D3.1

First assessment of baseband processing requirements for MaMi systems

Project number:	619086
Project acronym:	**MAMMOET**
Project title:	Massive MIMO for Efficient Transmission
Project Start Date:	1 January, 2014
Duration:	36 months
Programme:	FP7/2007-2013
Deliverable Type:	Report
Reference Number:	ICT-619086-D3.1
Workpackage:	WP 3
Due Date:	31 December, 2014
Actual Submission Date:	13 January, 2015
Responsible Organisation:	ULUND
Editor:	Ove Edfors
Dissemination Level:	PU
Revision:	1.0
Abstract:	Base-band processing requirements for Massive MIMO systems are discussed and outlined. Alternative transmission and reception strategies for Massive MIMO are discussed in conjunction with associated processing requirements and hardware platform/implementation options.
Keywords:	Massive MIMO, baseband processing, hardware, precoding, algorithms, accelerators, architecture

This project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 619086.

Editor

Ove Edfors (ULUND)

Contributors (ordered according to beneficiary numbers)

Emil Björnson, Christopher Mollén & Erik G. Larsson (LIU)
Claude Desset, André Bourdoux, Ubaid Ahmad & Liesbet Van der Perre (IMEC)
Ove Edfors, Liang Liu, Steffen Malkowsky, Hemanth Prabhu & Joao Vieira (ULUND)
Wim Dehaene (KUL)
Eleftherios Karipidis (EAB)

Executive Summary

Baseband processing is one of the critical parts of Massive MIMO systems and the efficiency at which it can be performed is an important factor. Due to the structure of Massive MIMO with many coherent transmit and receive streams at the base station antenna array, there are many options to consider when designing the baseband processing.

First, there is a certain amount of processing needed to perform initial synchronization, such as reciprocity calibration of transceivers. This category of processing is important but performed only at certain (well separated) time instants and, as such, not highly time-critical in its nature. After initial synchronization, there are several processing tasks that have to be performed in real-time while communication is ongoing. This includes channel estimation, design/calculation of transmit precoding and receive combining matrices, signal precoding, data detection, etc. Not only do we need to find appropriate and optimized algorithms for the different Massive MIMO processing tasks, but the available implementation options also have to be investigated for each algorithm. One important aspect in this context is that certain parts of the processing can be performed in a distributed way close to the antenna elements, which makes this processing highly scalable with the number of base station antennas, while other parts need to be performed at a central location where signals from multiple antenna elements are available. Further, appropriate hardware platforms have to be chosen so that various requirements on system flexibility and efficiency can be met.

To set the stage for future work in WP3, this deliverable addresses the above topics by collecting available knowledge among partners and results from preliminary investigations performed in the first few months of the MAMMOET project. Conclusions include:

- Preliminary investigations indicate that the difference between single-carrier and OFDM based Massive MIMO is not large. Neither in terms of resulting system performance nor in terms of processing complexity.
- Algorithms for Massive MIMO processing allow for many trade-offs to achieve high performance and low complexity, such as reduced-accuracy matrix inversions.
- The structure of Massive MIMO allows for new and efficient means to reduce power variations in the transmitted signals, without large sacrifices in performance, making it possible to use highly power-efficient non-linear amplifiers.
- Since Massive MIMO is still in its infancy, early implementations need to be flexible to allow changes when processing strategies improve. MAMMOET will start with highly flexible software-defined radio platforms and move towards more optimized implementations.
- Due to the processing of a large number of antenna streams, memory requirements and data shuffling capacity play an important role. It is both possible and important to use the specific advantages of Massive MIMO, such as imperfections being hidden/suppressed by averaging-effects over many antennas, to make appropriate optimizations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This deliverable constitutes a first assessment of the baseband processing requirements for MaMi systems. We approach the topic by first giving an algorithm overview, followed by a discussions on different processing hardware platforms and how to map algorithms to different hardware. Finally we summarize our initial assessments.

The basic concept of Massive MIMO is shown in Fig.1.1, where a base station is using M antennas to spatially multiplex $K \ll M$ single-antenna terminals. The success of such a spatial multiplex, in both up- and down-link, relies on several important concepts. The base station needs good enough propagation channel knowledge in both directions, on which efficient down-link precoders and up-link detectors can be based. Since acquisition of channel-state information (CSI) is generally infeasible in the down-link [34], massive MIMO systems typically rely on channel reciprocity, up-link channel estimation, and time-division duplex (TDD). With the massive number of channels to estimate between base station and mobile stations, a long-enough channel coherence time is needed to allow for efficient operation. The accuracy at which we can estimate the channel and the time interval over which it can be assumed constant bring fundamental limitations to massive MIMO [34].

Many of the algorithms required for massive MIMO are also found in other wireless communication systems, such as traditional MIMO systems, with the essential difference that a much larger number of transceiver chains have to be processed in parallel. While this expands the processing complexity in one dimension, properties of massive MIMO also allows many of the processing algorithms to be linear rather than non-linear, which helps to balance the massive increase of transceiver chains. The algorithms discussed and evaluated in Chapter 2 are all central in the context of massive MIMO.

When implementing any communication system, it is essential to select the correct hardware platforms. Depending on the requirements on flexibility and energy efficiency, different choices come into play. For the prototype development and proof-of-concept work in MAMMOET it is quite natural to use as flexible platforms as possible. A typical choice would include a combination of software defined radios (SDRs) complemented by additional computational resources. Due to the large number of transceiver chains and high requirements on synchronized real-time processing, often with an exchange of large amounts of data between processing nodes, it is important that the chosen platform has a high enough data shuffling capacity. These issues are discussed in more detail in Chapter 3.

An important part of selecting the appropriate hardware platform deals with how massive MIMO algorithms can and will be mapped onto computational hardware resources. In some cases it is quite sufficient that low-performing generic processors execute an algorithm, while in other cases much more advanced combinations of accelerators and/or specific computational

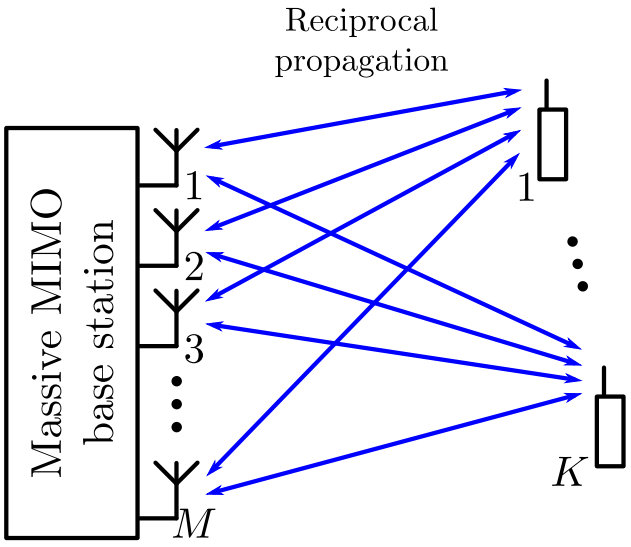


Figure 1.1: Massive MIMO base station using M antennas to perform spatial multiplex of K single-antenna mobile stations.

structures are required. An important part of the work in MAMMOET is to find algorithms that ensure high communication performance, which can be efficiently mapped onto appropriate hardware and thereby make massive MIMO a proven alternative for future communications standards. The first steps in this direction in MAMMOET are discussed in Chapter 4.

A short summary of Massive MIMO baseband processing requirements is given in Chapter 5.

Chapter 2

Algorithm Overview

A massive MIMO system relies on many different algorithms, for everything from initial system synchronization and parameter acquisition to precoding, detection, and user scheduling. Many of these will be addressed in the MAMMOET project and here we present an initial overview of some of the important algorithm categories, starting with reciprocity calibration and moving on to channel estimation, precoding, detection and scheduling of pilots and users. Finally we make a comparison of massive MIMO based on single- or multi-carrier techniques.

2.1 Reciprocity Calibration

Multi-user MIMO systems operating with a large number of base station (BS) antennas, render explicit downlink channel estimation as inefficient. Basically, one can not afford to transmit pilot symbols from every antenna in the downlink, receive them at the terminal side, and feed back the channel state information (CSI) to the BS so that it can calculate suitable precoding coefficients. Such a procedure would degrade the spectral efficiency significantly considering the amount of feedback information required, due to the large number of BS antennas.

An approach to compute proper precoding coefficients is to operate in time division duplex (TDD) mode, and rely on the reciprocity of the channel based on uplink pilots. However, it is generally agreed that the propagation channel is reciprocal, but the different transceiver radio frequency (RF) chains are not. Hence, in order to use reciprocity and calculate the precoding coefficients, one needs to know or estimate the differences in the (frequency) responses. Figure 2.1 illustrates how a typical duplex channel is experienced by a signal in a wireless transmission.

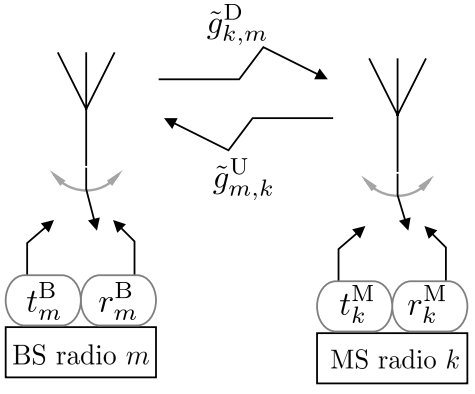


Figure 2.1: Illustration of uplink/downlink radio channels.

Let the uplink and downlink narrow-band radio channels between the BS and MS be denoted as

$$\begin{aligned} g^{\mathrm{U}}_{m,k} &= r^{\mathrm{B}}_{m}\ \tilde{g}^{\mathrm{U}}_{m,k}\ t^{\mathrm{M}}_{k} \\ g^{\mathrm{D}}_{k,m} &= r^{\mathrm{M}}_{k}\ \tilde{g}^{\mathrm{D}}_{k,m}\ t^{\mathrm{B}}_{m}, \end{aligned} \tag{2.1}$$

where $m \in [1, ..., M]$ is the BS antenna index, $k \in [1, ..., K]$ is the mobile station (MS) antenna index, r^{B} and r^{M} represent the BS and MS receiver RF chains, t^{B} and t^{M} represent the BS and MS transmitter RF chains, and $\tilde{g}^{\mathrm{U}}$ and $\tilde{g}^{\mathrm{D}}$ are the uplink and the downlink propagation channels, respectively. Note that all terms in (2.1) are complex random variables due to the narrow-band nature of the model.

Assuming perfect reciprocity of the propagation channel, i.e. $\tilde{g}^{\mathrm{U}}_{m,k} = \tilde{g}^{\mathrm{D}}_{k,m}$, a relation between the uplink and downlink radio channels can be established as

$$b_{m,k} = \frac{r^{\mathrm{M}}_{k}\ \tilde{g}^{\mathrm{D}}_{k,m}\ t^{\mathrm{B}}_{m}}{r^{\mathrm{B}}_{m}\ \tilde{g}^{\mathrm{U}}_{m,k}\ t^{\mathrm{M}}_{k}} = \frac{r^{\mathrm{M}}_{k}\ t^{\mathrm{B}}_{m}}{r^{\mathrm{B}}_{m}\ t^{\mathrm{M}}_{k}}. \tag{2.2}$$

Here we call $b_{m,k}$ the *calibration coefficient* between radios m and k, since if known, one can overcome the channel non-reciprocity and compute the downlink channel based on the uplink channel estimates.

Let us now introduce the channel between two BS radios as

$$h_{\ell,m} = r^{\mathrm{B}}_{\ell}\ \tilde{h}_{\ell,m}\ t^{\mathrm{B}}_{m} \tag{2.3}$$

where $\ell \neq m$, $\ell \in [1, ..., M]$, and $\tilde{h}_{\ell,m}$ is the propagation channel between the BS antennas ℓ and m. We introduce the calibration coefficient between BS radios as

$$h_{\ell,m} = b_{m\to\ell}\ h_{m,\ell}, \tag{2.4}$$

which by assuming perfect reciprocity yields[1]

$$b_{m\to\ell} = \frac{h_{\ell,m}}{h_{m,\ell}} = \frac{r^{\mathrm{B}}_{\ell} t^{\mathrm{B}}_{m}}{r^{\mathrm{B}}_{m} t^{\mathrm{B}}_{\ell}} = \frac{1}{b_{\ell\to m}}. \tag{2.5}$$

One of the main contributions from [58] was an internal reciprocity calibration method for a massive MIMO base station. The method has two main points as basis:

1. $$b_{m,k} = \frac{t^{\mathrm{B}}_{m}}{r^{\mathrm{B}}_{m}}\ \frac{r^{\mathrm{M}}_{k}}{t^{\mathrm{M}}_{k}} = \frac{r^{\mathrm{B}}_{n}\ t^{\mathrm{B}}_{m}}{r^{\mathrm{B}}_{m}\ t^{\mathrm{B}}_{n}}\ \frac{r^{\mathrm{M}}_{k}\ t^{\mathrm{B}}_{n}}{r^{\mathrm{B}}_{n}\ t^{\mathrm{M}}_{k}} = b_{m\to n} b_{n,k}. \tag{2.6}$$

   i.e., calibration between radios m and k can also be achieved if their forward and reverse channels to another BS radio n are jointly processed. Throughout this analysis, we set $n = 1$ for convenience and denote this radio as the reference radio.

2. As long as each downlink channel estimate from all BS antennas deviates from the real one by the same complex factor, the resulting downlink beam pattern shape does not change. Thus, since the transceiver response of any terminal shows up as a constant factor to all BS antennas, its contribution can be omitted from the calibration procedure.

[1]Note that we denote the calibration coefficients between two BS radios using "$\to$" to distinguish from the calibration coefficient between a BS radio and an MS which uses ",".

Combining (2.2) with the previous two points yields

$$g^{\mathrm{D}}_{k,m} = b_{m,k}\; g^{\mathrm{U}}_{m,k} \tag{2.7}$$

$$\overset{1)}{=} b_{m\to 1}\; b_{1,k}\; g^{\mathrm{U}}_{m,k} \tag{2.8}$$

$$\overset{2)}{\Leftrightarrow} g^{'\mathrm{D}}_{k,m} = b_{m\to 1} g^{\mathrm{U}}_{m,k} \tag{2.9}$$

where $g^{'\mathrm{D}}_{k,m}$ is a relative downlink channel that absorbs $b_{1,k}$. Thus relative downlink channels can be obtained by multiplying the respective uplink channels with their respective calibration coefficients to a reference radio. The authors in [56] took this approach one step forward in order to calibrate access points of a distributed MIMO network. A novelty in their approach was

$$g^{'\mathrm{D}}_{k,m} = b_{m\to 1}\; g^{\mathrm{U}}_{m,k} \tag{2.10}$$

$$\Leftrightarrow g^{''\mathrm{D}}_{k,m} = b_{m}\; g^{\mathrm{U}}_{m,k} \tag{2.11}$$

where $b_m = \frac{r^B_m}{t^B_m} = \frac{1}{b_{m\to 1}}\frac{t^B_1}{r^B_1}$, and $g^{''\mathrm{D}}_{k,m}$ is another relative downlink channel. This relative equivalence not only relaxes the double-indexing overhead, but allows different calibration coefficients to be treated as mutually independent.

Note that the absolute reference to the terminals was lost in the derivation step 2), which makes $b_{m\to 1}$ or b_m valid calibration coefficients up to a complex factor. Thus, downlink pilots still need to be broadcast through the beam to compensate for this uncertainty, as well as for the RF chain responses of the terminals. The overhead of these supplementary pilots is reported as very small [25]. The calibration coefficients can be valid over long periods of time if BS radios share the same synchronization references. For the case of the BS detailed in Chapter 3, this coherence time can range up to hours.

One way to estimate the calibration coeficients b_m is sounding the M antennas one-by-one by transmitting a pilot symbol from each one and receiving on the other $M-1$ silent antennas. For simplicity, we use a pilot symbol $p = 1$. Let $y_{m,\ell}$ denote the signal received at antenna m when transmitting p at antenna ℓ. It follows that the received signals between any pair of antennas can be written as

$$\begin{aligned}\begin{bmatrix} y_{\ell,m} \\ y_{m,\ell} \end{bmatrix} &= \tilde{h}_{\ell,m} \begin{bmatrix} r^{\mathrm{B}}_{\ell}\, t^{\mathrm{B}}_{m} \\ r^{\mathrm{B}}_{m}\, t^{\mathrm{B}}_{\ell} \end{bmatrix} + \begin{bmatrix} n_{\ell,m} \\ n_{m,\ell} \end{bmatrix} \\ &= \alpha_{\ell,m} \begin{bmatrix} b_{\ell} \\ b_{m} \end{bmatrix} + \begin{bmatrix} n_{\ell,m} \\ n_{m,\ell} \end{bmatrix},\end{aligned} \tag{2.12}$$

where $\alpha_{\ell,m} = t^B_\ell t^B_m \tilde{h}_{\ell,m} = t^B_\ell t^B_m \tilde{h}_{m,\ell}$ due to reciprocity, and $[n_{\ell,m}\; n_{m,\ell}]^T$ is a vector of independent zero-mean circularly symmetric complex Gaussian distributed random variables, each one with variance N_0.

To access the accuracy of a real-array calibration, we simulated the calibration of a 5 by 20 planar patch array (see [21] for array description). Figure 2.2 shows the different calibration performances obtained from different estimators [62] varying on the number of received signals used for calibration purposes.

We now provide a rough estimate of the calibration $\mathrm{SNR}_{\mathrm{Cal}}$ regime where a massive MIMO BS may operate. If such BS yields similar specifications as the BS in the LuMaMi testbed, the $\mathrm{SNR}_{\mathrm{Cal}}$ regime is given by:

$$\mathrm{SNR}_{\mathrm{cal}} = P_{RX} - N \approx 80\mathrm{dB}, \tag{2.13}$$

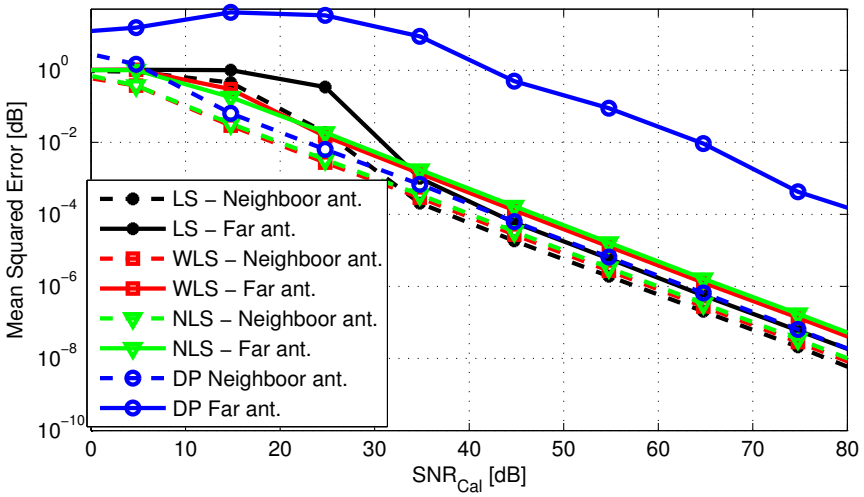


Figure 2.2: Mean squared error (MSE) of the calibration coefficients computed for the neighbor and the farthest antenna from the reference. See [62] for derivation of the different estimators.

where $P_{RX} = -15$ dBm is the maximum allowed receive power per RF-chain,

$$\mathrm{N} = 10\log_{10}(kBT_0) + N_F + G \approx -95\mathrm{dB}, \tag{2.14}$$

is the receiver noise power, k is Boltzmann's constant, $B = 20$ MHz is the channel bandwidth, $T_0 = 290°K$ is the standardized room temperature, $N_F = 6$ dB is the noise figure of the receiver chain, and $G = 0$ dB is a normalized amplifier gain. In practice, hardware limitations as ADC resolution and frequency harmonics will degrade the calibration performance. However, a margin of tens of dBs is still available to compensate for such impairments while still achieving "good enough" performance.

2.2 Channel estimation

Each BS uses its multitude of antennas for phase-coherent precoding in the downlink and receive combining in the uplink, as described in Sections 2.3 and 2.4 respectively. The main idea is to adaptively amplify desired signals to/from each user and simultaneously reject interfering signals. This requires some knowledge of the user channels. Such channel state information (CSI) is typically acquired by measuring the received uplink signals when the users send known pilot signals. This is a challenging task in cellular networks, where the transmission resources are reused across cells, because the pilot signals are then inevitably affected by inter-cell interference. This so-called *pilot contamination* limits the quality of the acquired CSI and the ability to reject inter-cell interference (unless intricate subspace methods can be used for decontamination, as initially described in [40]).

The impact of pilot contamination is usually studied under the assumption that exactly the same pilot signals are used in all cells. However, this is a theoretical simplification that needs to relaxed in practical implementations; particularly because using only a subset of the pilot signals can greatly improve the estimation quality and end performance [9]. Consequently, this section provides channel estimation results for arbitrary pilot allocation.

Let each coherence interval consist of S symbols. The pilot signals span B of these symbols, where $1 \leq B \leq S$, and are assumed to be sent in the beginning of the interval for notational

convenience. Each pilot signal can be represented by a deterministic vector $\mathbf{v} \in \mathbb{C}^B$ and the fixed per-symbol power implies that all entries have unit magnitude: $|[\mathbf{v}]_s| = 1$ for $s \in \{1, \ldots, B\}$. We assume that all pilot signals originate from a predefined *pilot book*

$$\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_B\} \quad \text{where} \quad \mathbf{v}_{b_1}^{\mathrm{H}} \mathbf{v}_{b_2} = \begin{cases} B, & b_1 = b_2, \\ 0, & b_1 \neq b_2. \end{cases} \tag{2.15}$$

Hence, the B pilot signals form an orthogonal basis and can, for example, be taken as the columns of a discrete Fourier transform (DFT) matrix [7].

Suppose there are J cells and K scheduled/active users per cell. Let $\mathbf{h}_{jlk} \in \mathbb{C}^N$ denote the channel response between user k in cell l and BS j. Moreover, let $\mathbf{v}_{i_{lk}}$ be the pilot signal allocated to user k in cell l, where $i_{lk} \in \{1, \ldots, B\}$ is the index in the pilot book $\mathcal{V}$. The received signal $\mathbf{Y}_j \in \mathbb{C}^{M \times B}$ at BS j from pilot signaling can then be modeled as

$$\mathbf{Y}_j = \sum_{l=1}^{J} \sum_{k=1}^{K} \sqrt{p_{lk}} \mathbf{h}_{jlk} \mathbf{v}_{i_{lk}}^{\mathrm{H}} + \mathbf{N}_j, \tag{2.16}$$

where p_{lk} is the average transmit power of user k in cell l and $\mathbf{N}_j$ is additive circularly complex Gaussian noise where each element is independent and has zero mean and variance N_0.

There is a variety of methods to estimate the unknown parameters from noise observations [23]. The classical methods assume that the unknown parameters are deterministic while the noise/interference is stochastic with some (semi-)known distributions, while the Bayesian methods assume that also the unknown parameters are stochastic with some (semi-)known distributions. Since the channel estimation in massive MIMO is intrinsically affected by pilot contamination, one needs to model the interfering channels as stochastic and thus it makes sense to also model the desired channels as stochastic. This section will therefore only deal with Bayesian estimation methods.

Suppose that $\mathbb{E}\{\mathbf{h}_{jlk}\} = \bar{\mathbf{h}}_{jlk}$ is the constant line-of-sight component of the channel $\mathbf{h}_{jlk}$ and that each element of $\mathbf{h}_{jlk}$ has the variance β_{jlk}, for each user k in cell l to BS j. If there is no line-of-sight component for a specific channel, then $\bar{\mathbf{h}}_{jlk} = \mathbf{0}$. Under these statistical assumptions, the linear minimum mean squared error (LMMSE) estimator of the effective channel $\mathbf{h}_{jlk}^{\text{eff}} = \sqrt{p_{lk}} \mathbf{h}_{jlk}$ at BS j is

$$\hat{\mathbf{h}}_{jlk}^{\text{eff}} = \sqrt{p_{lk}} \bar{\mathbf{h}}_{jlk} + p_{lk} \beta_{jlk} \left(\mathbf{v}_{i_{lk}}^{\mathrm{H}} \boldsymbol{\Psi}_j^{-1} \otimes \mathbf{I}_M \right) \mathrm{vec}(\tilde{\mathbf{Y}}_j), \tag{2.17}$$

where $\otimes$ is the Kronecker product, $\mathrm{vec}(\cdot)$ is the vectorization operator (i.e., stacking the columns of a matrix), and

$$\tilde{\mathbf{Y}}_j = \mathbf{Y}_j - \sum_{l=1}^{J} \sum_{k=1}^{K} \sqrt{p_{lk}} \bar{\mathbf{h}}_{jlk} \mathbf{v}_{i_{lk}}^{\mathrm{H}} \tag{2.18}$$

$$\boldsymbol{\Psi}_j = \sum_{\ell=1}^{J} \sum_{m=1}^{K} p_{\ell m} \beta_{j\ell m} \mathbf{v}_{i_{\ell m}} \mathbf{v}_{i_{\ell m}}^{\mathrm{H}} + N_0 \mathbf{I}_B. \tag{2.19}$$

The estimation error covariance matrix $\mathbf{C}_{jlk} \in \mathbb{C}^{M \times M}$ is given by

$$\begin{aligned} \mathbf{C}_{jlk} &= \mathbb{E}\left\{ (\mathbf{h}_{jlk}^{\text{eff}} - \hat{\mathbf{h}}_{jlk}^{\text{eff}})(\mathbf{h}_{jlk}^{\text{eff}} - \hat{\mathbf{h}}_{jlk}^{\text{eff}})^{\mathrm{H}} \right\} \\ &= p_{lk} \beta_{jlk} \left(1 - p_{lk} \beta_{jlk} \mathbf{v}_{i_{lk}}^{\mathrm{H}} \boldsymbol{\Psi}_j^{-1} \mathbf{v}_{i_{lk}} \right) \mathbf{I}_M \end{aligned} \tag{2.20}$$

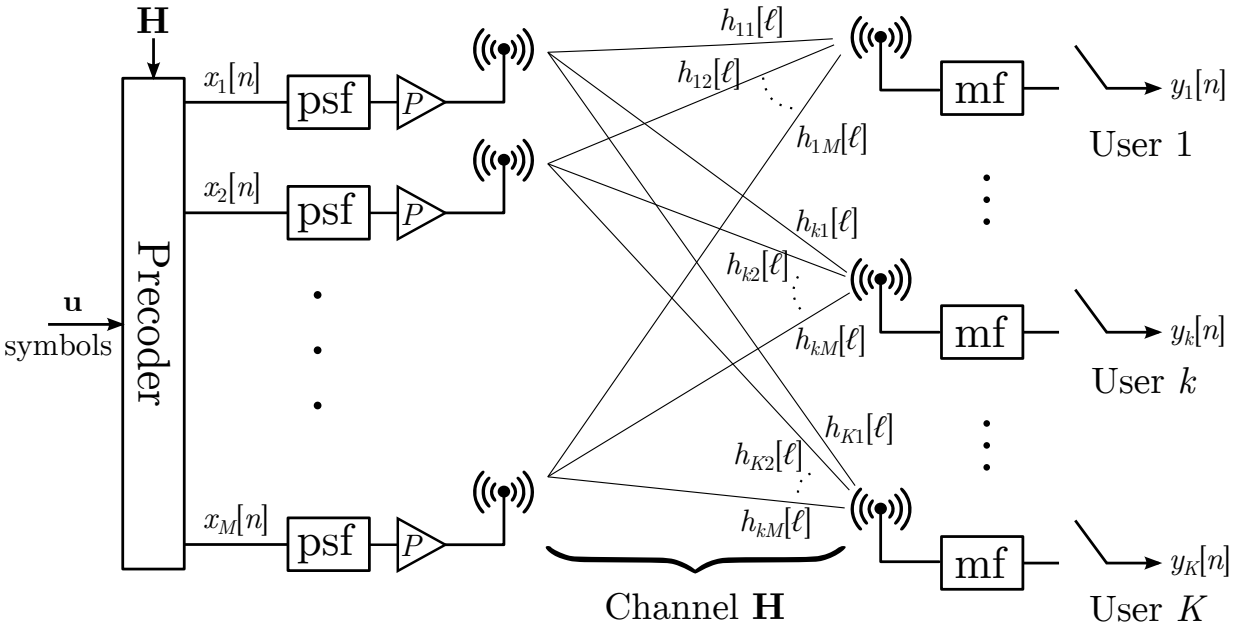


Figure 2.3: The downlink of a massive multi-user MIMO system.

and the mean-squared error (MSE) per element is

$$\mathrm{MSE}_{jlk} = \frac{1}{M}\mathrm{tr}(\mathbf{C}_{jlk}) = p_{lk}\beta_{jlk}\left(1 - p_{lk}\beta_{jlk}\mathbf{v}_{i_{lk}}^{\mathrm{H}}\boldsymbol{\Psi}_j^{-1}\mathbf{v}_{i_{lk}}\right). \tag{2.21}$$

Note that the LMMSE estimator in (2.17) is based on a minimal statistical characterization; only the mean value $\bar{\mathbf{h}}_{jlk}$ and the variance per channel element β_{jlk} is assumed to be known. These first and second order moments can be easily measured from the received signal and will typically vary at a much slower rate than the useful signal (e.g., 100 times slower according to the measurements in [63]). The stochastic distribution can be any that satisfies these main properties, and the channel responses can either be independent over the antennas or correlated.

If the channels would be circularly symmetric complex Gaussian as $\mathbf{h}_{jlk} \sim \mathcal{CN}(\bar{\mathbf{h}}_{jlk}, \beta_{jlk}\mathbf{I}_M)$, then (2.17) is also the minimum mean squared error (MMSE) estimator; that is, not only the *linear* estimator that minimizes the MSE but also the only estimator that minimizes the MSE. Notice that this statistical distribution is known as Rayleigh fading when $\bar{\mathbf{h}}_{jlk} = \mathbf{0}$ and Rician fading when $\bar{\mathbf{h}}_{jlk} \neq \mathbf{0}$.

2.3 Downlink precoding

A massive MIMO base station with M antennas, shown in Figure 2.3, is considered. It serves K single-antenna users over a frequency-selective channel modelled as an FIR filter with L taps. For simplicity in exposition, it is assumed in this section that the channel estimation in Section 2.2 provides the true channels. Signals are transmitted over a time interval $[-L, N-1]$, where N is the number of transmitted symbols.

The transmit signals at time n are denoted $\mathbf{x}[n] \triangleq (x_1[n], \ldots, x_M[n])^{\mathsf{T}}$, where $x_m[n]$ is the transmit signal at antenna m. The transmit signals $\mathbf{x}[-L], \ldots, \mathbf{x}[-1]$ are called the *prefix*. It is assumed that $\mathsf{E}[\,\|\mathbf{x}[n]\|^2\,] = 1$, $\forall n$. The transmit signals are pulse shape filtered with a

root-Nyquist pulse $p(t)$ into continuous-time signals

$$x_m(t) = \sum_{n=-L}^{N-1} x_m[n]p(t-nT), \quad \forall m, \tag{2.22}$$

and amplified to transmit power before being broadcast. It is assumed that $p(t)$ is time limited and that adjacent blocks of transmit signals do not interfere with each other.

The signals received at time n by the users, after matched filtering and sampling, are denoted $\mathbf{y}[n] \triangleq (y_1[n], \ldots, y_K[n])^{\mathsf{T}}$, where $y_k[n]$ is the signal received at user k. The signals received before $n = 0$ and after $n = N-1$ are discarded.

The channel is described by the $K \times M$-matrices $\mathbf{H}[\ell]$, $\ell = 0, \ldots, L-1$, whose (k,m)-th elements $\{h_{km}[0], \ldots, h_{km}[L-1]\}$ form the impulse response from antenna m to user k. In this section, it is assumed that the base station knows the channels perfectly and that each user knows the statistics of its channel. However, the estimated channel matrices from Section 2.2 could be used instead of the channel matrices $\mathbf{H}[\ell]$. The received signal vector at time n is given by

$$\mathbf{y}[n] = \sqrt{P} \sum_{\ell=0}^{L-1} \mathbf{H}[\ell]\mathbf{x}[n-\ell] + \mathbf{w}[n], \tag{2.23}$$

where $\mathbf{w}[n] \sim \mathcal{CN}(0, \mathbf{I}_K)$ is an i.i.d. zero-mean white Gaussian noise vector with covariance matrix $\mathbf{I}_K$ (the $K \times K$-identity matrix). The factor P thus represents the transmit power normalized by the noise variance.

Throughout this section, it is assumed that the prefix is *cyclic*:

$$\mathbf{x}[n] = \mathbf{x}[N+n], \quad \text{for } n = -L, \ldots, -1. \tag{2.24}$$

This results in a concise mathematical description of the channel, see for example [61]. More precisely, define the block-circulant $KN \times MN$-matrix

$$\mathbf{H} \triangleq \begin{pmatrix} \mathbf{H}[0] & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}[L-1] & \cdots & \mathbf{H}[1] \\ \mathbf{H}[1] & \mathbf{H}[0] & \mathbf{0} & & & & \mathbf{H}[2] \\ \vdots & & \ddots & & & & \vdots \\ \mathbf{0} & & \cdots & \mathbf{0} & \mathbf{H}[L-1] & \cdots & \mathbf{H}[0] \end{pmatrix} \tag{2.25}$$

then the input-output relation of the channel is given by

$$\mathbf{y} = \sqrt{P}\mathbf{H}\mathbf{x} + \mathbf{w}, \tag{2.26}$$

where

$$\mathbf{x} \triangleq (\mathbf{x}^{\mathsf{T}}[0], \ldots, \mathbf{x}^{\mathsf{T}}[N-1])^{\mathsf{T}}, \tag{2.27}$$

$$\mathbf{y} \triangleq (\mathbf{y}^{\mathsf{T}}[0], \ldots, \mathbf{y}^{\mathsf{T}}[N-1])^{\mathsf{T}}, \tag{2.28}$$

$$\mathbf{w} \triangleq (\mathbf{w}^{\mathsf{T}}[0], \ldots, \mathbf{w}^{\mathsf{T}}[N-1])^{\mathsf{T}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{KN}). \tag{2.29}$$

With a cyclic prefix, (2.23) is also easily given in the frequency domain. Let $\mathbf{F} \in \mathbb{C}^{N \times N}$ be the N-point discrete Fourier transform (DFT) transform with $\frac{1}{\sqrt{N}} e^{-j2\pi(n-1)(n'-1)/N}$ on its (n,n')-th position and define the two unitary matrices $\mathbf{F}_K \triangleq \mathbf{F} \otimes \mathbf{I}_K$ and $\mathbf{F}_M \triangleq \mathbf{F} \otimes \mathbf{I}_M$. Then

$$\tilde{\mathbf{H}} = \mathbf{F}_K \mathbf{H} \mathbf{F}_M^{\mathsf{H}} \tag{2.30}$$

is the block-diagonal matrix, whose diagonal blocks

$$\tilde{\mathbf{H}}[n] = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{L-1} \mathbf{H}[\ell] e^{-j2\pi\ell n/N}, \quad n = 0, \ldots, N-1, \tag{2.31}$$

are the DFTs of $\{\mathbf{H}[\ell]\}$. Let

$$\tilde{\mathbf{x}} \triangleq (\tilde{\mathbf{x}}^{\mathsf{T}}[0], \ldots, \tilde{\mathbf{x}}^{\mathsf{T}}[N-1])^{\mathsf{T}} = \mathbf{F}_M \mathbf{x} \tag{2.32}$$

$$\tilde{\mathbf{y}} \triangleq (\tilde{\mathbf{y}}^{\mathsf{T}}[0], \ldots, \tilde{\mathbf{y}}^{\mathsf{T}}[N-1])^{\mathsf{T}} = \mathbf{F}_K \mathbf{y} \tag{2.33}$$

be the DFTs of the transmit and receive signals respectively. The relation between $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is then

$$\tilde{\mathbf{y}}[n] = \sqrt{P}\tilde{\mathbf{H}}[n]\tilde{\mathbf{x}}[n] + \tilde{\mathbf{w}}[n], \quad n = 0, \ldots, N-1, \tag{2.34}$$

where $\tilde{\mathbf{w}}[n] \sim \mathcal{CN}(0, \mathbf{I}_K)$.

2.3.1 Linear precoding

The transmit signals have to be chosen such that the users receive the symbols intended for them, without being too disturbed by the symbols intended for other users. Each symbol duration $n \in \{0, \ldots, N-1\}$, a complex symbol $\sqrt{\rho}u_k[n]$ is transmitted to each user k. If the symbols have unit energy, such that $\mathsf{E}[\,|u_k[n]|^2\,] = 1$, then the positive factor ρ/P can be called the *array gain*. Denote the vector of all symbols at time n by $\mathbf{u}[n] \triangleq (u_1[n], \ldots, u_K[n])^{\mathsf{T}}$ and all these vectors by $\mathbf{u} \triangleq (\mathbf{u}^{\mathsf{T}}[0], \ldots, \mathbf{u}^{\mathsf{T}}[N-1])^{\mathsf{T}}$. A mapping $(\mathbf{u}, \mathbf{H}) \mapsto \mathbf{x}$ that ensures that the users simultaneously receive what they should is called a *precoder*.

Linear precoding can be done by weighting the symbols either in the time domain by a precoding matrix $\mathbf{G} \in \mathbb{C}^{MN \times KN}$:

$$\mathbf{x}_{\text{SC}} = \mathbf{G}\mathbf{u} \tag{2.35}$$

or in the frequency domain by a precoding matrix $\tilde{\mathbf{G}} \in \mathbb{C}^{MN \times KN}$ followed by a transform to the time domain

$$\mathbf{x}_{\text{OFDM}} = \mathbf{F}_M^{\mathsf{H}} \tilde{\mathbf{G}} \mathbf{u}. \tag{2.36}$$

The time domain transmission in (2.35) is referred to as *single-carrier (SC) transmission* and the frequency domain transmission in (2.36) as *orthogonal frequency-division multiplexing* (OFDM). For the linear precoders considered in this document, it holds that

$$\mathbf{G} = \mathbf{F}_M^{\mathsf{H}} \tilde{\mathbf{G}} \mathbf{F}_K. \tag{2.37}$$

Maximum-Ratio Transmission

In maximum-ratio transmission (MRT), the precoding matrix is given by

$$\begin{aligned} \mathbf{G}_{\text{MRT}} &= \alpha_{\text{MRT}} \mathbf{H}^{\mathsf{H}} \text{ or} \\ \tilde{\mathbf{G}}_{\text{MRT}} &= \alpha_{\text{MRT}} \tilde{\mathbf{H}}^{\mathsf{H}}, \end{aligned} \tag{2.38}$$

where α_{MRT} is a normalizing scalar. Since $\|\mathbf{H}\|_{\mathsf{F}} = \|\tilde{\mathbf{H}}\|_{\mathsf{F}}$, α_{MRT} is the same in both cases.

MRT maximizes the array gain of the transmission, but *interference* (undesired symbols intended to other users) will still be present in the received signal since there is no active interference mitigation. In typical scenarios (e.g., line-of-sight propagation and non-line-of-sight Rayleigh fading), MRT achieves interference suppression passively with higher number of base station antennas since the user channels are quasi-orthogonal in the limit of infinitely many antennas [34].

Practically for SC transmission, the precoding scheme results in M different L-tap filters that can be placed locally at each antenna, thus enabling distributed signal processing. For OFDM, the precoding has to be done in blocks, but it can still be done locally at each antenna.

Zero-Forcing Precoding

A precoding scheme that nulls all the interference, both intersymbol interference and interuser interference, is called *zero-forcing* (ZF). The precoding matrices of ZF are given by the pseudo-inverse of the channel

$$\begin{aligned} \mathbf{G}_{\mathrm{ZF}} &= \alpha_{\mathrm{ZF}} \mathbf{H}^{\mathsf{H}} (\mathbf{H}\mathbf{H}^{\mathsf{H}})^{-1} \text{ or} \\ \tilde{\mathbf{G}}_{\mathrm{ZF}} &= \alpha_{\mathrm{ZF}} \tilde{\mathbf{H}}^{\mathsf{H}} (\tilde{\mathbf{H}}\tilde{\mathbf{H}}^{\mathsf{H}})^{-1}, \end{aligned} \tag{2.39}$$

where α_{ZF} is a normalizing scalar.

The main difference between ZF and MRT is the matrix inversion, which provides the desired interference suppression. The computation of large inverses can be a major source of complexity, which requires an efficient hardware implementation. This is further discussed in Section 4.4.1

Regularized Zero-Forcing Precoding

There also exists a regularized version of the ZF precoder, *regularized zero-forcing* (RZF), whose precoding matrix is given by [10]

$$\begin{aligned} \mathbf{G}_{\mathrm{RZF}} &= \alpha_{\mathrm{RZF}} \mathbf{H}^{\mathsf{H}} (\mathbf{H}\mathbf{H}^{\mathsf{H}} + \tfrac{K}{P}\mathbf{I}_{KN})^{-1} \text{ or} \\ \tilde{\mathbf{G}}_{\mathrm{RZF}} &= \alpha_{\mathrm{RZF}} \tilde{\mathbf{H}}^{\mathsf{H}} (\tilde{\mathbf{H}}\tilde{\mathbf{H}}^{\mathsf{H}} + \tfrac{K}{P}\mathbf{I}_{KN})^{-1}, \end{aligned} \tag{2.40}$$

where α_{RZF} is a normalizing scalar. This precoder is also known as the *MMSE precoder* since, among all linear precoders, it is the precoder that minimizes $\mathsf{E}\left[\|\mathbf{u} - \frac{1}{\sqrt{\rho}}\mathbf{y}\|^2 \right]$ with respect to $\mathbf{x}$ and $\rho \in \mathbb{R}^+$, for unit-energy transmit vectors. The optimal linear precoder, with respect to a given performance metric, is generally very computationally expensive to compute, but it has a structure similar to RZF [10]; thus, RZF can be considered as the state-of-the-art linear precoder in terms of providing high performance with a reasonable computational complexity.

Due to the block-diagonalization property (2.30), the inversion of block-circulant matrices can be done in the frequency domain in a computationally simple way by inverting the smaller diagonal blocks. This makes the computations in (2.39) and (2.40) feasible, both for SC and OFDM transmission. Furthermore, in massive MIMO, not all N blocks on the diagonal have to be inverted. It is sufficient to invert a smaller number

$$N' \approx 50KL/M \ll N, \tag{2.41}$$

of blocks. The number N', that does not depend on N, becomes smaller the more antennas the base station is equipped with. More precisely, consider the λ-th superdiagonal block $\mathbf{G}[\lambda] \in \mathbb{C}^{M\times K}$ of $\mathbf{G}_{\mathrm{ZF}}$. Simulations have shown that the energy $\|\mathbf{G}[\lambda]\|_{\mathsf{F}}^2$ of the off-diagonal blocks rapidly falls off to zero for $\lambda \notin [0, L-1]$, see Figure 2.4. Therefore, precoding with respect to a

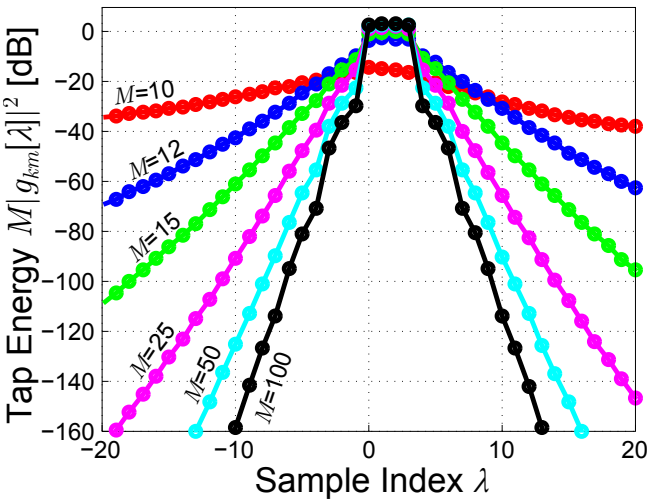


Figure 2.4: The normalized energy of the filter taps of a ZF precoder for SC transmission over a frequency-selective 4-tap channel. The base station serves $K = 10$ users.

matrix $\mathbf{H}' \in \mathbb{C}^{KN' \times MN'}$ of smaller dimension captures the most significant blocks close to the diagonal and will give a good approximation of $\mathbf{G}_{\text{ZF}}$ and $\mathbf{G}_{\text{RZF}}$. This has two implications:

1. The computational complexity of channel inversion does not depend on the block length in massive MIMO.

2. SC transmission with pre-equalization of the channel can be practically implemented in massive MIMO with FIR filters with a small number of taps.

The relation in (2.41) has only been observed in simulations. Intuitively, the rapid decay can be explained by studying the matrix multiplication $\mathbf{H}\mathbf{H}^{\mathsf{H}}$ and observing that, due to channel hardening, $\mathbf{H}[\ell]\mathbf{H}^{\mathsf{H}}[\ell] \approx M\mathbf{I}_K$ for i.i.d. Rayleigh fading and each element (k, k') in $\left[\mathbf{H}[\ell]\mathbf{H}^{\mathsf{H}}[\ell']\right]_{k,k'} \sim \mathcal{O}(\sqrt{M})$, for big M and $k \neq k'$. The product $\mathbf{H}\mathbf{H}^{\mathsf{H}}$ is therefore, with high probability, a diagonally dominant band-matrix, whose inverse also should be diagonally dominant with very small off-diagonal elements.

Energy Normalization and Control

To ensure that the power constraint on the transmit signal is fulfilled, the precoding matrices $\mathbf{G}$ and $\tilde{\mathbf{G}}$ have to be normalized. This is done by the factor α in (2.38), (2.39) and (2.40). There are two ways to choose α:

1. The same α is used for all channel realizations to ensure that $\mathsf{E}[\,\|\mathbf{G}\|_{\mathsf{F}}^2\,] = N$, so called *long-term power normalization*.

2. A new α is chosen for each new channel realization to ensure that $\|\mathbf{G}\|_{\mathsf{F}}^2 = N$, so called *short-term power normalization*.

In practice, there is no reason to do long-term power normalization, because it will lead to an increased variation in the transmit power over different channel realizations, whereas short-term power normalization will not. In the analysis of massive MIMO, long-term power normalization is nevertheless often assumed to simplify the mathematics, since the increase in power variations due to long-term power normalization is small due to channel hardening.

Looking at the block-diagonal precoding matrices $\tilde{\mathbf{G}}$ in OFDM, the n-th diagonal block can be expressed as $\tilde{\mathbf{G}}[n] = (\tilde{\mathbf{g}}_1[n] \cdots \tilde{\mathbf{g}}_K[n]) \in \mathbb{C}^{M \times K}$. The column $\tilde{\mathbf{g}}_k[n] \in \mathbb{C}^M$ describes:

1. The spatial directivity $\frac{\tilde{\mathbf{g}}_k[n]}{\|\tilde{\mathbf{g}}_k[n]\|}$ of the signal intended for user k.

2. The transmission power $\|\tilde{\mathbf{g}}_k[n]\|^2$ allocated for the signal intended for user k.

The MRT, ZF, and RZF precoding schemes mainly define the spatial directivities, while the power allocation is more implicit; MRT allocates power proportionally to the short-term channel gains (i.e., users with strong channels get more power), while ZF allocates power *inversely* proportional to the short-term channel gains (i.e., users with *weak* channels get more power).

The transmission power on this n-th subcarrier can be modified and controlled by a diagonal matrix $\mathbf{P}[n] = \text{diag}(p_1[n], \ldots, p_K[n])$ where $p_k[n]$ is a power allocation coefficient for user k. Consequently, each block of the precoding matrix in OFDM is then changed to

$$\tilde{\mathbf{G}}[n]\mathbf{P}^{1/2}[n] \tag{2.42}$$

and selecting α to satisfy $\mathsf{E}\left[\ \|\tilde{\mathbf{G}}\|_{\mathsf{F}}^2\ \right] = N$. The channel hardening implies that $\mathbf{P}[n]$ can usually be the same irrespective of n. The power allocation can be selected to maximize some performance metric, as exemplified in the next section. Although the description above was given for OFDM, similar power allocation concepts can be applied to SC transmission.

2.3.2 Linear precoders and power allocation

When comparing MRT, ZF and MMSE precoders, each of them is expected to provide its optimum performance in the absence of power normalization constraints on antennas, users, or subcarriers. This means that the pseudo-inverse or conjugate of the channel is applied without other constraint than the total transmitted power. However, the MRT precoder was found to be improved by applying additional power normalization constraints, especially when normalizing the OFDM air interface over subcarriers in order to require a constant output power spectral density, despite the reduction in degrees of freedom brought by this normalization constraint. This effect is visible in Figure 2.5 for a 64×4 massive MIMO set-up, where the received SNR is defined in expectation over the channel matrix assuming normalized total output power per user and non-coherent addition of the different transmitter antennas. This SNR definition illustrates the benefit of the beamforming gain from massive MIMO precoding, as the system can operate close to or even below 0 dB SNR. In order to understand the role of power normalization and further improve the MRT procoder, we have investigated how the MRT power allocation can be optimized over subcarriers.

The objective is to minimize the BER for MRT precoding, and see how far the performance of this simple precoder can be from the more complex ZF or MMSE precoders, in view of optimizing the trade-off between BER performance and digital baseband complexity.

Theoretical solution

In OFDM systems, a solution to minimize the BER already existed in the literature. However, this general solution must be adapted to the particular massive MIMO case. A BER minimization algorithm can be found in [47]. It has be chosen as a low-complexity suboptimal solution enabling a closed-form solution by using an approximation of the BER expression rather than the exact BER expression. Moreover, this suboptimal algorithm is has a low implementation complexity. This algorithm was initially developed in [47] for SIMO, assuming diversity recombination (MRC) at the receiver. It works exactly in the same way for SISO as far as the power allocation algorithm is concerned. The only difference is the different equivalent channel model

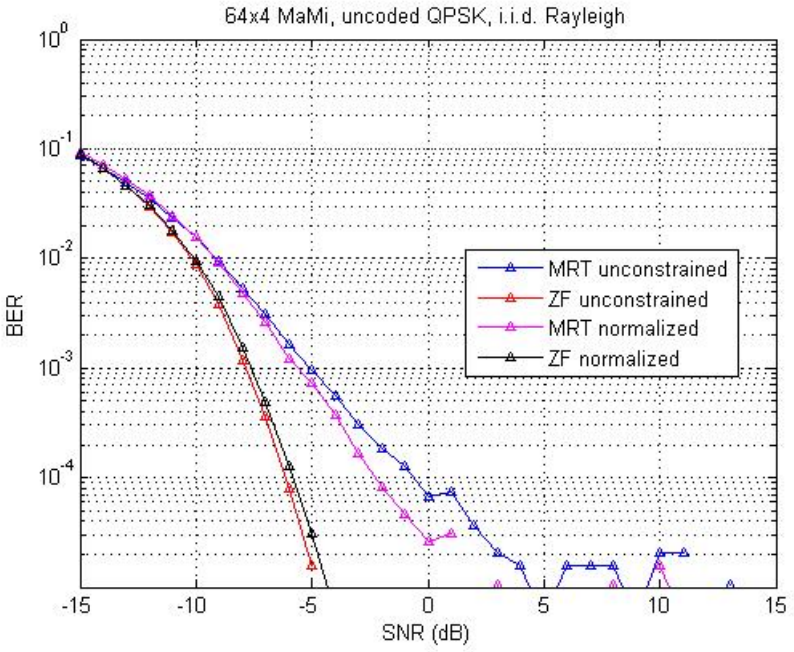


Figure 2.5: Comparison between ZF and MRT performance, with and without power normalization over antennas and subcarriers (64×4 case).

obtained in the SIMO case by combining the multiple diversity components. We summarize this algorithm first for a single-user SISO case, before adapting it to the massive MIMO case which for a single user corresponds to a MISO configuration.

The algorithm is based on an upper bound of the BER function:

$$f(\alpha_n p_n) \simeq aQ(\sqrt{b\alpha_n p_n}) \leq \frac{a}{2}\exp\left(-\frac{b}{2}\alpha_n p_n\right), \tag{2.43}$$

where the Q-function $Q(x) \triangleq \left(1/\sqrt{2\pi}\right)\int_x^\infty \exp\left(-t^2/2\right)\mathrm{d}t$ while a and b are normalization constants depending on the QAM constellation size $Q = 2^m$ for m bits per symbol:

$$a = \frac{2(\sqrt{Q}-1)}{m\sqrt{Q}} \tag{2.44}$$

$$b = \frac{3}{Q-1} \tag{2.45}$$

$\alpha_n p_n$ represents the SNR for the n^{th} subcarrier; $\alpha_n \triangleq |h_n^2|/N_0$ is the ratio between the channel gain of the n^{th} subcarrier and the noise power, while p_n is the power allocated on the n^{th} subcarrier. We can now substitute the upper bound of (2.43) into the Lagrange solution to the BER minimization system, expressed as:

$$\frac{1}{N}\frac{\mathrm{d}}{\mathrm{d}p_n} f\left(\alpha_n p_n\right) + \lambda = 0, \qquad n = 1, 2, ..., N, \tag{2.46}$$

where N is the number of subcarriers of the system. A closed-form solution can be obtained to the system combining the N equations from (2.46) and one additional equation expressing the total power constraint used to define the Lagrangian variable λ:

$$\sum_{n=1}^{N} p_n = N\overline{P}, \tag{2.47}$$

where $\overline{P}$ denotes the average transmit power per subcarrier when the total power is equally distributed amongst the N subcarriers. The details of the computations are not present in this deliverable but can be found in [47]. In the solution some p_n coefficients might be negative. In that case, the Kuhn-Tucker conditions are applied and the negative coefficients are set to zero, leading to the following solution:

$$p_n = \begin{cases} \frac{\lambda_0}{\alpha_n} - \left(\frac{2}{b}\right)\left(\frac{1}{\alpha_n}\right)\ln\left(\frac{1}{\alpha_n}\right), & \alpha_n \geq \exp\left(-\frac{b\lambda_0}{2}\right) \\ 0, & \alpha_n < \exp\left(-\frac{b\lambda_0}{2}\right) \end{cases}, \tag{2.48}$$

where λ_0 satisfies the power constraint (2.47) and is thus expressed as:

$$\lambda_0 = \frac{N\overline{P} + \left(\frac{2}{b}\right)\sum\limits_{n\in S}\left(\frac{1}{\alpha_n}\right)\ln\left(\frac{1}{\alpha_n}\right)}{\sum\limits_{n\in S}\left(\frac{1}{\alpha_n}\right)}, \tag{2.49}$$

where S is the subset containing all subcarrier indices n that meet the condition $\alpha_n \geq \exp\left(-\frac{b\lambda_0}{2}\right)$.

Practically, the algorithm is implemented in a recursive way. In a first phase, as initialization, λ_0 is computed on all the subcarriers: the subset S contains all N subcarriers.

Once the first power coefficients p_n are obtained, the recursive part of the implementation begins: all subcarriers with negative p_n value are set to zero and a new λ_0 is computed with only the subcarriers which had a positive p_n in the previous iteration. This process is iterated until no new negative power allocation coefficient is created.

Adaptation to the massive MIMO case

In the reference case [47], there is only one antenna at the transmitter and one at the receiver (possibly after recombination for receive diversity) and thus each subcarrier has a scalar channel response h_n. However, in the massive MIMO case, the situation is different: each user has one antenna (at the receiver side) but the transmitter has M antennas.

In the massive MIMO case, the power allocation algorithm is applied to the precoded channel, after summation over all transmit antennas. The precoding matrix is first computed in the usual way, i.e., by taking the conjugate transpose of the channel. Secondly, for each user, the contributions of the M antennas including their precoding coefficients multiplied by the corresponding channel coefficients are added in order to determine the equivalent combined channel amplitude for that user. This equivalent channel amplitude is used as channel coefficient h_n in the power allocation algorithm. The obtained power coefficient p_n is then applied to all coefficients of the precoding matrix affecting the selected user k.

Results

The impact of the proposed power allocation algorithm (PA in this section) is shown first in Figures 2.6, 2.7, 2.8 and 2.9 for single-user scenarios. A Rayleigh multi-tap channel has been used in all the simulations of this section. Single-user results are selected, given that the power allocation affects all the coefficients of a user over a given subcarrier in the same way, hence it does not modify the cross-correlation between precoded streams of multiple users. A multi-user validation is presented a the end of this section.

Figure 2.6 illustrates the algorithm for the SISO case. It can be seen that the power allocation algorithm deteriorates the BER performances at low SNR, which is not a useful region (the BER is too high), but enhances the performance of the system starting from an SNR of approximately

15 dB, where the BER gets below 5%. This cross-over point is around 10% for multiple-antenna cases (Figures 2.7 and 2.8). In the low SNR region, the degradation comes from the upper BER bound (2.43) being less tight and the larger number of suppressed subcarriers due to negative power allocation.

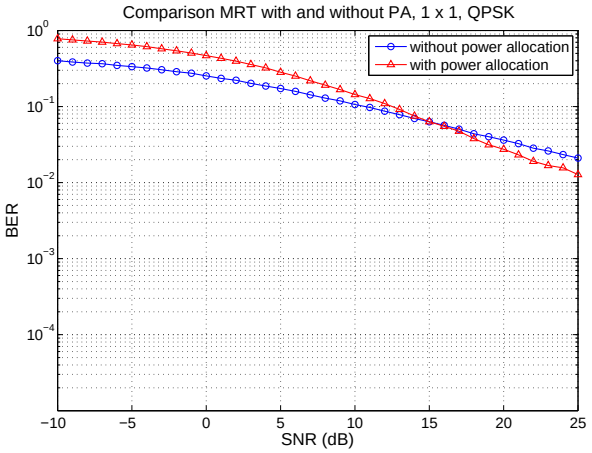


Figure 2.6: BER curves for MRT precoding with and without Power Allocation (1×1 system)

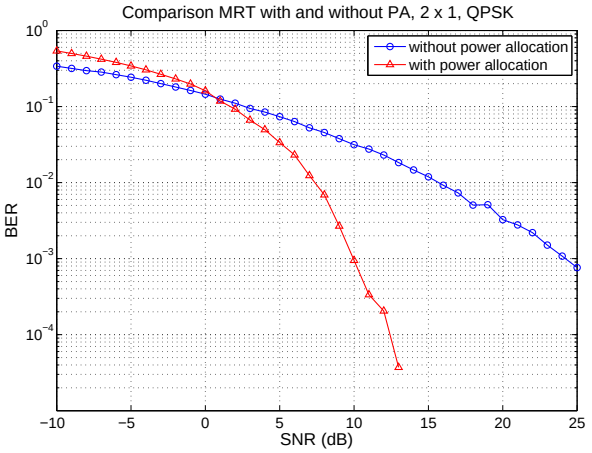


Figure 2.7: BER curves for MRT precoding with and without Power Allocation (2×1 system)

Figures 2.7 and 2.8 illustrate 2×1 and 8×1 systems, respectively. The optimized power allocation strongly improves the BER performance as compared to the non-optimized case. Figure 2.9 illustrates the fact that similar results are obtained even for a high-order constellation. Importantly, all those simulations have been performed for uncoded scenarios. When adding channel coding, we will revisit those conclusions in the future, as channel coding brings a dramatic performance improvement especially for OFDM systems in the presence of frequency-selective channels: in that case on top of the coding gain present in all systems, a diversity gain is also obtained through channel coding.

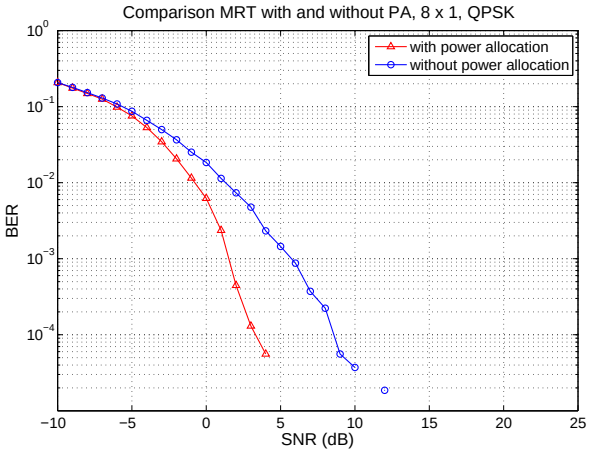


Figure 2.8: BER curves for MRT precoding with and without Power Allocation (8x1 QPSK system)

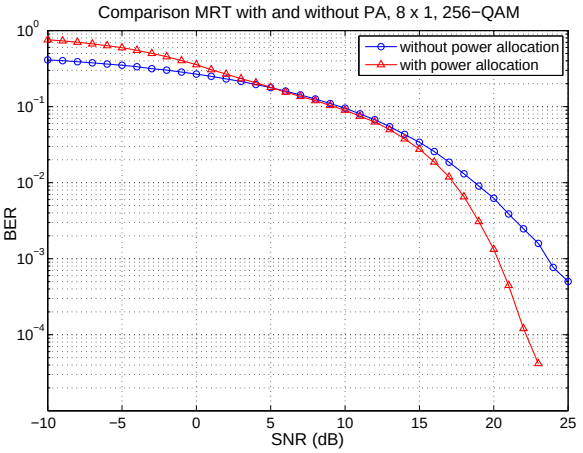


Figure 2.9: BER curves for MRT precoding with and without Power Allocation (8x1 256-QAM system)

Finally, Figure 2.10 compares the resulting performance to the curves of Figure 2.5 in a true

(64×4) massive MIMO case. The power allocation scheme performs almost exactly as the MRT curve with normalization over subcarriers and antennas. The reason is the following: by using

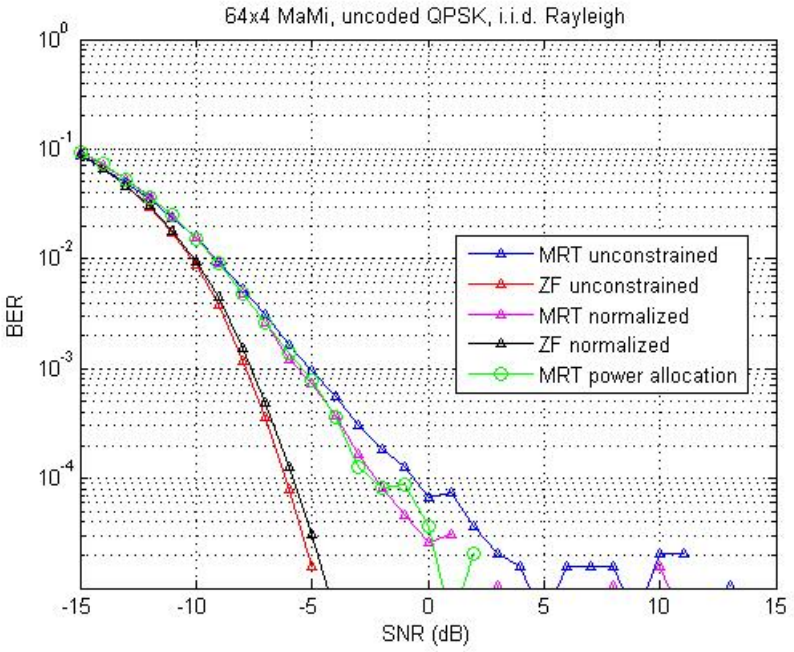


Figure 2.10: Positioning of power allocation result with respect to reference curves from Figure 2.5 (64×4).

the channel conjugate as a precoder, the MRT gives each precoder coefficient in the (antenna, user, subcarrier) three-dimensional space the same amplitude as the corresponding channel coefficient. This is proven to be optimal in the antenna dimension. Indeed, this maximizes the received SNR thanks to the inherent maximum-ratio combining which is performed when all antenna streams are added up at the receiver side. However, in the subcarrier dimension, no such combination is performed, as each subcarrier is processed independently from the others. Hence, in the uncoded case, the performance is dominated by the worst subcarriers and at high SNR performing an inverse waterfilling is shown to be optimal [47]. This is opposite to the capacity maximization problem where direct waterfilling is optimal.

2.3.3 Discrete-Time Constant-Envelope Precoding

The low-PAPR precoding scheme originally proposed in [36] and extended in [37,38], here called *discrete-time constant-envelope* (DTCE) precoding, is briefly described in this section.

Single-Carrier Transmission

The DTCE precoder proposed in [37, 38] finds transmit signals that minimize the difference between the received noise-free signal and the desired receive symbol under a fixed modulus constraint:

$$\mathbf{x} = \underset{|x_m[n]|=M^{-1/2},\forall m,n}{\arg\min} \|\mathbf{H}\mathbf{x} - \sqrt{\gamma}\mathbf{u}\|\,, \tag{2.50}$$

where $\gamma \in \mathbb{R}^+$ is a parameter that can be interpreted as the array gain. A low-complexity solver to this optimization problem is given in [37]. This solver minimizes (2.50) by cyclic

optimization: the norm is minimized with respect to one $x_m[n]$ at a time, keeping the other variables fixed. The parameter γ can be chosen to maximize some performance metric, for example, the sum-rate. If γ is too small, the received signal will drown in thermal noise at the user. If γ is too big, the precoder will not be able to produce the desired symbol at the user, which will make the norm in (2.50) excessively big.

OFDM Transmission

OFDM transmission with DTCE transmit signals can be done by using the same algorithm as for the SC transmission but by using $\mathbf{F}_K^{\mathsf{H}}\mathbf{u}$ (the inverse Fourier transform of the symbols) instead of $\mathbf{u}$. It can also be done with cyclic prefix by adapting the algorithm to minimize $\|\tilde{\mathbf{H}}\mathbf{F}_M\mathbf{x} - \sqrt{\gamma}\mathbf{u}\|$ instead. Since the prefix is cyclic the two schemes are equivalent, because $\|\tilde{\mathbf{H}}\mathbf{F}_M\mathbf{x} - \sqrt{\gamma}\mathbf{u}\| = \|\mathbf{H}\mathbf{x} - \sqrt{\gamma}\mathbf{F}_K^{\mathsf{H}}\mathbf{u}\|$.

2.3.4 SC Transmission vs. OFDM

Note that because of (2.37), the transmit signals of OFDM in (2.36) can be rewritten as

$$\mathbf{x}_{\text{OFDM}} = \mathbf{G}\mathbf{F}_K^{\mathsf{H}}\mathbf{u}. \tag{2.51}$$

Thus, precoding and transmitting $\mathbf{u}$ with OFDM is the same as precoding and transmitting $\mathbf{F}_K^{\mathsf{H}}\mathbf{u}$ with SC transmission. The massive MIMO OFDM system that has been described here does not do waterfilling over the subchannels $\tilde{\mathbf{H}}[n]$. We argue that the channel hardening phenomenon makes every subchannel close to equally good and that the rate loss of not doing waterfilling is negligible in massive MIMO. Analogously, one can show that a SC system with a detector that treats intersymbol interference, which seems to be the inherent drawback of SC transmission, as additional noise—it detects one symbol at a time and discards the possibility to do sequence detection—gives the same performance as the OFDM system that does not do waterfilling.

In SC transmission, the matched-filter detection is easy, the symbols are transmitted in the time domain and the user can detect the symbols directly. In OFDM in contrast, the symbols are transmitted in the frequency domain and the user has to await the whole OFDM symbol to perform a Fourier transformation before detection. OFDM will cause at least a delay of N, since precoding and detection are done block by block. SC transmission, on the other hand, can be implemented with FIR filters with much smaller delay in massive MIMO, as was illustrated for ZF in Figure 2.4.

The operational differences between SC and OFDM transmission are summarized in Table 2.1. Among the conventional linear precoders, MRT generally has the lowest implementation complexity since it only performs match filtering and a simple power scaling. Both ZF and RZF need to invert matrices of dimensions proportional to the number of users K, with a complexity proportional to K^3, and have essentially the same operational complexity. Hence, the higher communication performance with RZF precoding means that it is preferred over ZF, unless ZF can be implemented in much more efficient manner. DTCE precoding has a different architecture than the conventional linear precoeders, because it does not precompute a precoding matrix but send every symbol vector through a non-linear algorithm. This makes it hard to directly compare the complexity, but its complexity scales as $\mathcal{O}(MKL)$ operations, which certainly makes it more scalable with K than ZF and RZF.

Table 2.1: Precoding Schemes and Transmission Techniques for Massive MIMO

		SC $\longleftrightarrow$ (equivalent, when waterfilling and joint sequence detection is not done.)	OFDM
	General properties	Simple matched-filter detection Running precoding and detection, little delay Intersymbol interference not inherently suppressed due to (2.23)	Matched-filter detection in frequency domain (requires FFT at the users) Block-wise precoding and detection No intersymbol interference due to (2.70)
MRT	Maximizes array gain Enables local precoding High PAPR Low complexity Performs well at low data rates	Has intersymbol interference Has interuser interference L-tap FIR filter $G = M/K, I = 1$	Has interuser interference $G = M/K, I = 1$
ZF	Nulls interference No intersymbol interference High PAPR	Requires inversion of $K \times K$-matrices $\sim L$-tap FIR filter $G = \frac{M-K}{K}$, $I = 0$	Requires inversion of $K \times K$-matrices $G = \frac{M-K}{K}$, $I = 0$
RZF	State-of-the-art linear precoder High PAPR Estimate of user noise variance needed Performance-wise similar to ZF for small K, but better for big K	Requires inversion of $K \times K$-matrices $\sim L$-tap FIR filter	Requires inversion of $K \times K$-matrices
DTCE	Low PAPR Non-linear precoder low complexity per symbol	Filter with delay $\sim L$	

2.3.5 Distortion in Power Amplifiers

Massive MIMO will require simple, inexpensive and power efficient amplifiers in the downlink [26]. Next, a general model for power amplifiers and two distortion measures are introduced. This theory is then used to determine how amplifiers affect the transmission in massive MIMO.

Let $x(t)$ be the pulse shape filtered transmit signal (2.22) at one of the antennas, omitting the antenna index for simplicity. Further, let $a(t)$ be the non-linear amplification of $x(t)$. A common way to model the amplifier is to specify the AM-AM $g(|x(t)|)$ and AM-PM conversion $\Phi(|x(t)|)$. In this model, the amplitude and phase distortion of the amplified signal depend only on the amplitude of the original signal, see for example [44]. The amplifier output is thus given by

$$a(t) = g(|x(t)|)e^{j(\arg x(t)+\Phi(|x(t)|))}, \tag{2.52}$$

where j is the imaginary number.

Power Efficiency

The most basic class B amplifiers have the properties of being simple, inexpensive and power efficient [51], and could therefore potentially be suited for massive MIMO. The power efficiency of such an amplifier is given by [44]

$$\eta = \frac{\pi}{4}\frac{\mathsf{E}[\,g^2(|x(t)|)\,]}{a_{\max}\,\mathsf{E}[\,g(|x(t)|)\,]}, \tag{2.53}$$

where $a_{\max}$ is the highest possible output amplitude. Note that $\eta \leq \pi/4$ with equality only if the continuous-time input signal has perfectly constant envelope and the amplifier is operated at saturation.

To avoid non-linear amplification and distortion, the signal has to be backed off[2]; that is, its power has to be lowered to a suitable operation point, such that the signal amplitude most of the time stays in a region with sufficiently linear amplification. Since back-off decreases the efficiency, the efficiency in (2.53) is maximized by choosing the highest operation point that still results in acceptable distortion. This choice is usually done experimentally.

In-Band Distortion

The *Normalized Mean-Square-Error* (NMSE) is a measure of how much amplifier-caused in-band distortion the *users* experience. Similar measures for the distortion at the transmitter are defined in [24, 44]. Let $\tilde{r}$ be the received signal after matched filtering and sampling (at a random user at a random time), and let r be the signal at the same user at the same time that would have been received if no amplification had taken place. The NMSE is then given by

$$\mathsf{NMSE} \triangleq \frac{\mathsf{E}[\,|\tilde{r}-\lambda r|^2\,]}{\mathsf{E}[\,|\lambda r|^2\,]}. \tag{2.54}$$

The amplification factor λ is chosen to minimize the power of the expected distortion

$$\lambda = \frac{\mathsf{E}[\,r^*\tilde{r}\,]}{\mathsf{E}[\,|r|^2\,]}. \tag{2.55}$$

[2]In this document, a back-off b means that the amplifier is operated at a fraction b below the 1-dB compression point—the point, where the output signal is 1 dB weaker than what it would have been if the amplification had been perfectly linear.

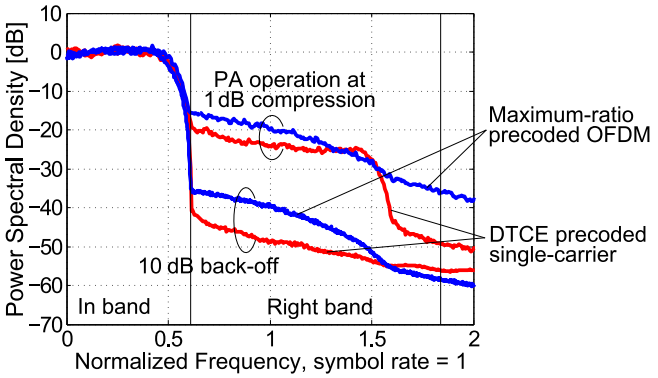


Figure 2.11: The power spectral densities after amplification of two signal types with PA operation at the 1 dB compression point and with PA operation well below saturation. The signals are from the system described in Table 2.2.

Out-of-Band Radiation

The out-of-band radiation can been quantified by the *Adjacent Channel Leakage Ratio* (ACLR), which is defined in terms of the power $P_{[-B/2,B/2]}$ of the power spectral density $S_a(f)$ of $a(t)$ in the useful band and the powers $P_{[-3B/2,-B/2]}$, $P_{[B/2,3B/2]}$ in the immediately adjacent bands:

$$\mathsf{ACLR} \triangleq \max\left(\frac{P_{[-3B/2,-B/2]}}{P_{[-B/2,B/2]}}, \frac{P_{[B/2,3B/2]}}{P_{[-B/2,B/2]}}\right), \tag{2.56}$$

where

$$P_{\mathcal{B}} \triangleq \int_{f\in\mathcal{B}} S_a(f)\mathrm{d}f. \tag{2.57}$$

The bandwidth B is the band occupied by the ideal signal after pulse shape filtering. For example, if a root-raised cosine filter with roll-off β has been used, then $B = 1+\beta$ (in units of the symbol rate).

In Figure 2.11, four power spectral densities are shown. Half the in-band spectrum is shown together with the whole right band. It can be seen that the signals that have not been backed off radiate more power into the right band than the backed off signals.

2.3.6 Comparison of Precoding Schemes

Next, the precoding schemes will be compared in terms of power efficiency and distortion. A simple amplifier model is the Rapp model [44]. In this model, the phase distortion is neglected, so $\Phi(|x|) = 0$, $\forall|x|$, and the amplitude conversion is given by

$$g(|x|) = a_{\max}\frac{|x|/x_{\max}}{(1+(|x|/x_{\max})^{2p})^{\frac{1}{2p}}}, \tag{2.58}$$

where the parameter $p = 2$ approximates a typical moderate-cost solid-state PA [5]. The parameter $a_{\max}$ is the maximum output amplitude and $x_{\max} = a_{\max}/g'(0)$ determines the slope of the asymptote that $g(|x|)$ approaches for small $|x|$. Here $x_{\max} = M^{-1/2}$ and $a_{\max} = x_{\max}\sqrt{P}/\lambda_0$, where λ_0 is the amplification factor (2.55) when $a_{\max} = x_{\max}$. The correction factor $1/\lambda_0$ is determined by the signal type and the back-off, it is chosen such that the total radiated power is P.

To see how this non-linear power amplifier affects the transmission, the system specified in Table 2.2 was simulated for different back-offs.

Table 2.2: Simulation Parameters

Number of tx-antennas	$M = 100$
Number of users	$K = 10$ and 50
Channel	$L = 4$-tap Rayleigh fading
Power delay profile	$h_{km}[\ell] \sim \mathcal{CN}(0, 1/L)$ i.i.d.
Pulse shape filter	Root-raised cosine, roll-off 0.22
Amplifier	Class B, see (2.53) and (2.58)

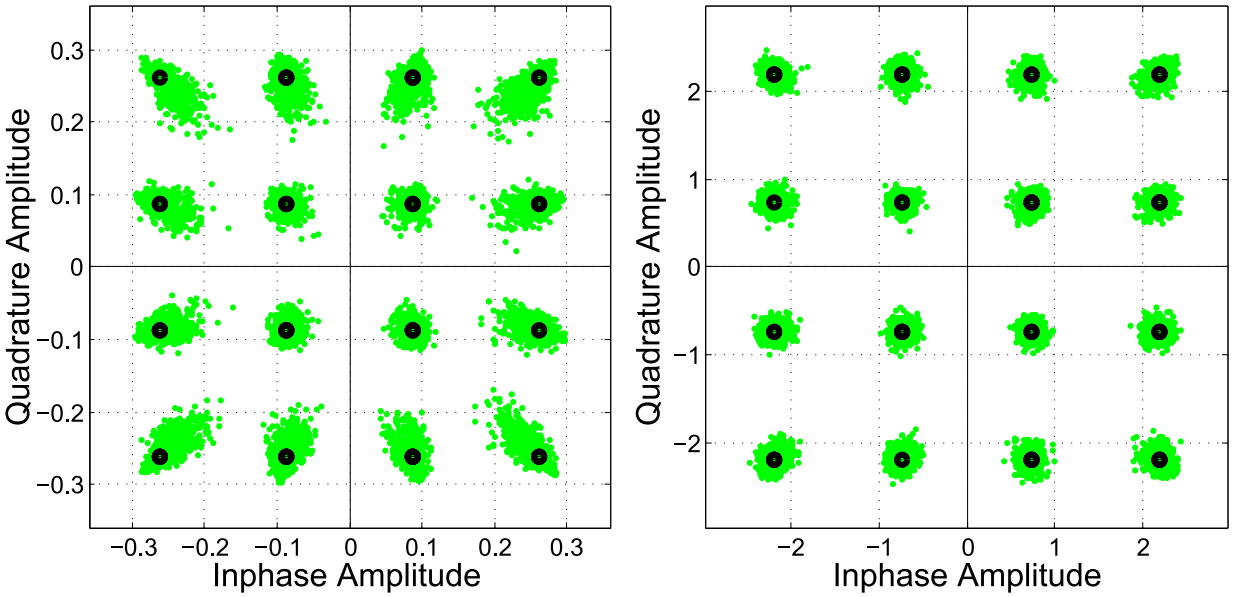


Figure 2.12: Received signal points after symbols from a 16-QAM constellation have been broadcast with 2.2 dB back-off and non-linear amplification over a MIMO channel by SC ZF precoding. The black dots show the desired symbols. The left plot shows a small MIMO system with 4 base station antennas serving 1 user. The right plot shows a massive MIMO system with 100 base station antennas serving 10 users.

Expected Power Efficiency and Distortion

In SISO and MIMO OFDM systems, the in-band distortion, seen at the users, is uncorrelated to the desired symbol, which means that the distortion can be regarded as uncorrelated additive noise [14]. In massive MIMO, a similar effect is observed—the noise can be treated as uncorrelated and additive for SC transmission too. Figure 2.12 shows the distribution of the distortion in two MIMO systems that use SC transmission. In the small MIMO system, the distortion is differently distributed for different constellation points. In the massive MIMO system, however, the distortion has approximately the same distribution for all constellation points. Since the precoded transmit signals are the sums of many independent symbols and the receive signals are the sums of many different transmit signals, it is intuitively understandable that the in-band distortion should be uncorrelated with any single symbol.

The efficiency in (2.53) was computed for several back-offs and averaged over different channel realizations. By treating the back-off as an intermediate variable, the NMSE can be given as a function of the efficiency, see Figure 2.14(a). Note that the efficiency is not a simple function of the back-off, but depends on the signal type.

Similarly, the ACLR was computed for several back-offs and averaged over different channel realizations. It can be seen in Figure 2.14(b) that the amount of energy radiated out-of-band is

monotonically decreasing with the back-off. A constraint on the ACLR will therefore constrain the maximum efficiency that the amplifier can operate at.

Because of their similar amplitude distributions, all the linear precoding schemes resulted in the same curves in both Figure 2.14(a) and 2.14(b). Therefore, only the results of OFDM MRT and ZF precoding are shown.

2.3.7 Consumed Power in Amplifiers

Using SC or OFDM transmission makes no significant difference on performance in massive MIMO. SC transmission was chosen for the DTCE scheme to show the feasibility of SC transmission in massive MIMO.

Let P_{cons} be the power that the amplifier consumes and η be its efficiency. An achievable rate is then given by

$$R = \max_{\gamma,\eta} \log_2 \left(1 + \frac{\eta P_{\text{cons}} G}{\eta P_{\text{cons}} (I + D(\eta)) + 1} \right), \tag{2.59}$$

where the maximization is over $\eta \in [0, \eta_{\max}]$, where $\eta_{\max}$ is the efficiency that corresponds to operating point of the amplifier with the maximum allowed out-of-band radiation. The array gain G and interference I will be functions of the transmit power $P = \eta P_{\text{cons}}$ and thus η, when RZF is used, and of the parameter γ when DTCE precoding is used. These functions have to be established through simulations. Note that the optimization is only done over γ when DTCE precoding is used, for the other precoders, the optimization is done over the single variable η. For MRT and ZF precoding, the array gain and interference are given by [48, 67]

$$G = \begin{cases} M/K, & \text{for MRT} \\ (M-K)/K, & \text{for ZF} \end{cases} \tag{2.60}$$

and

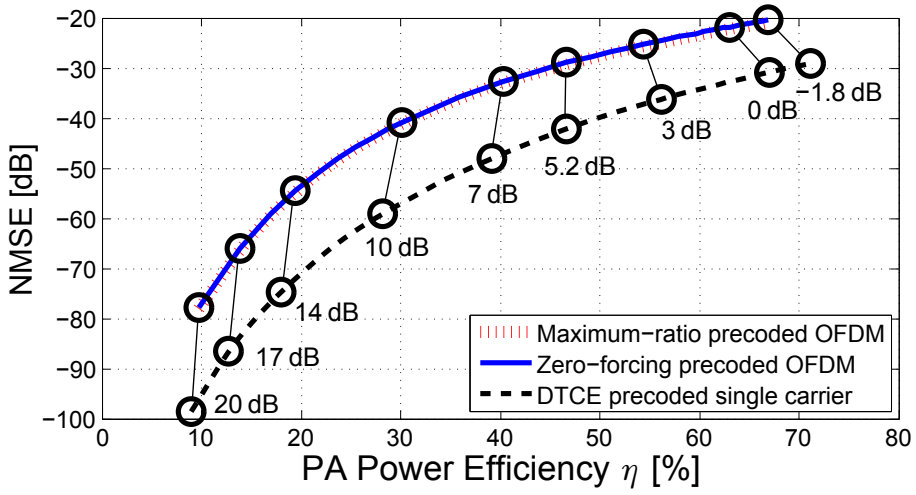
$$I = \begin{cases} 1, & \text{for MRT} \\ 0, & \text{for ZF} \end{cases}. \tag{2.61}$$

The relation in (2.59) thus constitutes a function between the power P_{cons} that the amplifiers consume and the data rate requirement R.

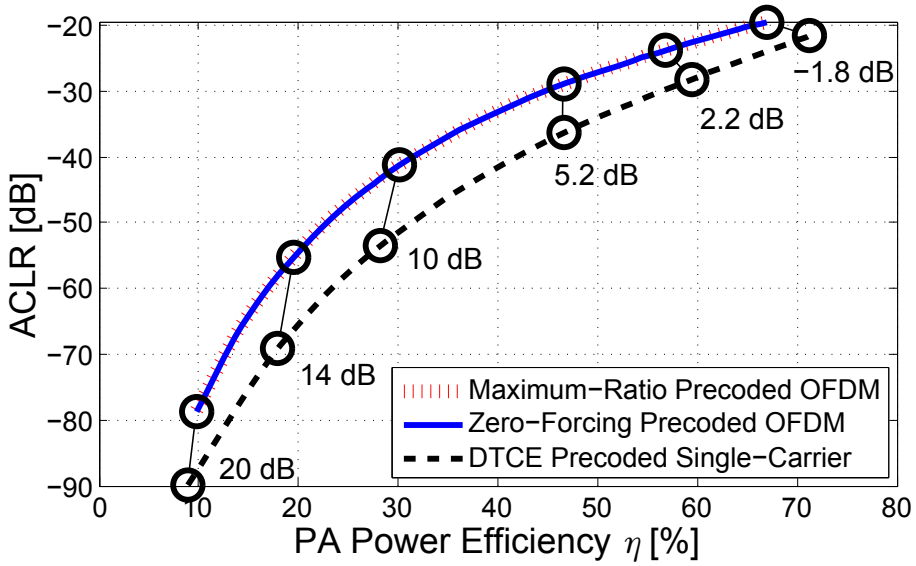
Evaluating (2.59) for the system specified in Table 2.2 for different P_{cons}, gives the estimated base station power consumption shown in Figure 2.14. It can be seen that maximum-ratio precoding works well for low rate requirements but is limited by interference to below a certain maximum rate. Further, it can be seen that RZF and ZF perform equally well when the number of users is small, but RZF has an advantage when the number of users is big.

DTCE precoding consumes more power than the optimal conventional precoder when the out-of-band radiation is not constrained, but seems to consume approximately the same amount of power in the range 1-2 bpcu per user. In the comparison, it is important to remember that different bounds have been used for the conventional and DTCE precoders and that the rate expression of the DTCE precoder might be pessimistic for low data rates.

The value of η that corresponds to the optimal operating point of the amplifiers is shown in Figure 2.15. When there is no constraint on the out-of-band radiation, it is optimal to operate the amplifiers at saturation for low rate requirements. For higher rate requirements, the amplifiers should be backed off to lower the in-band distortion. When the ACLR is constrained to



(a) In-band-distortion and efficiency at different back-offs.



(b) Out-of-band radiation and efficiency at different back-offs.

Figure 2.13: Some measurements of NMSE and ACLR for a Rapp-modelled ($p = 2$) class B amplifier with three signal types. The signals have been pulse shape filtered with a root-raised cosine, roll-off factor $\beta = 0.22$. The encircled points correspond to some selected operating points of the amplifier specified by the back-off from the 1-dB compression point.

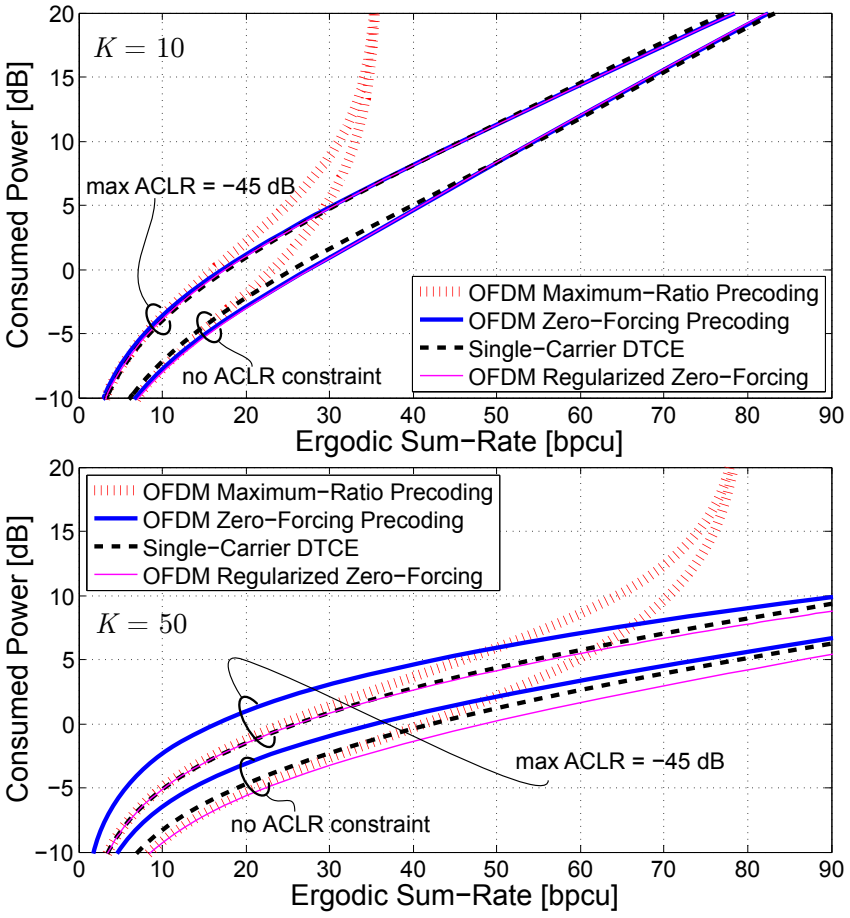


Figure 2.14: The estimated consumed power of a base station with $M = 100$ antennas required to serve $K = 10$ (above) and $K = 50$ (below) users with R bpcu over a frequency-selective channel with $L = 4$ taps with and without a constraint on the ACLR.

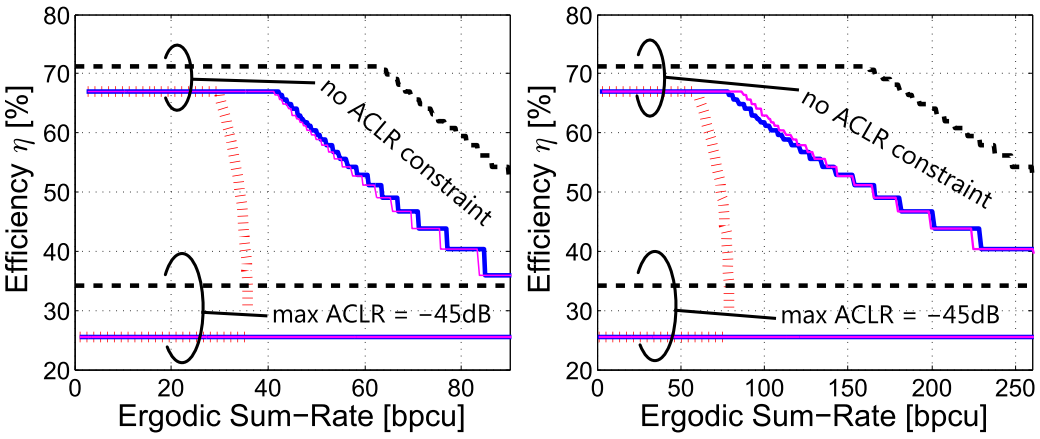


Figure 2.15: The power efficiency of the amplifiers at the optimal operating point for different sum-rate requirements. The legend in Figure 2.14 also applies here.

below −45 dB, the optimal efficiency of the amplifiers is $\eta_{\max}$, i.e. 34 % for DTCE precoding and 26 % for maximum-ratio and ZF precoding, over the whole range of required rates investigated, both when serving 10 and 50 users. This corresponds to a back-off of 8 dB and 12 dB respectively.

Finally, notice that only conventional ZF and RZF precoders were considered in this section. A third option is to modify the linear precoding to reduce the PAPR. For example, a feasible hardware implementation of PAPR reduction of linear precoders, based on antenna reservation, is described in Section 4.4.2.

2.4 Uplink detection

Each base station can use its multitude of antennas for phase-coherent receive combining, based on the acquired CSI. For simplicity in exposition, it is assumed in this section that the channel estimation in Section 2.2 provides the true channels. The receive combining can adaptively amplify desired signals and can reject interfering signals. Since the downlink and uplink transmissions in TDD systems take place over the same reciprocal channels, the same rates are typically achievable in both directions—this is known as uplink-downlink duality [8, 11, 64]. A key insight from the duality theory for linear processing is that power allocation needs to be used in the downlink, while the normalized precoding vector for a user in the downlink can be used as receive combining vector in the uplink. For this reason, the three main precoding schemes described in Section (2.3) (e.g., MRT, ZF, and RZF) have direct counterparts in the uplink detection.

Based on the input-output relation of the downlink channel in (2.26), we have the reciprocal counterpart

$$\mathbf{r} = \sqrt{\frac{P}{K}}\mathbf{H}^{\mathsf{H}}\mathbf{z} + \mathbf{w}, \tag{2.62}$$

where is the block-circulant $KN \times MN$-matrix $\mathbf{H}$ defined in (2.25). The transmitted signals from the K users at time $n \in [0, N-1]$ are defined as $\mathbf{z}[n] \triangleq (z_1[n], \ldots, z_K[n])^{\mathsf{T}}$, where $\mathbb{E}[|z_k[n]|^2] = 1$, and are gathered in the vector

$$\mathbf{z} \triangleq (\mathbf{z}^{\mathsf{T}}[0], \ldots, \mathbf{z}^{\mathsf{T}}[N-1])^{\mathsf{T}}. \tag{2.63}$$

Since the transmitted symbols are normalized, it is the factor P/K that determines the transmit power level per user. Note that P represents the *total* uplink transmit power normalized by the noise variance, and can take another value than the corresponding P in the downlink.

Similarly, the M-antenna array at the base stations receives signals $\mathbf{r}[n] \triangleq (r_1[n], \ldots, r_M[n])^{\mathsf{T}}$ at time $n \in [0, N-1]$ and these are gathered in

$$\mathbf{r} \triangleq (\mathbf{r}^{\mathsf{T}}[0], \ldots, \mathbf{r}^{\mathsf{T}}[N-1])^{\mathsf{T}}, \tag{2.64}$$

while the zero-mean white Gaussian noise over the N time instants is

$$\mathbf{w} \triangleq (\mathbf{w}^{\mathsf{T}}[0] \cdots \mathbf{w}^{\mathsf{T}}[N-1])^{\mathsf{T}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{MN}). \tag{2.65}$$

By using a cyclic prefix as in Section 2.3, (2.62) is also easily given in the frequency domain. Let $\mathbf{F} \in \mathbb{C}^{N \times N}$ be the N-point discrete Fourier transform (DFT) transform with $\frac{1}{\sqrt{N}} e^{-j2\pi(n-1)(n'-1)/N}$ on its (n, n')-th position and define the two unitary matrices $\mathbf{F}_K \triangleq \mathbf{F} \otimes \mathbf{I}_K$ and $\mathbf{F}_M \triangleq \mathbf{F} \otimes \mathbf{I}_M$. Recall from (2.30) that

$$\tilde{\mathbf{H}}^{\mathsf{H}} = \mathbf{F}_M \mathbf{H}^{\mathsf{H}} \mathbf{F}_K^{\mathsf{H}} \tag{2.66}$$

is the block-diagonal matrix, whose diagonal blocks

$$\tilde{\mathbf{H}}^{\mathsf{H}}[n] = \frac{1}{\sqrt{N}} \sum_{\ell=0}^{L-1} \mathbf{H}^{\mathsf{H}}[\ell] e^{j2\pi\ell n/N}, \quad n = 0, \ldots, N-1, \tag{2.67}$$

are the DFTs of $\{\mathbf{H}^{\mathsf{H}}[\ell]\}$. Let

$$\tilde{\mathbf{z}} \triangleq (\tilde{\mathbf{z}}^{\mathsf{T}}[0], \ldots, \tilde{\mathbf{z}}^{\mathsf{T}}[N-1])^{\mathsf{T}} = \mathbf{F}_M \mathbf{z} \tag{2.68}$$

$$\tilde{\mathbf{r}} \triangleq (\tilde{\mathbf{r}}^{\mathsf{T}}[0], \ldots, \tilde{\mathbf{r}}^{\mathsf{T}}[N-1])^{\mathsf{T}} = \mathbf{F}_N \mathbf{r} \tag{2.69}$$

be the DFTs of the uplink transmit and receive signals, respectively. The relation between $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{r}}$ is then

$$\tilde{\mathbf{r}}[n] = \sqrt{P} \tilde{\mathbf{H}}^{\mathsf{H}}[n] \tilde{\mathbf{z}}[n] + \tilde{\mathbf{n}}[n], \quad n = 0, \ldots, N-1, \tag{2.70}$$

where $\tilde{\mathbf{n}}[n] \sim \mathcal{CN}(0, \mathbf{I}_M)$.

2.4.1 Linear Receive Combining

Since we have $M \geq K$ in massive MIMO scenarios, the base stations have more observations in $\tilde{\mathbf{r}}[n] \in \mathbb{C}^M$ than there are unknown signals in $\tilde{\mathbf{z}}[n] \in \mathbb{C}^K$. When detecting the signal from user k, the base station can use the M degrees of freedom to amplify the desired signal and/or reject interfering signals from other users.

The computationally efficient *linear receive combining* schemes are based on selecting a matrix $\mathbf{C} \in \mathbb{C}^{MN \times KN}$ in the time domain and weight the received signals as

$$\hat{\mathbf{z}}_{\mathrm{SC}} = \mathbf{C}^{\mathsf{H}} \mathbf{r} \tag{2.71}$$

so that $\hat{\mathbf{z}}_{\mathrm{SC}}$ is similar to the transmitted time domain signal $\mathbf{z}$. Depending on which metric that is used to measure the similarity between $\hat{\mathbf{z}}_{\mathrm{SC}}$ and $\mathbf{z}$, different receive combining schemes arise.

In the frequency domain, a receive combining matrix $\tilde{\mathbf{C}} \in \mathbb{C}^{MN \times KN}$ is preceded by a transform to the time domain

$$\tilde{\mathbf{z}}_{\text{OFDM}} = \tilde{\mathbf{C}}^{\mathsf{H}} \mathbf{F}_M \mathbf{r} \tag{2.72}$$

As in the downlink in Section 2.3, the time domain transmission is referred to as *single-carrier (SC) transmission* and the frequency domain transmission as *orthogonal frequency-division multiplexing* (OFDM). For the linear receive combining matrices considered in this document, it holds that

$$\mathbf{C} = \mathbf{F}_K^{\mathsf{H}} \tilde{\mathbf{C}} \mathbf{F}_M. \tag{2.73}$$

2.4.2 Three Receive Combining Schemes

The uplink counterpart to MRT is called maximum-ratio combining (MRC) and the combining matrix is given by

$$\begin{aligned} \mathbf{C}_{\text{MRC}} &= \mu_{\text{MRC}} \mathbf{H}^{\mathsf{H}} \text{ or} \\ \tilde{\mathbf{C}}_{\text{MRC}} &= \mu_{\text{MRC}} \tilde{\mathbf{H}}^{\mathsf{H}}, \end{aligned} \tag{2.74}$$

where the normalizing scalar μ_{MRC} is not strictly needed (since the combining matrices are not subject to any power constraint) but can be useful to control the arithmetic range of the combined signals. Similar to MRT, MRC maximizes the array gain, but *interference* (undesired symbols intended to other users) will still be present in the combined signal since there is no active interference mitigation.

The ZF precoder has a direct counterpart in the ZF combiner, where the combining matrix is given by

$$\begin{aligned} \mathbf{C}_{\text{ZF}} &= \mu_{\text{ZF}} \mathbf{H}^{\mathsf{H}} (\mathbf{H}\mathbf{H}^{\mathsf{H}})^{-1} \text{ or} \\ \tilde{\mathbf{C}}_{\text{ZF}} &= \mu_{\text{ZF}} \tilde{\mathbf{H}}^{\mathsf{H}} (\tilde{\mathbf{H}}\tilde{\mathbf{H}}^{\mathsf{H}})^{-1}, \end{aligned} \tag{2.75}$$

where μ_{ZF} is an optional normalizing scalar. This receive combining scheme nulls all the interference, both intersymbol interference and interuser interference, by sacrificing part of the array gain.

Finally, the MMSE receive combining matrix is given by

$$\begin{aligned} \mathbf{C}_{\text{MMSE}} &= \mu_{\text{MMSE}} \mathbf{H}^{\mathsf{H}} (\mathbf{H}\mathbf{H}^{\mathsf{H}} + \tfrac{K}{P} \mathbf{I}_{KN})^{-1} \text{ or} \\ \tilde{\mathbf{C}}_{\text{MMSE}} &= \mu_{\text{MMSE}} \tilde{\mathbf{H}}^{\mathsf{H}} (\tilde{\mathbf{H}}\tilde{\mathbf{H}}^{\mathsf{H}} + \tfrac{K}{P} \mathbf{I}_{KN})^{-1}, \end{aligned} \tag{2.76}$$

for some optional scalar μ_{MMSE}. As the name suggests, this receive combining minimizes the mean squared error between the transmitted signals $\mathbf{z}$ and the processed received signal $\hat{\mathbf{z}}_{\text{SC}}$ (and similarly in the frequency domain) for $\mu_{\text{MMSE}} = 1$. This scheme is directly related to the RZF precoder.

The strong relationship between linear transmit precoding and linear receive combining implies that one typically have $\mathbf{C} = \mathbf{G}$ (or $\tilde{\mathbf{C}} = \tilde{\mathbf{G}}$), where the equality holds at least up to a scaling factor. Hence, the base station does not have to compute the precoder $\mathbf{G}$ and the receive combiner $\mathbf{C}$ separately, but only one of them—which reduces the computational complexity. The hardware implementation of precoding and combining is further discussed in Section 4.4.

2.5 Pilot and user scheduling

The quality of the channel estimation depends both on the SNR and on the level of inter-cell interference. More precisely, the channel estimation error covariance matrix in (2.20) in Section 2.2 can be rewritten as

$$\mathbf{C}_{jlk} = p_{lk}\beta_{jlk}\left(1 - \frac{p_{lk}\beta_{jlk}B}{\sum_{\ell=1}^{J}\sum_{m=1}^{K} p_{\ell m}\beta_{j\ell m}\mathbf{v}_{i_{lk}}^{\mathrm{H}}\mathbf{v}_{i_{\ell m}} + N_0}\right)\mathbf{I}_M, \tag{2.77}$$

which emphasizes that the error depends only on the noise variance N_0 (as compared to the effective channel variance $p_{lk}\beta_{jlk}$), on the pilot length B, and on UEs that have been allocated the same pilot signal; that is, which of the products $\mathbf{v}_{i_{lk}}^{\mathrm{H}}\mathbf{v}_{i_{\ell m}}$ that are non-zero. Consequently, the allocation of pilots across cells can have an important impact on the performance and there should preferably be a large difference in variances between users that reuse the same pilots. Since each BS is mainly interested in the channels to users in its own cell, this means that BS j would like to allocate orthogonal pilots among the users in each cell (this requires $B \geq K$) and also make sure that the ratio $\beta_{jjk}/\beta_{j\ell m}$ between the intra-cell variance β_{jjk} to its kth user and the inter-cell variance $\beta_{j\ell m}$ to user m in cell ℓ is large enough. This principle has been illustrated for one-dimensional networks in [28] and for two-dimensional networks with strong spatial correlation in [27,68], but there is certainly a need to develop new algorithms and scalable pilot allocation algorithms for practical networks that cannot rely on spatial correlation.

A simple and robust pilot allocation allocation was recently proposed in [9], based on classical cell planning for regular networks with hexagonal cells [16]. The idea is to have a $B = \alpha K$ pilot signals, where $\alpha \geq 1$ is the pilot reuse factor and, thus, every cell uses only the fraction $1/\alpha$ of the pilot signals. The worst-case ratio $\beta_{jjk}/\beta_{j\ell m}$ is 0 dB for $\alpha = 1$, while it becomes approximately 11 dB for $\alpha = 3$ and 18 dB for $\alpha = 7$ [45]. The average-case ratio might be much larger than this; particularly if one avoids to schedule cell edge users at the same time in the cells. These improvements in channel estimation quality seems to be worthwhile, because [9] shows that the highest spectral efficiency is usually achieved for $3 \leq \alpha \leq 7$.

2.5.1 Essence of Pilot Contamination

Although the LMMSE estimator in (2.17) allows for estimation of all channel vectors in the complete network, each BS can only resolve B different spatial dimensions in non-line-of-sight propagation since there are only B orthogonal pilot signals. To show this explicitly, we define the $M \times B$ matrix

$$\widehat{\mathbf{H}}_{\mathcal{V},j} = \left[\left(\mathbf{v}_1^{\mathrm{H}}\boldsymbol{\Psi}_j^{-1} \otimes \mathbf{I}_M\right)\mathrm{vec}(\tilde{\mathbf{Y}}_j), \ldots, \left(\mathbf{v}_B^{\mathrm{H}}\boldsymbol{\Psi}_j^{-1} \otimes \mathbf{I}_M\right)\mathrm{vec}(\tilde{\mathbf{Y}}_j)\right] \tag{2.78}$$

using each of the B pilot signals from $\mathcal{V}$. The channel estimate in (2.17) for UE k in cell l is parallel to the i_{lk}th column of $\widehat{\mathbf{H}}_{\mathcal{V},j}$; more precisely, if $\bar{\mathbf{h}}_{jlk} = \mathbf{0}$ then we have

$$\hat{\mathbf{h}}_{jlk}^{\mathrm{eff}} = p_{lk}\beta_{jlk}\widehat{\mathbf{H}}_{\mathcal{V},j}\mathbf{e}_{i_{lk}} \tag{2.79}$$

where $\mathbf{e}_i$ denotes the ith column of the $B \times B$ identity matrix $\mathbf{I}_B$. This is the essence of pilot contamination; BSs cannot tell apart UEs that use the same pilot signal and thus cannot reject the corresponding interference. In some cases (e.g., for slow changes in the user scheduling and high spatial channel correlation), user-specific statistical prior knowledge can be utilized to partially separate the UEs [68], but this will not be considered herein since we aim at establishing fundamental system properties that can be reliable applied in any propagation setup.

2.5.2 Mobility and Pilot Sharing

Each user might have different dimensions of its coherence block, defined by some coherence time $\tilde{T}_c$ and coherence bandwidth $\tilde{W}_c$, depending on the propagation environment and mobility. Suppose that $\tilde{T}_c = aT_c$ and $\tilde{W}_c = bW_c$ for a certain UE, where $a \geq 1$ and $b \geq 1$ since the frame structure was defined to fit into the coherence block of all UEs. Then, $\tau = \lfloor a \rfloor \lfloor b \rfloor$ is the total number of frames that fits into the coherence block of this particular UE. If $\tau > 1$, there is no need to transmit pilots in every frame; it sufficient to send pilots in $1/\tau$ of the frames. Consequently, multiple UEs with $\tau > 1$ can *share* a pilot signal without disturbing one another if the pilot is transmitted in different frames.

2.6 SC-FDE and OFDM with Precoding

2.6.1 Comparison of OFDM and SC-FDE combined with precoding

This section provides an overview of the possible modulation schemes for massive MIMO precoding. The assumption is that individual symbols from classical constellations (e.g. M-PSK or M-QAM) are transmitted. In other word, we do not address here modulation schemes with memory such as continuous phase modulation. Since the focus of this section is on the modulation scheme, we will do most of the discussions and mathematical representations for the SISO case, for clarity and when relevant. Indeed, for the MIMO case, we can most often straightforwardly extend the model to a "block" model with block-circulant, block-diagonal and block-Fourier matrices as introduced in the beginning of Section 2.3. Specific MIMO considerations will be added when needed. We will also neglect the receiver noise to make equations clearer.

Pure SC vs block transmission

There are two basic approaches to transmit sequences of symbols:

1. pure single carrier, in which a long sequence of symbols are transmitted in the time-domain. This is a pure time-domain approach; in case of multipath, the receiver usually mitigates inter-symbol interference by means of a time-domain equalizer in the form of a linear feedforward equalizer or a more complex decision-feedback equalizer consisting of a feedforward and a feedback section. The discrete-time baseband equivalent model is simply for the received signal
$$y[n] = h[n] * u[n] \tag{2.80}$$
and for the signal equalized with a feedforward equalizer $e[n]$:
$$\hat{u}[n] = e[n] * h[n] * u[n] \tag{2.81}$$

2. block transmission, in which the sequence of symbols is split into blocks of equal length N. A linear transformation can be applied on each block individually (for example but not necessarily a discrete inverse Fourier transform) and a length L guard interval comprising L symbols is prepended so that, if the channel length is smaller than L, the inter-block interference can be perfectly eliminated by discarding the received guard interval at the receiver. In most systems, the guard interval also plays the role of a cyclic extension, which enables simple frequency-domain equalization. We will discuss various types of

cyclic extension in detail below. The received signal vector $\mathbf{y}[n]$ and the equalized signal vector $\hat{\mathbf{u}}[n]$ are given by:

$$\mathbf{y}[n] = \mathbf{H}[n]\mathbf{u}[n] \tag{2.82}$$

$$\hat{\mathbf{u}}[n] = \mathbf{E}[n]\mathbf{H}[n]\mathbf{u}[n] \tag{2.83}$$

in which $\mathbf{H}[n]$ has size $(N+L-1)\times N$ and $\mathbf{E}[n]$, the equalizer matrix, has size $N\times(N+L-1)$.

Block transmission options and cyclic extension options

OFDM and SC-FDE. OFDM is probably the most well-known and most widely used block transmission scheme. It takes a block of N symbols, performs an IFFT of this block, adds a length L cyclic extension (note here that *extension* is more general than *prefix*) and transmits the resulting time-domain signal. At the receiver, the cyclic extension is removed and the signal is converted to the frequency domain (FD) for per sub-carrier equalization. SC-FDE is less common but is now in use in two prominent wireless standards (uplink of LTE/LTE-advanced [3] and IEEE802.11ad [20]). In SC-FDE, a block of N time-domain symbols receives directly a length L cyclic extension. The resulting length $N+L$ block is transmitted. At the receiver, the following operations are performed: an FFT, a per sub-carrier equalization and an IFFT.

Details of the cyclic extension. There are three common forms of cyclic extensions that are widely used and/or documented in the literature and that apply to OFDM and SC-FDE: cyclic prefix (CP), known symbol padding (KSP) and zero-padding (ZP).

Cyclic prefix. The cyclic prefix technique consists in taking the last L symbols or samples of a block and copying them in front of the block. Mathematically, the OFDM scheme with the cyclic prefix can be expressed as follows ($\mathbf{u}$ is the vector of information symbols and $\mathbf{x}$ is the vector of transmitted signals):

$$\mathbf{x} = \mathbf{T}_{CP}\mathbf{F}^H\mathbf{u} \tag{2.84}$$

$$\mathbf{r} = \mathbf{H}\mathbf{T}_{CP}\mathbf{F}^H\mathbf{u} \tag{2.85}$$

$$\begin{aligned}\hat{\mathbf{u}} &= \tilde{\mathbf{E}}\mathbf{F}\mathbf{R}_{CP}\mathbf{H}\mathbf{T}_{CP}\mathbf{F}^H\mathbf{u},\\ \hat{\mathbf{u}} &= \tilde{\mathbf{E}}\mathbf{F}\breve{\mathbf{H}}\mathbf{F}^H\mathbf{u},\\ \hat{\mathbf{u}} &= \tilde{\mathbf{E}}\tilde{\mathbf{H}}\mathbf{u}\end{aligned} \tag{2.86}$$

where the size $(N+L)\times N$ matrix $\mathbf{T}_{CP}$ and size $N\times(N+L)$ matrix $\mathbf{R}_{CP}$ are the matrices inserting and removing the cyclic prefix, the size $(N+L)\times(N+L)$ matrix $\mathbf{H}$ is the Toeplitz channel convolution matrix and $\mathbf{F}$ is a Fourier matrix. Note that the matrix $\breve{\mathbf{H}} = \mathbf{T}_{CP}\mathbf{H}\mathbf{T}_{CP}$ is circulant and that it is diagonalized by the Fourier matrix resulting in $\tilde{\mathbf{H}} = \mathbf{F}\breve{\mathbf{H}}\mathbf{F}^H$. Both the FD channel matrix $\tilde{\mathbf{H}}$ and the FD equalizer matrix $\tilde{\mathbf{E}}$ diagonal, which implies simple implementation of the equalizer.

The CP scheme can be used as such with SC-FDE. The system model can be written as follows:

$$\begin{aligned}\hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\mathbf{F}\mathbf{R}_{CP}\mathbf{H}\mathbf{T}_{CP}\mathbf{u},\\ \hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\mathbf{F}\breve{\mathbf{H}}\mathbf{F}^H\mathbf{F}\mathbf{u},\\ \hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\tilde{\mathbf{H}}\mathbf{F}\mathbf{u}\end{aligned} \tag{2.87}$$

Matrices $\tilde{\mathbf{E}}$ and $\tilde{\mathbf{H}}$ are identical to those in (2.86). Note that the matrix $\mathbf{F}$ in (2.87) appears from mathematical transformations but is not implemented at the TX side. The RX side, however needs to implement both an FFT and an IFFT. The FFT/IFFT size in CP-OFDM and CP-SC is the size of the block N, before adding the CP.

Known symbol padding. The KSP technique consists in appending always the same cyclic extension to all blocks instead of prepending some data symbols or samples. We will illustrate this for the SC-FDE case. The transmitted block of symbols consists of N data symbols and a length L vector of known symbols $\mathbf{p}$ which is the same for all blocks:

$$\mathbf{x} = \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}. \tag{2.88}$$

We need to consider the inter-block interference (IBI) channel matrix model [65] to introduce this cyclic scheme. The signal from the current block but including the interference from the previous block is as follows:

$$\begin{aligned} \mathbf{r} &= \mathbf{H}_{IBI} \begin{bmatrix} \mathbf{0}_{N\times 1} \\ \mathbf{p} \end{bmatrix} + \mathbf{H} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}, \\ \mathbf{r} &= \breve{\mathbf{H}} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}. \end{aligned} \tag{2.89}$$

Hence, the effect of the known symbol padding results also in a circulant channel matrix $\breve{\mathbf{H}}$ and the same receiver processing as in (2.87) can be applied for the FD equalization. An interesting feature of the KSP scheme is that, because the cyclic extension is known by the receiver, it can be used by the receiver after equalization for processing such as channel tracking or carrier/phase tracking. The FFT/IFFT size in KSP-SC is the size of the block N augmented with the cyclic extension length L. Thus it has size $N + L$.

The KSP scheme can in principle be used with OFDM (it is used in the DTMB broadcasting standard [59]). However, there is an inherent added complexity in KSP-OFDM which is that the length over which the channel appears circulant is $N + L$ whereas the FD OFDM symbols are encoded with a size N FFT. Hence, the receiver needs to perform a size $N + L$ FFT, equalization and IFFT followed by a size N FFT to recover the frequency domain symbols. It is therefore very complex at the receive side since it involves three (I)FFTs.

Other cyclic extension schemes. For the sake of completeness, we have to mention two other cyclic extension schemes: zero-padding (ZP) and unique-word OFDM (UW-OFDM). ZP can be applied to SC-FDE and OFDM. There are several receiver alternatives for ZP [41]. One of the drawbacks of ZP is that the transmitter amplitude varies sharply during the cyclic extension, which is not a desirable feature. Another major drawback is that, although the ZP symbols can be seen as a form of KSP ("zero" amplitude known symbols), they cannot be used for tracking purposes. Hence, additional overhead is needed for pilot signals.

UW-OFDM reserves certain frequency domain symbols to ensure that, after the IFFT, the first N samples of the time-domain signal are forced to a fixed sequence of N symbols (a sort of KSP but enforced in the frequency domain). This technique is very complex because the symbols that must be sent on the reserved tones are data-dependent and the complexity is significant for large FFT sizes.

Conclusion for the OFDM and SC-FDE schemes without precoding. The most attractive schemes for SISO without precoding are the CP-OFDM, CP-SC-FDE and KSP-SC-FDE. Without precoding, KSP-SC-FDE is preferred over CP-SC-FDE because it provides training symbols for free with each block. In the next section, we will see how these schemes can be used for (MIMO) precoding.

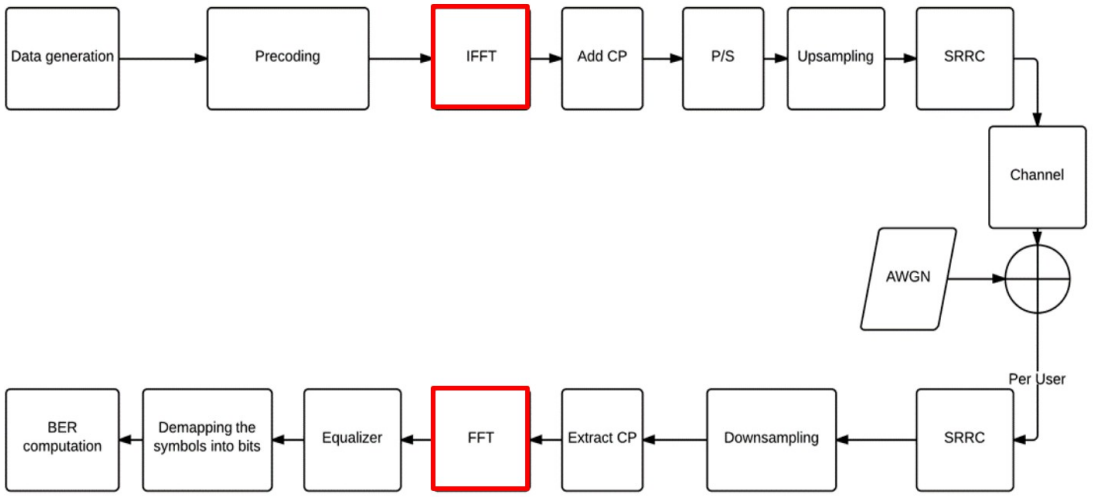


Figure 2.16: OFDM transmission block diagram.

Block Transmission and Precoding

Still assuming a SISO system, the FD precoding consists in pre-multiplying each transmitted block with a diagonal FD precoding matrix to pre-compensate the channel response. We will now go over the three preferred schemes (CP-OFDM, CP-SC-FDE and KSP-SC-FDE) and see how they can be used with precoding and MIMO.

Precoding with CP-OFDM. Precoding with CP-OFDM is straightforward. It can be represented as follows:

$$\begin{aligned} \hat{\mathbf{u}} &= \tilde{\mathbf{E}}\mathbf{F}\mathbf{R}_{CP}\mathbf{H}\mathbf{T}_{CP}\mathbf{F}^H\tilde{\mathbf{G}}\mathbf{u}, \\ \hat{\mathbf{u}} &= \tilde{\mathbf{E}}\mathbf{F}\breve{\mathbf{H}}\mathbf{F}^H\tilde{\mathbf{G}}\mathbf{u}, \\ \hat{\mathbf{u}} &= \tilde{\mathbf{E}}\tilde{\mathbf{H}}\tilde{\mathbf{G}}\mathbf{u} \end{aligned} \tag{2.90}$$

We end up with diagonal matrices at the TX and RX side and a diagonal channel matrix thanks to the cyclic prefix. This can be directly extended to MIMO precoding with block matrices, hence simple per-sub-carrier precoding and RX equalization.

The OFDM transmission block diagram is illustrated in Figure 2.16, highlighting the (I)FFT blocks.

Precoding with CP-SC-FDE. Precoding with CP-SC-FDE is similarly straightforward. It can be represented as follows (neglecting the receiver noise):

$$\begin{aligned} \hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\mathbf{F}\mathbf{R}_{CP}\mathbf{H}\mathbf{T}_{CP}\mathbf{F}^H\tilde{\mathbf{G}}\mathbf{F}\mathbf{u}, \\ \hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\mathbf{F}\breve{\mathbf{H}}\mathbf{F}^H\tilde{\mathbf{G}}\mathbf{F}\mathbf{u}, \\ \hat{\mathbf{u}} &= \mathbf{F}^H\tilde{\mathbf{E}}\tilde{\mathbf{H}}\tilde{\mathbf{G}}\mathbf{F}\mathbf{u} \end{aligned} \tag{2.91}$$

Compared to the SC-FDE case without precoding, we have here additional processing due to the FFT, diagonal precoding matrix and IFFT. Since it involves diagonal matrices, this scheme can also straightforwardly be extended to MIMO set-ups with block matrices and per sub-carrier operation. The down side is that we need two FFT/IFFT at both the TX and RX, hence the complexity is a bit higher.

The SC-FDE transmission block diagram is illustrated in Figure 2.17, highlighting the (I)FFT blocks.

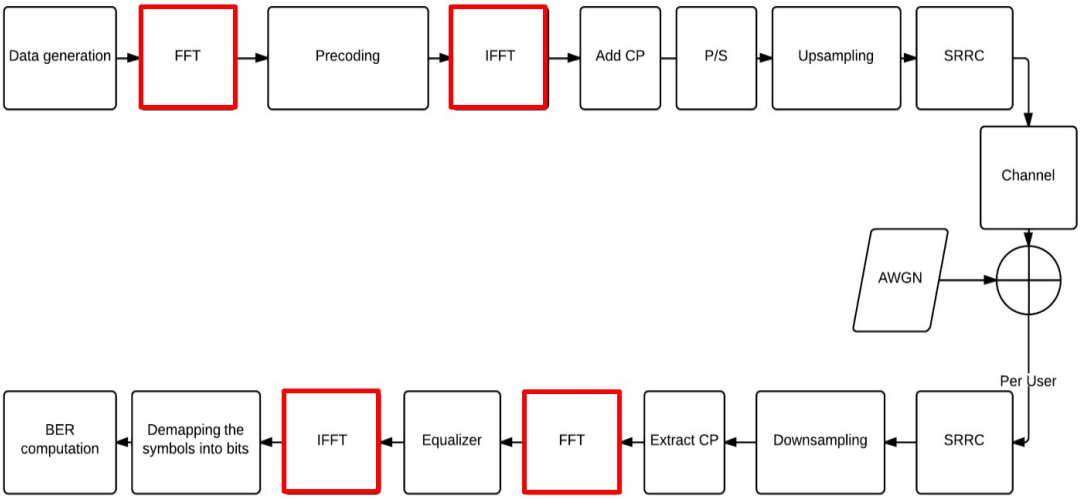


Figure 2.17: SC-FDE transmission block diagram.

Precoding with KSP-SC-FDE. For the precoding with KSP, the question arises as to where the KS must be appended at the TX side: before or after the precoding. Let us investigate both cases.

If the KS is appended after the precoding, the TX model reads:

$$\mathbf{x} = \begin{bmatrix} \mathbf{F}^H \tilde{\mathbf{G}} \mathbf{F} \mathbf{u} \\ \mathbf{p} \end{bmatrix} \tag{2.92}$$

This will indeed make the *propagation* channel matrix circulant but the channel seen by the receiver also includes the precoder (upper part of the bracketed expression in (2.92) which is not made circulant. Clearly, this does not work.

If the KS is inserted before the precoding, the TX model can be written as:

$$\mathbf{x} = \mathbf{F}^H \tilde{\mathbf{G}} \mathbf{F} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} \tag{2.93}$$

The transmitted signal in (2.93) will in general not have the desired structure of (2.88) which, for KSP-SC-FDE without precoding, made the channel appear to be circulant. A possible work around would be to compute a different value for $\mathbf{p}$ in (2.93) such that the TX signal $\mathbf{x}$ does have the structure of (2.88). This could be achieved as follows, by splitting the Fourier matrix into two parts (the columns multiplying $\mathbf{u}$ are collected in $\mathbf{F}_d$ and the columns multiplying $\mathbf{p}'$ are collected in $\mathbf{F}_{p'}$:

$$\begin{aligned} \begin{bmatrix} \mathbf{s}' \\ \mathbf{p} \end{bmatrix} &= \mathbf{F}^H \tilde{\mathbf{G}} \mathbf{F} \begin{bmatrix} \mathbf{u} \\ \mathbf{p}' \end{bmatrix} \\ \begin{bmatrix} \mathbf{s}' \\ \mathbf{p} \end{bmatrix} &= \begin{bmatrix} \mathbf{F}_d^H \\ \mathbf{F}_p^H \end{bmatrix} \tilde{\mathbf{G}} \begin{bmatrix} \mathbf{F}_d & \mathbf{F}_p \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p}' \end{bmatrix} \end{aligned} \tag{2.94}$$

The vector $\mathbf{p}'$ can then be calculated as:

$$\mathbf{p}' = (\mathbf{F}_p^H \tilde{\mathbf{G}} \mathbf{F}_p)^{-1} (\mathbf{p} - \mathbf{F}_p^H \tilde{\mathbf{G}} \mathbf{F}_d \mathbf{u}) \tag{2.95}$$

This scheme is not very attractive because the complexity is already significant in this SISO derivation. In a MIMO set-up, it would become even more complex with block matrices. The

vector $\mathbf{p}'$ is data dependent and, hence, must be recalculated for every block. A final argument against this approach is that the vector $\mathbf{p}$, which is artificially enforced after the precoding, does not undergo the same channel as the data vector $\mathbf{u}$. Hence, after RX equalization, $\mathbf{p}$ cannot be used directly for tracking. It should be noted that this approach is actually similar to the approach in UW-OFDM mentioned earlier in which we also concluded that it was too complex for practical systems.

Conclusion for cyclic extension. Block transmission for SISO or MIMO precoding is only viable using the cyclic prefix as a cyclic extension. KSP, although better when no precoding is applied, is too complex when precoding is applied because the known symbol cannot easily be appended to the block, neither before nor after the precoding. Comparing CP-OFDM and CP-SC-FDE, the latter is slightly more complex because two FFT/IFFTs are needed at both the TX and RX sides, against only one FFT/IFFT at both sides for CP-OFDM. Since FFTs can be implemented quite efficiently in deeply scaled CMOS technology, this complexity increase is moderate and other criteria will need to be evaluated to decide between CP-OFDM and CP-SC-FDE.

An argument in favor of CP-SC-FDE is that, if the channel is sufficiently pre-equalized by the precoding, the receiver "sees" a frequency flat channel and the RX FD equalization is possibly not needed; a simple time-domain equalizer with one or a very small amount of taps could be used at RX. This can be decided at design time or at run time.

2.6.2 SC Precoding without Cyclic Prefix

When no cyclic prefix is used, we need to use the convolution model to represent to channel effect and the precoding/equalization processes. This can be done as follows.

SISO SC precoding

The discrete-time baseband equivalent model of the precoded signal is

$$s[n] = h[n] * g[n] * d[n]. \tag{2.96}$$

In a MIMO set-up, this would become

$$\mathbf{x}[n] = \mathbf{H}[n] * (\mathbf{G}[n] * \mathbf{u}[n]) \tag{2.97}$$

where $\mathbf{H}[n]$ and ($\mathbf{G}[n]$ are matrices of impulse responses (each entry of the matrix is an impulse response).

SISO and MIMO SC MRT precoders

A possible precoding scheme is to resort to MRT; in this case, the SISO precoder is

$$g[n] = h^*[T - n] \tag{2.98}$$

and the MIMO precoder is

$$\mathbf{G}[n] = \mathbf{H}^H[T - n]. \tag{2.99}$$

in which we have introduced a delay T, greater than the channel length, to have causal precoders. This is just the transmit matched filter in the time-domain which is basically the complex-conjugate time-reversed of each impulse response appearing in the channel matrix $\mathbf{H}$.

It should be noted that we have not introduced a normalizing factor in these time-reversal precoders but it may be needed. Concerning implementation, a key point here is the impulse responses appearing in the precoder equations can be very long and that, implementation-wise, this can be challenging since it implies long FIR filters with programmable coefficients.

SISO and MIMO SC ZF precoders

For ZF SC precoders, we need to switch to a model with convolution matrices so that we can manipulate the matrices to compute e.g. inverses. For a SISO system, (2.96) becomes

$$\mathbf{x}[n] = \mathbf{HGu}[n] \tag{2.100}$$

where $\mathbf{H}$ and $\mathbf{G}$ are Toeplitz convolution matrices, with the impulse response $h[n]$ and $g[n]$ repeated in all columns with an index shift. If the impulse response $h[n]$ has length T and the vector $\mathbf{u}[n]$ contains N symbols, the convolution matrix $\mathbf{H}$ has size $(N+T-1)\times N$ if no precoding is applied. If a precoder $g[n]$ of length S is applied, then $\mathbf{G}$ has size $(N+S-1)\times N$ and $\mathbf{H}$ has size $(N+S+T-2)\times(N+S-1)$. Stricto sensu, this is also a block model as in Section 2.6 but it is not a block transmission scheme that eliminates inter-block interference by design (without equalizer).

The ZF precoder $\mathbf{G}$ can be computed as the pseudo-inverse of $\mathbf{G} = \mathbf{H}^{\dagger}$. If it is used in this way, it cannot be implemented as a convolution with an FIR filter because, $\mathbf{G}$ is not Toeplitz. A more implementation friendly solution is to take the central column of $\mathbf{G}$. This column implements a zero-forcing solution with the strongest response at an index corresponding to the chosen column index. It should be noted that, if the pseudo-inverse is computed on the baseband rate convolution matrix (thus without oversampling), the zero-forcing solution needs a very large number of taps for near-zero sidelobes. A much shorter response can be achieved with an oversampling factor of 2.

The size of the channel convolution matrix can be quite large (depending on the length of the channel impulse response); hence, the computation of $\mathbf{G}$ can be complex.

For a MIMO system, we still have a transmit model of the form

$$\mathbf{x}_M[n] = \mathbf{H}_M\mathbf{G}_M\mathbf{u}_M[n] \tag{2.101}$$

but here $\mathbf{H}_M$ and $\mathbf{G}_M$ are block Toeplitz matrices and $\mathbf{x}_M[n]$ and $\mathbf{u}_M[n]$ are vectors resulting from the concatenation of the TX and RX vectors, respectively, of the different users. For large MIMO systems, this approach is impractical because of the very large size of the channel convolution matrix $\mathbf{H}_M$ that has to be inverted. Even for an MRT transmitter, the number of multiplications involved in the product $\mathbf{G}_M\mathbf{u}_M$ is extremely large.

2.6.3 Power amplifier effect on OFDM and SC-FDE

We will analyze the effect of the power amplifier on OFDM and SC-FDE from the perspective of the BER and the PAPR.

BER analysis

We ran downlink simulations with 16QAM symbols and $K = 4$ users and $M = 4, 8, 16$ and 32 antennas. Different levels of back-off (with respect to the 1-dB compression point) were applied. The precoding scheme was zero-forcing. The results are shown in Figure 2.18 and 2.19 for OFDM and SC-FDE, respectively. A quick inspection of Figure 2.18 and 2.19 reveals that:

- both systems have very similar performances without PA non-linearity
- the PA impact is similar on both systems,
- both systems are very sensitive to PA non-linearities at full system load. They do not work properly for 0 or 3dB back-off,

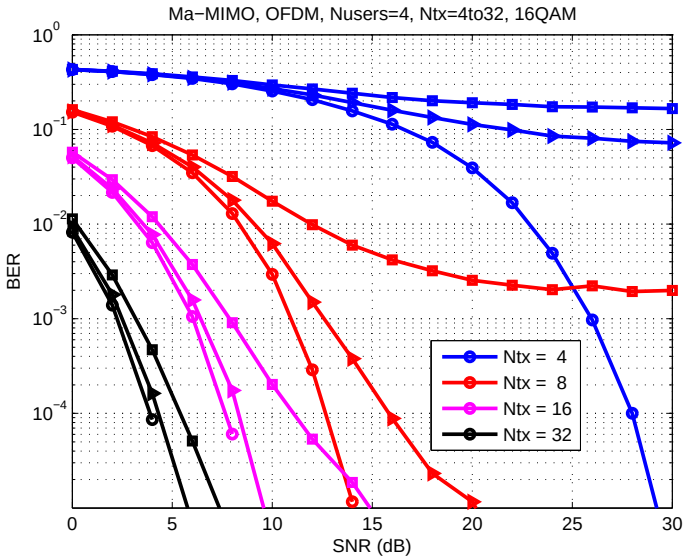


Figure 2.18: BER performance for OFDM 16QAM (circles: no PA; squares: PA 0dB back-off; triangles: PA 3dB back-off).

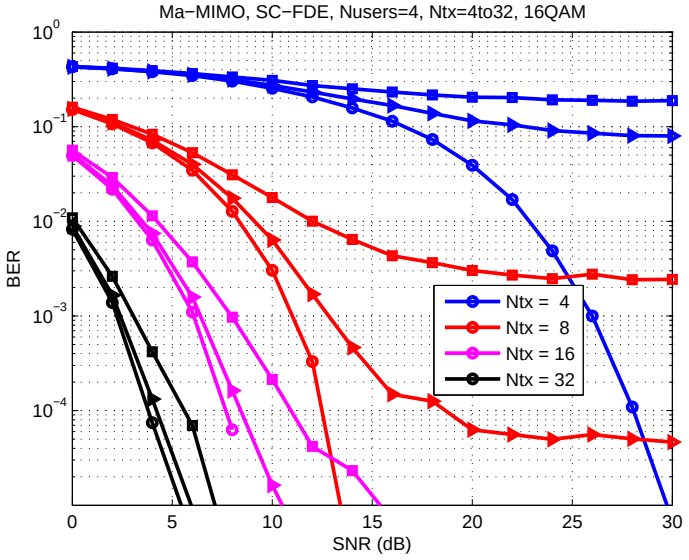


Figure 2.19: BER performance for OFDM 16QAM (circles: no PA; squares: PA 0dB back-off; triangles: PA 3dB back-off).

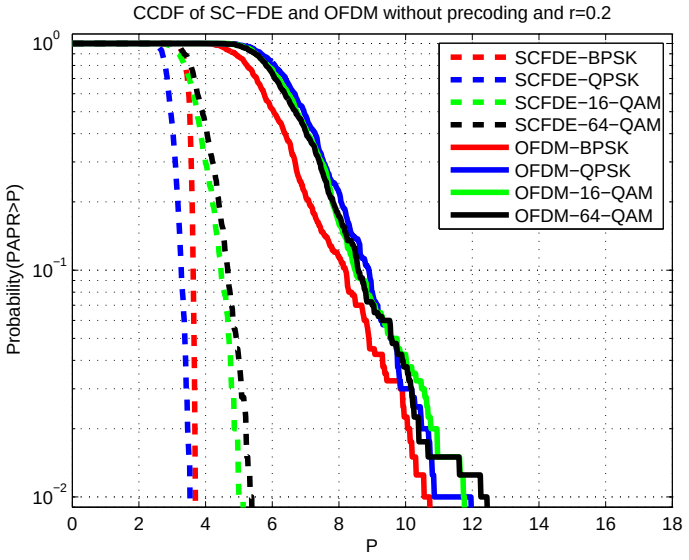


Figure 2.20: PAPR for OFDM and SCFDE without precoding.

- both systems can be operated at half system load if sufficient back-off is applied (3 dB or more),
- from 25% system load and below, very little back-off is needed.

This confirms the "averaging" effect of non-idealities when they are uncorrelated. It is interesting to note that one of the advantages of SC-FDE over OFDM in SISO systems - the lower sensitivity to PA non-linearity - is lost with precoding. This is due to the fact that the FD precoding destroys the good PAPR property of SC modulation, especially for low order constellations.

PAPR analysis

We can confirm the BER impact analysis by looking at the PAPR complementary cumulative distribution function (CCDF) of CP-SC-FDE and CP-OFDM with and without precoding, which are shown in Figure 2.20 for the case without precoding and in Figure 2.21 for the case with precoding (ZF, 4 users, 32 TX antennas). We observe that the PAPR advantage of SC-FDE without precoding completely vanishes when precoding is applied. We have verified this for other load scenarios (100%, 50% and 25%): the PAPR CCDF are almost identical to the results shown in Figure 2.21.

2.6.4 Conclusions

The key conclusions of this section are as folows. CP is the only viable cyclic extension option for OFDM and SC-FDE when precoding is applied. Precoded CP-OFDM and CP-SC-FDE have very similar BER performance under ideal conditions and very similar BER performance degradation when PA non-linearity is applied. The "averaging" effect on the non-linearity degradation is clearly visible when the system load is not too high (<25%) and the PAPR of precoded CP-OFDM and CP-SC-FDE are very similar. Pure SC precoding without CP has a reasonable complexity for MRT precoding but cannot be realistically implemented for ZF or MMSE precoding. CP-SC-FDE needs two (I)FFTs at TX side; it also needs two (I)FFTs at

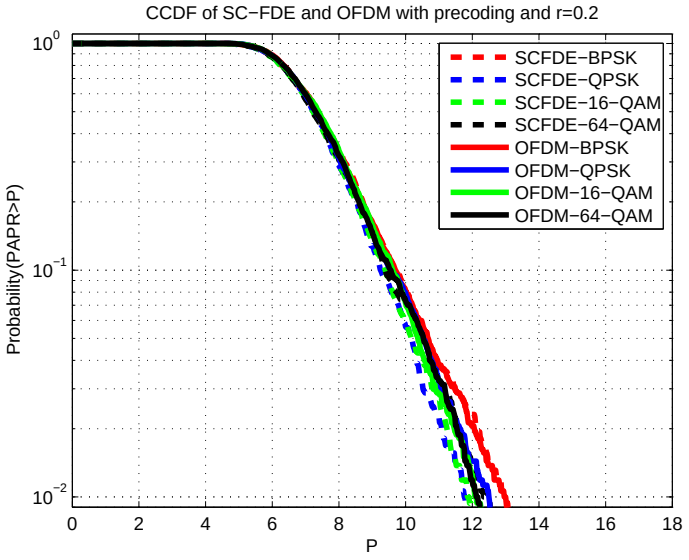


Figure 2.21: PAPR for OFDM and SCFDE with precoding, ZF, 4 users, 32 TX antennas.

RX side if an RX equalizer is needed. Interestingly, the RX equalizer for CP-SC-FDE may not be needed if the channel is pre-equalized (i.e. made frequency flat) by the TX precoding; this can save significant terminal side complexity.

Chapter 3

Processing hardware

In this chapter, we present isolated hardware components or building blocks that can be used in order to perform specific massive MIMO DSP operations, as well as platforms combining a number of those components at a higher level in order to provide the complete required functionality.

3.1 Hardware components and accelerators

When hardware components have to be designed or selected in order to support a given functionality, multiple dimensions come into play, with conflicting high-level objectives:

- Cost (including silicon area impact)
- Power consumption (peak as well as low-load or sleeping)
- Flexibility (covering multiple modes or standards)
- Throughput (data processing speed)
- Time to market

As far as digital sub-components are concerned, the following categories are available, with specific advantages as well as drawbacks.

ASICs are specialized integrated circuits designed for a well-defined application. They are best in power consumption as well as throughput, thanks to very specific design optimziation. Their main drawback is the lack of flexibility they offer. Their cost is low per unit but very large as overhead (design and NRE, masks...). This is especially the case in the latest deeply-scaled technologies. Hence, they are only relevant for mass production.

FPGAs offer programmability based on pre-structured hardware templates that can be tuned in order to reproduce any functional behavior. While being more expensive than ASICs and less power-efficient, they offer a very convenient flexibility for prototyping, and also strongly reduce the overhead of ASICs when targeting a faster design at lower risk.

DSPs are the most generic components for signal processing. Hence, they offer a lot of flexibility and a very fast time-to-market, given that only software programming is needed. However, they are much less efficient than other components in power consumption, and also suffer from a lower throughput due to the overhead of being generic.

ASIPs are more recent than the other categories. They are processors, offering flexibility as DSPs do, but tuned to a specific domain of application. This enables a much more optimized

design, reaching throughput and power consumption figures closer to the ASIC designs, while keeping the necessary flexibility in the targeted domain of application.

At platform level, **software-defined radios** (SDRs) offer an advantageous flexibility as described in Section 3.2.1. If they are custom designed for massive MIMO they can use hybrid architectures, combining multiple of the above components in order to get exactly the required flexibility and performance while minimizing cost, time-to-market and power consumption.

3.2 Massive MIMO platforms

3.2.1 SDR architectures

A key technology that enables seamless wireless connectivity in a flexible way is Software Defined Radio (SDR). A software defined radio is a wireless communication system in which some of the functionality is implemented on a programmable and/or reconfigurable platform [35, 53].

The importance of SDR solutions is driven by two facts. First, to enable seamless connectivity the mobile devices have to support communications to various wireless networks such as GSM, WCDMA, LTE, WiFi (IEEE 802.11a/b/g/n/ac) or Bluetooth [31]. In a traditional approach, each standard is implemented using a dedicated ASIC. However, supporting multiple standards using dedicated ASICs results in a costly design in terms of power and area. Besides, all the standards are not active simultaneously all the time, so having a dedicated chip for each standard is an over-designed system. Thus, a more cost-effective solution is to have a flexible architecture that can support multiple standards and can switch between them over time.

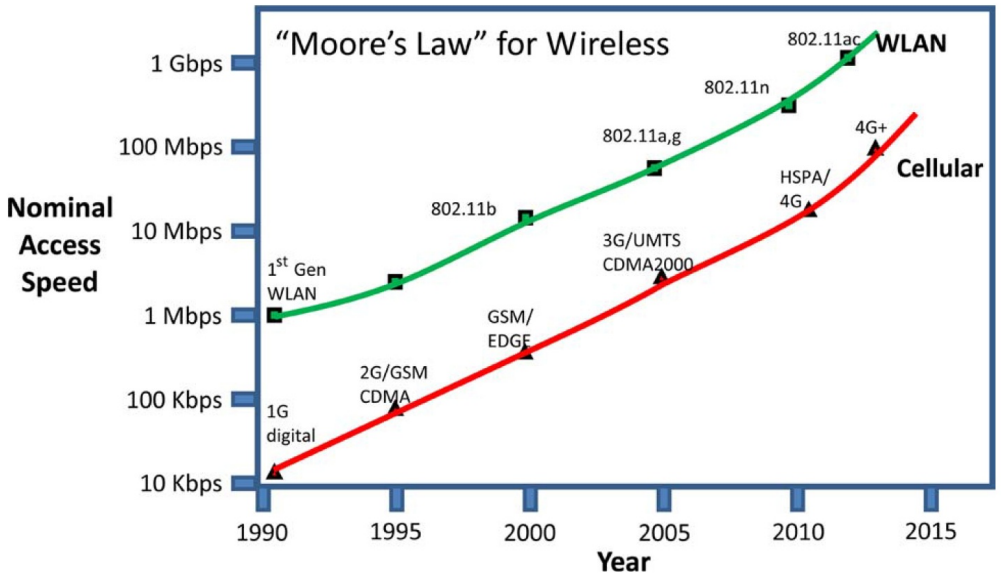


Figure 3.1: Evolution of wireless standards over the past 15 years [55].

Secondly, the number of wireless standards is evolving at a tremendous rate in order to satisfy the ever growing user demands and application scenarios. According to Edholm's law [13], the data rate of communications increases about 100 times every 10 years. Figure 3.1 shows the evolution of WLAN and cellular wireless standards, illustrating that almost every 5 years there is a new wireless standard being proposed and within each standard new releases are drafted almost every year [33]. For instance, after the release of 3GPP-LTE Rel. 8 in 2008, the 3GPP standard body has released significantly new features every year or two: Rel. 10 in 2011, Rel. 11

in 2012, Rel. 12 in 2014 and currently work is under progress for Rel. 13 [1]. Such a fast evolution makes SDR a very attractive solution, as it enables to reuse software and hardware architecture templates. Ideally, with a programmable architecture, new releases of a current standard can be updated by a software upgrade and only major changes require the introduction of a new chip. As a result time-to-market can be reduced and the time-in-market can be increased. This is beneficial not only for mobile devices, but also for infrastructure such as base stations, for which hardware updates are very costly operations.

As an example, both industry and academia have spent a significant effort in MIMO-SDR design. Given that MIMO-OFDM is the underlying technology in LTE and 802.11n/ac, an SDR solution can ideally be designed to support both standards. However, creating practical SDR baseband solutions providing high flexibility, reusability, and a high energy efficiency still remains a significant challenge. This requires MIMO baseband signal processing algorithms to be implemented on programmable architectures. The area and energy efficiency of such programmable architectures is typically much lower than ASIC, simply because ASICs are designed and optimized for a fixed functionality. An SDR baseband processor has a lot of overhead in order to enable programmability, such as an instruction decoder, instruction memory hierarchy, programmable interconnect, general purpose data memory and various functional units to support many different baseband algorithms. However, this efficiency gap can be reduced by implementing scalable algorithms that can adapt to the varying channel conditions or user requirements. In short, scalable algorithms providing multiple modes of operations to trade-off bit-error-rate (BER) performance and computational complexity are required, to increase the average efficiency of the system.

3.2.2 State of the art of SDR baseband processors

In order to meet the programmability and flexibility requirements of SDR solutions, many companies and universities have proposed programmable baseband processors. In general there is no agreed benchmark set in industry or academia to evaluate and compare SDR solutions [4]. For instance, a particular SDR solution might support both GSM and LTE standards while another one supports DVB-T/H and 802.11a standards, so a common benchmark cannot be set. Moreover, within each standard different algorithms can be employed for the same signal processing task, which can differ in BER performance, throughput, power and area requirements. Therefore, we are limited to give an overview of different SDR architectures. The purpose of this state-of-the-art overview is to familiarize the reader with the trends in architecture design of SDR baseband solutions. As proposed in [4], SDR architectures can be categorised into three different types: 1) DSP centered with hardware accelerators, 2) Multi-core and 3) Reconfigurable coarse grain arrays (CGA).

DSP centered with hardware accelerators

LeoCore by CoreSonic [32] is an ASIP for baseband processing targeting hand held devices. It is a processor centered design with SIMD cores or ASIC accelerators. In LeoCore the algorithms are evaluated for complexity and required operations and then mapped to suitable SIMD cores. It has been demonstrated for DVB-T/H and WiMax implementation. It delivers a throughput of 31.67 Mbps with 70 mW power consumption.
Sandblaster by SandBridge [60] is a multi-core multi-threaded vector processor. The main focus in their work has been the support of high level language and compiler optimization for DSP applications. The compiler analyses the C code and appropriately generates the SIMD vector operation to enable DLP. The chip SB3500 has 3 cores, each capable of executing SIMD

instructions with 4 threads. It has been demonstrated with implementation of 2×2 LTE Cat. 2 baseband processing.
ConnX BBE by Tensilica [57] is an SIMD processor with VLIW instructions. It uses Tensilica's Xtensa processor template to generate a baseband processor. Different processor configurations according to the application requirements can be generated using the Xtensa Processor Generator and Tensilica Instruction Extension (TIE). The ConnX baseband engine (BBE) has been synthesized for 65 nm and runs at 400 MHz. Essentially it is a VLIW processor with SIMD instructions, the differentiating factor is that TIE can automatically vectorize the code with little or no human intervention.
EVP by NXP [6] is a VLIW processor supporting both scalar and vector operations. The main data path supports 16 bit operations while 8 and 32 bit operations are also supported on EVP (Embedded Vector Processor). The application code is written in EVP-C which is an extension of ANSI-C. It has been demonstrated for a MMSE MIMO detector implementation for 2×2 802.11n standard [46]. The core can run at 300 MHz with 300 mW power consumption when synthesized for 90 nm.

Multi-core architectures

SODA [30] is a multi-core processor with separate processors for data and control operations. The data processor has 4 PE (Processing Elements) that supports both scalar and vector operations. 32 bit wide SIMD with 16 bit data width are used in each PE. The data processor executes computationally intensive kernels like FFT, FEC kernels and equalization while the control processor performs system operations and manages the data processor. In SODA (signal processing of demand), inter-kernel data communication is enabled by a global scratch pad memory. It has been demonstrated for the implementation of complete physical layer of W-CDMA and 802.11a standards [30]. The power consumption is 450 mW for 90 nm technology.
Ardbeg by ARM [66] is a commercial prototype based on SODA designed by ARM. The main changes in Ardbeg compared to SODA consist of an optimized wide SIMD design, related VLIW support for SIMD instructions and algorithm specific hardware acceleration. Ardbeg is also a multicore architecture, with a control processor and multiple PEs. Special ASIC accelerators are added for specific algorithms like turbo encoding/decoding. Each PE has a local scratch pad memory and shares a global memory as well. C-language support is provided which can take the C model directly from Matlab for compilation. It runs at 350 MHz in 90 nm technology, and dissipates 500 mW. This is demonstrated with the implementation of major kernels in DVB-T/H, W-CDMA and 802.11a [66].
Tomahawk 2 [43] is a multiprocessor System-on-Chip (MPSoC) with a heterogeneous array of PEs. As many other solutions it also exploits instruction, data and task level parallelism. The MPSoC has 8 Duo-PEs and dedicated units for MIMO detection and decoding. Each Duo-PE comprises a vector DSP and a RISC core, connected to a shared local memory. The distinguishing factor in Tomahawk 2 is its dynamic run-time manager (CoreManage, CM) which adapts to the dynamically varying workload of wireless applications. The CM analyses at runtime the scheduling requests and exploits the results to maximize data locality and to configure the dynamic voltage and frequency scaling (DVFS) of the PEs according to current system load, priorities and deadlines [43]. To accelerate computationally intensive SDR baseband algorithms, two programmable application-specific cores perform MIMO detection and FEC. It has been demonstrated to support LTE, WiMax and 802.11n. For 4×4 MIMO LTE baseband it achieves a throughput of 396 Mbps with 74.6 mW power dissipation running at 445 MHz.
X-Gold by Infineon [54] is a multi-processor SDR that combines an SIMD sub-system for physical layer signal processing and an ARM sub-system for control and communication to

upper layers [52]. The SIMD sub-system has 3 SIMD clusters and each SIMD cluster has 4 SIMD cores which can be programmed individually. Furthermore, each SIMD core has 4 PEs for vector operations. To execute standard specific algorithms that do not require flexibility such as Viterbi/turbo decoding, hardware accelerators are provided. The SDR20 instance has been fabricated in a 65 nm process and can be clocked at 300 MHz. It has been demonstrated to support GSM, UMTS Rel. 99, and 2×2 LTE downlink [52] while it is also claimed to be software upgradable to support 802.11a/b/g/n and DVB-T/H [2].

Reconfigurable coarse grain array architectures

Montium by Recore Systems [18] is a coarse grain array architecture supporting both fixed point and floating point operations. The FUs in Montium are arranged as tiles and connected by 10 global buses to provide the interconnect flexibility. The distinguishing feature of Montium is its multi-level ALU. Each ALU has two levels, one for general purpose computing and another for specific functions like FFT and filtering. These levels can be bypassed according to the needs of the algorithm. It has been demonstrated to support various algorithms such as, correlation, FIR filtering, DCT and FFT [18]. It comes with its own design tool called Montium Sensation Suite which has a compiler, a simulator and an editor. The compiler uses its proprietary language called Montium Configuration Design Language (CDL) for reconfiguration. It is demonstrated with implementation of W-CDMA and HiperLan2 [17].
HERS [42] stands for HEterogeneous Reconfigurable System. The basic idea is to divide and distribute the kernels (algorithms) among the reconfigurable engines (REs) based on the required computations. Each RE further consists of homogeneous PEs, optimized for specific operations. The PEs are connected as 8×8 array. To provide the inter-engine communication there is a high speed bus. In [42] it is demonstrated with the implementation of DVB-T/H and 802.11a/g, by employing 2 REs. It can be clocked at 250 MHz when synthesized using 90 nm technology.
ADRES by IMEC [12] is a reconfigurable CGA processor designed with the Dynamically Reconfigurable Embedded Systems Compiler (DRESC) tool suite. The platform consist of VLIW FUs and a CGA. SIMD operations are supported on the CGA while VLIW performs scalar operations. It can be operated in a VLIW mode or as a CGA processor and it can switch between the modes at run-time. The CGA can be reconfigured to support different array sizes up to 4×4 and SIMD widths. The architecture is C-programmable and the code is compiled with the DRESC compiler. Special intrinsics functions can be designed required by a specific algorithm which are then implemented by the DRESC compiler. The processor, designed in 90 nm process achieves a clock frequency of 400 MHz in worst case conditions and consumes maximally 310 mW. The mapping of complete baseband processing required in 4×4 LTE Cat.5 is demonstrated in [29] for an advanced MIMO receiver using an ADRES instance with 2 cores.

Although the SDR processors differ a lot in the architecture style, they are fundamentally programmable processors providing DLP, ILP and TLP. This requires algorithms that are specifically designed to exploit parallelism and scalability. Many wireless standards have been demonstrated with SDR implementation. However, the power consumption of these implementations is still relatively high as compared to ASIC-based solutions. Therefore, there is a clear need of designing MIMO detection algorithms that can be implemented on SDR processors while providing a near-optimal BER performance, high throughput and high area/energy efficiency comparable to ASIC/FPGA design.

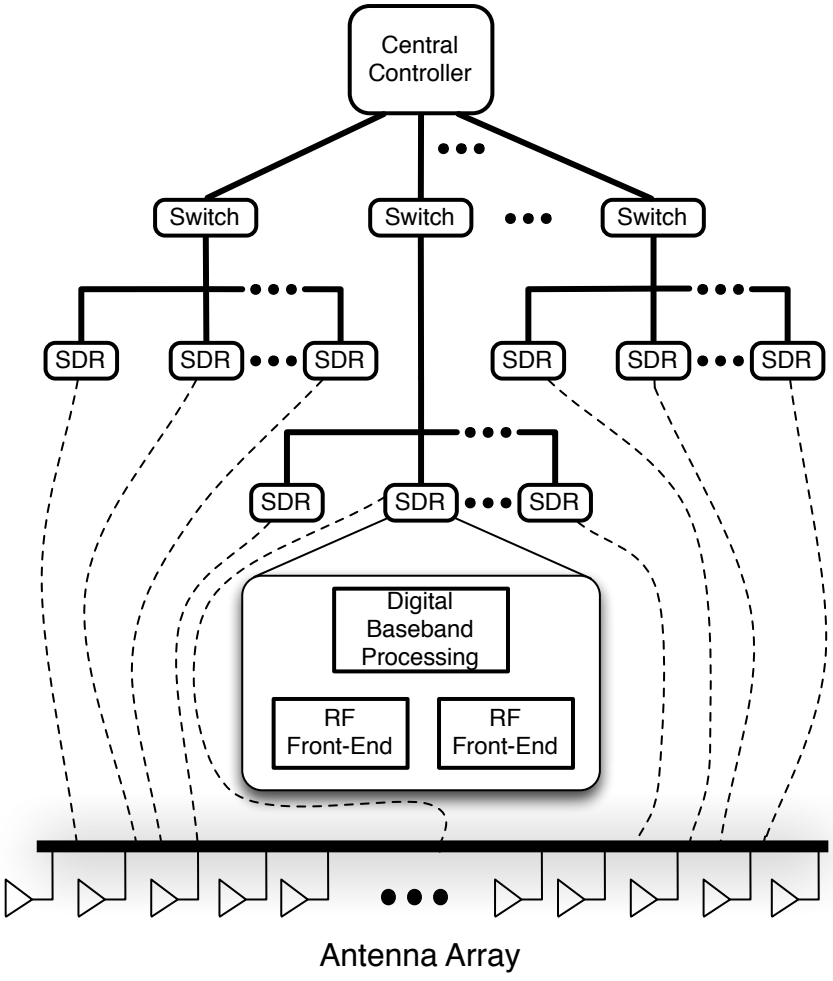


Figure 3.2: Hierarchical Overview of the LuMaMi testbed BS.

3.2.3 FPGA-based LuMaMi testbed

When exploring real-life opportunities and limitations of massive MIMO, at an early development stage, it is beneficial to have a high degree of freedom to make adjustments. A massive MIMO FPGA array system can provide such a freedom, while processing performance is high enough for real-time operation. Plenty of different architecture choices may be made. Here, a tree-architecture approach will be discussed.

Generic star architecture

A generic star-architecture may be employed as seen in Figure 3.2. A central controller, responsible for upper-layer protocol implementation, link evaluation, radio interfacing, bit-file deployment builds the starting point connecting to switches on the leafs. Through these switches the central controller sources and sinks user data, e.g. HD video stream and performs link quality evaluation using metrics like BER, EVM etc.

The switches on the other hand connect to SDRs routing data between central controlling unit and SDRs using, e.g. DMA transfers. Instead of applying several switches as shown in Figure 3.2, a suitable big switch connecting to all SDRs may be used. However, both setup should, additionally to streaming between central unit and SDR, support peer-to-peer streaming between different SDRs to allow flexibility in distribution of processing blocks.

Table 3.1: Components of LuMaMi

Generic	Component	Amount
Central Controller	NI PXIe-8135	1
Switch	NI 1085 PXIe	4
SDR	NI USRP-RIO 2943R	50
Antenna Array	In-house developed	1

Each SDR on the BS serves either one or several antennas depending on the actual number of RF chains integrated. Special attention has to be paid to (1) streaming rate limits between all components, (2) maximum parallel input/output streaming links in the devices, especially the SDRs and (3) maximum aggregated data rate inside the switches.

Hierarchical structure of LuMaMi testbed BS

Following aforementioned tree structure approach, the LuMaMi (Lund University Massive MIMO) testbed [21] was developed in cooperation with National Instruments. Mapping of the generic entities to actual implemented hardware components is shown in Table 3.1. As central controller implementing the interface to the BS, an NI PXIe-8135 without real-time calculation capability, running 64-bit Windows 7 was implemented. Three switches, NI 1085 PXIe, each connecting up to 18 USRP-RIO 2943R with the central controlling unit allowing overall support of 50 SDRs are incorporated. Each USRP-RIO SDR provides two RF-chains serving up to 40 MHz bandwidth and tuneable center frequency from 1.2 GHz to 6 GHz while delivering maximum transmit power of 15 dBm. This setup supports 100 antenna elements at the BS. An in-house developed antenna array, supporting different antenna array arrangements with 160 dual-polarized elements connects to the BS to wirelessly transmit and receive data.

Inherent property of the tree architecture is the distributed processing requirement since no centralized processing node is available. To remedy this requirement and to add higher flexibility, the central controlling unit itself was placed in a switch which concedes addition of parallel co-processing FPGAs at the central architecture node.

All interconnections between components are limited by different maximum streaming rates. Interconnection from central controller to switches allows 3.2 Gbps bidirectional traffic whereas the SDRs connect with up to 800 Mbps shared among 13 available DMA channels. Each slot in the switches handles up to 3.2 Gbps bidirectional traffic with aggregated total traffic of 32 Gbps. Additional co-processor FPGA units, implemented by NI PXIe-7976R, handle up to 3.2 Gbps bidirectional streaming rate on up to 32 channels.

Figure 3.3 shows the assembled LuMaMi testbed BS.

3.3 Flexibility requirements

In this section we investigate the requirements that are most critical for the selection and design of a MaMi digital baseband platform. Especially, the need for flexibility is dominant in the decision process.

Figure 3.3: Assembled LuMaMi Testbed BS.

3.3.1 Prototyping requirements

For prototyping, the needs are clearly specific as compared to final product design. The dominant focus is on flexibility, in order to be able to efficiently test multiple solutions and optimize them. Time to market is also an asset when using flexible platforms, i.e., building on software development instead of hardware development in order to speed up the exploration phase.

The prototyping platform needs to be built from existing hardware components. Dedicated hardware components might be designed as part of prototyping some components for key functionality, but such components cannot be used when building the exploration platform. This is even more the case when exploring a system as complex as a MaMi base station, where many high-speed data streams across many antennas have to be processed in real-time, and platform assembly from proven state-of-the-art components is already challenging enough.

As a consequence, as illustrated in Section 3.2.3, state-of-the-art SDR platforms building mostly on FPGAs as well as possibly DSP co-processors have to be used and assembled into a demonstrator platform, in order to maximize the flexibility and exploration speed required in such a study.

3.3.2 Product requirements

The hardware selection for a final product is expected to sacrifice a bit in flexibility in order to gain in cost and power efficiency of the final design. The difficult part is to evaluate how much

flexibility should still be present in such a product.

Run-time flexibility is needed in order to address multiple expected scenarios, for example changes in number of users, system load, signal constellations, user scheduling, etc. One approach is to design the system for the worst-case, and make it possible to down-scale when the peak performance is not needed. If this is supported by a flexible platform, it can lead to power savings when not operating under the peak conditions, while a static design would unnecessarily burn power by always operating at maximum complexity. Another approach is to target a performance which is not worst-case along all the dimensions. For example, the system might not operate simultaneously at maximum number of antennas and users, at full load, with the highest spectral efficiency and the largest mobility, but reconfigure depending on which one is most demanding. This approach is often acceptable for a user device, which may either provide the fastest throughput or support high-mobility, depending on conditions, while the full worst-case would lead to overdesign. However, it is less likely to be an acceptable trade-off for a base station which is expected to provide a specified level of performance on all the relevant metrics, whatever the operating conditions.

Next to run-time flexibility, reconfigurability can also be relevant in order to adapt to modifications in the communication standard and air interface definition. Especially for a technique as recent as massive MIMO, for which no agreed communication standard exists yet, early deployment will need to build on flexible hardware in order to support the expected modifications to come. On the other hand, keeping too much flexibility in the system would prevent the design from being sufficiently cheap, power-efficient, and powerful enough in terms of required processing.

For those reasons, we expect the first implementations of baseband signal processing for MaMi to rely on SDRs or other reconfigurable platforms, combining specific accelerators for the most critical operations with more generic reconfigurable components such as FPGAs. The specific components performing those critical operations could be dedicated ASIPs where the functional scope is clear enough to allow for such a design. In the longer term, a second generation of base stations could benefit from the experience gained and target more custom hardware in order to further cut down on costs and power consumption.

3.3.3 Signal processing operations for the main air interfaces

In order to identify implementation bottlenecks and select hardware accordingly, we have to assess the expected functionality and quantify its complexity.

Three main air interfaces can be identified for massive MIMO. The first one is OFDM, where MaMi precoding/decoding is applied at subcarrier level. The second one is SC with FD precoding; it presents a number of similarities with the OFDM case, but requires a few more FFTs. The expected benefit is a lower PAPR of the transmitted signals. The third one is fully operated in time domain, building on time-reversal; its main drawback is that only MRT/MRC processing is possible, but it can provide a more efficient implementation for that specific processing type.

A number of functional operations have to be performed for any air interface, such as channel encoding/decoding, constellation mapping/demapping, up-/down-sampling and transmit/receive filtering, etc. Some other operations are only present in some of the air interfaces, such as FFT/IFFT for SC and OFDM solutions. Finally, some operations will have a different form depending on the selected air interface, such as channel estimation, precoding, and combining which could be implemented in time-domain or frequency-domain, possibly combined with PAPR reduction techniques. Compensation of some non-idealities from the analog

front-end may also be implemented either in time domain or in frequency domain.

In this project we will quantify the complexity of those functional blocks in order to determine the implementation bottlenecks and hence select hardware accordingly.

A last element concerns the decision between centralized processing and distributed processing, i.e., whether to use a large central processing unit or to perform most of the DSP operations at the level of the different antennas. This has implications on the overall platform design, already in the case of a co-located antenna array due to the overhead of interfacing the different components, but even much more in case of distributed antenna system, which is a possible architecture for MaMi systems. This constraint can impact the choice of an air interface and linear processing scheme, given that some processing schemes such as MRT/MRC are more easily implemented in a distributed way while the ZF or MMSE schemes require centralized processing.

Chapter 4

Algorithm/hardware mapping

One of the important steps in the process of realizing massive MIMO is to decide how algorithms, such as those discussed in Chapter 2, are best mapped onto hardware to strike a good balance between flexibility and complexity/energy efficiency. This is one of the core tasks of the MAMMOET project and much of the work is still in its infancy. The description below is therefore based on giving several different points of view, from estimation of power consumption in the digital processing of massive MIMO, a discussion on algorithm-platform co-design/co-optimization, and detailed experiences when implementing massive MIMO in the largely FPGA-based LuMaMi testbed to specific examples of hardware accelerators developed for massive MIMO precoding.

4.1 Algorithmic operations and digital power consumption

Given the fundamental differences between massive MIMO base stations and traditional (macro) base stations, estimating the power consumption of such a base station requires a significant modeling effort, taking into account the different components in order to assess the overall system power consumption and its energy efficiency. In [15] a global power modeling approach for a massive MIMO base stations was presented. The corresponding power modeling approach is described in the MAMMOET deliverable D1.1 on System scenarios and requirements specifications, as part of the presentation of the relevant metrics and scaling rules in order to evaluate massive MIMO systems. In this section of deliverable D3.1, we focus on the way to exploit the approach of [15] specifically towards modeling the the baseband signal processing complexity and power consumption.

Digital components can be modeled by counting the number of floating-point operations performed and translating the results into power consumption figures. This is one of the possible power modeling approaches for digital signal processing. This approach is often selected as it represents a fair trade-off between modeling effort and modeling accuracy, which can be estimated to be a factor 2× to 10× off from the actual power consumption, depending on the effort spent and the accuracy of the assumptions used. Simpler power modeling techniques would not provide any useful absolute results but only coarse relative comparisons. On the other hand, getting very accurate power modeling results, less than a factor 2x off from the actual power consumption, require much more modeling effort. For most complex systems, this is only possible by going through the complete design of the digital platform and in case of programmable hardware the complete mapping flow as well. This amount of effort is not

Table 4.1: Reference complexity of digital sub-components, per antenna and per user for 20 MHz and 6 bps/Hz (64-QAM, coding rate 1).

Subcomponent	Downlink [GOPS]	Uplink [GOPS]	Estimation [GOPS]
Filtering	6.7	6.7	6.7
Up/Down-sampling	2	2	2
FFT/IFFT	0.5	0.5	0.5
MIMO processing	.04	.04	0
Synchronization	0	2	0
Channel estimation	0	0	.01
OFDM Mod/Demod	1.3	2.7	2.7
Mapping/Demapping	1.3	2.7	2.7
Channel coding	1.3	8	0
Control	2.7	1	1
Network	8	5.3	0

feasible at the level of pre-design system and platform exploration.

Reference complexity values $C_{i,\text{ref}}$ are provided in Table 4.1. Those values estimate the number of billion complex floating-point arithmetic operations required per second for each specific digital signal processing block. The average power consumption will be obtained by weighted averaging between downlink, uplink and pilot transmission (channel estimation) phases, based on the frame definition. The values in the table have already been multiplied by an overhead factor in order to take into account the data transfers (memories and registers) which play a large role in the power consumption of digital systems. This factor has been selected at a value of 2.5, hence not only arithmetic operations are taken into account.

The reference scenario does not need to be representative of massive MIMO scenarios. For example, the reference value of 6 bps/Hz is most likely too large for massive MIMO systems, but the model was designed such that it can scale complexity and power consumption values according to the selected scenario, hence the outcome will correspond to the desired scenario.

Reference values for filtering, up/down-sampling, synchronization, modulation/demodulation, mapping/demapping and channel coding are taken as similar to small base stations in a reference scenario. The idea is that the performance expectations are similar for massive MIMO systems as for small cell base stations, while large base stations suffer from more overhead. Specific values are introduced for massive MIMO specific components, i.e., FFT/IFFT, MIMO precoding/combining and channel estimation. Those are estimated based on [67]. They are further scaled in order to match the reference scenario of Table 4.1, i.e., per-antenna and per-user numbers assuming 6 bps/Hz and 20 MHz. FFTs are used in all phases of the frame; precoding is used in downlink but also as detection matched filter in uplink. Finally, channel estimation is used in pilot transmission phase only. A MRT is assumed, which requires no additional computations besides estimating the channel, i.e., no need for matrix inversion.

Per antenna and per user, the reference complexity values for specific massive MIMO components are relatively small. However, the precoding and channel estimation terms scale up linearly with both the number of antennas and the number of users, while most other digital sub-components only scale with one of the two, as can be seen from Table 4.2. This makes the massive MIMO specific terms relatively more important when scaled up to a realistic scenario. This scaling with input parameters is applied as follows, denoting $\mathcal{I}_{\text{Baseband}}$ the set of sub-components i (filtering, up/down-sampling...), $\mathcal{X}_{\text{Baseband}}$ the set of scenario parameters x

having each a reference value x_{ref} and an actual value x_{act}, $s_{i,x}$ the scaling exponent for sub-component i with respect to parameter x, $P_{i,\text{ref}}$ the reference power of sub-component i and P_{Baseband} the computed total digital power:

$$P_{\text{Baseband}} = \sum_{i \in \mathcal{I}_{\text{Baseband}}} P_{i,\text{ref}} \prod_{x \in \mathcal{X}_{\text{Baseband}}} \left(\frac{x_{\text{act}}}{x_{\text{ref}}} \right)^{s_{i,x}} \tag{4.1}$$

Scenario parameters $x \in \mathcal{X}_{\text{Baseband}} = \{W, \text{SE}_\text{u}, M, \Upsilon, K, Q\}$ are the system bandwidth W, spectral efficiency per user SE_u (function of constellation order and coding rate), number of BS antennas M, system load Υ (in frequency-domain), number of users K, and the digital quantization Q (number of bits/word). In the reference case of Table 4.1, reference parameters are set to 20 MHz for bandwidth, 6 bps/Hz for spectral efficiency, 1 for BS antennas and users given that reference values are provided per antenna and per user, and 1 for the (full) load.

In order to obtain reference power values $P_{i,\text{ref}}$, The last step is to convert complexity numbers $C_{i,\text{ref}}$ into power consumption. The proposed approach is based on an average intrinsic efficiency η_{Baseband} in GOPS/W, i.e., $P_{i,\text{ref}} = C_{i,\text{ref}}/\eta_{\text{Baseband}}$. The reference conversion factor is set to $\eta_{\text{Baseband}} = 8$ GOPS/W for the year 2010, assuming dedicated hardware components which are more efficient than general-purpose ones. It assumes a reference quantization parameter of 24 bits/word. This factor increases when a reduced accuracy is required, which is the case for massive MIMO: the quantization parameter Q actually improves the intrinsic hardware efficiency factor η_{Baseband} thanks to simpler operations, while the other scenario parameters such as number of users or antennas influence the number of GOPS to perform. Next to reduced quantization, extrapolation to future technology generations is a second way to improve the hardware efficiency. It is applied by considering a factor 2 improvement every 2 years thanks to silicon technology evolution, which is a conservative estimate taking into account Moores's law as well as the more and more complex technological challenges in order to keep the benefits from technology scaling.

A remaining essential digital scaling parameter is quantization. Indeed, the selected resolution of digital computations impacts their power consumption severely. Unlike the reference scenario of Table 4.1 and the related intrinsic efficiency η_{Baseband}, the average required resolution for a massive MIMO system is expected to be only 4 bits, in contrast with large and small cell base stations modeled to use an average of 24 and 16 bits, respectively. Massive MIMO systems tolerate such a low resolution because the many antennas average out the impairments caused by the limited accuracy of individual antenna chains. This low resolution results in a significant reduction in power consumption, thanks to the scaling with respect to quantization parameter.

Next to the digital signal processing as such, some additional digital functions are required in a complete system. They perform platform control (activating different parts of the system, managing data flow, ...), network processing (higher-layer protocols) and backhauling to the core network. Control and network processing are modeled similarly to the digital baseband components. The corresponding numbers are provided at the bottom of tables 4.1 and 4.2.

In future deliverables we will revisit the complexity tables and reference values selected in order to model the baseband power consumption in view of the selected system scenarios and functional algorithms within MAMMOET and provide a closer assessment of the expected digital power consumption in the project.

Table 4.2: Scaling exponents $s_{i,x}$ for digital sub-components, as function of the bandwidth (W), spectral efficiency per user (SE_u), number of antennas (M), system load (Υ), number of users and digital quantization resolution (Q).

Sub-component	Bandwidth	SE/user	BS antennas	Load	UEs	Quantization
Notation	W	SE_u	M	Υ	K	Q
Filtering	1	0	1	0	0	1.2
Up/Down-sampling	1	0	1	0	0	1.2
FFT/IFFT	1.2	0	1	0	0	1.2
MIMO precoding	1	0	1	1	1	1.2
Synchronization	0	0	1	0	0	1.2
Channel estimation	1	0	1	.5	1	1.2
OFDM Mod/Demod	1	0	1	.5	0	1.2
Mapping/Demapping	1	1.5	0	1	1	1.2
Channel coding	1	1	0	1	1	1.2
Control	0	0	.5	0	.2	.2
Network	1	1	0	1	0	0

4.2 Algorithm-platform co-design and co-optimization

The design of a wireless baseband platform starts from a set of specifications from the wireless standard. These specifications define certain performance objectives that have to be met in both functionality and implementation. In terms of functionality, the minimum and peak throughput requirement and the BER or PER set by the wireless standard are the most important ones. In terms of implementation, these specifications have to be met with the highest possible energy and chip area efficiency. Therefore, wireless baseband design poses a significantly difficult challenge, i.e., to find a solution that achieves low BER performance with low chip area and high energy efficiency while still providing flexibility and programmability. This is true for SDR platforms in general but also more specifically for dedicated wireless processors of ASIPs. In general, flexibility, low chip area and high energy efficiency are conflicting design objectives. To achieve the goal of meeting performance requirements under power and area constraints, we use a system level design approach considering both the algorithm and architecture. The main software/hardware optimizations applied in this approach are shown in Figure 4.1 and are detailed below.

Data and Instruction Level Parallelism

Although, different SDR baseband processors differ from each other in many aspects, they are essentially parallel programmable processors. The majority of the SDR baseband processors are Very Long Instruction Word (VLIW) processors that provide data level parallelism (DLP) with Single Instruction Multiple Data (SIMD) operations, instruction level parallelism (ILP) and task level parallelism (TLP). This multiple level parallelism can potentially provide high throughput with a moderate chip area and high energy efficiency. However, in practice most of the signal processing algorithms are designed keeping functionality aspects in mind while details on exploiting parallelism are often ignored. If an algorithm with an indeterministic data and control flow is implemented on a processor providing DLP/ILP, the hardware resources will be under utilized and the end results would be an inefficient implementation. Hence, both DLP/ILP parallelism have to be explicitly enabled at the first stage of the design flow, during algorithm design, to avoid problems in the subsequent design stages (Figure 4.1).

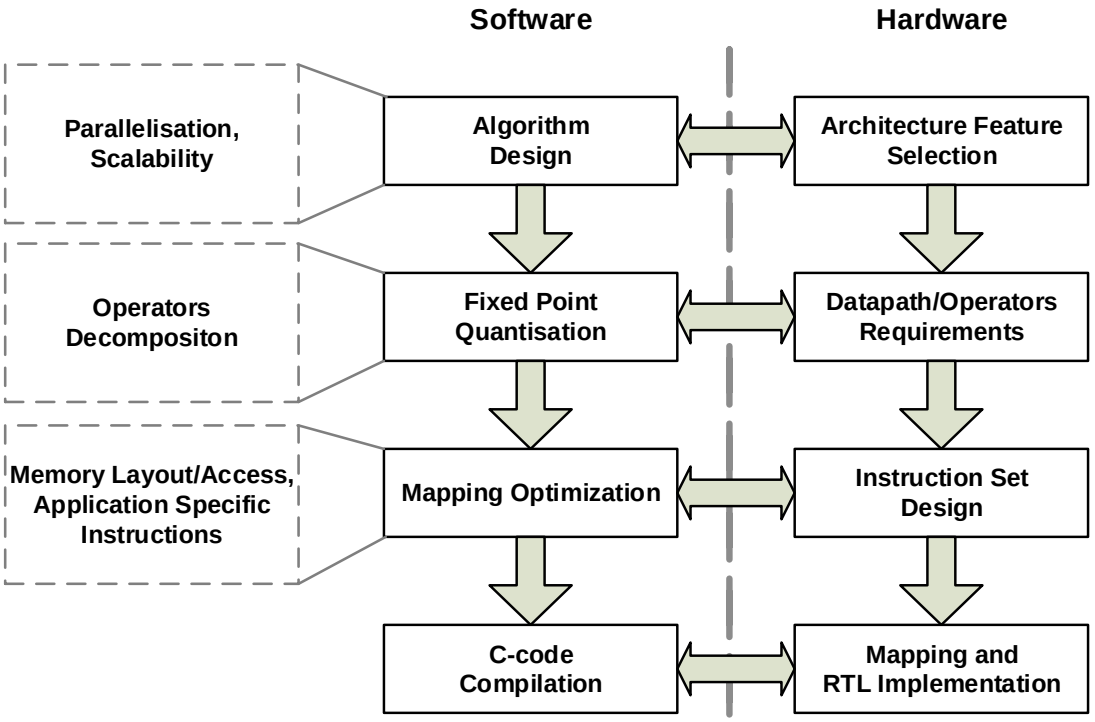


Figure 4.1: Design flow for algorithm and architecture co-design targeting SDR baseband solutions.

For instance, considering a VLIW processor with SIMD operations, MIMO detection can be implemented with several algorithms, providing different BER performance. The sphere detector is known to provide near-optimal performance, with a low memory requirement. However, it cannot be efficiently implemented on a parallel processor due to its indeterministic data-flow. In contrast, the K-best algorithm has a larger memory requirement, but thanks to its deterministic flow it can exploit parallelism which can lead to an energy efficient implementation. Algorithm design choices at the highest level can be used to determine the architecture features, such as, DLP (SIMD width), memory requirements, memory access, arithmetic operators, etc.

Run Time Scalability

SDR baseband processors are programmable such that the multiplexing of data path and memory is more flexible than in ASICs. Exploiting this flexibility to increase average energy efficiency requires scalable algorithms that can adapt to the varying wireless channel and the user requirements. For instance, when the SNR is high even a sub-optimal MIMO detector can provide a good enough BER performance, whereas at low SNR a near-optimal MIMO detector is required. Obviously, this impacts the required power for processing and hence energy efficiency. Such dynamic features can be exploited in the SDR baseband to reduce the energy efficiency gap to ASICs. We can do so by using software parameters such as a scalable threshold value to enable early termination of an algorithm, adjusting the number of iterations, or using exit conditions extracted from the wireless channel. Run-time scalable algorithms can operated in multiple modes providing a trade-off between BER performance, throughput and energy efficiency. To enable such scalability effectively, design considerations have to be made in both the algorithm

and the architecture.

Design Time Scalability

SDR baseband processors are usually template-based so that with the release of a new wireless standard the template can be reconfigured to support the new release. Design time scalability is enabled so that algorithms can be easily reconfigured, by changing a few parameters, to support various standards or modes within a standard and even different architecture instances. Generic baseband algorithms can be designed for implementation on programmable VLIW/SIMD processors, which can eventually reduce the design time cost. Common features such as the number of iterations and DLP can be kept scalable to provide design time reconfigurability.

Fixed Point Quantization

Fixed point quantization is another important aspect of the overall system design. Data of signal processing algorithms is often represented by fixed point integers in baseband platforms. Numerical stability and the word length requirements have to be particularly considered as this can influence both the functionality aspect as well as the power consumption and area of the design. A particular algorithm that might have a lower complexity in terms of the number of operations but be numerically unstable. For example, the ZF equalizer computation can be implemented with the direct inversion of the Gramian of the channel matrix, with a Gaussian elimination, or by first performing a QR decomposition of the channel matrix and then using back substitution to calculate the inverse. The Gaussian elimination requires the least operations but it is numerically unstable; this problem is strongly improved by including minor pivoting, but it makes the flow non-deterministic and hence less easy to parallelize. On the other hand, QR-based inversion offers the best numerical stability and has a smaller word length requirement. The choice of word length (or data type) for a particular signal can lead to potential opportunities of operator decomposition, which essentially determines the design area and power. For instance, if a signal can be represented by 4 bits instead of 16 bits, a multiplier can be replaced by simple bit-shift-add operations. Therefore, fixed point quantization is an important aspect considered during the algorithm design.

Application Specific Instructions

In many cases, baseband algorithms are designed to be implemented on a pre-designed baseband architecture. If the design is based on an architecture template tuning the architecture to support specific algorithms or a class of algorithms, it can potentially bring significant gains. This requires creating application specific instructions (ASIs). In order to achieve a high throughput with low chip area and high energy efficiency, ASIs should be based on low cost arithmetic operators. The design of an ASI is closely linked to the type of arithmetic operators, word length requirements and the structure of the algorithm. Decomposing complex arithmetic operations into simplified equivalent or sometimes approximate operations, can potentially bring a gain in energy efficiency at a low or negligible performance degradation cost. For instance, multiplication can be decomposed into shift-add operations which enable low area and energy efficient implementation. Generally, ASIs should be designed in such a way that they can be used for various computations in the same algorithm or even for multiple algorithms. For example, if division is followed by a rounding operation to a finite set of integers, then it can be implemented by bit-shift and comparison to the integer set. In this way both division and round operation can be avoided and only one ASI for the bit-shift and comparison would need to be

implemented. The choice of operators during algorithm design is very crucial to the ASIs. In order to deliver the maximum potential of ASIs, they need to be enabled during the algorithm design at the Matlab level and not only at the C code level just before mapping.

Memory Layout/Access

Another important aspect that is considered throughout this section is the deterministic data flow of the algorithms that significantly influence memory layout and access. For instance, an algorithm with a reduced number of memory accesses might not result in an efficient implementation if the memory elements are accessed in an un-predictable or irregular manner. In contrast, a regular data flow and deterministic memory access pattern might lead to an efficient implementation even with a higher number of accesses. These considerations about memory access are considered during the algorithm design because it is tightly coupled with the supported architecture features such as, SIMD, memory layout and address generation. The proposed algorithms in this thesis are explicitly optimized to have a deterministic and regular memory access.

4.3 Mapping of Massive MIMO System into the FPGA-based Platform

In this section, we show the example of mapping massive MIMO baseband processing to an FPGA-based platform, LuMaMi. We start with the complexity profile of the targeting system, which is an LTE-like OFDM based massive MIMO system. The analysis together with the knowledge on the platform features enable the optimization of algorithm mapping. Finally, we give the detailed mapping scheme and some initial implementation results.

4.3.1 Algorithm Profile and Complexity Assessment

Figure 4.2 shows the processing blocks for the developed prototype testbed LuMaMi. On the left side and right side, host processing as well as analog front-end can be seen, respectively.

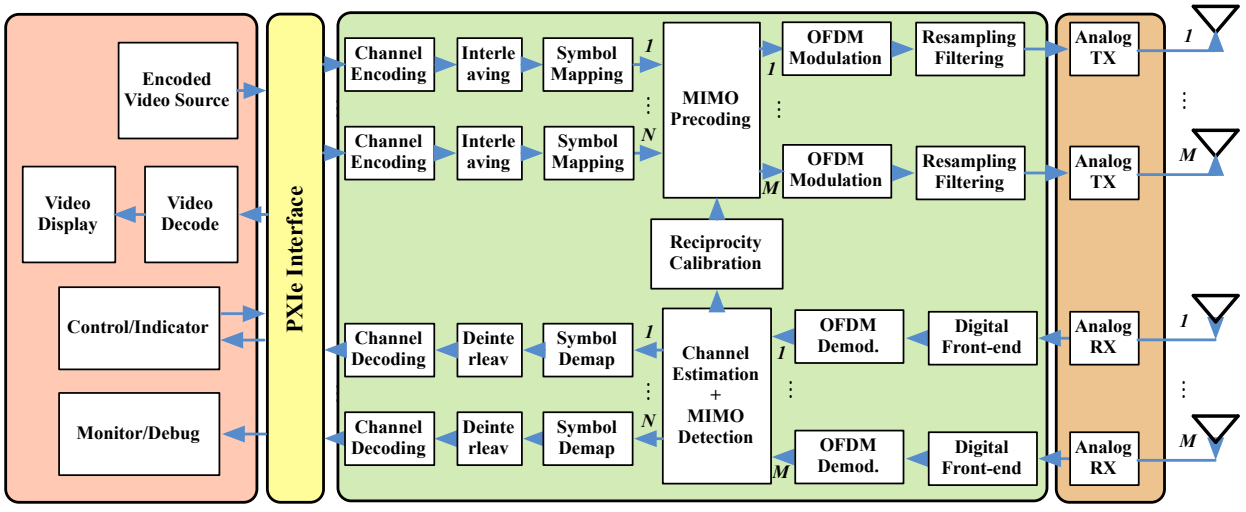


Figure 4.2: Block diagram for Massive MIMO baseband processing.

Resampling Filter and Digital Front-end are standard blocks as used in wireless system. Interleaving and Deinterleaving do not add computational complexity but rather latency since blocks have to be buffered to be able to shuffle the data. Moreover, Symbol Mapping has no real computational complexity.

To facilitate complexity analysis several simplifying assumptions were made:

- Only multiplications and divisions were considered as complex operations.
- Multiplication has same complexity as division.
- Additions have complexity *one*.
- Analysis was performed for an uncoded system, as the simplifying assumption would lead to a complexity of 1 when implying, e.g. Turbo coders.

While accurate complexity estimates are still to be performed for the massive MIMO baseband processing, rough estimates for important processng blocks are listed in Table 4.3. N_{FFT} and N_{SUB} are the length of the FFT and number of subcarriers for the OFDM, M the number of BS antennas and N is the number of users. Operations are divided in four different columns depending when and how often they have to be performed. First column are operations required after each uplink pilot symbol, i.e. every time channel estimation is performed. Second and third column give complexities of operations for each uplink and downlink symbol, respectively. The last column is devoted to operations which have to be done less frequently as for example the reciprocity calibration.

In the normal case, there is a strong correlation between channel attenuations at nearby OFDM subcarriers. This is exploited by transmitting pilots only on selected subcarriers for each user. Channel estimates between pilots are obtained through interpolation and initially we assume a simple first-order interpolation. MIMO precoding schemes, such as MRT and ZF, include element-wise multiplication of each channel matrix entry with the corresponding reciprocity calibration weights. Moreover, for MRC, multiplication with power scaling factors to compensate for large-scale fading was added. MRC scheme for detection also includes normalization for each of the N users. Complexity in uplink and downlink columns for detection and precoding are basically the matrix-vector product of channel and symbols required for each subcarrier. For complexity of matrix inversion, as required in ZF, an approximation using Neumann-Series was assumed [49].

4.3.2 Mapping to the FPGA Array

The LuMaMi testbed operates with many parameters, see Table 4.4, similar to those in LTE OFDMA. Using OFDM in TDD mode allows to separate overall bandwidth into sub-chunks, to efficiently lower processing requirements of single blocks and to distribute processing over the whole array.

Hence, the processing is distributed over six similar subsystems consisting of eight FPGAs leaving two single FPGAs as a mini-subsystem. The overall received 20 MHz bandwidth is split into eight chunks. Figure 4.3 shows the first four subsystems as applied in the LuMaMi testbed. The purple boxes on the left hand side mark FPGAs, the orange box in the middle is the central controller and the blue boxes on the right hand side are other subsystems. Each FPGA is connected to two antennas and implements OFDM RX/TX functionality.

Upper part of the subsystem handles uplink transmission by first combining samples received from all 16 subsystem antennas followed by splitting bandwidth into eight chunks. Seven of

Table 4.3: Complexities for massive MIMO BB processing

Block	After channel estimate	Uplink sym.	Downlink sym.	Seldom
OFDM				
Modulation			$\mathcal{O}(2 \cdot M \cdot N_{FFT} \cdot \log_2(N_{FFT}))$	
Demodulation		$\mathcal{O}(2 \cdot M \cdot N_{FFT} \cdot \log_2(N_{FFT}))$		
Reciprocity calibration				TO BE INVESTIGATED
Channel Estimation	$\mathcal{O}(M \cdot N_{SUB})$			
Channel Interpolation	$\mathcal{O}(2 \cdot M \cdot (N_{SUB} - \frac{N_{SUB}}{M}))$			
MIMO Detection				
Zero-Forcing	$\mathcal{O}(2 \cdot N_{SUB} \cdot (N^3 + N^2 \cdot M))$	$\mathcal{O}(N_{SUB} \cdot M \cdot N)$		
MRC	$\mathcal{O}(N_{SUB} \cdot N * M)$	$\mathcal{O}(N_{SUB} \cdot M \cdot N)$		
MIMO Precoding				
Zero-Forcing	$\mathcal{O}(2 \cdot N_{SUB} \cdot (N^3 + N^2 \cdot M) + N \cdot M)$		$\mathcal{O}(N_{SUB} \cdot M \cdot N)$	
MRT	$\mathcal{O}(3 \cdot N_{SUB} \cdot N \cdot M)$		$\mathcal{O}(N_{SUB} \cdot M \cdot N)$	

Table 4.4: High-level system parameters of LuMaMi testbed.

Parameter	Variable	Value
Bandwidth	W	20 MHz
Carrier frequency	f_c	3.7 GHz
Sampling rate	F_s	30.72 MS/s
FFT size	N_{FFT}	2048
# Used subcarriers	N_{used}	1200
Slot time	T_S	0.5 ms
Sub-Frame time	T_{sf}	1 ms
Frame time	T_f	10 ms
# UEs	K	10
# BS antennas	M	100

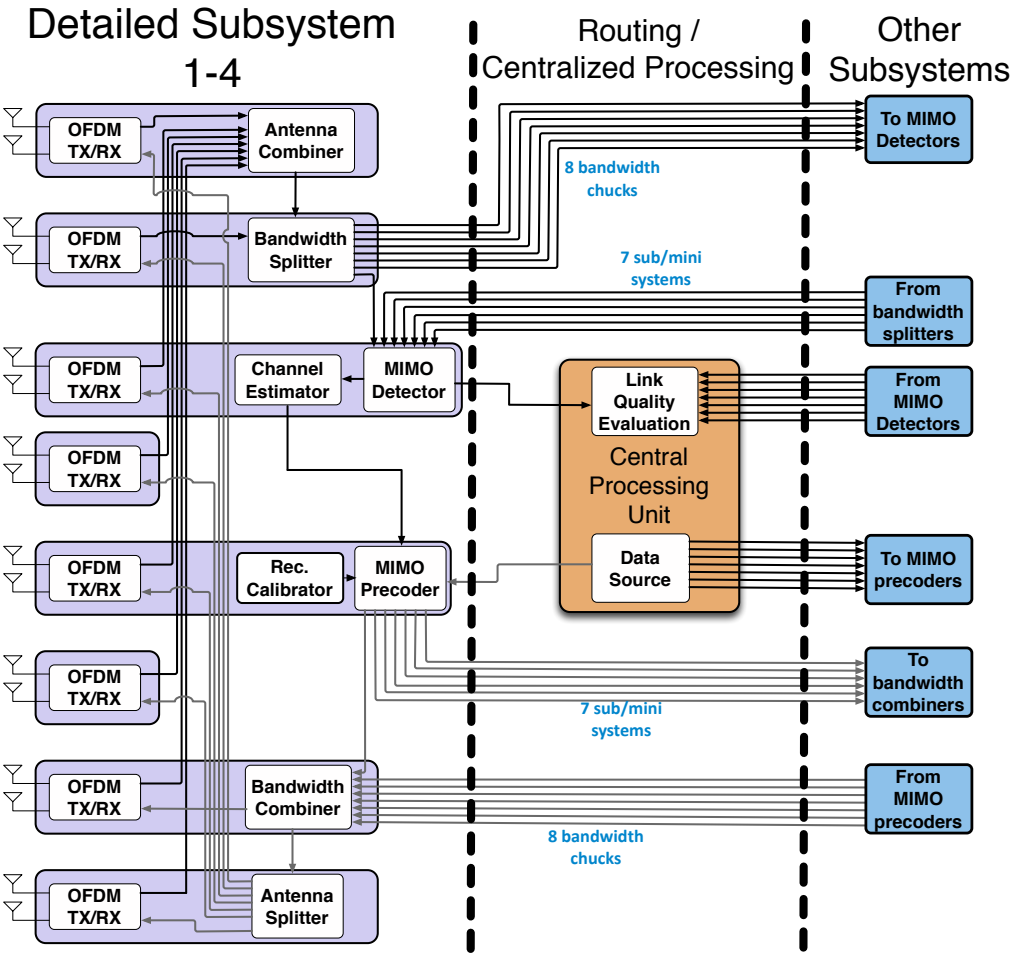


Figure 4.3: Subsystem 1-4 of the LuMaMi testbed BS.

these chunks are sent to other subsystems keeping the eighth chunk, assigned to the current subsystem for MIMO detection. MIMO detection block receives its bandwidth chunks from the other 7 subsystems and performs detection and channel estimation before sending decoded data to the central controlling unit.

Lower part of the subsystem is responsible for downlink transmission. MIMO precoder block

receives symbols to be send and uses estimated channel to precode symbols, also taking into account reciprocity calibration weights, to mitigate deviation due to non-reciprocal behavior of hardware at the RF chains. Precoded symbols are sent to bandwidth combiner of the current subsystem as well as the seven other subsystems. Bandwidth combiner receives bandwidth chunk from its own precoder and precoders of the seven other subsystem to build a whole 20 MHz bandwidth signal and forwards it to antenna splitter. Antenna splitter distributes the signal to be sent to all FPGAs which perform OFDM TX functionality to trigger antennas.

Since the BS consists of only six subsystems plus a mini-subsystem, however, has to handle eight different bandwidth chunks, two subsystems have to implement additional MIMO detector, MIMO precoder and channel estimator functionality. These additional function blocks are added in the fifth and sixth subsystem as shown in Fig. 4.4. Supplemental hardware blocks

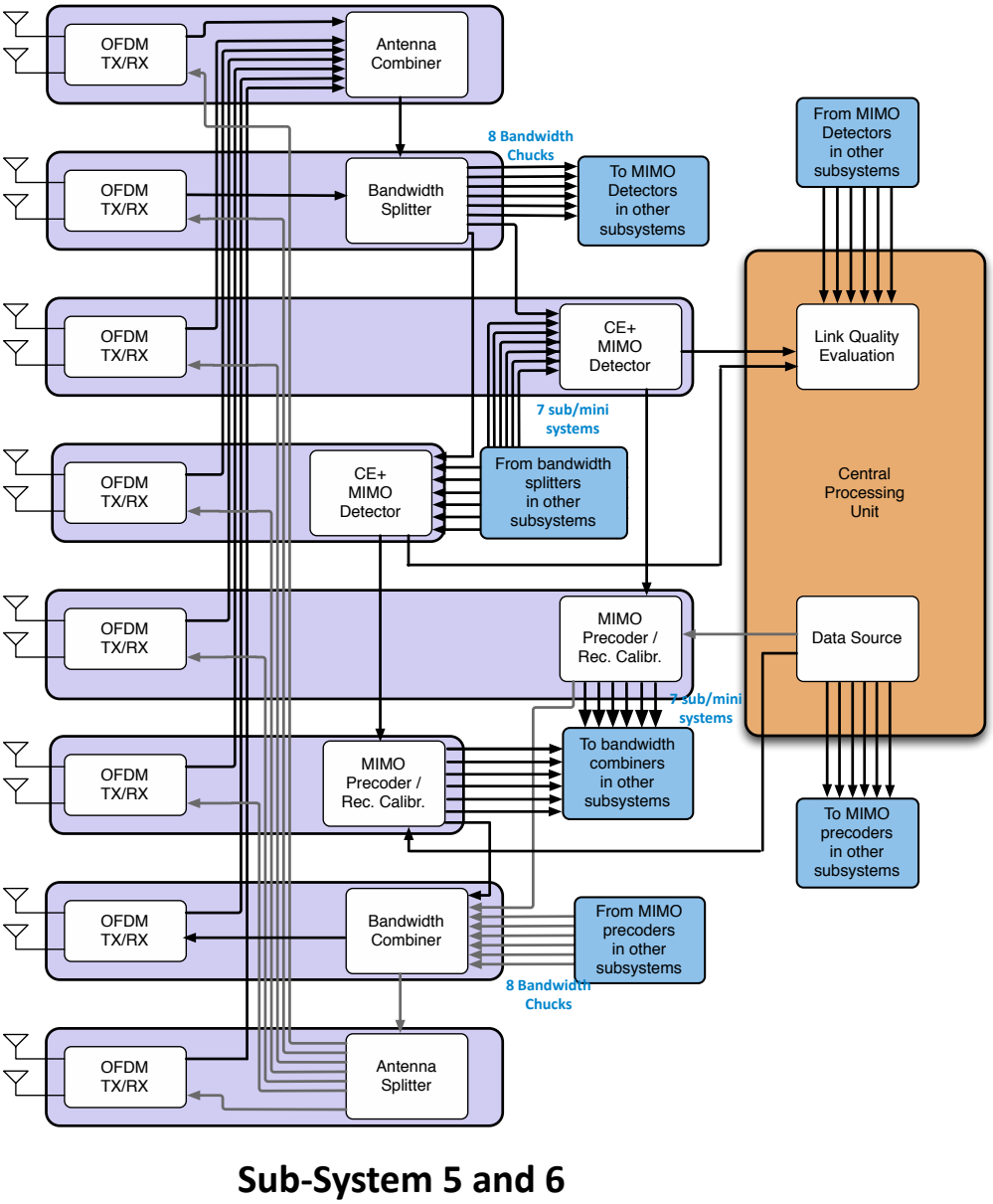


Figure 4.4: Subsystem 5-6 of the LuMaMi testbed BS.

in FPGA four and six integrate a second detection and precoding chain in these subsystem leading to an overall processing of eight bandwidth chunks in 6 subsystems.

The presented six subsystems occupy 48 FPGAs leaving two FPGAs arranged as a mini-subsystem as shown in Figure 4.5. The mini-subsystems handles only four antennas employing

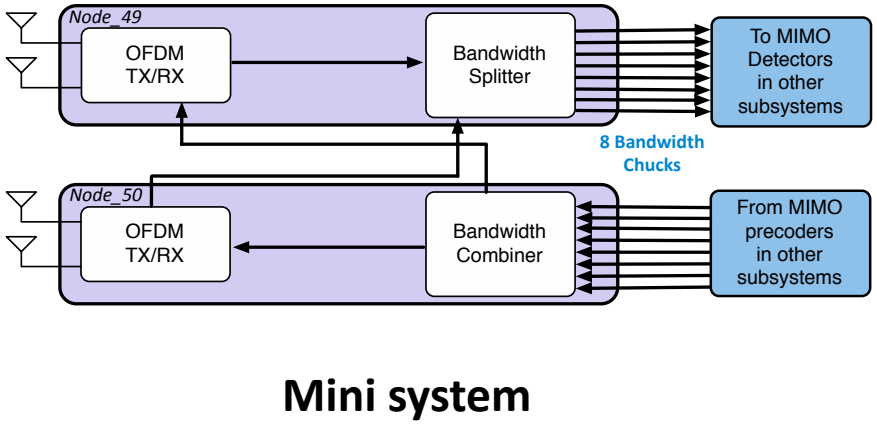


Figure 4.5: Node 49 and 50 of the LuMaMi testbed BS.

a bandwidth splitter on one FPGA and a bandwidth combiner on the other transmitting and receiving samples to or from other MIMO detectors/precoders in other subsystems. Bandwidth splitter and bandwidth combiner also include antenna combiner and antenna splitter for the four antennas in this subsystem.

A massive MIMO BS requires time synchronization and phase coherency between the RF chains. This is achieved using a reference clock and timing/trigger distribution network as shown in Fig. 4.6. The network consists of eight OctoClock modules in a tree structure with

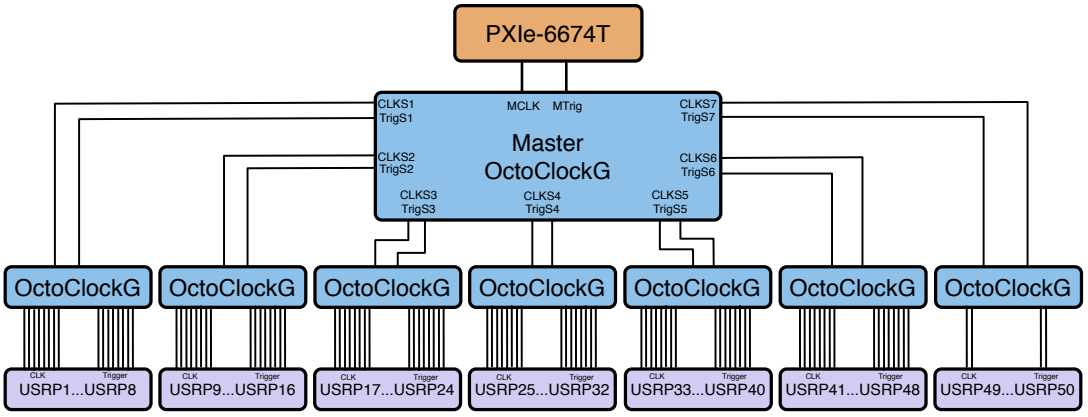


Figure 4.6: Clock distribution in the LuMaMi testbed BS.

a master OctoClock feeding seven secondary OctoClocks. Low skew buffering circuits and matched-length transmission cables ensure that there is low skew between the reference clock input at each SDR. The source clock for the system is an oven-controlled crystal oscillator within an NI PXIe-6674T timing module. Triggering is achieved by instigating a start pulse within the Master SDR via a software trigger. This trigger is then output from an output port on the master and input to the NI PXIe-6674T, which conditions and amplifies the trigger. The trigger is then propagated to the master OctoClock and distributed down the tree to each SDR in the system (including the master itself). This signal sets the reference clock edge to use for start of acquisition for the TX and RX within each channel. Initial results show that reference clock skew is within 100 ps and trigger skew is within 1.5 ns.

Looking at Fig. 4.3 sample rates for the different uplink blocks can be identified as shown in Table 4.5. One sample is the I- and Q-channel data of two antennas. The samples of the

Table 4.5: Sample rates at uplink blocks.

Block	**Input** [MS/s]	**Output** [MS/s]
OFDM RX	30.72	16.8
Antenna Combiner	7*16.8	117.72
Bandwidth Splitter	117.72 16.8	8*16.8
MIMO detector	7*16.8	

antennas arrive with a rate of 30.72 MS/s and reduce to 16.8 MS/s after CP and guard band removal, respectively. The antenna combiner combines the streams of seven incoming FPGAs to an output stream of 117.72 MS/s. Note that the samples from FPGA two are fed in to the bandwidth splitter locally, i.e. without going through the antenna combiner. Afterwards, the bandwidth splitter splits the overall bandwidth into eight streams with a rate of 16.8 MS/s each. MIMO detector block receives seven times the same rate at the input and feeds received data to the central controller. Sample rates for the downlink are similar. Initial results show that reference clock skew is within 100 ps and trigger skew is within 1.5 ns.

4.4 Hardware Accelerators

Massive MIMO inherently demands more signal processing due to the large number of antennas at the BS and the larger number of users. Furthermore, the downlink signal processing needs to performed faster than the channel coherence time, hence requiring a high throughput and more importantly low latency hardware solutions. Different approaches need to be employed to implement the signal processing hardware, ranging from highly reconfigurable processors to high throughput application specific accelerators. Considering the critical challenge to combine the high-throughput low-latency requirement with power efficiency, hardware accelerators are essential to implement some key signal processing blocks in Massive MIMO systems. Specified algorithm-hardware co-optimization can then be conducted to archive the design target. In the following section we will describe briefly the accelerator design for different processing algorithms which are likely candidates for the Massive MIMO system.

4.4.1 Zero-Forcing Precoding

As mentioned in Sec. 2.3.1 and Sec. 2.4.2, zero-forcing can be used in both precoding and detection in the context of Massive MIMO. In this section, we demonstrate how ZF processing can be efficiently implemented. ZF consists of the bulk of the processing complexity, which mainly requires handling large channel matrices. In-terms of latency precoding is more critical compared to detection, since detection can be performed by buffering the symbols. The ZF precoding/detection requires a pseudo-inverse of the channel matrix, which requires two matrix multiplications and one matrix inversion operation.

Matrix Multiplication

A traditional matrix multiplication has a cubic order of complexity, and there are other divide-and-conquer algorithms that have lower complexity; e.g., Strassen's - $\mathcal{O}(N^{2.8})$. However, these

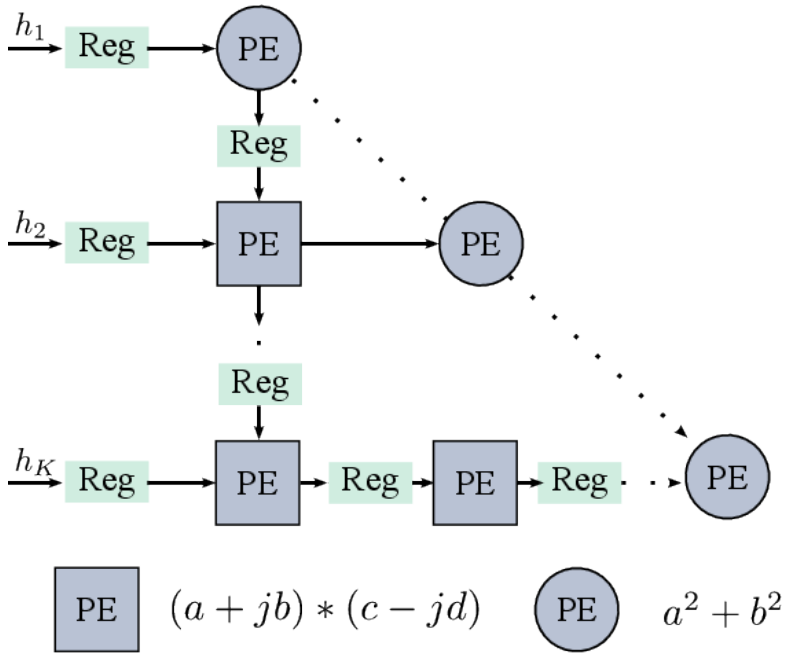


Figure 4.7: Systolic array to perform hermitian symmetric matrix multiplication.

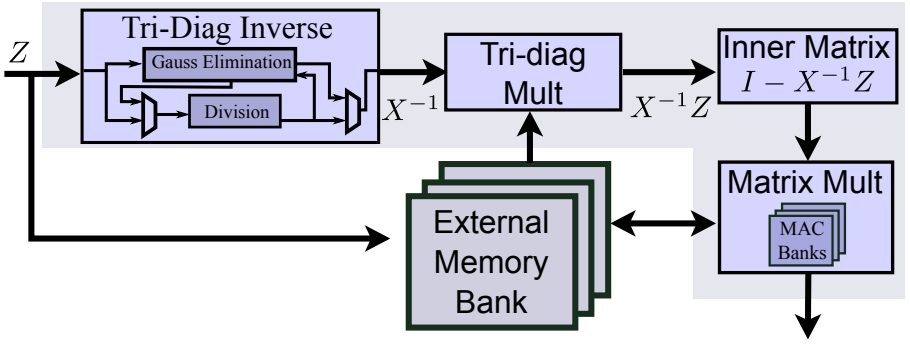


Figure 4.8: Neumann series based matrix inversion with tri-diagonal matrix as initial condition.

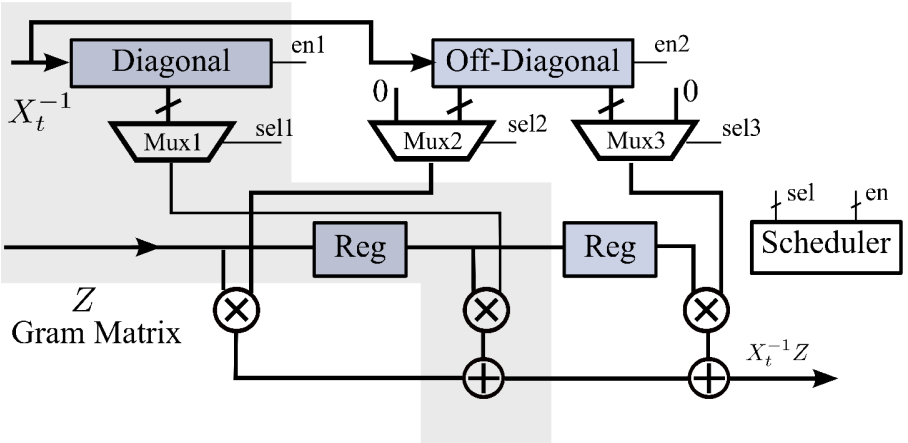


Figure 4.9: Circuit description of Tri-diagonal matrix multiplication.

Table 4.6: Hardware Details for matrix multiplication.

	Systolic Array	**MAC-unit Based**
Matrix Size	10×100	10×100
# of multipliers	200	40
# of adders	200	40
# Internal Accumulators	110	20
Memory Port	1	2
Latency (cycles)	120	1000

algorithms have a very high (routing) overhead in hardware, and has been shown to be efficient only for very large matrices even for processor based architectures. In case of pseudo-inverse, the matrix multiplication is to compute a Gram matrix (Hermitian symmetric matrix), hence the complexity can be reduced by half. This can be implemented with the traditional MAC-units and a controller to handle the data-flow. However, if one memory is used (*i.e.* $\mathbf{H}^H$ not stored explicitly) a slightly more complicated scheduling is required. Another solution to exploit this is to use high-throughput systolic arrays as shown in Fig. 4.7. The hardware cost for the multiplication unit is detailed in Table 4.6.

Matrix Inversion

Although matrix inversion in theory has same order of complexity as matrix multiplication, it is much more expensive to implement in hardware. The matrix inversion operation can be divided into three approaches: explicit computation, implicit computation, and polynomial expansion. In the first approach the matrix inversion is performed explicitly, whereas in the second approach the inversion is computed as the solution to e linear system of equations. The complexity has a crossover point after which performing explicit inversion would have lower complexity. However, the implicit inversion would have a lower initial latency, since a full matrix inversion is avoided. The third approach is based on rewriting the inversion as a matrix polynomial, which allows for complexity reductions if the order of the polynomial is truncated. In the following, brief details of these three approaches are provided.

Explicit Inversion Explicit inversions can be performed using various methods like QR-decomposition, LU-decomposition, Cholesky-decomposition, Gauss-elimination etc. Another approach for explicit inversion is the Neumann series [49], which uses the special channel properties arising in massive MIMO. This is a strong candidate for performing inversions since it requires only a series of matrix multiplications, which are highly parallelizable and efficient in hardware.

The top-level architecture for the Neumann series based precoder is shown in Figure 4.8. The Neumann series convergence is improved by using tri-diagonal matrix as initial condition. The inversion of tri-diagonal matrix is performed using Gauss Elimination. The tri-diagonal multiplication is implemented in a FIR filter like structure as shown in 4.9. The architecture has been implemented using RTL (register transfer level), and compatible with both ASIC and FPGA flow. The ASIC implementation results in 65nm CMOS technology is provided in Table 4.7 and Table 4.8.

Implicit Inversion Implicit inversion can be performed by using standard linear-solvers like conjugate-gradient, coordinate-descent etc. Apart from lower initial latency, these approaches

Table 4.7: Hardware cost breakup for Neumann series.

	Area [in mm^2]	**Latency [in cycles]**
Matrix Mult	0.11	200
Division	0.02	1
Tri-diag Inv	0.04	60
Tri-diag mult	0.02	3
Inner Mult	0.01	1

Table 4.8: Hardware Details for Neumann series based matrix inversion.

Matrix Size	$K \times K, K \in (2, 16)$
Technology	65 nm
Gate Count	$104K$
Max. Freq (MHz)	420
Throughput (Inversions per sec)	$0.051M$

also have a lower memory requirements.

Polynomial Expansion The inverse of any invertible $M \times M$ matrix $\mathbf{A}$ can be expressed as a matrix polynomial expansion of order M:

$$\mathbf{A}^{-1} = \mathbf{A}^{M} + c_{M-1}\mathbf{A}^{M-1} + \ldots + c_1\mathbf{A} + c_0\mathbf{I} \tag{4.2}$$

where $c_0, \ldots, c_{M-1}$ are the coefficients of the characteristic polynomial of the matrix. This is a consequence of the Cayley-Hamilton theorem, which also provides the exact expressions for the coefficients as sums and products of the eigenvalues of $\mathbf{A}$. Computing all the coefficients would be more complex than an explicit inversion of the matrix, but various polynomial approximations are available in the literature [19, 22, 39]. Only a handful of the terms in the polynomial matrix expansions are needed to obtain a good approximation of the matrix inversion, and the number of terms can be tuned to provide a tradeoff between inversion accuracy and end-performance. The key to reduced complexity is an iterative implementation where the inverse is never computed explicitly, but only as a inner products with different data symbol vectors [39].

4.4.2 Low complexity PAPR aware precoding

OFDM is known to suffer from high PAPR, and requires a linear PA with high dynamic range to avoid out-of-band components due to non-linearity and signal clipping. One way to reduce the PAPR is to apply the DTCE precoding algorithm described in Section 2.3.3, which is directly designed for low PAPR. Alternatively, one can modify conventional linear precoding schemes to reduce their PAPR.

A low-complexity approach to reduce the PAPR of linear precoding is described in [50]. It performs a simple clipping of the signals before they are sent to the antennas, which creates a deliberate clipping distortion. This distortion is mitigated by reserving a certain subset of the antennas to counteracting the clipping. To some extent it is similar to tone-reservation in OFDM systems, where subcarriers are reserved for mitigation of PAPR rather than data transmission. The difference, however, is that reserving tones on OFDM has a linear impact

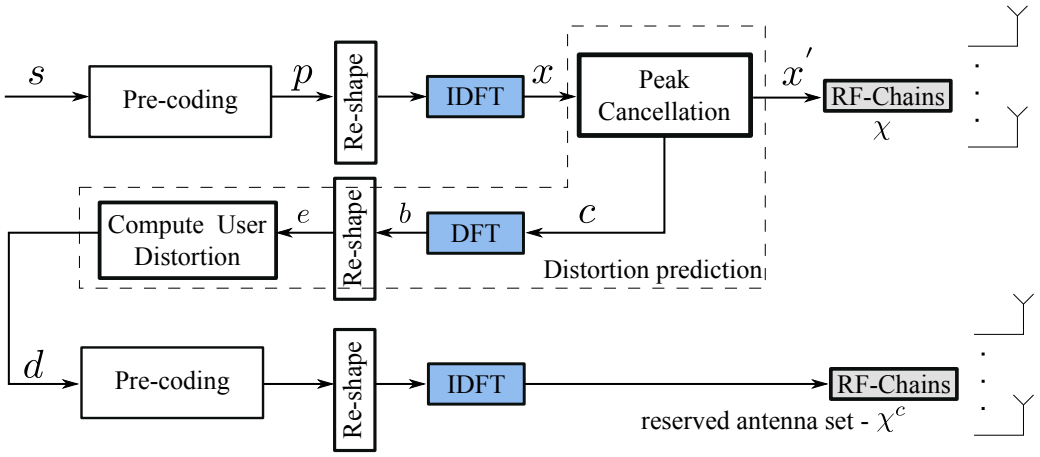


Figure 4.10: Top level description of PAPR aware precoding.

on the user rates, while reserving antennas in massive MIMO only has a logarithmic influence on the user rates. The PAPR aware precoding scheme from [50] is referred to as "antenna-reservation" and its top level architecture is depicted in Fig. 4.10. As an example, with 100 antennas at the BS, we can extend the range of output power by 4 dB, as compared to no PAPR reduction, with only 15% increase in precoding complexity.

Chapter 5

Summary of processing requirements

The initial assessment of baseband processing requirements for Massive MIMO presented in this deliverable points to several important aspects. Depending on the chosen transmission scheme, baseband processing requirements vary both in terms of the amount operations that need to be done and in terms of which hardware platforms/architechtures that are most efficient. Since it is an initial assessment, several of the results are incomplete and need further refinements.

5.1 Single-carrier vs. OFDM

The choice between single-carrier and OFDM has the potential to significantly influence both performance and baseband processing complexity. Investigations show that, in the massive MIMO regime, both single-carrier and OFDM based schemes have similar BER performance and there are only minor differences in their processing requirements. At this stage, no major advantage of one scheme over the other has been identified from this perspective.

An important non-technical aspect is that most recent standards are based on OFDM, such as LTE and WiFi. Massive MIMO is under discussion in several standardization bodies and massive MIMO research related to OFDM based systems may therefore have a larger short-term impact than research focused on single-carrier ones.

A third option would be to work in single-carrier but without frequency-domain processing. While this approach has limitations in order to implement some of the massive MIMO air interface options and algorithms, it can provide a low-performance and low-complexity solution by relying only on time-domain (time-reversal) processing in the MRT/MRC case, covering some of the massive MIMO scenarios.

5.2 Massive MIMO algorithms

A number of basic properties of massive MIMO influence the system complexity. The main one relates to the averaging of noise and other non-correlated impairments over the different antennas. This strongly relaxes the specifications that need to be achieved on each individual antenna. At the digital side, the most direct impact is on quantization, where the resolution can be strongly reduced. It also enables the use of simple algorithms such as MRT/MRC thanks to the coherent combination of the useful signal over the antennas while interference is non-coherently added. The asymmetry between the number of base station antennas and number of users makes it important to acquire CSI only in the uplink and exploit channel reciprocity to use this CSI also in the downlink. Since only the propagation channels are reciprocal, not

the transceivers, a dedicated non-reciprocity calibration procedure was presented.

While the dimension of massive MIMO matrices can be large and lead to heavy computations, many optimization steps are possible. For example, it was shown that reduced-accuracy inversion of the channel matrix is sufficient for good performance when using a ZF precoder, which strongly reduces complexity.

The exploitation of channel coherence in time and frequency also impacts all computations related to channel estimation and precoder computation, given that the same precoder can be used over a broader time-frequency range.

5.3 Linear vs. non-linear precoding

A classical argument for massive MIMO is that linear precoding performs almost as well as optimal precoding. This is attributed to the excess number of BS antennas, compared to the number of user terminals, and favourable propagation conditions. In theory this works well, resulting signal-power variations on the antennas in combination with amplifier non-linearities have the potential to reduce power efficiency, since large amplifier back-offs may be required to fulfill requirements on out-of-band emissions. Two conceptually different non-linear precoding schemes have been investigated, where signal-power variations are reduced. The most efficient scheme, from a power-variation point of view, is based on selecting constant envelope signals for all antennas in the discrete-time domain, through an optimization procedure. The computational complexity of this scheme has not been fully investigated at this point. The other scheme is similar to tone-reservation used to reduce power-variations in OFDM systems. Linearly precoded antenna signal are clipped to reduce power variations (in the discrete-time domain) and compensation signals are transmitted on a small subset of antennas reserved for that particular purpose. The second approach has very small complexity increase compared to the linear precoding on which it is based.

Presented results show that Massive MIMO opens up new opportunities for computationally efficient non-linear precoding techniques, allowing power amplifiers to work at small back-offs. While complexity assessments are rudimentary at this stage, initial results encourage further investigation in this direction. These investigations should also use realistic amplifier models. This will both improve realism/accuracy of evaluations as well as provide a means to find out which amplifier characteristics are the critical ones when designing for a massive MIMO application.

5.4 Computational platforms and architectures

Selecting a hardware platform for a given system is always a trade-off between different dimensions. The three main aspects are the system performance (throughput as well as signal quality), the system cost (area and power consumption) and the system flexibility (reconfigurability and time-to-market).

At one extreme of the hardware range, ASIC components offer the best trade-off between cost and performance, although the cost is only favourable in large volumes. The full ASIC approach lacks the necessary flexibility to be selected for massive MIMO: the field is not mature enough, various algorithmic solutions are still explored and run-time flexibility is expected depending on variations in channel propagation, users, traffic, scenarios, etc. Moreover, no standard is available yet, making a certain level of hardware flexibility even more necessary in the first implementations.

At the other extreme, fully-flexible general-purpose processors lack the throughput necessary to implement a complete massive MIMO system, or will only do so at a prohibitive cost and power consumption.

We propose the use of software-defined radios dedicated to the massive MIMO field. By this we mean platforms combining different basic components offering just the required amount of flexibility while still scoring good enough in the cost-performance plane. Such platforms can build on ASIP components, which thanks to the restriction of their flexibility to the dedicated field, enable to operate much closer to the intrinsic ASIC efficiency while keeping sufficient flexibility. They could be complemented by some more flexible components such as FPGAs, especially in the first prototypes. Once some of the key digital signal processing blocks are frozen in the future evolution of massive MIMO, those dominating the system area and power consumption could benefit from being redesigned as ASIC accelerators.

When using SDR platforms, the design of the architecture and the algorithms has to be performed in combination, co-design being the way to obtain the optimal performance of the system by adapting architecture to the algorithmic requirements without over-designing and similarly selecting algorithms that suit well to the implementation, e.g., regular flows and inherent parallelism.

5.5 Memory requirements and data shuffling

For most digital systems, memories typically count for half the system area and power consumption. In case of simple systems, this can be handled as a basic overhead factor. However, in the case of massive MIMO systems, the data storage and shuffling aspects become more crucial because of the large number of antenna chains having some physical distance between each other. As can be clearly seen from the LuMaMi testbed, the amount of data to shuffle over the system is huge and requires dedicated technical solutions. This is a strong incentive in order to distribute the processing as far as possible and reduce the amount of data transfers

Another element with huge impact on memories as well as computations is the required digital resolution, i.e., how many bits should we use to represent the different signals. Massive MIMO systems are expected to provide good performance even at a low resolution, thanks to the combination of all antennas which averages the quantization noise. Our first results tend to confirm this assumption but it will be explored more systematically in the future, in order to dimension more accurately the system.

5.6 Future assessment

A number of points will benefit from further study in the project.

5.6.1 Centralized versus distributed processing

In order to decide between centralized processing and distributed processing close to each antenna, we have to consider several elements. The main motivations for distributed processing is a reduced data communication with the central part of the base station and higher system scalability. The corresponding drawbacks could be limitations in the choice of algorithmic solutions, given that some of them are only applicable in a centralized way. This trade-off should be quantified in order to recommend one of the options.

5.6.2 Quantizing complexity-performance trade-offs

Some of those trade-offs have to be assessed numerically and not only as trends. Complexity and performance will be explored in the project by combining channel measurements for realistic assessment, performance simulations for key scenarios, testbed experiments, design of new algorithms and modeling of their complexity. A global power consumption model is also required.

List of Abbreviations

ACLR	Adjacent Channel Leakage Ratio
ASI	Application-Specific Instruction
ASIC	Application-Specific Integrated Circuit
ASIP	Application-Specific Instruction-set Processor
BER	Bit Error Rate
BS	Base Station
CB	Conjugate Beamforming
CP	Cyclic Prefix
CE	Cyclic Extension
CSI	Channel Statie Information
DLP	Data-Level Parallelism
DSP	Digital Signal Processor (or Digital Signal Processing)
DTCE	Discrete-Time Constant-Envelope
EC	European Commission
FD	Frequency-Domain
FDD	Frequency-Division Duplex
FDE	Frequency-Domain Equalizer
FIR	Finite Impulse Response
FPGA	Field-Programmable Gate Array
ILP	Instruction-Level Parallelism
KSP	Known-Symbol Padding
LMMSE	Linear Minimum Mean-Squared Error
MIMO	Multiple-Input Multiple-Output
MMSE	Minimum Mean-Squared Error
MRC	Maximum-Ratio Combining
MRT	Maximum-Ratio Transmission
MS	Mobile Station
MSE	Mean-Squared Error
NMSE	Normalized Mean-Square-Error
NRE	Non-Recurring Engineering

OFDM	Orthogonal frequency-division multiplexing
PA	Power Amplifier
QAM	Quadrature Amplitude Modulation
PAPR	Peak-to-Average Power Ratio
PER	Packet Error Rate
RF	Radio Frequency
RZF	Regularized Zero-Forcing
SC	Single-Carrier
SDR	Software-Defined Radio
SIMD	Single-Instruction Multiple-Data
SISO	Single-Input Single-Output
SNR	Signal-to-Noise Ratio
TD	Time-Domain
TDD	Time-Domain Duplex
TDE	Time-Domain Equalizer
TLP	Task-Level Parallelism
VLIW	Very Long Instruction Word
ZF	Zero Forcing
ZP	Zero Padding

Bibliography

[1] 3GPP-LTE:http://www.3gpp.org/technologies/keywords-acronyms/97-lte-advanced.

[2] Intel Website http://www.intel.com/support/wireless/sb/CS-032244.htm.

[3] 3GPP LTE-Advanced. Overview of 3GPP Release 10 v0.1.1 (2011-06). http://www.3gpp.org/ftp/Specs/html-info/FeatureList–Rel-10.htm, June 2011.

[4] Omer Anjum, Tapani Ahonen, Fabio Garzia, Jari Nurmi, Claudio Brunelli, and Heikki Berg. State of the art baseband DSP platforms for Software Defined Radio: A survey. *EURASIP Journal on Wireless Communications and Networking*, 2011(1):1–19, 2011.

[5] N. Benvenuto, R. Dinis, D. Falconer, and S. Tomasin. Single carrier modulation with nonlinear frequency domain equalization: An idea whose time has come—again. *Proceedings of the IEEE*, 98(1):69–96, 2010.

[6] Kees Van Berkel, Frank Heinle, Patrick PE Meuwissen, Kees Moerman, and Matthias Weiss. Vector processing as an enabler for software-defined radio in handheld devices. *EURASIP Journal on Advances in Signal Processing*, 2005(16):906408, 2005.

[7] M. Biguesh and A. B. Gershman. Downlink channel estimation in cellular systems with antenna arrays at base stations using channel probing with feedback. *EURASIP J. Appl. Signal Process.*, 2004(9):1330–1339, 2004.

[8] E. Björnson and E. Jorswieck. Optimal resource allocation in coordinated multi-cell systems. *Foundations and Trends in Communications and Information Theory*, 9(2-3):113–381, 2013.

[9] E. Björnson, E. G. Larsson, and M. Debbah. Optimizing multi-cell massive MIMO for spectral efficiency: How many users should be scheduled? In *Proc. IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014.

[10] Emil Björnson, Mats Bengtsson, and Björn Ottersten. Optimal multiuser transmit beamforming: A difficult problem with a simple solution structure. *IEEE Signal Processing Magazine*, 31(4):142–148, 2014.

[11] H. Boche and M. Schubert. A general duality theory for uplink and downlink beamforming. In *Proc. IEEE VTC-Fall*, pages 87–91, 2002.

[12] B. Bougard, B. De Sutter, S. Rabou, D. Novo, O. Allam, S. Dupont, and L. Van der Perre. A coarse-grained array based baseband processor for 100Mbps+ software defined radio. In *Design, Automation and Test in Europe, 2008. DATE '08*, pages 716–721, March 2008.

[13] S. Cherry. Edholm's law of bandwidth. *Spectrum, IEEE*, 41(7):58–60, July 2004.

[14] Elena Costa, Michele Midrio, and Silvano Pupolin. Impact of amplifier nonlinearities on OFDM transmission system performance. *IEEE Communications Letters*, 3(2):37–39, 1999.

[15] Claude Desset, Björn Debaillie, and Filip Louagie. Modeling the hardware power consumption of large scale antenna systems. In *Invited at IEEE OnlineGreenComm*, November 2014.

[16] V.H. Mac Donald. The cellular concept. *Bell System Technical Journal*, 58(15-41):113–381, 1979.

[17] PM Heysters. Coarse-grained reconfigurable computing for power aware applications. In *in Proceedings of the 2006 International Conference on Engineering of Reconfigurable Systems Algorithms*, June 2006.

[18] P.M. Heysters and G. J M Smit. Mapping of DSP algorithms on the MONTIUM architecture. In *Parallel and Distributed Processing Symposium, 2003. Proceedings. International*, pages 6 pp.–, April 2003.

[19] J. Hoydis and M. Debbah. Polynomial Expansion Detectors for Large Antenna Arrays with Antenna Correlation. 2011.

[20] IEEE 802.11ad. IEEE P802.11ad, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 3: Enhancements for Very High Throughput for Operation in the 60 GHz Band. http://www.ieee802.org/11/, December 2012.

[21] J. Vieira, S. Malkowsky, K. Nieman, Z. Miers, N. Kundargi, L. Liu, I. Wong and V. Öwall and O. Edfors and F. Tufvesson. A flexible 100-antenna testbed for Massive MIMO. In *IEEE GLOBECOM 2014 Workshop on Massive MIMO: from theory to practice, 2014-12-08*. IEEE, 2014.

[22] A. Kammoun, A. Müller, E. Björnson, and M. Debbah. Linear precoding based on polynomial expansion: Large-scale multi-cell MIMO systems. *IEEE J. Sel. Topics Signal Process.*, 8(5):861–875, 2014.

[23] S.M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.

[24] Per Landin, Magnus Isaksson, and Peter Händel. Comparison of evaluation criteria for power amplifier behavioral modeling. In *IEEE MTT-S International Microwave Symposium Digest*, pages 1441–1444, 2008.

[25] Erik G. Larsson, Ove Edfors, Fredrik Tufvesson, and Thomas L. Marzetta. Massive MIMO for next generation wireless systems. *IEEE Communications Magazine*, 52(2):186–195, 2014.

[26] Erik G. Larsson, Ove Edfors, Fredrik Tufvesson, and Thomas L. Marzetta. Massive MIMO for next generation wireless systems. *IEEE Communications Magazine*, 52(2):186–195, 2014.

[27] M. Li, S. Jin, and X. Gao. Spatial orthogonality-based pilot reuse for multi-cell massive MIMO transmission. In *Proc. WCSP*, 2013.

[28] M. Li, Y.-H. Nam, B.L. Ng, and J. Zhang. A non-asymptotic throughput for massive MIMO cellular uplink with pilot reuse. In *Proc. IEEE Globecom*, 2012.

[29] Min Li, A Amin, R. Torrea, U. Ahmad, R. Appeltans, A Dejonghe, and L. Van Der Perre. Processor based 20Mhz 4×4 Cat-5 LTE MIMO receiver with advanced detectors. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 2669–2673, May 2013.

[30] Yuan Lin, Hyunseok Lee, M. Woh, Y. Harel, S. Mahlke, T. Mudge, C. Chakrabarti, and K. Flautner. SODA: a high-performance DSP architecture for software-defined radio. *Micro, IEEE*, 27(1):114–123, Jan 2007.

[31] D. Liu, A. Nilsson, E. Tell, Di Wu, and J. Eilert. Bridging dream and reality: Programmable baseband processors for software-defined radio. *Communications Magazine, IEEE*, 47(9):134–140, September 2009.

[32] D. Liu, A Nilsson, E. Tell, Di Wu, and J. Eilert. Bridging dream and reality: Programmable baseband processors for software-defined radio. *Communications Magazine, IEEE*, 47(9):134–140, September 2009.

[33] Keith Mallinson. The 2020 vision for LTE. June 2012. `http://www.fiercewireless.com/europe/story/mallinson-2020-vision-lte/`.

[34] Thomas L. Marzetta. Noncooperative cellular wireless with unlimited numbers of base station antennas. *IEEE Transactions on Wireless Communications*, 9(11):3590–3600, 2010.

[35] J. Mitola. The software radio architecture. *Communications Magazine, IEEE*, 33(5):26–38, May 1995.

[36] Saif Mohammed and Erik G. Larsson. Single-user beamforming in large-scale MISO systems with per-antenna constant-envelope constraints: The doughnut channel. *IEEE Transactions on Wireless Communications*, 11(11):3992–4005, 2012.

[37] Saif Mohammed and Erik G. Larsson. Constant-envelope multi-user precoding for frequency-selective massive MIMO systems. *IEEE Wireless Communications Letters*, 2(5):547–550, 2013.

[38] Saif Mohammed and Erik G. Larsson. Per-antenna constant envelope precoding for large multi-user MIMO systems. *IEEE Transactions on Communications*, 61(3):1059–1071, 2013.

[39] A. Müller, A. Kammoun, E. Björnson, and M. Debbah. Efficient linear precoding for massive MIMO systems using truncated polynomial expansion. In *Proc. IEEE SAM*, 2014.

[40] R. Müller, M. Vehkaperä, and L. Cottatellucci. Blind pilot decontamination. In *Proc. ITG Workshop on Smart Antennas (WSA)*, 2013.

[41] B. Muquet, Zhengdao Wang, G.B. Giannakis, M. de Courville, and P. Duhamel. Cyclic prefixing or zero padding for wireless multicarrier transmissions? *Communications, IEEE Transactions on*, 50(12):2136–2148, Dec 2002.

[42] A Niktash, H.T. Parizi, and N. Bagherzadeh. Application of a heterogeneous reconfigurable architecture to OFDM wireless systems. In *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pages 2586–2589, May 2007.

[43] B. Noethen, O. Arnold, E. Perez Adeva, T. Seifert, E. Fischer, S. Kunze, E. Matus, G. Fettweis, H. Eisenreich, G. Ellguth, S. Hartmann, S. Hoppner, S. Schiefer, J.-U. Schlusler, S. Scholze, D. Walter, and R. Schuffny. 10.7 A 105GOPS $36mm^2$ heterogeneous SDR MPSoC with energy-aware dynamic scheduling and iterative detection-decoding for 4G in 65nm CMOS. In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, pages 188–189, Feb 2014.

[44] Hideki Ochiai. An analysis of band-limited communication systems from amplifier efficiency and distortion perspective. *IEEE Transactions on Communications*, 61(4):1460–1472, 2013.

[45] K. Pahlavan and P. Krishnamurthy. *Principles of Wireless Networks: A Unified Approach*. Prentice Hall, 2002.

[46] O. Paker, K. Van Berkel, and K. Moerman. Hardware and software implementations of an MMSE equalizer for MIMO-OFDM based WLAN. In *Signal Processing Systems Design and Implementation, 2005. IEEE Workshop on*, pages 1–6, Nov 2005.

[47] Chang Soon Park and Kwang Bok Lee. Transmit power allocation for ber performance improvement in multicarrier systems. *Communications, IEEE Transactions on*, 52(10):1658–1663, 2004.

[48] Antonios Pitarokoilis, Saif Khan Mohammed, and Erik G. Larsson. On the optimality of single-carrier transmission in large-scale antenna systems. *IEEE Wireless Communications Letters*, 1(4):276–279, 2012.

[49] H. Prabhu, O. Edfors, J. Rodrigues, Liang Liu, and F. Rusek. Hardware efficient approximative matrix inversion for linear pre-coding in massive mimo. In *Circuits and Systems (ISCAS), 2014 IEEE International Symposium on*, pages 1700–1703, June 2014.

[50] H. Prabhu, O. Edfors, J. Rodrigues, Liang Liu, and F. Rusek. A low-complex peak-to-average power reduction scheme for ofdm based massive mimo systems. In *Communications, Control and Signal Processing (ISCCSP), 2014 6th International Symposium on*, pages 114–117, May 2014.

[51] Frederick H. Raab, Peter Asbeck, Steve Cripps, Peter B. Kenington, Zoya B. Popovic, Nick Pothecary, John F. Sevic, and Nathan O. Sokal. Power amplifiers and transmitters for RF and microwave. *IEEE Transactions on Microwave Theory and Techniques*, 50(3):814–826, 2002.

[52] W. Raab, J. Berthold, U. Hachmann, D. Langen, M. Schreiner, H. Eisenreich, J.-U. Schluessler, and G. Ellguth. Low power design of the X-GOLD; SDR 20 baseband processor. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pages 792–793, March 2010.

[53] U. Ramacher. Software-Defined Radio prospects for multistandard mobile phones. *Computer*, 40(10):62 –69, oct. 2007.

[54] U. Ramacher, W. Raab, U. Hachmann, D. Langen, J. Berthold, R. Kramer, A Schackow, C. Grassmann, M. Sauermann, P. Szreder, F. Capar, G. Obradovic, W. Xu, N. Bruls, Kang Lee, E. Weber, R. Kuhn, and J. Harrington. Architecture and implementation of a software-defined radio baseband processor. In *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, pages 2193–2196, May 2011.

[55] D. Raychaudhuri and Narayan B. Mandayam. Frontiers of wireless and mobile communications. *Proceedings of the IEEE*, 100(4):824–840, April 2012.

[56] R. Rogalin, O. Bursalioglu, H. Papadopoulos, G. Caire, A. Molisch, A. Michaloliakos, V. Balan, and K. Psounis. Scalable synchronization and reciprocity calibration for distributed multiuser MIMO. *submitted to Wireless Communications, IEEE Transactions on*, PP(99):1–17, 2014.

[57] C. Rowen, P. Nuth, and S. Fiske. A DSP architecture optimized for wireless baseband. In *System-on-Chip, 2009. SOC 2009. International Symposium on*, pages 151–156, Oct 2009.

[58] Clayton Shepard, Hang Yu, Narendra Anand, Erran Li, Thomas Marzetta, Richard Yang, and Lin Zhong. Argos: Practical many-antenna base stations. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 53–64, New York, NY, USA, 2012. ACM.

[59] Standardization Administration of the People's Republic of China. Chinese National Standard GB 20600-2006: Framing Structure, Channel Coding and Modulation for Digital Television Terrestrial Broadcasting System. http://www.codeofchina.com/gb/communications/201105/25-4952.html, Aug. 2006.

[60] Zhenyu Tu, Meng Yu, D. Iancu, M. Moudgill, and J. Glossner. On the performance of 3GPP LTE baseband using SB3500. In *System-on-Chip, 2009. SOC 2009. International Symposium on*, pages 138–142, Oct 2009.

[61] Allert van Zelst and Tim C. W. Schenk. Implementation of a MIMO OFDM-based wireless LAN system. *IEEE Transactions on Signal Processing*, 52(2):483–494, 2004.

[62] Joao Vieira, Fredrik Rusek, and Fredrik Tufvesson. Reciprocity calibration methods for massive MIMO based on antenna coupling. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, Dec 2014.

[63] I. Viering, H. Hofstetter, and W. Utschick. Spatial long-term variations in urban, rural and indoor environments. In *COST273 5th Meeting, Lisbon, Portugal*, 2002.

[64] P. Viswanath and D.N.C. Tse. Sum capacity of the vector Gaussian broadcast channel and uplink-downlink duality. 49(8):1912–1921, 2003.

[65] Zhendao Wang and G.B. Giannakis. Wireless multicarrier communications. *Signal Processing Magazine, IEEE*, 17(3):29–48, May 2000.

[66] M. Woh, Yuan Lin, Sangwon Seo, S. Mahlke, T. Mudge, C. Chakrabarti, R. Bruce, D. Kershaw, A Reid, M. Wilder, and K. Flautner. From SODA to scotch: the evolution of a wireless baseband processor. In *Microarchitecture, 2008. MICRO-41. 2008 41st IEEE/ACM International Symposium on*, pages 152–163, Nov 2008.

[67] Hong Yang and Thomas L. Marzetta. Performance of conjugate and zero-forcing beamforming in large-scale antenna systems. *IEEE Journal on Selected Areas in Communications*, 31(2):172–179, 2013.

[68] H. Yin, D. Gesbert, M. Filippou, and Y. Liu. A coordinated approach to channel estimation in large-scale multiple-antenna systems. 31(2):264–273, Feb. 2013.