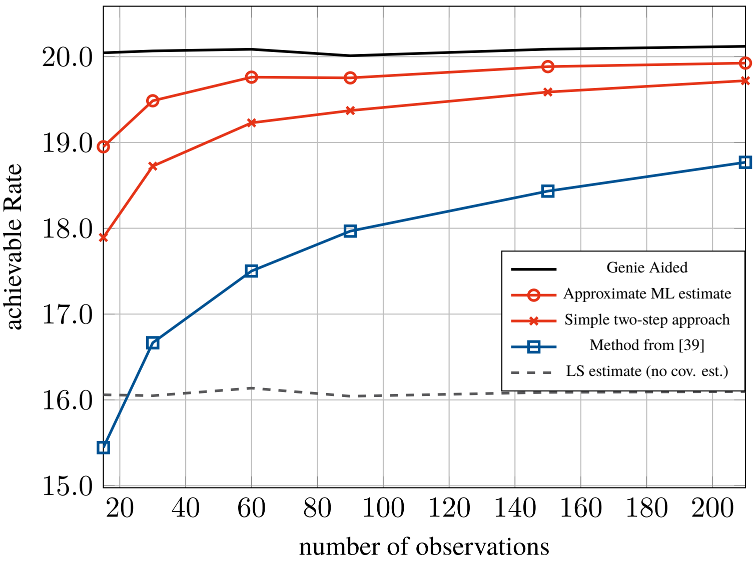


Figure 4.8: Achievable uplink rate of a single user, when the estimated channel covariance matrix is used to calculate the LMMSE filter. We assume perfect knowledge of the statistics at the decoder. We have $M = 64$ antennas at the base station and the effective training SNR is $\rho_{\text{tr}} = -10\text{dB}$.

4.3.3 Simulation Results

To showcase the performance of our algorithms, we present simulation results for the same two scenarios we used in Section 3.9. We simulate the multi-cell setup depicted (again) in Fig. 4.9. We have 5 users per cell for a total of $K = 15$ users.

We compare our approaches to the ones in [39], which use extra pilots specifically for covariance matrix estimation. The authors of [39] discuss two different pilot designs, which can be seen as a specific dynamic pilot allocation as discussed in Section 4.3. One method directly estimates the covariance matrices only using the extra pilots. The other method combines the pilots used for channel estimation with the extra pilots in a similar way as what we propose in Section 4.3. Of course, for the first pilot design, we could also combine the observations from the extra pilots and the observations used for channel estimation to get better results.

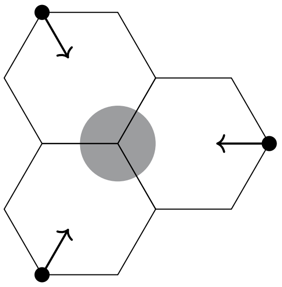


Figure 4.9: Small network with three hexagonal cells. The base stations are positioned at the corners and the users are uniformly distributed in the shaded circular area in the center.

The cell throughput with respect to the coherence interval of the covariance matrices is depicted in Fig. 4.8. We show results for $T_{\text{tr}} = 6$ orthogonal pilot sequences. For the method in [39], 5 of those sequences are reused in all cells. The remaining pilot sequence is used to generate additional observations that help to estimate the covariance matrices. In each coherence interval of the covariance matrices, we need at least one additional observation per user. Thus, for the given parameters, the method in [39] requires the coherence interval to be at least $T_C = K = 15$.

4.4 Summary

We first focus on separate maximum likelihood (ML) estimation of the covariance matrices. We start with the simple case where we have interference free observations to estimate the covariance matrices of the users. We can thus focus on a single user. We drop the user index and write the independent observations as

$$\boldsymbol{\varphi}_i = \boldsymbol{h}_i + \boldsymbol{v}_i^{\mathrm{tr}} \quad i = 1, \ldots, n. \tag{4.112}$$

where $\boldsymbol{h}_i \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{h}})$ and $\boldsymbol{v}_i^{\mathrm{tr}} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{v}})$. Since noise and channel vectors are independent we have $\boldsymbol{\varphi}_i \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\varphi}})$ with

$$\boldsymbol{C}_{\boldsymbol{\varphi}} = \boldsymbol{C}_{\boldsymbol{h}} + \boldsymbol{C}_{\boldsymbol{v}}. \tag{4.113}$$

The ML estimate for the covariance matrix of the observation is given by

$$\widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} = \frac{1}{n} \sum_i \boldsymbol{\varphi}_i \boldsymbol{\varphi}_i^{\mathrm{H}}. \tag{4.114}$$

The positive definite ML estimate for the channel covariance matrix is

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}} = \boldsymbol{C}_{\boldsymbol{v}}^{1/2} P_{\mathbb{S}}\Big(\boldsymbol{C}_{\boldsymbol{v}}^{1/2} \widehat{\boldsymbol{C}}_{\boldsymbol{\varphi}} \boldsymbol{C}_{\boldsymbol{v}}^{1/2} - \mathbf{I}\Big) \boldsymbol{C}_{\boldsymbol{v}}^{1/2}. \tag{4.115}$$

where $P_{\mathbb{S}}(\cdot)$ denotes the orthogonal projection onto the cone of positive semi-definite matrices.

If the covariance matrices are diagonal and $\boldsymbol{C}_{\boldsymbol{v}} = \sigma^2 \, \mathbf{I}$ we get

$$\widehat{\boldsymbol{C}}_{\boldsymbol{h}} = \mathrm{diag}(\hat{\boldsymbol{c}}_{\boldsymbol{h}}) \tag{4.116}$$

where

$$\hat{\boldsymbol{c}}_{\boldsymbol{h}} = \left[\hat{\boldsymbol{c}}_{\boldsymbol{\varphi}} - \sigma^2 \mathbf{1}\right]_+ \tag{4.117}$$

and

$$\hat{\boldsymbol{c}}_{\boldsymbol{\varphi}} = \frac{1}{n} \sum_i |\boldsymbol{\varphi}_i|^2 . \tag{4.118}$$

There exist different iterative approaches that are needed when the covariance matrices have more complicated structure, e.g., Toeplitz structure. In our experience, the gains over diagonal structure are marginal.

Significant gains can be obtained from prior information for the covariance matrices. We model the channel covariance matrix $\boldsymbol{C}_{\boldsymbol{\delta}}$ as a function of the random parameters $\boldsymbol{\delta} \sim p(\boldsymbol{\delta})$. That is, $\boldsymbol{h}_i | \boldsymbol{\delta} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\delta}})$.

The MMSE estimator for the channels $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n]$ from all observations $\boldsymbol{\Phi} = [\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_n]$ is given by

$$\hat{\boldsymbol{H}} = \widehat{\boldsymbol{W}}(\boldsymbol{\Phi})\ \boldsymbol{\Phi} \tag{4.119}$$

where

$$\widehat{\boldsymbol{W}} = \frac{\int \boldsymbol{W}_{\boldsymbol{\delta}} p(\boldsymbol{\delta}) \exp\big(\operatorname{tr}(\boldsymbol{W}_{\boldsymbol{\delta}}\boldsymbol{S}) + b_{\boldsymbol{\delta}}\big) d\boldsymbol{\delta}}{\int p(\boldsymbol{\delta}) \exp\big(\operatorname{tr}(\boldsymbol{W}_{\boldsymbol{\delta}}\boldsymbol{S}) + b_{\boldsymbol{\delta}}\big) d\boldsymbol{\delta}} \tag{4.120}$$

with

$$\boldsymbol{W}_{\boldsymbol{\delta}} = \boldsymbol{C}_{\boldsymbol{\delta}} \boldsymbol{C}_{\boldsymbol{\varphi}}^{-1} = \boldsymbol{C}_{\boldsymbol{\delta}} \left(\boldsymbol{C}_{\boldsymbol{\delta}} + \boldsymbol{C}_{\boldsymbol{v}}\right)^{-1}. \tag{4.121}$$

We approximate $\widehat{\boldsymbol{W}}(\cdot)$ with a convolutional neural network (CNN). Specifically we have the CNN-estimator

$$\boldsymbol{W}_{\mathcal{A}^{\mathrm{C}}_{\boldsymbol{Q}}}(\boldsymbol{\Phi}) = \boldsymbol{Q} \operatorname{diag}(\boldsymbol{w}_{\mathrm{CNN}}(\boldsymbol{s}(\boldsymbol{\Phi})))\boldsymbol{Q}^{\mathrm{H}} \tag{4.122}$$

where $\boldsymbol{Q} \in \mathbb{C}^{M \times S}$ is a fixed matrix and

$$\boldsymbol{s}(\boldsymbol{\Phi}) = \frac{1}{\sigma_2} \sum_i \left|\boldsymbol{Q}^{\mathrm{H}} \boldsymbol{\varphi}_i\right|^2. \tag{4.123}$$

The function $\boldsymbol{w}_{\mathrm{CNN}}(\cdot) : \mathbb{R}^S \mapsto \mathbb{R}^S$ is a two-layer CNN. The parameters of the CNN are optimized in an off-line learning procedure.

The structure of the estimator $\boldsymbol{W}_{\mathcal{A}^{\mathrm{C}}_{\boldsymbol{Q}}}(\boldsymbol{\Phi})$ is motivated by typical spatial channel models. The preferred choice of the matrix $\boldsymbol{Q}$ depends on the array geometry.

If the observations are subject to interference, things get more complicated. We consider a general dynamic pilot allocation. We use the matrix $\boldsymbol{\Pi}_t \in \{0,1\}^{K \times T_{\mathrm{tr}}}$ with $\boldsymbol{\Pi}_t \boldsymbol{1} = \boldsymbol{1}$ to define the allocation of pilots to users in coherence interval t. With $\boldsymbol{\Pi}_t = [\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_{T_{\mathrm{tr}}}]$, the observations in coherence interval t can then be expressed as

$$\boldsymbol{\varphi}_{pt} = \boldsymbol{H}_t \boldsymbol{\pi}_{pt} + \boldsymbol{v}_{pt}, \ \ p = 1, \ldots, T_{\mathrm{tr}}, \ t = 1, \ldots, T_C \tag{4.124}$$

where $\boldsymbol{H}_t = [\boldsymbol{h}_{1t}, \ldots, \boldsymbol{h}_{Kt}]$ and T_C is the coherence interval of the channel covariance matrices.

For the covariance matrices we get the relation

$$\boldsymbol{C}_{\boldsymbol{\varphi}_{pt}} = \sum_k \boldsymbol{C}_{\boldsymbol{h}_k} [\boldsymbol{\pi}_{pt}]_k + \boldsymbol{C}_{\boldsymbol{v}}. \tag{4.125}$$

To get a convenient matrix-vector notation, we vectorize all covariance matrices, i.e., we have $\boldsymbol{c}_{\boldsymbol{\varphi}_{pt}} = \operatorname{vec}(\boldsymbol{C}_{\boldsymbol{\varphi}_{pt}})$ and $\boldsymbol{c}_{\boldsymbol{v}} = \operatorname{vec}(\boldsymbol{C}_{\boldsymbol{v}})$ and

$$\boldsymbol{C}_{\boldsymbol{H}} = [\operatorname{vec}(\boldsymbol{C}_{\boldsymbol{h}_1}), \ldots, \operatorname{vec}(\boldsymbol{C}_{\boldsymbol{h}_K})]. \tag{4.126}$$

We get

$$\boldsymbol{c}_{\boldsymbol{\varphi}_{pt}} = \boldsymbol{C}_{\boldsymbol{H}} \boldsymbol{\pi}_{pt} + \boldsymbol{c}_{\boldsymbol{v}}. \tag{4.127}$$

If we iterate through S different allocations $\widetilde{\boldsymbol{\Pi}}_i$, $i = 1, \ldots S$ we get observations with ST_{tr} different covariance matrices $\boldsymbol{C}_{\boldsymbol{\varphi}_{pi}}$ which are related to the channel covariance matrices via

$$\boldsymbol{C}_{\boldsymbol{H}}[\widetilde{\boldsymbol{\Pi}}_1, \ldots, \widetilde{\boldsymbol{\Pi}}_S] = [\boldsymbol{c}_{\boldsymbol{\varphi}_{11}}, \ldots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}1}}, \ldots, \boldsymbol{c}_{\boldsymbol{\varphi}_{T_{\text{tr}}S}}] - \boldsymbol{c}_{\boldsymbol{v}} \boldsymbol{1}^{\mathrm{T}}. \tag{4.128}$$

If we have estimates of the covariance matrices $\boldsymbol{C}_{\boldsymbol{\varphi}_{pi}}$, we can solve for the channel covariance matrices as long as $\widetilde{\boldsymbol{\Pi}} = [\widetilde{\boldsymbol{\Pi}}_1, \ldots, \widetilde{\boldsymbol{\Pi}}_s]$ has full row-rank. We can find such a matrix if

$$S \geq \frac{K-1}{T_{\text{tr}} - 1}. \tag{4.129}$$

We also introduce a method that directly finds an approximate ML estimate of the channel covariance matrices. The estimation is similar to the two-step approach, but instead of applying the Moore-Penrose pseudo inverse of $\widetilde{\boldsymbol{\Pi}}$ to the right hand side of (4.128), we use a different pseudo inverse of the form $\boldsymbol{D}\widetilde{\boldsymbol{\Pi}}^{\mathrm{T}}(\widetilde{\boldsymbol{\Pi}}\boldsymbol{D}\widetilde{\boldsymbol{\Pi}}^{\mathrm{T}})^{-1}$ with a diagonal matrix $\boldsymbol{D}$.

Optimally, we use a different matrix $\boldsymbol{D}_m$ for each row m of (4.128) where $\boldsymbol{D}_m$ depends on a previous estimate of the m-th row of $\boldsymbol{C}_{\boldsymbol{H}}$.

Chapter 5

Conclusion and Outlook

So, what did we learn so far? We saw the impact of different models for channel uncertainty on algorithms for transceiver design and resource allocation. We saw how asymptotic results can be used to optimize power allocation based on channel statistics instead of instantaneous channel realizations. We realized that obtaining the channel statistics may also be non-trivial, but that for typical channel models, efficient and accurate algorithms can be developed.

If we have an accurate model for the channel, noise, and the available observations, we often succeed in finding efficient algorithms with close to optimal performance by combining asymptotic approximations with established optimization methods. For more complicated models, it pays off to be aware of various tools from statistical signal procssing, such as the variational bayesian inference used in Chapter 3 or the convolutional neural networks used in Chapter 4.

Many extensions of the system setup are possible, some were mentioned throughout the thesis. For example, the extensions of the downlink resource allocation algorithms to a multi-cell setup. Since modern mobile phones typically have four antennas and are capable of non-linear processing, this could also be taken into account in an extended model. To handle multiple antennas at the mobile phone, we might need a more general concept of uplink-downlink duality, such as the one presented by Dotzler et al. in [16]. Finally, extending the new learning-based channel estimation approach to more general system models might also be worthwhile.

In most of this work we tried to keep the assumptions on the physical model as general as possible, only incorporating assumptions that helped to significantly reduce the computational complexity. Adding more details to the channel model is better done by the product developer than the academic researcher. Since so many aspects of algorithm design depend on the channel model and hardware constraints, it might in practice not always be possible to directly apply the algorithms described in this thesis. Nevertheless, I hope that the presented algorithm designs and analyses provide fellow engineers with methods and tools that help to tackle similar problems.

Appendix A

Information Theory Preliminaries

Consider the simple scalar channel model with input symbols s, which have variance one, and output

$$y = gs + v \tag{A.1}$$

where the channel g is known, and we have some additive noise v.

The maximal amount of information that can be transmitted over the channel is given by the mutual information

$$I(y; s) = h(s) - h(s|y) \tag{A.2}$$

where $h(\cdot)$ denotes the differential entropy. The mutual information depends on the distribution of the input s and the noise v.

In the following, we assume that s follows the normal distribution $s \sim \mathcal{N}_{\mathbb{C}}(0, 1)$. In this case, we have $h(s) = \log_2(\pi e)$. If the noise is independent of s and also normally distributed with $v \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^2)$, we get

$$I(y; s) = \log_2(1 + |g|^2 / \sigma^2) \tag{A.3}$$

which is in fact the maximal mutual information with respect to the input distributions, since for additive white Gaussian noise, Gaussian inputs are optimal [49].

If the noise is not Gaussian distributed (s is still Gaussian) and possibly also not independent of the input, it might not be possible to find a closed form expression for the conditional entropy $h(s|y)$. However, if we know the variance var$(s|y)$, we can use the bound [14, 15]

$$h(s|y) \leq \mathrm{E}[\log_2(\pi e \,\mathrm{var}(s|y))]. \tag{A.4}$$

This step is missing in the original proof in [15], where they simply assume that var$(s|y) = \mathrm{E}[\mathrm{var}(s|y)]$, which does not hold in general (it holds for jointly Gaussian distributed random variables).

We then use Jensen's inequality to get

$$h(s|y) \leq \log_2(\pi e \,\mathrm{E}[\mathrm{var}(s|y)]). \tag{A.5}$$

We further know that

$$\begin{aligned}\mathrm{E}[\mathrm{var}(s|y)] &= \mathrm{E}[(s-\mathrm{E}[s|y])(s-\mathrm{E}[s|y])^*] && \text{(A.6)}\\ &\leq \mathrm{E}[(s-f(y))(s-f(y))^*] && \text{(A.7)}\end{aligned}$$

for any function $f(y)$, since $\mathrm{E}[s|y]$ is the MMSE estimate of s from y. We can use the *linear* MMSE estimate of s for f,

$$f(y) = \frac{\mathrm{cov}(s,y)}{\mathrm{var}(y)} y \tag{A.8}$$

to get

$$\begin{aligned}\alpha = \mathrm{E}[(s-f(y))(s-f(y))^*] &= 1 - \frac{|\mathrm{cov}(y,s)|^2}{\mathrm{var}(y)} && \text{(A.9)}\\ &= 1 - \frac{|g+\mathrm{cov}(v,s)|^2}{|g|^2 + g\,\mathrm{cov}(v,s) + g^*\,\mathrm{cov}(s,v) + \mathrm{var}(v)} && \text{(A.10)}\\ &= \frac{\mathrm{var}(v) - |\mathrm{cov}(v,s)|^2}{|g|^2 + g\,\mathrm{cov}(v,s) + g^*\,\mathrm{cov}(s,v) + \mathrm{var}(v)}. && \text{(A.11)}\end{aligned}$$

A lower bound for the mutual information is thus given by

$$\begin{aligned}I(y;s) = h(s) - h(s|y) &\geq \log_2(\pi e) - \mathrm{E}[\log_2(\pi e \,\mathrm{var}(s|y))]\\ &\geq \log_2(\pi e) - \log_2(\pi e \alpha) = \log_2(1/\alpha)\\ &= \log_2\left(1 + \frac{|g+\mathrm{cov}(v,s)|^2}{\mathrm{var}(v) - |\mathrm{cov}(v,s)|^2}\right). && \text{(A.12)}\end{aligned}$$

If the noise v is uncorrelated with the input s, the bound simplifies to

$$I(y;s) \geq \log_2\left(1 + \frac{|g|^2}{\mathrm{var}(v)}\right), \tag{A.13}$$

which is simply the mutual information with equivalent Gaussian noise.

We can apply this bound to the scenario with imperfect CSI. We have the side information φ on the channel at the receiver and want to calculate the conditional mutual information $I(y; s|\varphi)$. To apply the bound we split the channel into an estimate $\hat{g}$, which is deterministic given φ and the resulting estimation error $\tilde{g} = g - \hat{g}$. We have

$$y = gs + v \tag{A.14}$$

$$y = \hat{g}s + \underbrace{(g - \hat{g})s + v}_{v_{\text{eff}}}. \tag{A.15}$$

That is, $\hat{g}$ takes the role of g in (A.12) and v_{eff} takes the role of v:

$$I(y; s) \geq \log_2 \left(1 + \frac{|\hat{g} + \text{cov}(v_{\text{eff}}, s|\varphi)|^2}{\text{var}(v_{\text{eff}}|\varphi) - |\text{cov}(v_{\text{eff}}, s|\varphi)|^2}\right). \tag{A.16}$$

We assume that the additive noise v is independent of s and g, but not necessarily Gaussian. The total effective noise v_{eff} is clearly no longer Gaussian and also not independent of the signal s. Thus, we need to calculate $\text{var}(v_{\text{eff}}|\varphi)$ and $\text{cov}(v_{\text{eff}}, s|\varphi)$ to evaluate the bound in (A.12) for the imperfect CSI scenario. With $\bar{g} = \text{E}[g|\varphi]$ we get

$$\begin{aligned}\text{var}(v_{\text{eff}}|\varphi) &= \text{var}(v) + \text{E}[|g - \hat{g}|^2 \,|\varphi] \\ &= \text{var}(v) + \text{var}(g|\varphi) + |\bar{g}|^2 - \bar{g}\hat{g}^* - \hat{g}\bar{g}^* + |\hat{g}|^2 \\ &= \text{var}(v) + \text{var}(g|\varphi) + |\bar{g} - \hat{g}|^2 \end{aligned} \tag{A.17}$$

and

$$\text{cov}(v_{\text{eff}}, s|\varphi) = \bar{g} - \hat{g}. \tag{A.18}$$

The bound evaluates to

$$I(y; s|\varphi) \geq \log_2 \left(1 + \frac{|\bar{g}|^2}{\text{var}(v) + \text{var}(g|\varphi)}\right). \tag{A.19}$$

Note, that the bound is independent of the choice of $\hat{g}$ and only depends on the MMSE estimate $\bar{g}$ and the corresponding estimation error $\text{var}(g|\varphi)$. If we directly chose $\hat{g} = \bar{g}$, the effective noise is uncorrelated with the signal s and we get the same result

$$I(y; s|\varphi) \geq \log_2 \left(1 + \frac{|\bar{g}|^2}{\text{var}(v_{\text{eff}}|\varphi)}\right) = \log_2 \left(1 + \frac{|\bar{g}|^2}{\text{var}(v) + \text{var}(g|\varphi)}\right). \tag{A.20}$$

Appendix B

Spatial Channel Model

In general, the channel from a single user to the base station is modeled as complex Gaussian $\boldsymbol{h} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{\delta}})$ where the covariance matrix depends on parameters $\boldsymbol{\delta}$ of the physical environment.

Most channel models for cellular networks assume that the antenna array at the base station is in the far-field of the impinging waves. In this case, the covariance matrix is given by

$$\boldsymbol{C}_{\boldsymbol{\delta}} = \int g(\theta; \boldsymbol{\delta}) \boldsymbol{a}(\theta) \boldsymbol{a}(\theta) d\theta \tag{B.1}$$

where $\boldsymbol{a}(\theta)$ is the array manifold vector and $g(\theta; \boldsymbol{\delta})$ describes the distribution of the incoming power over the angle of arrival. In the following we derive asymptotic approximations for the uniform linear and uniform rectangular array geometries. We also quickly discuss a simple model for distributed antennas.

B.1 Uniform Linear Array

For a uniform linear array (ULA) with half-wavelength spacing at the base station, the steering vector is given by

$$\boldsymbol{a}(\theta) = \big[1,\ \exp(i\pi \sin\theta),\ \ldots,\ \exp(i\pi(M-1)\sin\theta)\big]^{\mathrm{H}}. \tag{B.2}$$

Consequently, the covariance matrix has Toeplitz structure with entries

$$[\boldsymbol{C}_{\boldsymbol{\delta}}]_{mn} = \int_{-\pi}^{\pi} g(\theta; \boldsymbol{\delta}) \exp(-i\pi(m-n)\sin\theta) d\theta. \tag{B.3}$$

If we substitute $\omega = \pi \sin \theta$, we get

$$[\boldsymbol{C_\delta}]_{mn} = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\omega; \boldsymbol{\delta}) \exp(-i(m-n)\omega) d\omega \tag{B.4}$$

with

$$f(\omega; \boldsymbol{\delta}) = 2\pi \frac{g(\arcsin(\omega/\pi); \boldsymbol{\delta}) + g(\pi - \arcsin(\omega/\pi); \boldsymbol{\delta})}{\sqrt{\pi^2 - \omega^2}} \tag{B.5}$$

where we extended g periodically beyond the interval $[-\pi, \pi]$. That is, the entries of the channel covariance matrix are Fourier coefficients of the periodic spectrum $f(\omega; \boldsymbol{\delta})$.

An interesting property of the Toeplitz covariance matrices is that we can define a circulant matrix $\tilde{\boldsymbol{C}}_{\boldsymbol{\delta}}$ with the eigenvalues $f(2\pi k/M; \boldsymbol{\delta})$, $k = 0, \dots, M-1$, such that $\tilde{\boldsymbol{C}}_{\boldsymbol{\delta}} \asymp \boldsymbol{C_\delta}$ [27]. That is, to get the elements of the circulant matrices we approximate the integral in (B.4) by the summation (cf. [50])

$$[\tilde{\boldsymbol{C}}_{\boldsymbol{\delta}}]_{mn} = \frac{1}{M} \sum_{k=0}^{M-1} f(2\pi k/M; \boldsymbol{\delta}) e^{-i(m-n)2\pi k/M}. \tag{B.6}$$

B.2 Uniform Rectangular Array

To work with a two-dimensional array, we need a three-dimensional channel model. That is, in addition to the azimuth angle θ, we also need an elevation angle ϕ to describe a direction of arrival. Under the far-field assumption, the covariance matrix is given by

$$\boldsymbol{C_\delta} = \int_{-\pi/2}^{\pi/2} \int_{-\pi}^{\pi} g(\theta, \phi; \boldsymbol{\delta}) \boldsymbol{a}(\theta, \phi) \boldsymbol{a}(\theta, \phi)^{\mathrm{H}} d\theta d\phi. \tag{B.7}$$

For a uniform rectangular array (URA) with half-wavelength spacing at the base station, we have $M = M_H M_V$ antenna elements, where M_H is the number of antennas in horizontal direction and M_V the number of antennas in vertical direction. The correlation between the antenna element at position (m, p) and the one at (n, q), given the parameters $\boldsymbol{\delta}$, is given by

$$\begin{aligned}
&\mathrm{E}[h_{(m,p)}h^*_{(n,q)}|\boldsymbol{\delta}] \\
&= \int\limits_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int\limits_{-\pi}^{\pi} g(\theta, \phi; \boldsymbol{\delta}) e^{i\pi((n-m)\sin\theta + (q-p)\cos\theta\sin\phi)} d\theta d\phi && \text{(B.8)} \\
&= \int\limits_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int\limits_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \tilde{g}(\theta, \phi; \boldsymbol{\delta}) e^{i\pi((n-m)\sin\theta + (q-p)\cos\theta\sin\phi)} d\theta d\phi && \text{(B.9)}
\end{aligned}$$

where

$$\tilde{g}(\theta, \phi; \boldsymbol{\delta}) = g(\theta, \phi; \boldsymbol{\delta}) + g(\pi - \theta, \phi; \boldsymbol{\delta}). \tag{B.10}$$

We can map the square $[-\pi/2, \pi/2]^2$ bijectively onto the circle with radius π with the substitution $\omega = \pi \sin\theta$ and $\nu = \pi\cos\theta\sin\phi$. The transformed integral can be written as

$$\begin{aligned}
&\mathrm{E}[h_{(m,p)}h^*_{(n,q)}|\boldsymbol{\delta}] \\
&\qquad = \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} f(\omega, \nu; \boldsymbol{\delta}) e^{-i\pi((m-n)\omega + (p-q)\nu)} d\omega\, d\nu && \text{(B.11)}
\end{aligned}$$

with

$$f(\omega, \nu; \boldsymbol{\delta}) = \begin{cases} \tilde{f}(\omega, \nu; \boldsymbol{\delta}), & \text{for } \omega^2 + \nu^2 \leq \pi^2, \\ 0, & \text{otherwise.} \end{cases} \tag{B.12}$$

The non-zero entries of the two dimensional spectrum are given by

$$\tilde{f}(\omega, \nu; \boldsymbol{\delta}) = \frac{\tilde{g}(\arcsin(\omega/\pi), \arcsin(\nu/(\pi\sqrt{1-\omega^2}))}{\sqrt{(\pi^2 - \omega^2)(\pi^2 - \omega^2 - \nu^2)}}. \tag{B.13}$$

That is, for a URA, the entries of the channel covariance matrix are two-dimensional Fourier coefficients of the periodic spectrum $f(\omega, \nu; \boldsymbol{\delta})$.

We can use the results for the ULA case to show that the URA covariance matrix is asymptotically equivalent to a nested circulant matrix with the eigenvalues $f(2\pi m/M_H, 2\pi p/M_V; \boldsymbol{\delta})$ where $m = 0$, …, $M_H - 1$ and $p = 0$, …, $M_V - 1$. The eigenvectors of the nested circulant matrix are given by $\boldsymbol{F}_{M_H} \otimes \boldsymbol{F}_{M_V}$ where $\boldsymbol{F}_M$ denotes the M-dimensional DFT matrix. Clearly, the asymptotic equivalence only holds if M_H and M_V both go to infinity.

B.3 Distributed Antennas

In principle, distributed antennas offer much higher worst-case SINRs than a single base station with the same number of antennas that has to cover the same area. If we distribute many antennas, there will be some antennas nearby at every location in the cell, i.e., cell-edge users do not have the same disadvantage as with a compact antenna array. However, there are many additional challenges with distribute antennas, e.g., the backhaul link and synchronization. The typical channel model for distributed antennas is quite simple: all channel coefficients are assumed to be independent. Consequently, the channel covariance matrices are diagonal and we have one variance c_{km} for each user/antenna pair. If d_{km} is the distance between user k and antenna m, then the variance c_{km} is a decreasing function of d_{km}.

Typically, we have something like

$$c_{km} = \min(c_{\max}, c_0 d_{km}^{-\alpha})$$

with a path-loss coefficient $\alpha \in [2, 4]$.

If we reasonably assume that c_{km} is bounded as the distance goes to zero, this channel model fulfills Condition 3.1. Users at different positions have different distances to the antennas and thus Condition 3.2 is also fulfilled for practical scenarios.

Appendix C

Useful Lemmas

C.1 SINR Uplink-Downlink Duality

Lemma C.1. *Consider downlink SINRs of the form*

$$\gamma_k^{dl} = \frac{\boldsymbol{w}_k^{\mathrm{H}} \boldsymbol{c}_k \boldsymbol{c}_k^{\mathrm{H}} \boldsymbol{w}_k}{1 + \sum_n \boldsymbol{w}_n^{\mathrm{H}} \boldsymbol{B}_{nk} \boldsymbol{w}_n}$$

and corresponding uplink SINRs

$$\gamma_k^{ul} = \frac{\lambda_k \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{c}_k \boldsymbol{c}_k^{\mathrm{H}} \boldsymbol{g}_k}{\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{Q} \boldsymbol{g}_k + \sum_n \lambda_n \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{B}_{kn} \boldsymbol{g}_k}$$

with positive semi-definite $\boldsymbol{B}_{kn}$ and positive definite $\boldsymbol{Q}$.

For any $\boldsymbol{g}_k$ and $\lambda_k \geq 0$ such that $\sum_k \lambda_k = P$ we can find $\boldsymbol{w}_k$ such that $\sum_k \boldsymbol{w}_k^{\mathrm{H}} \boldsymbol{Q} \boldsymbol{w}_k = P$ and $\gamma_k^{dl} = \gamma_k^{ul}$ for all k. Conversely, for any $\boldsymbol{w}_k$ with $\sum_k \boldsymbol{w}_k^{\mathrm{H}} \boldsymbol{Q} \boldsymbol{w}_k = P$ we can find $\boldsymbol{g}_k$ and $\lambda_k \geq 0$ such that $\sum_k \lambda_k = P$ and $\gamma_k^{dl} = \gamma_k^{ul}$ for all k.

Proof. Given uplink filters $\boldsymbol{g}_k$ and power allocations $\lambda_k \geq 0$ with $\sum_k \lambda_k = P$ which achieve certain uplink SINRs γ_k^{ul}, we choose precoding vectors

$$\boldsymbol{w}_k = \sqrt{p_k} \boldsymbol{g}_k$$

and consider whether the system of equations $\gamma_k^{\mathrm{dl}} = \gamma_k^{\mathrm{ul}}$ for all k has a feasible solution $\boldsymbol{p} \geq 0$. With some manipulation we get the linear system of equations

$$\lambda_k = p_k \left(\boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{Q} \boldsymbol{g}_k + \sum_n \lambda_n \boldsymbol{g}_k^{\mathrm{H}} \boldsymbol{B}_{kn} \boldsymbol{g}_k \right) - \sum_n p_n \lambda_n \boldsymbol{g}_n^{\mathrm{H}} \boldsymbol{B}_{nk} \boldsymbol{g}_n \; \forall k$$

which we can write in matrix-vector notation as

$$\boldsymbol{\lambda} = \boldsymbol{\Phi}\boldsymbol{p}$$

with

$$[\boldsymbol{\Phi}]_{kn} = \begin{cases} \boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{Q}\boldsymbol{g}_k + \sum_{\ell \neq k} \lambda_\ell \boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{B}_{k\ell}\boldsymbol{g}_k, & \text{for } k = n \\ -\lambda_n \boldsymbol{g}_n^{\mathrm{H}}\boldsymbol{B}_{nk}\boldsymbol{g}_n, & \text{else.} \end{cases}$$

Since $\boldsymbol{\Phi}$ is column-wise diagonally dominant with positive diagonal entries and negative off-diagonal entries (M-matrix), $\boldsymbol{\Phi}^{-1}$ exists and has non-negative entries. Consequently, $\boldsymbol{p} = \boldsymbol{\Phi}^{-1}\boldsymbol{\lambda}$ is non-negative. Additionally, we have

$$P = \mathbf{1}^{\mathrm{T}}\boldsymbol{\lambda} = \mathbf{1}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{p} = \sum_k p_k \boldsymbol{g}_k^{\mathrm{H}}\boldsymbol{Q}\boldsymbol{g}_k = \sum_k \boldsymbol{w}_k^{\mathrm{H}}\boldsymbol{Q}\boldsymbol{w}_k$$

which completes the first part of the proof.

For the converse we start with some $\boldsymbol{w}_k$ such that $\sum_k \boldsymbol{w}_k^{\mathrm{H}}\boldsymbol{Q}\boldsymbol{w}_k = P$. We choose $\boldsymbol{g}_k = \boldsymbol{w}_k$ and again consider the system of equations $\gamma_k^{\mathrm{dl}} = \gamma_k^{\mathrm{ul}}$ for all k. We get the linear system of equations

$$\boldsymbol{w}_k^{\mathrm{H}}\boldsymbol{Q}\boldsymbol{w}_k = \lambda_k(1 + \sum_n \boldsymbol{w}_n^{\mathrm{H}}\boldsymbol{B}_{nk}\boldsymbol{w}_k) - \sum_n \lambda_n \boldsymbol{w}_k \boldsymbol{B}_{kn}\boldsymbol{w}_k.$$

The remaining steps of the proof are analogous to the first part. □

The point of this duality is that we can use the uplink SINRs to optimize downlink precoding. In the uplink, the optimal filters $\boldsymbol{g}_k$ can be calculated analytically. We have the optimal filters

$$\boldsymbol{g}_k^\star = (\boldsymbol{Q} + \sum_n \lambda_n \boldsymbol{B}_{kn})^{-1}\boldsymbol{c}_k$$

with corresponding optimal SINR

$$\gamma_k^\star = \lambda_k \boldsymbol{c}_k^{\mathrm{H}}(\boldsymbol{Q} + \sum_n \lambda_n \boldsymbol{B}_{kn})^{-1}\boldsymbol{c}_k.$$

Thus, we only have to optimize the power allocation $\boldsymbol{\lambda}$, i.e., the number of variables reduces to one real-valued scalar per user.

C.2 Fourth Order Moments

Lemma C.2. *For two jointly Gaussian distributed vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ with zero mean, we have*

$$\mathrm{E}[|\boldsymbol{x}^{\mathrm{H}}\boldsymbol{y}|^2] = \mathrm{tr}(\mathrm{E}[\boldsymbol{x}\boldsymbol{x}^{\mathrm{H}}]\,\mathrm{E}[\boldsymbol{y}\boldsymbol{y}^{\mathrm{H}}]) + |\mathrm{tr}(\mathrm{E}[\boldsymbol{y}\boldsymbol{x}^{\mathrm{H}}])|^2\,.$$

Proof. Let $\tilde{\boldsymbol{z}} = [\boldsymbol{x}^{\mathrm{T}}, \boldsymbol{y}^{\mathrm{T}}]^{\mathrm{T}}$ with $\mathrm{E}[\tilde{\boldsymbol{z}}\tilde{\boldsymbol{z}}^{\mathrm{H}}] = \boldsymbol{C}_{\tilde{\boldsymbol{z}}}$. We have $\boldsymbol{x} = \boldsymbol{S}_{\boldsymbol{x}}\tilde{\boldsymbol{z}}$ and $\boldsymbol{y} = \boldsymbol{S}_{\boldsymbol{y}}\tilde{\boldsymbol{z}}$ with

$$\boldsymbol{S}_{\boldsymbol{x}} = [\mathbf{I}, \mathbf{0}] \quad \text{and} \quad \boldsymbol{S}_{\boldsymbol{y}} = [\mathbf{0}, \mathbf{I}]$$

and thus

$$\begin{aligned} \mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{y}\boldsymbol{y}^{\mathrm{H}}\boldsymbol{x}] &= \mathrm{E}[\tilde{\boldsymbol{z}}^{\mathrm{H}}\boldsymbol{S}_{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{S}_{\boldsymbol{y}}\tilde{\boldsymbol{z}}\tilde{\boldsymbol{z}}^{\mathrm{H}}\boldsymbol{S}_{\boldsymbol{y}}^{\mathrm{T}}\boldsymbol{S}_{\boldsymbol{x}}\tilde{\boldsymbol{z}}] \\ &= \mathrm{E}[\boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}^{\mathrm{H}}\boldsymbol{z}] \\ &= \mathrm{tr}(\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}]\boldsymbol{W}^{\mathrm{H}}) \qquad \text{(C.1)} \end{aligned}$$

where $\boldsymbol{z}$ has i.i.d. entries with zero-mean and unit-variance and

$$\boldsymbol{W} = \boldsymbol{C}_{\tilde{\boldsymbol{z}}}^{1/2}\boldsymbol{S}_{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{S}_{\boldsymbol{y}}\boldsymbol{C}_{\tilde{\boldsymbol{z}}}^{1/2}.$$

We obtain

$$\begin{aligned} \boldsymbol{e}_i^{\mathrm{T}}\,\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}]\boldsymbol{e}_j &= \mathrm{E}[z_i z_j^* \boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}\boldsymbol{z}] \\ &= \sum_{r,c} w_{rc}\,\mathrm{E}[z_i z_j^* z_r z_c^*] \\ &= \begin{cases} w_{ij} & \text{for} \quad i \neq j \\ w_{ii} + \sum_r w_{rr} & \text{for} \quad i = j. \end{cases} \end{aligned}$$

Incorporating this result into (C.1) yields

$$\begin{aligned} \mathrm{E}[\left|\boldsymbol{x}^{\mathrm{H}}\boldsymbol{y}\right|^2] &= \mathrm{tr}(\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}\boldsymbol{W}\boldsymbol{z}\boldsymbol{z}^{\mathrm{H}}]\boldsymbol{W}^{\mathrm{H}}) \\ &= \mathrm{tr}((\boldsymbol{W} + \mathrm{tr}(\boldsymbol{W})\,\mathbf{I})\boldsymbol{W}^{\mathrm{H}}) \\ &= \mathrm{tr}(\boldsymbol{W}\boldsymbol{W}^{\mathrm{H}}) + |\mathrm{tr}(\boldsymbol{W})|^2 \\ &= \mathrm{tr}(\boldsymbol{S}_{\boldsymbol{x}}\boldsymbol{C}_{\boldsymbol{z}}\boldsymbol{S}_{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{S}_{\boldsymbol{y}}\boldsymbol{C}_{\boldsymbol{z}}\boldsymbol{S}_{\boldsymbol{y}}^{\mathrm{T}}) + \left|\mathrm{tr}(\boldsymbol{S}_{\boldsymbol{y}}\boldsymbol{C}_{\boldsymbol{z}}\boldsymbol{S}_{\boldsymbol{x}}^{\mathrm{T}})\right|^2 \end{aligned}$$

which is the desired result. □

C.3 Asymptotic Analysis

Lemma C.3. *For two n-dimensional zero-mean, jointly Gaussian distributed random vectors $\boldsymbol{x}^{(n)} \in \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{x}}^{(n)})$ and $\boldsymbol{y}^{(n)} \in \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{y}}^{(n)})$ we have*

$$\lim_{n\to\infty} \frac{\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}}{n} - \frac{\mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}]}{n} = 0,$$

as long as the covariance matrices $\boldsymbol{C}_{\boldsymbol{x}}$, $\boldsymbol{C}_{\boldsymbol{y}}$ and the matrix $\boldsymbol{A}$ have bounded spectral norm.

Proof. With Lemma C.2 we calculate the variance

$$\begin{aligned}\mathrm{var}(\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}) &= \mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}\boldsymbol{y}^{\mathrm{H}}\boldsymbol{A}^{\mathrm{H}}\boldsymbol{x}] - \left|\mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}]\right|^2 \\ &= \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{x}}\boldsymbol{A}\boldsymbol{C}_{\boldsymbol{y}}\boldsymbol{A}^{\mathrm{H}}) + \left|\mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}]\right|^2 - \left|\mathrm{E}[\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}]\right|^2 \\ &= \mathrm{tr}(\boldsymbol{C}_{\boldsymbol{x}}\boldsymbol{A}\boldsymbol{C}_{\boldsymbol{y}}\boldsymbol{A}^{\mathrm{H}}).\end{aligned}$$

Thus

$$\begin{aligned}\mathrm{var}(\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}/n) &= \frac{1}{n^2}\,\mathrm{var}(\boldsymbol{x}^{\mathrm{H}}\boldsymbol{A}\boldsymbol{y}) \\ &\leq \frac{\|\boldsymbol{C}_{\boldsymbol{x}}\|\,\|\boldsymbol{A}\|^2\,\|\boldsymbol{C}_{\boldsymbol{y}}\|\,\mathrm{tr}(\mathbf{I})}{n^2} = \frac{\|\boldsymbol{C}_{\boldsymbol{x}}\|\,\|\boldsymbol{A}\|^2\,\|\boldsymbol{C}_{\boldsymbol{y}}\|}{n}\end{aligned}$$

which goes to zero for $n \to \infty$ if the matrices have bounded norm. □

Appendix D

Projected Gradient Methods

Since several optimization problems discussed in this work are solved with projected gradient methods, we want to provide a short summary of such methods.

D.1 Basic Projected Gradient Method

The projected gradient method is a straightforward extension of the classical gradient descent method to optimization problems with simple constraints. With simple constraints we mean a constraint set, for which the orthogonal projection of a point onto the set can be calculated efficiently. Preferably, we are also able to calculate the projection onto the tangent cone at any point on the boundary of the constraint set.

Suppose we have an optimization problem

$$\min_{\boldsymbol{x}\in\mathcal{C}} f(\boldsymbol{x}) \tag{D.1}$$

with the convex constraint set $\mathcal{C} \subset \mathbb{R}^n$. We denote the orthogonal projection onto the constraint set by

$$P_{\mathcal{C}}(\boldsymbol{y}) = \operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{C}} \|\boldsymbol{x}-\boldsymbol{y}\|^2 . \tag{D.2}$$

Further, $P_{\mathcal{T}_\mathcal{C}}(\boldsymbol{g};\boldsymbol{x})$ denotes the projection of a vector $\boldsymbol{g}$ onto the tangent cone of $\mathcal{C}$ at point $\boldsymbol{x} \in \mathcal{C}$. For notational convenience we introduce the gradient $\boldsymbol{g}(\boldsymbol{x}) = \nabla f(\boldsymbol{x})$ and the projected gradient $\boldsymbol{g}_\mathcal{C}(\boldsymbol{x}) = P_{\mathcal{T}_\mathcal{C}}(\boldsymbol{g}(\boldsymbol{x});\boldsymbol{x})$, which is the gradient at $\boldsymbol{x}$ projected onto the tangent cone of $\mathcal{C}$ at $\boldsymbol{x}$.

The projected gradient method starts from an initial $\boldsymbol{x} \in \mathcal{C}$ with iterative updates of the form

$$\boldsymbol{x} \leftarrow P_{\mathcal{C}}(\boldsymbol{x} - s\boldsymbol{g}(\boldsymbol{x})) \tag{D.3}$$

with a suitable step-size s.

Algorithm 5 Backtracking line-search with Armijo's rule

```
Require: Inital, sufficiently large $s$ and constants $\beta, \sigma \in (0, 1)$, e.g.
  $\beta = 0.5$ and $\sigma = 0.001$.
  while $h_{\boldsymbol{x}}(s) - h_{\boldsymbol{x}}(0) > \sigma h'_{\boldsymbol{x}}(0)s$ do
      $s \leftarrow \beta s$
  end while
```

The step-size is typically found by an approximate minimization of the one-dimensional function

$$h_{\boldsymbol{x}}(s) = P_{\mathcal{C}}(\boldsymbol{x} - s\boldsymbol{g}(\boldsymbol{x})). \tag{D.4}$$

In our implementation we use a backtracking line-search with Armijo's rule to calculate the step-size which is described in Alg. 5. This method requires the derivative $h'_{\boldsymbol{x}}(s)$ at $s = 0$ which is given by

$$h'_{\boldsymbol{x}}(0) = -\boldsymbol{g}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{g}_{\mathcal{C}}(\boldsymbol{x}) = -\left\|\boldsymbol{g}_{\mathcal{C}}(x)\right\|^2. \tag{D.5}$$

D.2 Projected Quasi-Newton Methods

For unconstrained problems (Quasi-)Newton methods can be significantly more efficient than the simple gradient descent method. It is possible to formulate projected versions of those methods that can then be applied to constrained problems instead of the projected gradient method. The iterative update of the current optimizer $\boldsymbol{x}$ is similar:

$$\boldsymbol{x} \leftarrow P_{\mathcal{C}}(\boldsymbol{x} + s\boldsymbol{d}(\boldsymbol{x})) \tag{D.6}$$

for a search direction $\boldsymbol{d}(\boldsymbol{x})$.

For unconstrained Quasi-Newton methods, the search direction is of the form $\boldsymbol{d}(\boldsymbol{x}) = -\boldsymbol{B}\boldsymbol{g}(\boldsymbol{x})$, where $\boldsymbol{B} \succ \boldsymbol{0}$, which guarantees an improving direction since $\boldsymbol{g}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d}(\boldsymbol{x}) < 0$. In the constrained case, we use $\boldsymbol{d}(\boldsymbol{x}) = -\boldsymbol{B}\boldsymbol{g}_{\mathcal{C}}(\boldsymbol{x})$. If we define a one-dimensional function equivalent to before

$$h_{\boldsymbol{x}}(s) = P_{\mathcal{C}}(\boldsymbol{x} - s\boldsymbol{d}(\boldsymbol{x})) \tag{D.7}$$

we get

$$h'_{\boldsymbol{x}}(0) = -\boldsymbol{d}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{g}_{\mathcal{C}}(\boldsymbol{x}) < 0. \tag{D.8}$$

That is, the direction $\boldsymbol{d}(\boldsymbol{x}) = -\boldsymbol{B}\boldsymbol{g}_{\mathcal{C}}(\boldsymbol{x})$ based on the projected gradient and a positive definite matrix $\boldsymbol{B}$ is an improving direction. Thus, we can again apply the backtracking line-search from Alg. 5 to find a suitable step-size.

In Alg. 6 we summarize the steps of a projected Quasi-Newton algorithm. This method is an extension of earlier work on Quasi-Newton methods with simple box constraints [51] to a more general class of constraints. We state the method here without proof of convergence or convergence rate, but we think that such an analysis could be done analogously to that in the cited earlier work. Note that the matrix $\boldsymbol{B}_k$ in the algorithm does not have to be formed explicitly. We can for example use the L-BFGS update [52] that stores an implicit representation of $\boldsymbol{B}_k$ that only requires a small amount of memory.

Algorithm 6 Constrained Quasi-Newton algorithm that makes use of projections onto the constraint set and projections onto the tangent cone of the constraint set

```
Require: Initial $\boldsymbol{x}_1 \in \mathcal{C}$ and $\boldsymbol{B}_1 \succ \mathbf{0}$
  for $k = 1, 2, \dots$ do
    Calculate the search direction using $\boldsymbol{B}$
        $\boldsymbol{d}_k \leftarrow -\boldsymbol{B}_k\boldsymbol{g}_{\mathcal{C}}(\boldsymbol{x}_k)$
    Use backtracking line search to get the step-size $s_k$, start with $s_k \leftarrow 1$
    Update the optimizer
        $\boldsymbol{x}_{k+1} \leftarrow P_{\mathcal{C}}(\boldsymbol{x}_k + s_k\boldsymbol{d}_k)$
    Calculate (or update) the matrix $\boldsymbol{B}_{k+1}$ using $\boldsymbol{x}_1, \dots, \boldsymbol{x}_{k+1}$ and $\boldsymbol{g}(\boldsymbol{x}_1), \dots, \boldsymbol{g}(\boldsymbol{x}_{k+1})$
  end for
```

D.3 Common Constraint Sets

The two constraint sets which are relevant for this work are the ball $\mathcal{B}_r = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\| \leq r\}$ and the simplex $\Delta_p = \{\boldsymbol{x} \in \mathbb{R}^n : \mathbf{1}^{\mathrm{T}}\boldsymbol{x} =$

$p, \boldsymbol{x} \geq \mathbf{0}\}$. The projection onto the ball is pretty straightforward. We have

$$P_{\mathcal{B}_r}(\boldsymbol{x}) = \begin{cases} \boldsymbol{x} & \text{if } \|\boldsymbol{x}\| \leq r \\ r\boldsymbol{x}/\|\boldsymbol{x}\| & \text{otherwise.} \end{cases} \tag{D.9}$$

Since the tangent cone at a point $\boldsymbol{x}$ on the boundary of $\mathcal{B}_r$ is simply the half space $\{\boldsymbol{g} : \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x} \leq 0\}$, the projection onto the tangent cone is given by

$$P_{\mathcal{T}_{\mathcal{B}_r}}(\boldsymbol{g}; \boldsymbol{x}) = \begin{cases} \boldsymbol{g} & \text{if } \|\boldsymbol{x}\| < r \text{ or } \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x} \leq 0 \\ \boldsymbol{x}\frac{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{g}}{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}} & \text{if } \|\boldsymbol{x}\| = r \text{ and } \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x} > 0. \end{cases} \tag{D.10}$$

Projection onto the simplex Δ_p is slightly more complicated. There exists a semi-analytical solution to the projection problem, which is similar to the waterfilling procedure. The method is described in Alg. 7.

Algorithm 7 Iterative, waterfilling-like procedure for the projection onto the simplex Δ_p

```
Require: Vector $\boldsymbol{x} \in R^n$ that is projected and parameter $p > 0$
  $\boldsymbol{x}' \leftarrow$ entries of $\boldsymbol{x}$ sorted in ascending order
  for $i$ from 1 to $n$ do
    $w^\star \leftarrow \frac{\sum_{j=i}^{n}[\boldsymbol{x}']_j - p}{n-i+1}$
    if $[\boldsymbol{x}']_i > w^\star$ then
      break
    end if
  end for
  $\boldsymbol{x} \leftarrow [\boldsymbol{x} - w^\star\mathbf{1}]_+$
```

The projection onto the tangent cone follows along similar lines and is described in Alg. 8. Since we are not aware that this projection onto the tangent cone of a simplex has been previously discussed, we will derive the method in the following. The tangent cone of the simplex Δ_p at a point $\boldsymbol{x}$ is given by

$$\mathcal{T}_{\Delta_p}(\boldsymbol{x}) = \{\boldsymbol{g} : \mathbf{1}^{\mathrm{T}}\boldsymbol{g} = 0, [\boldsymbol{g}]_i \geq 0 \; \forall i \in \mathcal{A}(\boldsymbol{x})\} \tag{D.11}$$

with the index-set of active constraints $\mathcal{A}(\boldsymbol{x}) = \{i : [\boldsymbol{x}]_i = 0\}$.

The optimization problem corresponding to the orthogonal projection onto the tangent cone is thus of the form

$$\min_{\boldsymbol{g}} \|\boldsymbol{g} - \boldsymbol{y}\|^2, \text{s.t. } \mathbf{1}^{\mathrm{T}}\boldsymbol{g} = 0, [\boldsymbol{g}]_i \geq 0 \; \forall i \leq k, \tag{D.12}$$

Algorithm 8 Iterative procedure for the projection onto the tangent cone of $\mathcal{T}_{\Delta_p}$ at $\boldsymbol{x}$

```
Require: Vector $\boldsymbol{x} \in \Delta_p$ and a vector $\boldsymbol{g} \in \mathbb{R}^n$ that is projected
Require: Bijective map $\pi : \{1, \dots, |\mathcal{A}(\boldsymbol{x})|\} \mapsto \mathcal{A}(\boldsymbol{x})$ such that
  $[\boldsymbol{x}]_{\pi(i)} \leq [\boldsymbol{x}]_{\pi(j)}$ for $i < j \leq |\mathcal{A}(\boldsymbol{x})|$.
  for $i$ from 1 to $|\mathcal{A}(\boldsymbol{x})|$ do
      $w^\star \leftarrow \frac{\sum_{j \notin \mathcal{A}(\boldsymbol{x})} [\boldsymbol{x}]_j + \sum_{j=i}^{|\mathcal{A}(\boldsymbol{x})|} [\boldsymbol{x}]_{\pi(j)} - p}{n-i+1}$
      if $[\boldsymbol{x}']_i > w^\star$ then
          break
      end if
  end for
  $[\boldsymbol{x}]_i \leftarrow \max([\boldsymbol{x}]_i - w^\star, 0)\ \forall i \in \mathcal{A}(\boldsymbol{x})$
  $[\boldsymbol{x}]_i \leftarrow [\boldsymbol{x}]_i - w^\star\ \forall i \notin \mathcal{A}(\boldsymbol{x})$
```

where we fixed the index set to $\mathcal{A}(\boldsymbol{x}) = \{1, \dots, k\}$ without loss of generality. We will further assume that for $i < j \leq k$ we have $[\boldsymbol{y}]_i \leq [\boldsymbol{y}]_j$. We also do not lose generality with this assumption, because of the symmetry of the constraints. To find the optimal solution for a general vector, we permute the elements such that they fulfill the requirements, solve the problem in (D.12), and then rearrange the elements of the optimal solution with the inverse of the initial permutation.

The KKT conditions of (D.12) are given by

$$[\boldsymbol{g}]_i \geq 0\ \forall i \leq k \tag{D.13}$$

$$\mathbf{1}^{\mathrm{T}} \boldsymbol{g} = 0 \tag{D.14}$$

$$[\boldsymbol{g}]_i - [\boldsymbol{y}]_i + \lambda - \mu_i = 0\ \forall i \leq k \tag{D.15}$$

$$[\boldsymbol{g}]_i - [\boldsymbol{y}]_i + \lambda = 0\ \forall i > k \tag{D.16}$$

$$[\boldsymbol{g}]_i \mu_i = 0\ \forall i \in \mathcal{A}. \tag{D.17}$$

Since the problem is strictly convex, there exists a tuple $(\boldsymbol{g}^\star, \lambda^\star, \boldsymbol{\mu}^\star)$ that fulfills the KKT conditions with a unique optimizer $\boldsymbol{g}^\star$.

Given the optimal dual variable $\lambda^\star$ (let's call it the water level) we have

$$[\boldsymbol{g}^\star]_i = [\boldsymbol{y}]_i - \lambda^\star\ \forall i > k \tag{D.18}$$

and

$$[\boldsymbol{g}^\star]_i = \left[[\boldsymbol{y}]_i - \lambda^\star\right]_+\ \forall i \leq k. \tag{D.19}$$

We note that, if the constraint $[\boldsymbol{g}]_i \geq 0$ is active for some $i \leq k$, then the constraints for all $j < i$ are active, too.

Now suppose we know the constraints that are active at the optimum. That is, we know the index $k^\star \leq k$ such that $[\boldsymbol{g}^\star]_i = 0 \; \forall i \leq k^\star$ and $[\boldsymbol{g}^\star]_i > 0 \; \forall k^\star < i \leq k$. Since $\mathbf{1}^\mathrm{T}\boldsymbol{g}^\star = 0$ we can evaluate the optimal water level

$$\lambda^\star = \frac{1}{n - k^\star} \sum_{i=k^\star+1}^{n} [\boldsymbol{y}]_i. \tag{D.20}$$

Now the algorithm iteratively evaluates the water-level for all hypotheses $k^\star = 0, \ldots, k$ starting from $k^\star = 0$. If as expected $[\boldsymbol{y}]_{k^\star+1} - \lambda^\star > 0$ the optimal water-level is found, otherwise $k^\star$ is increased by one.

Bibliography

[1] B.M. Hochwald, T.L. Marzetta, and Vahid Tarokh. Multiple-antenna channel hardening and its implications for rate feedback and scheduling. *IEEE Transactions on Information Theory*, 50:1893–1909, September 2004.

[2] David Tse and Pramod Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[3] Ezio Biglieri, Robert Calderbank, Anthony Constantinides, Andrea Goldsmith, Arogyaswami Paulraj, and H. Vincent Poor. *MIMO Wireless Communications*. Cambridge University Press, January 2007.

[4] J. Vieira, F. Rusek, O. Edfors, S. Malkowsky, L. Liu, and F. Tufvesson. Reciprocity Calibration for Massive MIMO: Proposal, Modeling, and Validation. *IEEE Transactions on Wireless Communications*, 16:3042–3056, May 2017.

[5] Martin Schubert and Holger Boche. QoS-Based Resource Allocation and Transceiver Optimization. *Foundations and Trends® in Communications and Information Theory*, 2:383–529, July 2006.

[6] M. Schubert and H. Boche. A Generic Approach to QoS-Based Transceiver Optimization. *IEEE Trans. Commun.*, 55:1557–1566, August 2007.

[7] A. Adhikary, A. Ashikhmin, and T. L. Marzetta. Uplink Interference Reduction in Large-Scale Antenna Systems. *IEEE Transactions on Communications*, 65:2194–2206, May 2017.

[8] Trinh Van Chien, Emil Björnson, and Erik G. Larsson. Joint Pilot Design and Uplink Power Allocation in Multi-Cell Massive MIMO Systems. *arXiv:1707.03072 [cs, math]*, July 2017.

[9] Emil Björnson, Jakob Hoydis, and Luca Sanguinetti. Massive MIMO has Unlimited Capacity. *arXiv:1705.00538 [cs, math]*, May 2017.

[10] F. Rusek, D. Persson, Buon Kiong Lau, E.G. Larsson, T.L. Marzetta, O. Edfors, and F. Tufvesson. Scaling Up MIMO: Opportunities and Challenges with Very Large Arrays. *IEEE Signal Process. Mag.*, 30:40 –60, January 2013.

[11] Lizhong Zheng and D.N.C. Tse. Communication on the Grassmann manifold: A geometric approach to the noncoherent multiple-antenna channel. *IEEE Trans. Inf. Theory*, 48:359–383, February 2002.

[12] David Neumann, Thomas Wiese, Michael Joham, and Wolfgang Utschick. A Generalized Matched Filter Framework for Massive MIMO Systems. *arXiv:1707.09940 [cs, math]*, July 2017.

[13] E. Björnson, E. G. Larsson, and M. Debbah. Massive MIMO for Maximal Spectral Efficiency: How Many Users and Pilots Should Be Allocated? *IEEE Trans. Wireless Commun.*, 15:1293–1308, February 2016.

[14] M. Medard. The effect upon channel capacity in wireless communications of perfect and imperfect knowledge of the channel. *IEEE Trans. Inf. Theory*, 46:933–946, May 2000.

[15] B. Hassibi and B.M. Hochwald. How much training is needed in multiple-antenna wireless links? *IEEE Transactions on Information Theory*, 49:951 – 963, April 2003.

[16] Andreas Dotzler, Maximilian Riemensberger, and Wolfgang Utschick. Minimax Duality for MIMO Interference Networks. 7:19, March 2016.

[17] D. Neumann, A. Gruendinger, M. Joham, and W. Utschick. CDI Rate-balancing with Per-base-station Constraints. In *Workshop on Smart Antennas*, pages 1–6, March 2016.

[18] P. Viswanath and D. N. C. Tse. Sum capacity of the vector Gaussian broadcast channel and uplink-downlink duality. *IEEE Transactions on Information Theory*, 49:1912–1921, August 2003.

[19] E. Visotsky and U. Madhow. Optimum beamforming using transmit antenna arrays. In *1999 IEEE 49th Vehicular Technology Conference (Cat. No.99CH36363)*, volume 1, pages 851–856 vol.1, July 1999.

[20] F. Rashid-Farrokhi, L. Tassiulas, and K. J. R. Liu. Joint optimal power control and beamforming in wireless networks using antenna arrays. *IEEE Transactions on Communications*, 46:1313–1324, October 1998.

[21] Haifan Yin, David Gesbert, Miltiades Filippou, and Yingzhuang Liu. A Coordinated Approach to Channel Estimation in Large-Scale Multiple-Antenna Systems. *IEEE J. Sel. Areas Commun.*, 31:264 –273, February 2013.

[22] A Liu and V. Lau. Hierarchical Interference Mitigation for Massive MIMO Cellular Networks. *IEEE Trans. Signal Process.*, 62:4786–4797, September 2014.

[23] A. Adhikary, Junyoung Nam, Jae-Young Ahn, and G. Caire. Joint Spatial Division and Multiplexing – The Large-Scale Array Regime. *IEEE Trans. Inf. Theory*, 59:6441–6463, October 2013.

[24] D. Neumann, M. Joham, and W. Utschick. On MSE Based Receiver Design for Massive MIMO. In *Conf. Syst. Commun. and Coding*, pages 1–6, February 2017.

[25] H. Q. Ngo and E. G. Larsson. No Downlink Pilots Are Needed in TDD Massive MIMO. *IEEE Transactions on Wireless Communications*, 16:2921–2935, May 2017.

[26] D. Neumann, K. Shibli, M. Joham, and W. Utschick. Joint covariance matrix estimation and pilot allocation in massive MIMO systems. pages 1–6, May 2017.

[27] Robert M. Gray. Toeplitz and Circulant Matrices: A Review. *Foundations and Trends® in Communications and Information Theory*, 2:155–239, January 2006.

[28] Hien Quoc Ngo, A. Ashikhmin, Hong Yang, E.G. Larsson, and T.L. Marzetta. Cell-Free Massive MIMO: Uniformly great service for everyone. In *Int. Workshop Signal Process. Adv. Wireless Commun.*, pages 201–205, June 2015.

[29] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta. Cell-Free Massive MIMO Versus Small Cells. *IEEE Trans. Wireless Commun.*, 16:1834–1850, March 2017.

[30] E. Bjornson and B. Ottersten. A Framework for Training-Based Estimation in Arbitrarily Correlated Rician MIMO Channels With Rician Disturbance. *IEEE Trans. Signal Process.*, 58:1807–1820, March 2010.

[31] R.D. Yates. A framework for uplink power control in cellular radio systems. *IEEE Journal on Selected Areas in Communications*, 13:1341–1347, September 1995.

[32] Hien Quoc Ngo and E.G. Larsson. EVD-based channel estimation in multicell multiuser MIMO systems with very large antenna arrays. In *Int. Conf. Acoust., Speech, Signal Process.*, March 2012.

[33] R.R. Müller, L. Cottatellucci, and M. Vehkapera. Blind Pilot Decontamination. *IEEE J. Sel. Areas Commun.*, 8:773–786, October 2014.

[34] David Neumann, Michael Joham, and Wolfgang Utschick. Channel Estimation in Massive MIMO Systems. *arXiv:1503.08691 [cs, math]*, March 2015.

[35] A.K. Jagannatham and B.D. Rao. Whitening-rotation-based semi-blind MIMO channel estimation. *IEEE Trans. Signal Process.*, 54:861–869, March 2006.

[36] Charles W. Fox and Stephen J. Roberts. A tutorial on variational Bayesian inference. *Artificial Intelligence Review*, 38:85–95, August 2012.

[37] 3GPP. Spatial channel model for Multiple Input Multiple Output (MIMO) simulations (Release 12). Technical report, 3GPP, 2014.

[38] David Neumann, Andreas Gründinger, Michael Joham, and Wolfgang Utschick. Pilot Coordination for Large-Scale Multi-Cell TDD Systems. In *Workshop on Smart Antennas*, pages 1–6, March 2014.

[39] E. Björnson, L. Sanguinetti, and M. Debbah. Massive MIMO with imperfect channel covariance information. In *2016 50th*

Asilomar Conference on Signals, Systems and Computers, pages 974–978, November 2016.

[40] David Neumann, Michael Joham, Lorenz Weiland, and Wolfgang Utschick. Low-Complexity Computation of LMMSE Channel Estimates in Massive MIMO. In *Workshop on Smart Antennas*, pages 1–6, March 2015.

[41] A. Dembo. The relation between maximum likelihood estimation of structured covariance matrices and periodograms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34:1661–1662, December 1986.

[42] T. W. Anderson. Asymptotically Efficient Estimation of Covariance Matrices with Linear Structure. *The Annals of Statistics*, 1:135–141, 1973.

[43] J. P. Burg, D. G. Luenberger, and D. L. Wenger. Estimation of structured covariance matrices. *Proceedings of the IEEE*, 70:963–974, September 1982.

[44] S. Haghighatshoar and G. Caire. Massive MIMO Channel Subspace Estimation From Low-Dimensional Projections. *IEEE Trans. Signal Process.*, 65:303–318, January 2017.

[45] S. Haghighatshoar and G. Caire. Low-complexity massive MIMO subspace tracking from low-dimensional projections. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–7, May 2017.

[46] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[47] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 315–323, Fort Lauderdale, FL, USA, 2011.

[48] Michael Frigge, David C. Hoaglin, and Boris Iglewicz. Some Implementations of the Boxplot. *The American Statistician*, 43:50–54, 1989.

[49] C. E. Shannon. A Mathematical Theory of Communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5:3–55, January 2001.

[50] Charles L. Epstein. How well does the finite Fourier transform approximate the Fourier transform? *Communications on Pure and Applied Mathematics*, 58:1421–1435, October 2005.

[51] Dongmin Kim, Suvrit Sra, and Inderjit S Dhillon. *A new projected quasi-newton approach for the nonnegative least squares problem*. 2006.

[52] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, August 1989.