Realizations and Applications of MIMO Channel Decomposition Algorithms

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Jing Wang

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Professor Gerald E. Sobelman, Advisor

May, 2012

Acknowledgements

Foremost, I would like to express my deepest gratitude to my advisor, Professor Gerald E. Sobelman, for his guidance, support, caring and encouragement during my Ph.D. study at the University of Minnesota. I'm very grateful to have the opportunity and an excellent atmosphere for doing research. This thesis would not have been possible without him.

I would also like to thank Professor Keshab K. Parhi, Professor Marc Riedel and Professor Antonia Zhai for serving in my dissertation committee. Their reviews and suggestions helped improve the dissertation greatly.

It is my sincere pleasure to thank my friend and collaborator, Yi Jiang, whose work inspired this dissertation. The discussions with him have always been truly enlightening, which not only helped with this work, but will also continue to benefit me in the future.

Additionally, I would like to thank the Department of Electrical & Computer Engineering for the financial support for my Ph.D. study. The last five years here is an invaluable experience for me.

Finally, I want to give special thanks to my parents, for loving me and standing by me through the good times and bad.

Abstract

As one of the major breakthroughs in the physical layer evolution, multiple-input multiple-output (MIMO) is a promising technology in present and future wireless communication systems, boosting up the overall throughput/reliability by opening up the spatial dimension.

MIMO systems can be used to achieve diversity gain, multiplexing gain or a combination of the two. For MIMO systems seeking multiplexing gain, an essential objective is to cancel the interference among antenna pairs in order to obtain the original transmitted data. This interference-cancelling process can be viewed as decomposing the channel into multiple independent subchannels, each transmitting its own data without intruding on one another.

This research focuses on the realizations and applications of MIMO channel decomposition algorithms. The first part of the work, Chapters 2 and 3, are based on three channel decomposition methods proposed by Yi Jiang et. al., including the Geometric Mean Decomposition (GMD), the Uniform Channel Decomposition (UCD) and the Tunable Channel Decomposition (TCD). We first present hardware design of closed-loop MIMO transceivers based on the GMD and the UCD, which can both decompose the communications channel into identical subchannels, but the UCD is superior in that it is capacity lossless. We then discuss the application of the TCD in cognitive radio systems, which can control the individual gains of the decomposed subchannels and is suitable for

satisfying different quality-of-service (QoS) constraints. We present a reconfigurable MIMO transceiver design based on the TCD.

In the second part of this dissertation, Chapter 4, we investigate MIMO transceiver designs over channels where inter-symbol interference (ISI) is present. We propose a new fast iterative algorithm to obtain the minimum-mean-square-error decision feedback equalizer (MMSE-DFE) for MIMO single carrier systems. This algorithm is based on the QR decomposition of an augmented channel matrix. It outperforms other time domain MMSE-DFE algorithms in terms of complexity and flexibility; moreover, it can be converted into hybrid DFE where the feedforward part is in frequency domain, which makes it especially suitable for heterogeneous networks.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 MIMO systems: gains introduced by multiple antennas

Multiple-input multiple-output (MIMO) wireless systems have received great attention in recent years [1] [2] [3]. As its name indicates, multiple antennas are deployed both at the transmitter and the receiver, while traditional single-input single-output (SISO) systems have only one antenna at each side. There are also single-input multiple-output (SIMO) and multiple-input single-output (MISO) systems, as illustrated in Fig. 1.1.

To understand the popularity of MIMO communications, let us look at the gains that can be obtained by introducing additional antennas at the transmitting and/or receiving end. A SISO system has only one single signal communication path between transmitter and receiver, and the reliability of communication solely depends on the channel strength of this path. While communicating in a fading channel where the channel gain varies due to shadowing or multipath, there is a significant probability that this single signal path is in a “deep fade”, which translates to insufficient channel strength for the targeted communication. In order to achieve more reliable communication, a natural solution is to transmit the information across multiple independently fading signal

paths. This technique is called diversity.

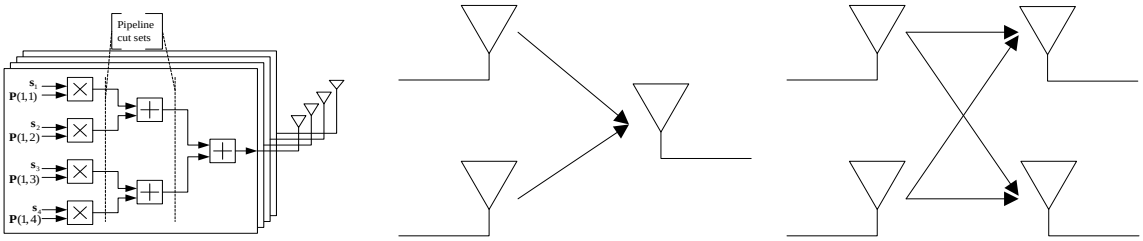


Fig. 1.1: Different types of systems
(a) 1 × 2 SIMO system (b) 2 × 1 MISO system (c) 2 × 2 MIMO system

Diversity gain can be obtained over time, space or frequency. Deploying multiple antennas at the transmitter and/or the receiver can exploit diversity over space, given that there is scattering and that the antennas are spaced sufficiently far apart so that the signal paths experience independent fading. In a rich multipath environment, the channel decorrelates over a short distance, and from one-half to one carrier wavelength is sufficient antenna separation. In a MIMO system with M_t transmitting antennas and M_r receiving antennas, the maximal diversity gain is $M_t \times M_r$, which is the total number of fading gains that can be averaged over.

Besides diversity gain, SIMO/MISO systems can also achieve additional power gain or array gain. For SIMO systems, by coherently combining the multiple signal copies at the receiver, the effective total received signal power increases linearly with the number of receiving antennas. For MISO systems, given the channel state information at the transmitter (CSIT), the so-called "transmit beamforming" strategy can be used to achieve the same power gain. This strategy maximizes the received signal-to-noise ratio

(SNR) by aligning the signal phase and allocating proper amounts of power to the transmit antennas depending on the channel information.

Moreover, in MIMO systems where multiple antennas are present both at the transmitter and the receiver, additional degrees-of-freedom can be exploited from the expanded spatial dimension. The degrees-of-freedom is defined as the dimension of the received signal space, which largely affects the system throughput since it determines the number of different signals that can be reliably distinguished at the receiver. While diversity is a means for combating fading, the extra degrees-of-freedom are in fact a desirable side effect of fading. If the paths between individual transmit–receive antenna pairs fade independently, the MIMO channel matrix is well conditioned with high probability, and multiple independent parallel spatial subchannels are created. This can be viewed through a channel decomposition perspective, as discussed in the next section. As a result, we can achieve a large increase in system throughput by spatially multiplexing different data streams onto the MIMO channel. This effect is known as the multiplexing gain.

One way to quantify the MIMO multiplexing gain is to look at the spectral efficiency. The spectral efficiency that can be achieved by a SISO system in an additive white Gaussian noise (AWGN) channel is:

$$C = \log_2(1+\mathrm{SNR}) \quad \text{bps/Hz} \tag{1.1}$$

where SNR is defined as the average received signal energy divided by the noise energy per symbol time. With the same total input power, a MIMO system with M_t transmitting antennas and M_r receiving antennas can achieve a spectral efficiency up to [4]

$$C = \min(M_r, M_t)\log_2(\text{SNR}) + O(1) \text{ bps/Hz} \quad (1.2)$$

This boost in spectral efficiency is especially desirable, given the ever-increasing demands on data rate in modern wireless applications, and the fact that the available spectrum resources are both limited and expensive. Indeed, the spatial multiplexing gain is arguably a more important benefit than either diversity gain or array again in MIMO systems.

Not surprisingly, with these promising gains, MIMO systems have been under intense research over the past decade. Several space-time coding methods have been developed in [5] [6] [7] [8] for maximizing diversity gain, while the works in [9] [10] [11] [12] [13] focus on multiplexing gain.

In the context of exploiting multiplexing gain, transceiver design can be viewed as a way to decompose the MIMO channel into multiple parallel subchannels, as mentioned above. The performances of different designs depend on the various decompositions of the MIMO channel matrix which can be applied. In this dissertation, we focus on the realizations and applications of these MIMO channel decomposition algorithms.

1.2 MIMO communications from a channel decomposition perspective

In this section, we first introduce the MIMO system model, followed by two representative ways of decomposing the MIMO channel, namely the QR decomposition

and the SVD decomposition, each of which leads to a well-known transceiver design.

Fig. 1.2 shows a MIMO system with M_t transmitting antennas and M_r receiving antennas in a flat fading channel, where the bandwidth of the input is considerably less than the channel's coherence bandwidth and where a single tap is sufficient to represent each channel component.

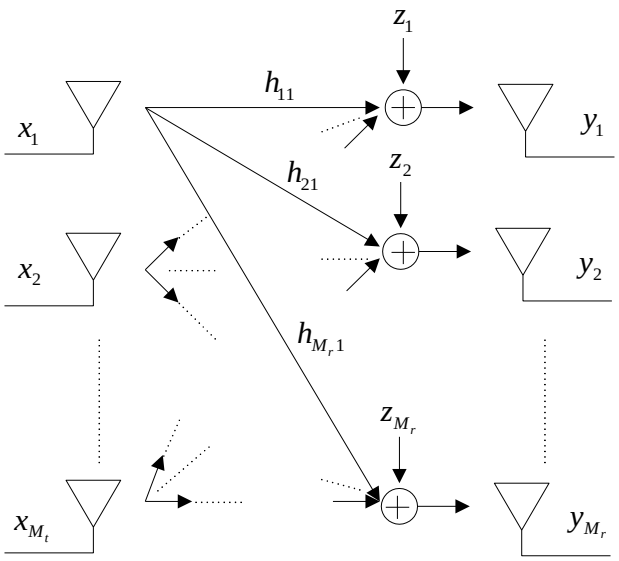


Fig. 1.2: MIMO system model.

The system input-output relationship can be modeled as:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{M_r} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1M_t} \\ h_{21} & h_{22} & \cdots & h_{2M_t} \\ \vdots & \ddots & \ddots & \vdots \\ h_{M_r1} & \cdots & \cdots & h_{M_rM_t} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{M_t} \end{bmatrix} + \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{M_t} \end{bmatrix} \tag{1.3}$$

Or, more compactly, as:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z} \tag{1.4}$$

where the channel $\mathbf{H} \in C^{M_r \times M_t}$ is a rank K matrix whose $(i, j)^{\text{th}}$ element represents the

channel impulse response between the j^{th} transmitting and the i^{th} receiving antenna, $\mathbf{x} \in C^{M_t \times 1}$ and $\mathbf{y} \in C^{M_r \times 1}$ are the transmitted and received signals and $E[\mathbf{x}\mathbf{x}^*] = \sigma_x^2 \mathbf{I}$. $\mathbf{z} \sim N(0, \sigma_z^2 \mathbf{I}_{M_r})$ is zero-mean circularly symmetric complex Gaussian noise vector.

It is apparent that the received signal at each receiving antenna contains information from all the transmitting antennas. With proper processing at the receiver and/or the transmitter, the primary goal of the MIMO transceiver is to separate and decode these simultaneously transmitted symbols. This can be realized based on MIMO channel matrix decomposition algorithms. Depending on the availability of the channel state information (CSI) at the transmitter (CSIT), different channel decomposition methods can be developed, which in turn results in corresponding transceiver designs and performances.

We review two typical MIMO channel decomposition methods below. One is the QR decomposition, which is the basis of the well-known Vertical-Bell Laboratories Layered Space-Time (VBLAST) receiver [10]. It does not assume CSIT, so it is referred to as an open-loop design. The other decomposition method is the singular value decomposition (SVD), which achieves the MIMO channel capacity if it is combined with the so-called "water-filling" power allocation method. CSIT is required in this case, which allows collaborations between the transmitter and the receiver, and thus it is known as a closed-loop or joint transceiver design [15].

Let us start with the VBLAST open-loop design. Without CSIT, the optimum power allocation strategy is to transmit independent data streams with equal power across all transmit antennas. At the receiving end, a sequential nulling and canceling process is

used to separate these data streams. Data symbols from the independent data streams are detected in a certain order, and the symbols detected previously are canceled out from the received signal to facilitate the current symbol estimation. It can be used with either a zero-forcing (ZF) or minimum-mean-square-error (MMSE) estimator. When combined with a ZF estimator, the decoding scheme can be described through the QR decomposition of the channel matrix **H**.

Suppose $M_t \leq M_r$ and rank $K = M_t$, the QR decomposition of **H** is:

$$\mathbf{H} = \mathbf{QR} \tag{1.5}$$

where **Q** is an $M_r \times K$ matrix with orthonormal columns and **R** is a $K \times K$ upper triangular matrix. Then, (1.4) can be rewritten as:

$$\mathbf{y} = \mathbf{QRx} + \mathbf{z} \tag{1.6}$$

At the receiver, left multiply $\mathbf{Q}^*$ with **y** to obtain:

$$\tilde{\mathbf{y}} = \mathbf{Q}^*\mathbf{y} = \mathbf{Rx} + \tilde{\mathbf{z}} \tag{1.7}$$

or

$$\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_K \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1K} \\ 0 & r_{22} & \cdots & r_{2K} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{KK} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{bmatrix} + \begin{bmatrix} \tilde{z}_1 \\ \tilde{z}_2 \\ \vdots \\ \tilde{z}_K \end{bmatrix} \tag{1.8}$$

Due to the upper triangular structure of **R**, the sequential nulling and cancelling process can be performed as follows:

$$\begin{aligned} &\text{for } i = K:-1:1 \\ &\quad \hat{x}_i = C\left[(\tilde{y}_i - \sum_{j=i+1}^{K} r_{ij}\hat{x}_j) / r_{ii} \right] \\ &\text{end} \end{aligned} \tag{1.9}$$

where $\hat{x}_i$ is the estimate of x_i, and the notation $C[\ ]$ indicates mapping to the nearest symbol in the modulation constellation. Incorporating the widely used simplifying assumption of no error-propagation, the MIMO channel is equivalently decomposed into K independent subchannels:

$$\tilde{y}_i = r_{ii} x_i + \tilde{z}_i, i = 1, 2, ..., K \tag{1.10}$$

However, ZF-VBLAST can only achieve about 72% of the MIMO channel capacity [10]. This is mainly due to the lack of precoding at the transmitter: equal-rate transmission on all subchannels makes the overall channel capacity limited by the worst subchannel.

If the channel is slowly time-varying, CSIT may also be available through feedback or if time-division duplex (TDD) is used. This enables the closed-loop/joint transceiver design, with precoding at the transmitter and equalization at the receiver. In this way, the performance of the system can be greatly enhanced. Suppose the precoder at the transmitter is denoted by $\mathbf{F} \in C^{M_t \times M_t}$, then the system equation becomes:

$$\mathbf{y} = \mathbf{HFx} + \mathbf{z} \tag{1.11}$$

Optimizing **F** and the corresponding receiver is the goal of closed-loop transceiver design.

It is widely understood that the singular value decomposition (SVD) and water-filling method can be combined to achieve the MIMO channel capacity [4]. Assume the SVD of the channel is $\mathbf{H} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^*$, where $\mathbf{\Lambda}$ is a non-negative diagonal matrix containing the singular values of **H** in a non-increasing order, and **U** and **V** are unitary matrices consisting of the right and left singular vectors of **H**, respectively. The number of the

positive entries of $\mathbf{\Lambda}$ is just *K*, the rank of the channel matrix **H**. Suppose that the total power constraint is P.

The capacity-achieving **F** is

$$\mathbf{F} = \mathbf{V}\mathbf{\Phi}^{1/2} \tag{1.12}$$

where $\mathbf{\Phi}$ is a diagonal matrix whose k^{th} diagonal element ϕ_k indicates the power loaded to the k^{th} subchannel and is determined by the water-filling method:

$$\phi_k = \left(\mu - \frac{\alpha}{\lambda_{H,k}^2} \right)^+ \tag{1.13}$$

Here $\alpha = \sigma_z^2 / \sigma_x^2$, $\lambda_{H,k}$ is the k^{th} diagonal entry of $\mathbf{\Lambda}$, $(a)^+ = \max\{0, a\}$ and μ is the chosen water-filling "level" such that:

$$\sum_{k=1}^{K} \phi_k = \mathrm{P} \tag{1.14}$$

The water-filling method is illustrated in Fig. 1.3. The red lattice bars are $\alpha / \lambda_{H,k}^2$ for each subchannel (so that a smaller bar indicates better subchannel gain), and the blue horizontally lined bars are ϕ_k, the power loaded into each subchannel. It is evident that this process is like filling the profile constructed by the red bars with a constant amount of water (i.e., the total power). More power is put into the stronger subchannels and less or no power is allocated to the weaker ones.

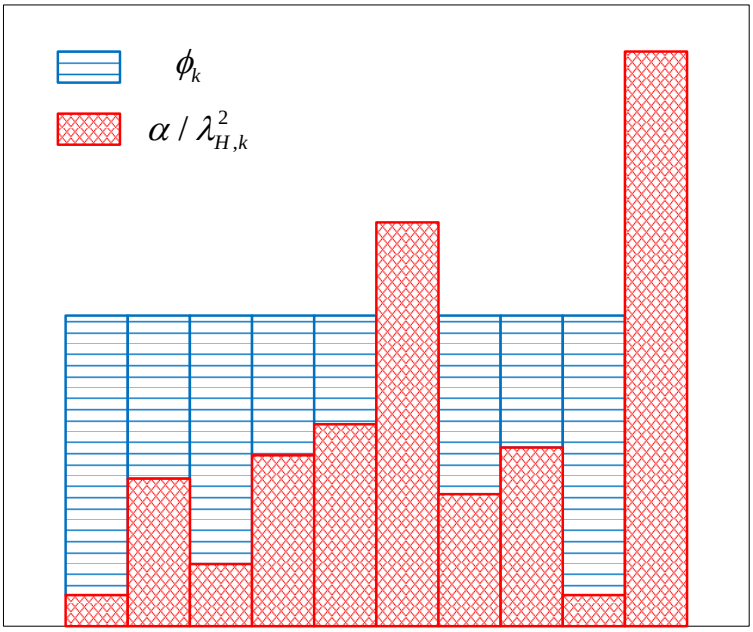


Fig. 1.3: Water-filling method.

By multiplying $\mathbf{U}^*$ and the received signal at the transmitter, we obtain:

$$\tilde{\mathbf{y}} = \mathbf{\Lambda \Phi}^{1/2}\mathbf{x} + \tilde{\mathbf{z}} \tag{1.15}$$

so the MIMO channel is decomposed into K independent subchannels. The overall channel capacity is:

$$C = \sum_{k=1}^{K} \log_2(1 + \frac{\phi_k}{\alpha} \lambda_{H,k}^2) \text{ bps/Hz} \tag{1.16}$$

which is shown to be the MIMO channel capacity [4]. To achieve this capacity, however, one has to adjust the modulation size/format in accordance with $\lambda_{H,k}$, which is called bit-loading. In realistic implementations, capacity loss is inevitable due to the granularity of the constellations. Moreover, careful power allocation and bit-loading requires additional system complexity and processing power [17] [18]. On the other hand, if we want to avoid bit-loading and use the same modulation scheme for all subchannels, there is a

tradeoff between the channel throughput and the bit-error-rate (BER) performance, since more power should be allocated to the poorer channels if a certain BER constraint needs to be satisfied.

1.3 MIMO channel decomposition algorithms: GMD, UCD and TCD

To resolve the above dilemma, it would be desirable if we can decompose the channel into parallel subchannels that have the same channel gain. Yi Jiang et. al. proposed a class of channel decomposition methods for MIMO transceiver design, including the geometric mean decomposition (GMD) [19], the uniform channel decomposition (UCD) [20] and the tunable channel decomposition (TCD) [21]. Among them, the GMD can decompose the MIMO channel into *K* identical subchannels (*K* is the rank of the channel matrix), and it is asymptotically optimal at high SNR. On the other hand, the UCD can decompose the channel into an arbitrarily large number of identical subchannels, and is also strictly capacity lossless at any SNR. Furthermore, the TCD can decompose the channel, in a capacity lossless manner, into independent subchannels with prescribed channel gains. We briefly review these three methods in this section.

The GMD theorem [22] states that, for a channel matrix **H** described in (1.2) with non-zero singular values $\{\lambda_{H,k}\}_{k=1}^{K}$, there exists an upper triangular matrix $\mathbf{R} \in C^{K\times K}$ and semi-unitary matrices $\mathbf{Q} \in C^{M_r\times K}$ and $\mathbf{P} \in C^{M_t\times K}$ such that:

$$\mathbf{H} = \mathbf{QRP}^{*} \tag{1.17}$$

where **R** has identical diagonal elements $R_{ii} = \left(\prod_{k}^{K} \lambda_{H,k} \right)^{1/K}$. (Note that a semi-unitary matrix is a matrix having orthonormal columns.)

The GMD can be viewed as an extended QR decomposition since R is also an upper triangular matrix. Thus, ZF-VBLAST can be used in the receiver design, with **P** functioning as a linear precoder at the transmitter. Since the diagonal elements of **R** have identical values, the MIMO channel is effectively decomposed into *K* identical subchannels, which naturally bypasses the need to make tradeoffs between the throughput and the BER performance.

As shown in [19], the GMD scheme is asymptotically optimal for high SNR. However, for lower SNR, it suffers from performance loss due to the embedded zero-forcing operations. To improve on this aspect, a subsequent channel decomposition method, the UCD, was proposed in [20].

The UCD method is also based on the GMD theorem. Instead of using ZF-VBLAST at the receiver, however, it utilizes the MMSE-VBLAST detector. It shares the desirable property of GMD in that it decomposes the MIMO channel into identical subchannels. Moreover, UCD achieves the following three improvements over the GMD: first, it is strictly capacity lossless at any SNR (due to the information lossless property of MMSE-VBLAST); second, UCD has the maximal diversity gain; and third, it can decompose the MIMO channel into an arbitrarily large number of independent subchannels (i.e., not constrained by *K*) by introducing an additional semi-unitary matrix in the precoder. This can be particularly useful when high throughput transmission is to be achieved with small constellation modulation schemes.

Taking one step further, the TCD method, which is based on the generalized triangular decomposition algorithm developed in [23], introduces an additional desirable feature: it can decompose a MIMO channel into independent subchannels with prescribed channel gains. The TCD method can be beneficially applied to multi-application MIMO communication systems having various QoS constraints.

A major focus of this dissertation is the efficient hardware realization of MIMO transceivers based on the above channel decomposition algorithms, which are presented in Chapters 2 and 3.

1.4 MIMO transceiver design in ISI channels

All of the discussion above is based on the assumption that the MIMO channel is flat-fading and that no ISI is present. However, with the data rate requirements of wireless systems continually increasing over time, this assumption is likely to be invalid. When the data rate is high enough so that the input bandwidth is larger than the channel coherence bandwidth, the channel exhibits frequency-selective fading. In this case, the MIMO channel needs to be modeled with multiple taps, and the received signal at a receiving antenna at a certain symbol time is affected by all the transmitted data symbols not only at the current symbol time, but also from previous symbol times within the channel length. This leads to inter-symbol interference (ISI).

In order to remove ISI, equalization needs to be performed, which is essentially the compensation of the distortion caused by channel frequency selectivity. In traditional

SISO systems, equalization can be performed either in the time domain or in the frequency domain. For practical time domain equalization (TDE), the finite-length MMSE decision-feedback equalizer (MMSE-DFE) is a favored approach since its complexity is much lower than that of the maximum likelihood (ML) equalizer, while still having near-optimal performance [24]. However, the equalization complexity per symbol time increases at least linearly with the channel length (i.e., the ISI span), making it less appealing for longer channels.

Frequency domain equalization (FDE) offers a lower-complexity (per symbol time) solution for equalization with long ISI span. Given both the received signal and the channel response in the frequency domain, the equalization can be greatly simplified due to the fact that convolution in time domain corresponds to multiplication in frequency domain. In order to convert the signal into the frequency domain, FDE operates on blocks of symbols using the Discrete Fourier Transform (DFT), which is efficiently implemented by the Fast Fourier Transform (FFT). This leads to an equalization complexity that grows only logarithmically with the channel length, which makes FDE a favored option over TDE in modern wireless communication standards.

FDE can be performed with both multicarrier modulation and single carrier modulation. Orthogonal frequency division multiplexing (OFDM) is a widely adopted multicarrier transmission scheme, due to its low complexity and close-to-optimum performance [25]. At the transmitter, information is modulated in the frequency domain, converted by an inverse DFT into the time domain and then transmitted. The receiver converts the signal back into frequency domain, and then performs equalization and

demodulation. However, the OFDM waveform has large envelope fluctuations, which leads to a high peak-to-average power ratio (PAPR). This leads to low power efficiency, which is not a desirable characteristic in handheld devices.

Single carrier FDE (SC-FDE) [26] has emerged as an alternative that effectively addresses this issue. Information is modulated and transmitted in the time domain; the received signal is converted to the frequency domain and equalization is performed, and then it is converted back to the time domain again for detection. Thus, by moving the inverse DFT to the receiver, single carrier transmission can also make use of simplified equalization, as in OFDM. Moreover, it solves the high PAPR problem, which makes it a strong candidate for mobile uplink transmissions. However, the performance of SC-FDE can only compare to that of OFDM or time domain MMSE-DFE if a DFE is used, which is called hybrid DFE (HDFE) since the DFE portion is usually implemented in the time domain [27].

Compared to SISO systems, MIMO systems require a lot more complexity in equalization, with each channel tap being a matrix rather than a scalar. Which approach is then a better single-carrier scheme for MIMO transmissions, TD MMSE-DFE or HDFE? In terms of performance, both schemes can be capacity achieving. In terms of complexity, for longer channels, the equalization complexity per symbol time of TD MMSE-DFE is higher than that of HDFE; however, for shorter channels, TD MMSE-DFE is likely to be simpler without the overhead of FFT operations. In heterogeneous networks where cells of different sizes coexist, it would be desirable if we can exploit the merits of both equalization methods.

In this dissertation, we propose a fast finite-length TD MMSE-DFE which can be converted to HDFE when the channel length becomes sufficiently long. This algorithm is also based on the QR decomposition of an augmented channel matrix.

1.5 Contributions of the thesis

In Chapter 2, we present joint MIMO transceiver designs based on the GMD and UCD. We find that the hardware resources required for GMD-based and UCD-based designs are comparable, which makes the UCD method more appealing due to its superior BER performance. Both transceiver designs can run at 400 MHz with a 16-QAM constellation on a Xilinx Virtex-4 FPGA, and can achieve a data throughput of 12.8 Gbps in an 8x8 MIMO system.

In Chapter 3, we discuss the possible application of the TCD in software defined radio (SDR) communications. Subject to limited spectrum resources and specified quality-of-service (QoS) constraints, we map the required services onto the available frequency bands, using the TCD to partition the MIMO channel into subchannels having prescribed gains. We then present a reconfigurable closed-loop transceiver design which can adapt to various partitioning scenarios. This design has the potential to deliver the required services with reduced transmitting power, and has approximately the same area as its non-configurable counterpart.

Chapter 4 proposes a new fast iterative algorithm to obtain the optimal finite impulse response (FIR) TD MMSE-DFE for MIMO over ISI channels, based on the QR

decomposition of an augmented Toeplitz channel matrix. The proposed algorithm is not only computationally efficient but also very frugal in its memory usage. The algorithm applies to MIMO systems with any number of transmitting and receiving antennas, either balanced or unbalanced. The iterative feature of the proposed algorithm enables a flexible choice of N_f, the length of the feedforward filter (FFF), which translates to a controllable tradeoff between complexity and performance for real implementations. We also propose a conversion from TDE to HDFE for lower complexity when N_f becomes large. Complexity analysis suggests significant improvement over the prior art on TDE.

The conclusions are given in Chapter 5, together with suggestions for future related work.

Chapter 2

Joint MIMO Transceiver Design Based on GMD and UCD

In Chapter 1, we reviewed that the MIMO channel capacity can be achieved by combing the SVD of the channel matrix with the water-filling method. However, the water-filling power allocation procedure requires different modulation schemes to be used for subchannels with different qualities. This bit-loading method not only has additional complexity, but also leads to capacity loss in actual implementations due to the granularity of constellation sizes. If the same modulation scheme is used for all subchannels, then there is an inevitable tradeoff between throughput and BER performance, arising from the vastly different subchannel gains. Given the channel matrix, one has no control over these subchannel gains using the SVD, as they are determined by the singular values of the channel matrix.

In this chapter, we present joint MIMO transceiver designs based on two recently introduced channel decomposition methods, GMD [19] and UCD [20]. Their shared advantage is that the MIMO channel becomes decomposed into multiple identical subchannels, thereby eliminating the dilemma faced with the SVD. Furthermore, the UCD is capacity lossless and it can decompose the channel into an arbitrary number of subchannels, regardless of the rank of the channel matrix.

We use the Xilinx blocksets in Matlab Simulink to build our designs, and then transform them into hardware using System Generator [28]. The synthesis results show that both the GMD-based and UCD-based closed-loop transceiver designs can support a data throughput of up to 12.8 Gbps with a 400 MHz clock rate for an 8×8 MIMO, 16-QAM system design.

2.1 GMD and UCD

In this section, we will give an overview of the GMD and UCD methods, the full details of which can be found in [19] and [20], respectively.

2.1.1 Geometric Mean Decomposition

Recall from Chapter 1 that a MIMO system with M_t transmitting and M_r receiving antennas in a flat fading channel can be modeled as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z}, \tag{2.1}$$

where the channel $\mathbf{H} \in C^{M_r \times M_t}$ is a rank K matrix whose $(i, j)^{\text{th}}$ element represents the channel impulse response between the j^{th} transmitting and the i^{th} receiving antenna, $\mathbf{x} \in C^{M_t \times 1}$ and $\mathbf{y} \in C^{M_r \times 1}$ are the transmitted and received signals and $E[\mathbf{x}\mathbf{x}^*] = \sigma_x^2 \mathbf{I}$. $\mathbf{z} \sim N(0, \sigma_z^2 \mathbf{I}_{M_r})$ is zero-mean circularly symmetric complex Gaussian noise vector.

Suppose the same constellation size is used for all subchannels in order to reduce the system complexity. The overall BER performance is then limited by the worst subchannel. Aiming at maximizing the smallest subchannel gain, consider the following

optimization problem, assuming CSI at both the transmitter and the receiver:

$$\begin{aligned} &\max_{\mathbf{Q},\mathbf{P}} \min \ \{r_{ii} : 1 \le i \le K\} \\ &\text{subject to } \mathbf{R} = \mathbf{Q}^{*}\mathbf{H}\mathbf{P} \\ &\qquad \mathbf{R} \in \mathbb{R}^{K\times K},\ r_{ij} = 0 \text{ for } i > j \\ &\qquad r_{ii} > 0 \text{ for } 1 \le i \le K \\ &\qquad \mathbf{Q}^{*}\mathbf{Q} = \mathbf{P}^{*}\mathbf{P} = \mathbf{I}_{K} \end{aligned} \tag{2.2}$$

where **P** and **Q** are semi-unitary matrices indicating the linear operations performed at the transmitter and the receiver. It is proved in [22] that the GMD of **H** is the solution to this problem.

The geometric mean decomposition [22]: For any rank K matrix $\mathbf{H} \in C^{M_r \times M_t}$ with singular values $\lambda_{H,1} \ge \lambda_{H,2} \ge \cdots \ge \lambda_{H,k} > 0$, there exists an upper triangular matrix $\mathbf{R} \in \mathbb{R}^{K\times K}$ with identical diagonal elements

$$r_{ii} = \bar{\lambda}_H \equiv \left(\prod_{k}^{K} \lambda_{H,k} \right)^{1/K}, 1 \le i \le K \tag{2.3}$$

and semi-unitary matrices $\mathbf{Q} \in C^{M_r \times K}$ and $\mathbf{P} \in C^{M_t \times K}$ such that $\mathbf{H} = \mathbf{Q}\mathbf{R}\mathbf{P}^{*}$.

The name GMD is due to the characteristic of the decomposition that the diagonal elements of **R** are all equal to the geometric mean of the singular values of **H**. An algorithm for computing GMD is also proposed in [22].

The GMD of **H** can be readily combined with ZF-VBLAST in a joint transceiver design. At the transmitter, we first encode the information symbol $\mathbf{s} \in C^{K\times 1}$ via the linear precoder **P** to be $\mathbf{x} = \mathbf{P}\mathbf{s}$. Equation (2.1) is now equivalent to:

$$\mathbf{y} = \mathbf{Q}\mathbf{R}\mathbf{s} + \mathbf{z} \tag{2.4}$$

At the receiving end, multiplying on the left by Q^{*} gives:

$$\tilde{\mathbf{y}} = \mathbf{R}\mathbf{s} + \tilde{\mathbf{z}} \tag{2.5}$$

or

$$\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_K \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1K} \\ 0 & r_{22} & \dots & r_{2K} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & r_{KK} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_K \end{bmatrix} + \begin{bmatrix} \tilde{z}_1 \\ \tilde{z}_2 \\ \vdots \\ \tilde{z}_K \end{bmatrix} \tag{2.6}$$

Now, ZF-VBLAST can be used to decode the transmitted information symbol. It estimates the last symbol first, and then cancels it out from the received signal vector, then estimates the preceding symbol, and so on.

Assuming there is no error-propagation, the GMD decomposes a MIMO channel into multiple identical subchannels

$$\tilde{y}_i = \bar{\lambda}_H s_i + \tilde{z}_i, \quad i = 1, ..., K. \tag{2.7}$$

Fig. 2.1 depicts the joint transceiver structure using the GMD method. The GMD scheme has been shown to be optimal asymptotically at high SNR in terms of both the information rate and the BER performance [19].

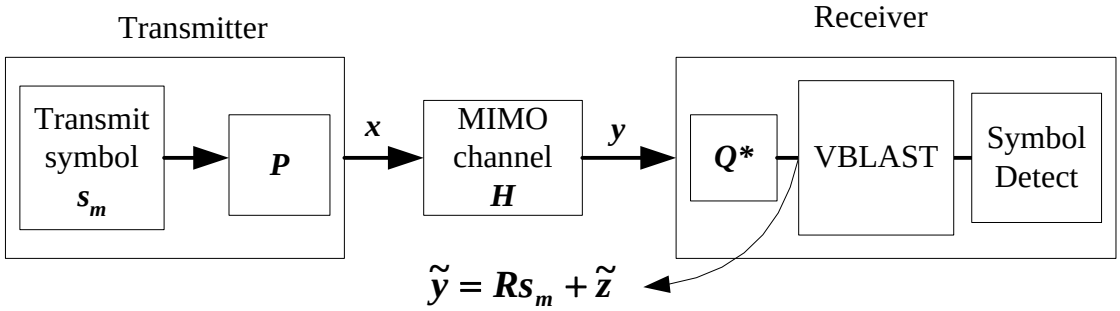


Fig. 2.1: Joint transceiver design based on the GMD.

While the GMD gives an attractive method of channel decomposition, it has the

drawback that its performance degrades when the SNR becomes low. This is due to the fact that the ZF-VBLAST decoder is inherently suboptimal. In the following subsection we will describe the UCD, which is a further improved channel decomposition method.

2.1.2 Uniform Channel Decomposition

The UCD method is proposed in [20] to eliminate the capacity loss at low SNR while preserving all the desirable properties of the GMD. In addition, it can decompose the channel into an arbitrary number L of subchannels, no longer constrained by the rank value K. It extends the GMD matrix decomposition algorithm and it incorporates MMSE-VBLAST rather than ZF-VBLAST.

In the UCD, we still have the same input-output relationship as in equation (2.1), except that the information symbol $\mathbf{s} \in C^{L\times 1}$. We first compute the SVD of the channel matrix H as:

$$\mathbf{H} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^{*} \tag{2.8}$$

Then, the precoder matrix F is defined as:

$$\mathbf{F} = \mathbf{V}\boldsymbol{\Phi}^{1/2}\boldsymbol{\Omega}^{*} \tag{2.9}$$

where $\boldsymbol{\Omega} \in C^{L\times K}$ is a semi-unitary matrix and $\boldsymbol{\Phi}$ is a diagonal matrix whose k^{th} diagonal element determines the power loaded into the k^{th} subchannel, which is computed using the standard water-filling method. Since MMSE-VBLAST is information lossless and $\boldsymbol{\Omega}$ is semi-unitary, $\mathbf{F}$ is a precoder maximizing the MIMO channel throughput. However, introducing $\boldsymbol{\Omega}$ brings greater flexibility than the precoder in (1.12).

Combining the precoder $\mathbf{F}$ and the channel $\mathbf{H}$, the virtual channel that the

information symbol **s** passes through will be:

$$\mathbf{G} \equiv \mathbf{HF} = \mathbf{U\Lambda\Phi}^{1/2}\mathbf{\Omega}^{*} \equiv \mathbf{U\Sigma\Omega}^{*} \tag{3.0}$$

Now we consider the augmented matrix

$$\mathbf{G}_a = \begin{bmatrix} \mathbf{U\Sigma\Omega}^{*} \\ \sqrt{\alpha}\mathbf{I}_L \end{bmatrix} \tag{3.1}$$

where $\alpha = \sigma_z^2 / E[\mathbf{s}^{*}\mathbf{s}]$. As proved in [20], we can always find an $\mathbf{\Omega}$ (using the GMD method) such that the QR decomposition of $\mathbf{G}_a$ yields an upper triangular matrix $\mathbf{R}_J$ with equal diagonal elements, that is:

$$\mathbf{G}_a = \mathbf{Q}_{\mathbf{G}_a}\mathbf{R}_J \tag{3.2}$$

We define $\mathbf{Q}^{u}_{\mathbf{G}_a}$ as the matrix containing the first M_r rows of $\mathbf{Q}_{\mathbf{G}_a}$. The nulling vectors of the MMSE-VBLAST are

$$\mathbf{w}_i = r_{J,ii}^{-1}\mathbf{q}_{\mathbf{G}_a,i}, \quad i = 1, 2, ..., L \tag{3.3}$$

where $r_{J,ii}$ is the i^{th} diagonal element of $\mathbf{R}_J$, and $\mathbf{q}_{\mathbf{G}_a,i}$ is the i^{th} column of $\mathbf{Q}^{u}_{\mathbf{G}_a}$. $\mathbf{w}_i$ is then used at the receiver to decode the information symbol, also through a sequential nulling process.

As analyzed in [20], Matlab simulations show that the UCD method requires 2 dB less SNR than the GMD at a 10^{-4} BER when using 16-QAM in a 4×4 MIMO system, due to its use of an MMSE detector.

2.2 GMD-Based Joint Transceiver Design

In this section, we present the design details of the GMD transmitter and the receiver. Various system configurations have been implemented, but we will use the 4 × 4 16-QAM system as an illustrative example. In our implementations, we assume that the channel is slowly time-varying, so that the channel decomposition calculations can be computed in software.

The transmitter precodes the symbol $\boldsymbol{s}$ with $\mathbf{P}$, generating the transmitted signal $\mathbf{x} = \mathbf{P}\mathbf{s}$, as depicted in Fig. 2.2. The precoder Xilinx FPGA synthesis results will be discussed in Section 2.5.

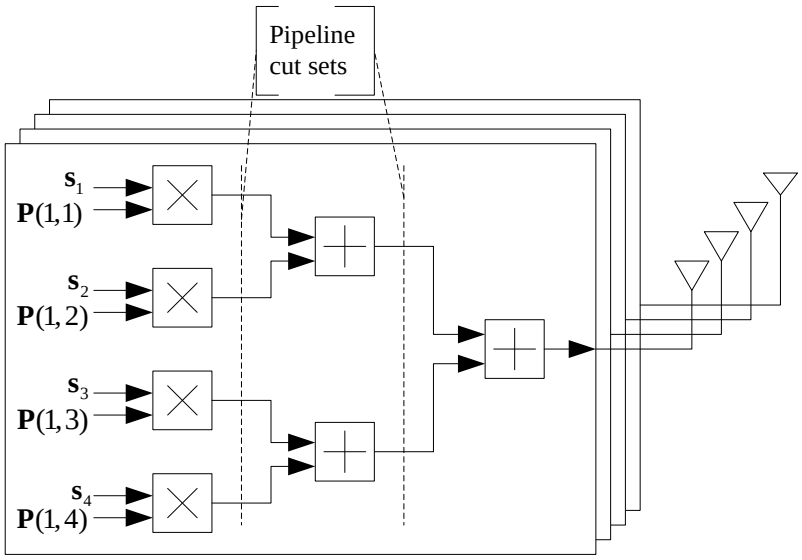


Fig. 2.2: Precoder design at the transmitter.

The receiver is composed of three parts: space equalization, VBLAST and symbol decision.

The space equalization module multiplies $\mathbf{y}$ with $\mathbf{Q}^*$, outputting $\tilde{\mathbf{y}} = \mathbf{Rs} + \tilde{\mathbf{z}}$. Therefore, this module shares the same structure as the precoder at the transmitter. All of the multipliers are maximally pipelined in our design in order to speed up the clock.

Following the space equalization step is the VBLAST module, which removes the interference introduced by the upper triangular elements of $\mathbf{R}$. We first estimate s_K from $\tilde{y}_K$ and r_{KK}:

$$\hat{s}_K = C[\tilde{y}_K / r_{KK}] \tag{3.4}$$

where C[ ] stands for mapping onto the nearest symbol in the symbol constellation. Next, cancel the value from $\tilde{y}_{K-1}$ and estimate s_{K-1}:

$$\hat{s}_{K-1} = C[(\tilde{y}_{K-1} - r_{K-1,K}\hat{s}_K) / r_{K-1,K-1}] \tag{3.5}$$

The above operation is repeated until all symbols are decoded. The whole process can be described as follows:

$$\begin{aligned} &\text{for } i = K:-1:1 \\ &\hat{s}_i = C\left[\left(\tilde{y}_i - \sum_{j=i+1}^{K} r_{ij}\hat{s}_j\right) / r_{ii}\right] \\ &\text{end} \end{aligned} \tag{3.6}$$

The VBLAST structure of this 4×4 design is given in Fig. 2.3 and the decision block is shown in Fig. 2.4.

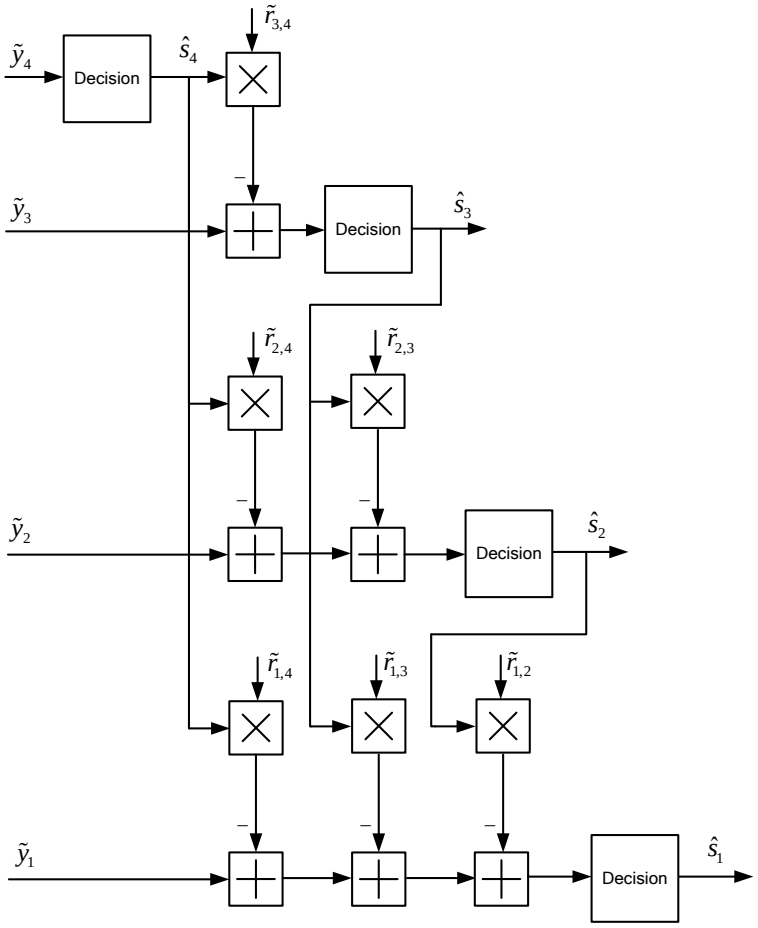


Fig. 2.3: VBLAST structure in the 4 ×4 GMD design
(pipeline registers not shown)

sb is sign bit of input.
b_2 is the 2nd bit to the left of fractional point.

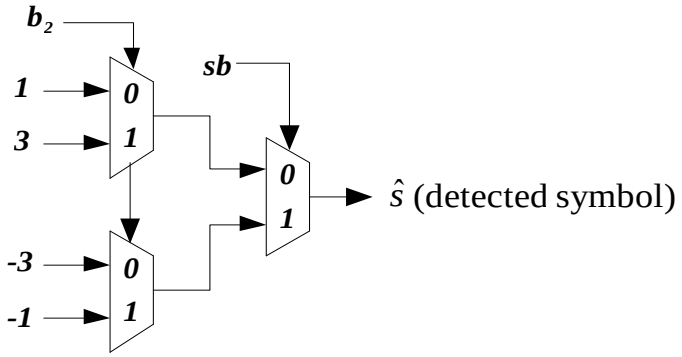


Fig. 2.4: 16-QAM symbol detector with GMD channel decomposition

Due to the sequential nulling and cancellation process, VBLAST may introduce error propagation of the noise from the channel. If one of the symbols was not correctly detected, it will affect the detection of all lower indexed symbols. One possible option is to perform the sequential nulling and cancellation at the transmitter, i.e. the "dirty paper" precoder, as described in [19]. However, this method is not used in our design since it would increase the size and complexity of the transmitter, which contradicts our initial intention to decrease the complexity of the transmitter by avoiding power allocation and bit-loading.

2.3 UCD-Based Joint Transceiver Design

The UCD transmitter here is almost the same as in the GMD case, except that a different matrix **F** is used to precode the information symbol vector **s**. The output is $\mathbf{x} = \mathbf{F}\mathbf{s}$. After

passing through the channel, the received signal is $\mathbf{y} = \mathbf{HFs} + \mathbf{z}$.

The receiver, however, is quite different from the GMD design. As discussed in Section 2.2, $\mathbf{w}_i$ $(i = 1,2,...,L)$ are the nulling vectors input to the MMSE-VBLAST decoder. First, we multiply vector $\mathbf{y}$ with $\mathbf{w}_L$ to get the estimate of s_L:

$$\hat{s}_L = C[\mathbf{y} \times \mathbf{w}_L{}'] \tag{3.7}$$

Then, we "pass" $\hat{s}_L$ through the precoder and channel again, and cancel the result out from $\mathbf{y}$ to obtain $\mathbf{y}1$. Assuming $\tilde{\mathbf{H}} = \mathbf{HF}$, we have:

$$\mathbf{y}1 = \mathbf{y} - \tilde{\mathbf{H}}_{:,L} \times \hat{s}_L \tag{3.8}$$

Now we can estimate s_{L-1} as:

$$\hat{s}_{L-1} = C[\mathbf{y}1 \times \mathbf{w}_{L-1}{}'] \tag{3.9}$$

Again, multiply $\hat{s}_{L-1}$ with the $(L\text{-}1)^{\text{th}}$ column of $\tilde{\mathbf{H}}$ and cancel it out from $\mathbf{y}1$ to get $\mathbf{y}2$, and then estimate s_{L-2}:

$$\mathbf{y}2 = \mathbf{y}1 - \tilde{\mathbf{H}}_{:,L-1} \times \hat{s}_{L-1} \tag{3.10}$$

$$\hat{s}_{L-2} = C[\mathbf{y}2 \times \mathbf{w}_{L-2}{}'] \tag{3.11}$$

The above operations are repeated until all of the symbols have been decoded.

The process can be summarized as follows:

$$\begin{aligned} &\text{for } i = L:-1:1 \\ &\hat{s}_i = C\left[\mathbf{y}[L-i] \times \mathbf{w}_i{}'\right] \\ &\mathbf{y}[L-i+1] = \mathbf{y}[L-i] - \tilde{\mathbf{H}}_{:,i} \times \hat{s}_i \\ &\text{end} \end{aligned} \tag{3.12}$$

Fig. 2.5 shows the receiver structure for a 4 × 4 UCD design. Fig. 2.6 shows the

fully pipelined structure used in our implementation, where pipelines inside blocks are not shown. This design enables very high system throughput as presented in Section 2.4, but also introduces higher latency as a tradeoff. In actual applications, the degree of pipelining can be adjusted according to the system requirements.

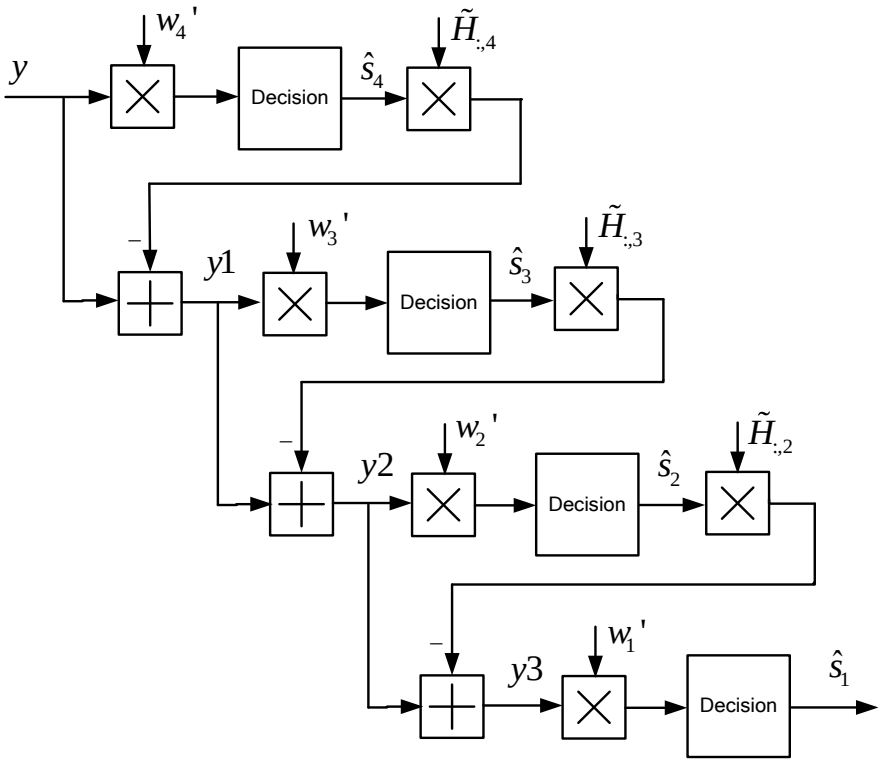


Fig. 2.5: Receiver structure in the 4×4 UCD design

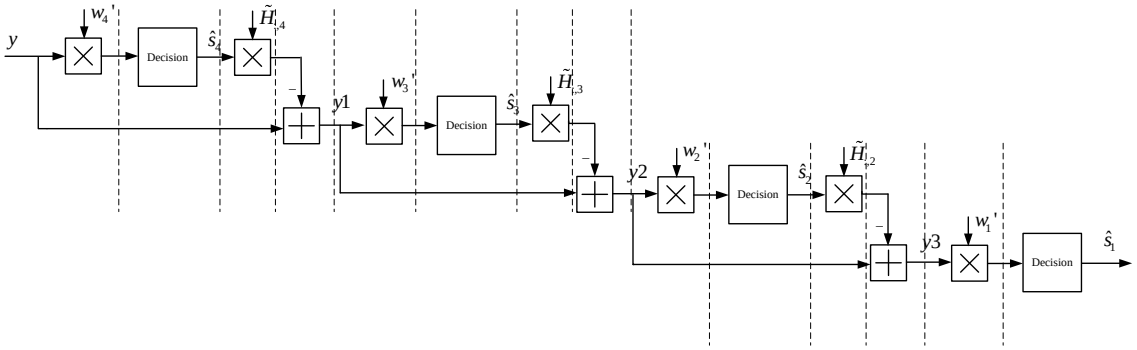


Fig. 2.6: Receiver structure in the 4 × 4 UCD design (fully pipelined)
(pipelines inside blocks are not shown)

2.4 Results and Comparisons

The design flow uses Matlab Simulink as the model builder, followed by the Xilinx System Generator to transform the Simulink model into an RTL description, which then is synthesized and mapped onto a Xilinx xc4vlx200ff1513-12 Virtex-4 FPGA.

We use Matlab to calculate the precoder filter **P**, space equalizer **Q** and channel gain **R** in the GMD design, as well as the precoder filter **F** and nulling vectors w_i in the UCD design, which are then read into Simulink. We assume that all the channel taps are zero mean, unit variance, complex Gaussian random variables. We realized MIMO configurations from 2 × 2 through 8 × 8 using a 16-QAM constellation and from 2 × 2 through 4 × 4 using a 64-QAM constellation under each of the two channel decomposition methods.

2.4.1 Matlab fixed-point simulation

To determine the appropriate bit-length required for satisfactory BER performance, both floating point and fixed point simulations have been done in Matlab. Fig. 2.7 shows the results for 4 × 4 MIMO, 16-QAM UCD, where, for example, the number "15 10" indicates a total bit-length of 15 bits with a fractional part of 10 bits. We can determine from these results that a 5-bit integer part and a 10-bit fractional part give a performance nearly the same as that of floating point. The figure also shows that if the fractional part is reduced to 8 bits, the performance is only slightly degraded at higher SNR.

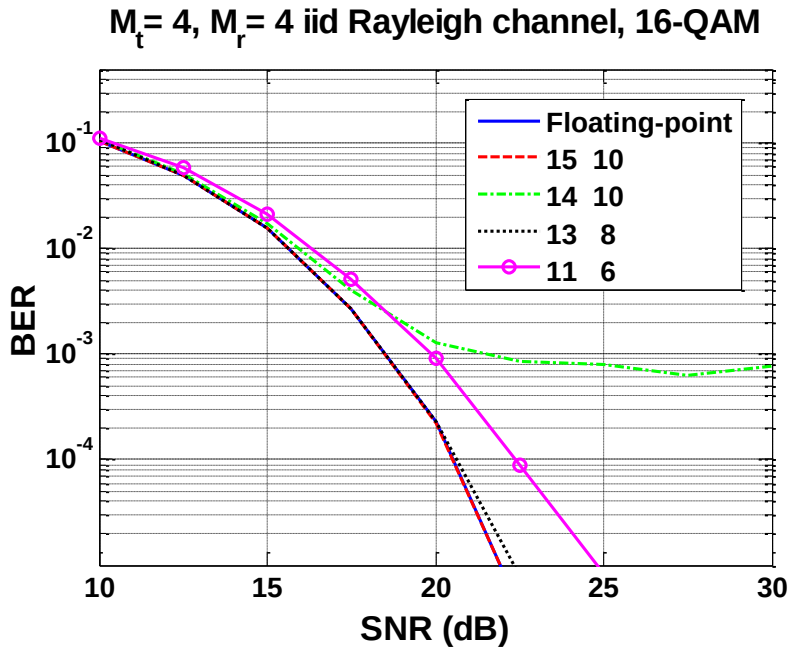


Fig. 2.7: BER performance with different bit-length of the 4 ×4 16-QAM UCD design

Similar simulations have been done for other configurations of both GMD and UCD which also show that using a word length of 15 bits is adequate to keep the quantization error small. Therefore, we have set the word length to be 15 bits in all of our

designs.

2.4.2 FPGA results

We compare the area results for the GMD and UCD designs in Tables 2.1 and 2.2. Note that the two designs share the same precoder structure; only their precoder matrix values differ. We also compare the area of 15-bit and 13-bit realizations for selected configurations of UCD in Table 2.3, in order to give a sense for how the complexity changes for different bit lengths. The speeds of the designs are listed in Table 2.4, where the GMD-based and UCD-based designs (for both 16-QAM and 64-QAM) share the same speed since they are all pipelined at a fine-grained level.

Table 2.1: Area results for 16-QAM (slices)

Configurations	Precoder	Receiver (GMD)	Receiver (UCD)
2 x 2	568	2134	2178
3 x 3	1302	4674	5280
4 x 4	2336	8196	9724
5 x 5	3670	12700	15560
6 x 6	5304	18186	22758
7 x 7	7238	24654	31360
8 x 8	9472	32104	41336

Table 2.2: Area results for 64-QAM (slices)

Configurations	Precoder	Receiver (GMD)	Receiver (UCD)
2 x 2	704	2238	2294
3 x 3	1608	4830	5472
4 x 4	2880	8404	10004

Table 2.3: Area results for different bit-lengths (slices)

Configurations		15 bits (UCD)	13 bits (UCD)	Area Saved (%)
4 x 4 16-QAM	Precoder	2336	2072	11.30
	Receiver	9724	7900	18.76
8 x 8 16-QAM	Precoder	9472	8400	11.32
	Receiver	41336	33592	18.73
4 x 4 64-QAM	Precoder	2880	2376	17.50
	Receiver	10004	8144	18.59

Table 2.4: Speed results

	Max frequency
Precoder	411MHz
Receiver	400MHz

From the area results in Tables 2.1 and 2.2 we can see that, even though in UCD-based designs the decoded symbols have to be "passed through" the precoder and channel again, not many additional hardware resources are consumed compared to the GMD-based designs. This is because in GMD-based designs, the space equalization part occupies a large number of slices. Thus, the UCD-based designs are evidently more favorable since they offer better BER performance. From Table 2.3, we can see that the area decreases by approximately 11-17 percent for the precoder part and by about 19 percent for receiver part. Since less than 1 dB is sacrificed at a BER of 10^{-5}, it may therefore be favorable to use 13 bits instead of 15 bits in applications where area is a critical issue.

2.4.3 Comparisons with other detection schemes

Comparisons with other joint transceiver schemes have been investigated in [19] [20]. As mentioned before, the main advantages of the GMD and UCD methods are that they do not require bit allocation, nor do they need to deal with the trade-off between capacity and BER performance; GMD has been proved to be asymptotically optimal for (moderately) high SNR, and UCD is capacity lossless at any SNR; they can decompose the MIMO channel into identical subchannels, and UCD can even decompose it to an arbitrary number of subchannels.

For open-loop designs, [29] compared the most commonly used MIMO detection algorithms in terms of their performance and complexity, including linear and nonlinear detection schemes and linear adaptive detection. Among them, the Maximum Likelihood

detection has the best performance, but its complexity grows exponentially with the number of antennas. The Sphere Decoding (SD) algorithm [30] [31] has been developed to achieve comparable performance at reduced complexity. However, it is difficult to achieve a fixed throughput with the SD algorithm since its search radius is a function of the channel. Fixed Sphere Decoders (FSD) have been investigated to address this problem [32], but they may exhibit a performance degradation. What's more, our designs give much higher throughputs which are fixed, as shown in Table 2.5. On the other hand, the decomposition schemes used here do require the additional precoder hardware at the transmitting end.

Table 2.5: Comparison with SD designs (4 × 4 16-qam)

	Slices of Receiver	Frequency (MHz)	Throughput
[30](SE)	3880	251	81.5 Mbps
[30](VB)	5614	257	36.8 Mbps
[32]	13743	102	800 Mbps
GMD	7612	400	6.4 Gbps
UCD	9724	400	6.4 Gbps

2.5 Conclusions

We have designed MIMO baseband transceivers based on two advanced channel

decomposition methods, namely the GMD and the UCD. Xilinx synthesis results have been obtained for various MIMO system configurations for both 16-QAM and 64-QAM constellations. Given that the UCD can considerably outperform the GMD in terms of the bit error rate, the UCD-based designs seem to be more appealing since their hardware complexities are comparable to those of the GMD-based designs. The 16-QAM systems run at up to 400 MHz, which gives a data throughput of 6.4 Gbps for a 4×4 MIMO system. The same system configuration under a 64-QAM constellation can achieve a data throughput of 9.6 Gbps at a clock rate of 400 MHz.

Chapter 3

QoS-oriented MIMO transceiver design using TCD in Cognitive Radio

3.1 Introduction

With the ever-increasing demand for more versatile and multi-functional mobile devices, new types of services are being integrated into modern wireless systems, which will require additional bandwidth and power consumption. However, the available spectrum resources are scarce and very expensive, thus placing considerable limitations on the services that can be added. What's more, these services usually have different quality-of-service (QoS) constraints requiring different transmission bit-rates (for example, voice/video/file downloading). How can one satisfy these disparate QoS constraints in a power efficient way using the limited spectrum resources? In this chapter, we address these challenges by using the Tunable Channel Decomposition (TCD) in MIMO systems combined with cognitive radio.

A cognitive radio is adaptable in the sense that it is aware of its environment and can adjust its operation according to dynamic changes in the local spectral conditions [33]. Cognitive radio may utilize a software defined radio (SDR) in which some or all of the

physical layer functions are determined with software or other reconfigurable techniques [34] [35]. In the foreseeable future, we can expect a handheld wireless device being able to adapt to different channels, operating modes, etc. [36]. In the system described here, we assume that it can determine the currently available frequency bands and adapt itself accordingly. In other words, our system can make use of certain frequency bands found through spectrum sensing during a specific time period, i.e. when they are not occupied in the local area by the licensed or primary users [37], or we can control the total power of our system on these frequency bands so that interference to these users are kept under the regulated power level.

Given one or more available frequency bands, the next step is to accommodate the required services onto these bands in a power efficient way. Considering several services sharing the same channel, one possible solution is to exploit the multiplexing gain offered by MIMO as described in Chapter 1. We can map the services on to the independent subchannels obtained by decomposing the MIMO channel matrix. Recall that although SVD and the water-filling technique can be combined together to achieve the MIMO channel capacity, the subchannel gains are determined by the singular values of the channel matrix and cannot be adjusted by the user. This aspect makes it less suitable for transmitting the QoS-constrained services [38]. The GMD and UCD transceivers discussed in Chapter 2 decompose the MIMO channel into identical subchannels, which also don't exactly match the envisioned scenario.

In this chapter, we make use of an alternative method, namely the TCD, which was proposed in [21] and which can decompose a MIMO channel into an arbitrary number of

subchannels having prescribed capacities. It is also capacity lossless, provided that those capacities satisfy a majorization constraint. The TCD scheme can ensure minimal power consumption for transmitting a group of QoS-constrained services in a single MIMO system [21].

We will first discuss the MIMO subchannel allocation problem under a cognitive radio scenario utilizing the TCD method. Assuming several available frequency bands obtained through spectrum sensing, a fixed number of antennas at both the transmitter and the receiver, and the services and their QoS attributes that need to be handled, we seek to achieve an optimized scheme so that the total transmitting power is minimized. Given that the TCD scheme can offer the minimal power solution in a single MIMO system, what we need to do in our system is to find the best way to: 1) group antennas together onto different frequency bands to form subsystems, if necessary; 2) partition the required services onto the subsystems. Then, applying the TCD scheme to each subsystem will result in a power efficient solution.

Next, we will present the hardware architecture design for such a system. Specifically, we will present a reconfigurable MIMO transceiver which can operate as either a single MIMO system or as two MIMO subsystems, all of them using the TCD. Thus, this transceiver can provide the flexible hardware that is necessary for achieving power efficient operation.

The remainder of this chapter is organized as follows: Section 3.2 briefly reviews the TCD algorithm and the resulting TCD scheme which can always achieve minimal power in a single MIMO system under QoS constraints. In Section 3.3, the subchannel

allocation in MIMO systems under an SDR scenario is discussed according to the two aforementioned steps. Section 3.4 presents the reconfigurable MIMO transceiver design based on TCD. Conclusions are given in Section 3.5.

3.2 The TCD Scheme

We consider a MIMO system with M_t transmitting and M_r receiving antennas in a flat fading channel. The system input-output relationship can be modeled as

$$\mathbf{y} = \mathbf{HFx} + \mathbf{z}, \tag{3.1}$$

where $\mathbf{H} \in \mathbf{C}^{M_r \times M_t}$ is the channel matrix with rank *K*, $\mathbf{x} \in \mathbf{C}^{L \times 1}$ is the information symbols precoded by the linear precoder $\mathbf{F} \in \mathbf{C}^{M_t \times L}$, $\mathbf{y} \in \mathbf{C}^{M_r \times 1}$ is the received signal, and $\mathbf{z} \sim N(0, \sigma_z^2 \mathbf{I}_{M_r})$ is a zero-mean circularly symmetric complex Gaussian noise vector. The SNR is defined as

$$\rho = \frac{E[\mathbf{x}^* \mathbf{F}^* \mathbf{F} \mathbf{x}]}{\sigma_z^2} = \frac{\sigma_x^2}{\sigma_z^2} \mathrm{Tr}\{\mathbf{F}^* \mathbf{F}\} \equiv \frac{1}{\alpha} \mathrm{Tr}\{\mathbf{F}^* \mathbf{F}\} \tag{3.2}$$

Denote the SVD of the channel matrix *H* as $\mathbf{H} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^*$, where $\mathbf{\Lambda}$ is a non-negative diagonal matrix containing the singular values $\{\lambda_{H,k}\}_{k=1}^K$ of $\mathbf{H}$ in a non-increasing order, and where * is the complex conjugate transpose. In a conventional SVD scheme, $\mathbf{F} = \mathbf{V} \mathbf{\Phi}^{1/2}$ where $\mathbf{\Phi}$ is a diagonal matrix whose diagonal elements $\{\phi_k\}_{k=1}^K$ indicate the power allocation (not necessarily water filling, but depending on the specific scheme).

Then, the MIMO channel is transformed into K orthogonal subchannels with capacities

$C_k = \log_2(1+\frac{\lambda_{H,k}^2 \phi_k}{\alpha})$ bps/Hz, $k = 1,2,...,K$.

3.2.1 TCD algorithm

In the TCD algorithm, the precoder is modified to become $\mathbf{F} = \mathbf{V}\mathbf{\Phi}^{1/2}\mathbf{\Omega}^{\mathrm{T}}$, where $\mathbf{\Omega} \in \mathbf{R}^{L\times K}$ with $L \geq K$ satisfies $\mathbf{\Omega}^{\mathrm{T}}\mathbf{\Omega} = \mathbf{I}$, and where the superscript T denotes the transpose. Thus, without changing the overall capacity, greater flexibility can be introduced.

According to the TCD Theorem given in [19], for any $L \geq K$, let $\mathbf{c} \in \mathbf{R}^L$ be a zero vector with its first K elements replaced by $\{C_k\}_{k=1}^K$. Given any rates $\{R_k\}_{k=1}^L$, we can find an orthonormal matrix $\mathbf{\Omega} \in \mathbf{R}^{L\times K}$ such that the combination of the linear precoder $\mathbf{F} = \mathbf{V}\mathbf{\Phi}^{1/2}\mathbf{\Omega}^{\mathrm{T}}$ and the MMSE-VBLAST detector yields L subchannels with capacities $\{R_k\}_{k=1}^L$ if and only if $\{R_k\}_{k=1}^L \prec_+ \mathbf{c}$, where majorization $\mathbf{x} \prec_+ \mathbf{y}$ is defined as follows [20]:

For $\mathbf{x}, \mathbf{y} \in \mathbf{R}^n$, if $\sum_{i=1}^{j} x_{[i]} \leq \sum_{i=1}^{j} y_{[i]}$, $1 \leq j \leq n$ with equality holding for $j = n$, where the subscript $[i]$ denotes the i^{th} largest element of the sequence, we say that $\mathbf{x}$ is majorized by $\mathbf{y}$ and denote it as $\mathbf{x} \prec_+ \mathbf{y}$ or, equivalently, $\mathbf{y} \succ_+ \mathbf{x}$.

3.2.2 TCD scheme under QoS constraints

Under various QoS constraints, Ref. [21] gives an optimization scheme to decompose a MIMO channel so that the minimal transmitted power can be achieved. This TCD scheme deals with two types of scenarios, described below.

The first type of scenario is where the majorization constraint is satisfied. Suppose $\{C_k^1\}_{k=1}^K$ are the subchannel capacities which are obtained by applying the SVD plus water filling to a rank K MIMO channel. Since the water filling method is capacity lossless, $\sum_{i=1}^K C_i^1$ will be equal to the MIMO channel capacity. We can use the TCD algorithm to convert these K subchannels into $L \geq K$ subchannels with capacities $R_1, R_2, ..., R_L$ if and only if $(C_1^1, ..., C_K^1, 0, ..., 0) \in \mathbf{R}_+^L$ majorizes $(R_1, R_2, ..., R_L)$. What's more, $\sum_{i=1}^K C_i^1 = \sum_{i=1}^L R_i$, i.e. under this scenario, the TCD scheme is capacity lossless.

The other type of scenario occurs if the required capacities don't satisfy the majorization constraint with $\{C_k^1\}_{k=1}^K$, i.e. they are too disparate. In this case, the TCD scheme will apply multi-level water filling to achieve the optimum power allocation. However, this will suffer from overall capacity loss. To resolve this issue, we can split the large-valued substreams into smaller ones so that the majorization constraint is satisfied.

3.3 MIMO Subchannel Allocation in SDR

3.3.1 Power minimization procedure

From the above discussion, we know that the TCD scheme in [21] already gives the minimum power needed to transmit the required services in a single MIMO system. When

the given rates $\{R_k\}_{k=1}^{L}$ satisfy the majorization constraint, the TCD algorithm can be applied. It is capacity-lossless and therefore automatically gives the minimum power needed to transmit the required data. On the other hand, when the majorization constraint is not satisfied, the TCD scheme still achieves minimal power by applying multi-level water filling. Thus, in this case, we only have to perform Step 1), as discussed in Section 3.1. That is, among several available frequency bands detected through spectrum sensing, decide which band or bands will be used. If more than one frequency band is to be used, i.e. the whole system is to be divided into several subsystems, we have to decide which of the services will be assigned to each subsystem. Once this has been done, applying the TCD scheme to each subsystem will suffice. In the following discussion, we consider an illustrative example in which the number of services to be transmitted is the same as the number of antenna pairs in the MIMO system, and where two frequency bands are available.

As shown in Fig. 3.1, we consider K pairs of transmitting/receiving antennas and K services to be transmitted, with required bit rates $\{B_k\}_{k=1}^{K}$, given in descending rate order. Suppose the two available spectrum slots are F_1 and F_2 having bandwidths W_1 and W_2, where $W_1 > W_2$. Their noise levels are α_1 and α_2, as defined in (2). Group the antennas into two subsystems so that each group operates in one spectrum slot. K_1 and K_2 are the numbers of antenna pairs assigned to F_1, F_2, respectively, where $K_1 + K_2 = K$.

Suppose $\mathbf{H}_1$ and $\mathbf{H}_2$ are the channel matrices of the resulting MIMO subsystems. Define $\mathbf{H}_{F_1}$ to be the channel matrix when all antenna pairs are assigned to F_1, and $\mathbf{H}_{F_2}$ to be the channel matrix when all antennas are put into F_2. In a flat fading environment, it is reasonable to assume that the channel does not change during a certain time period. During this period, $\mathbf{H}_1$ and $\mathbf{H}_2$ are just partitions of $\mathbf{H}_{F_1}$ and $\mathbf{H}_{F_2}$, since the transmitting coefficients for a given antenna pair remain unchanged in the same frequency band.

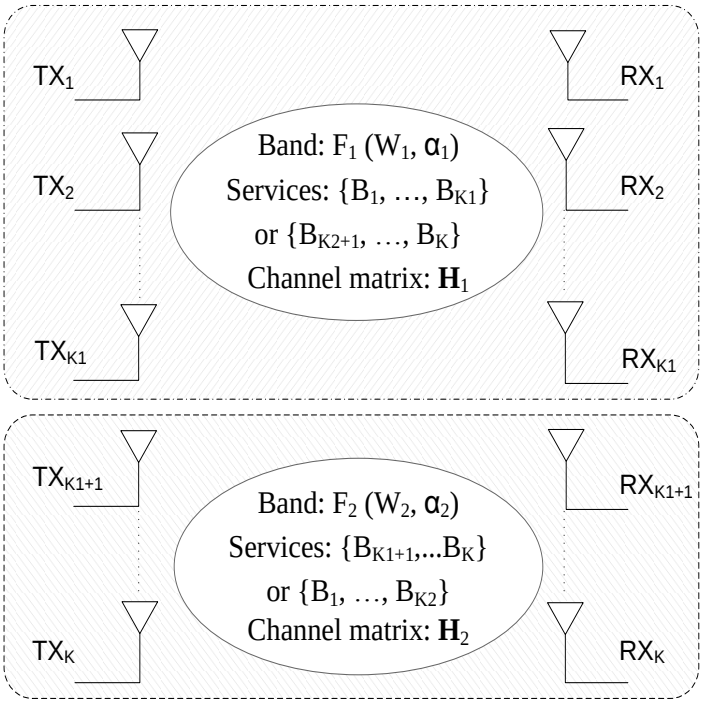


Fig. 3.1: Possible division of a MIMO system with *K* antenna pairs.

The channel capacities of the two subsystems are given as follows:

$$\begin{aligned} C_{1,k} &= \log_2(1+\frac{\lambda_{H_1,k}^2 \phi_{1,k}}{\alpha_1}), k = 1,...,K_1 \\ C_{2,k} &= \log_2(1+\frac{\lambda_{H_2,k}^2 \phi_{2,k}}{\alpha_2}), k = 1,...,K_2 \end{aligned}, \tag{3.3}$$

where $\{\lambda_{H_i,k}\}_{k=1}^{K_i}, i = 1,2$ are the singular values of $\mathbf{H}_1$ and $\mathbf{H}_2$, and where $\{\phi_{i,k}\}_{k=1}^{K_i}, i = 1,2$ are the power allocations for the two subsystems obtained through SVD and water filling. If the majorization constraint is satisfied for each subsystem, which means that the two subsystems are capacity-lossless, we have:

$$\sum_{k=1}^{K_1} C_{1,k} = \sum_{k=1}^{K_1} \frac{B_k}{W_1}, \sum_{k=1}^{K_2} C_{2,k} = \sum_{k=K_1+1}^{K} \frac{B_k}{W_2}, \tag{3.4}$$

or

$$\sum_{k=1}^{K_1} C_{1,k} = \sum_{k=K_2+1}^{K} \frac{B_k}{W_1}, \sum_{k=1}^{K_2} C_{2,k} = \sum_{k=1}^{K_2} \frac{B_k}{W_2}, \tag{3.5}$$

In (3.4), we place services that require larger data rates into F_1 (which will be referred to as "descending order" in the remainder of this chapter), while in (3.5) we put them into F_2 (which will be referred to as "ascending order"). In both cases we intuitively group services having adjacent data rates together, so that the data rates will be not too disparate from each other. In this way, the majorization constraint is more likely to be satisfied so as to achieve capacity-lossless transmissions.

Given $\mathbf{H}_1, \mathbf{H}_2$, we seek to find the K_1, K_2 pair and the descending/ascending order that minimize the total transmitting power (which, if the majorization constraint is

satisfied, will be equal to $\sum_{i=1}^{K_1}\phi_{1,i} + \sum_{i=1}^{K_2}\phi_{2,i}$). The channel matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ are generated randomly and simulations are done to obtain empirical solutions, as described in the following section.

3.3.2 Simulation Results

In our simulations, we use the bit rates required for several widely used applications, including video (128 kbps, 384 kbps), DAB (Digital Audio Broadcasting) service (192 kbps), standard audio (128–160 kbps), FM radio (96 kbps), AM radio (32 kbps), video phone (16 kbps) and telephone (8 kbps). In each trial, K + 1 subchannel division options (i.e., ranging from $K_1 = 0$ to $K_1 = K$) were simulated. Two K by K random matrices were generated as $\mathbf{H}_{F_1}$ and $\mathbf{H}_{F_2}$ for each trial; then, in each division, the corresponding partitions of them were assigned to be $\mathbf{H}_1$ and $\mathbf{H}_2$. Both descending order and ascending order were simulated in every division and the one having the smaller total power was retained. Finally, the particular division which required the minimal total power was selected for that trial.

In Table 3.1 (simulation case 1), we use the values K = 8, W_1 = 30 kHz, W_2 = 10 kHz, and B = [384 128 96 32 16 8 8 8] kbps. 100,000 trials were carried out for each pair of chosen noise levels α_1 and α_2, where α_1 and α_2 are in dB. From the results we can see that, in most cases, the lowest power is obtained when all of the services are assigned to a single band. When α_1/α_2 is 10, we have a maximum of 0.072% of the cases where it is

necessary to divide the services between the two frequency bands. We also observe that when α_1/α_2 is 1 or 2, in most of the cases, all services are assigned to F_1 . This is understandable since F_1 has three times as much bandwidth as F_2 , thus requiring a lower SNR to attain the required throughput. However, when α_1/α_2 is 3, about half of the realizations suggest that all of the services should be assigned to F_2 . When α_1/α_2 is larger than 3, F_2 becomes the more favorable choice.

In Table 3.2 (case 2), we randomly draw bit-rates out of a pool of [32 96 128 160 192 8 16 384 64] kbps, while the other parameters remain the same. For each α_1/α_2, 100 realizations of *B* are generated, each with 1000 channel trials. The results show increasing necessity to divide the services, with a maximum of 1.27% when α_1/α_2 is 10. Table 3.3 (case 3) shows the result for the same simulation conditions as for Table 3.2 except that *K* = 4. The maximum percentage is 5.98% when α_1/α_2 = 9, which is a considerable increase compared to *K* = 8. Table 3.4 (case 4) changes the bandwidths to W_1 = 300 kHz and W_2 = 100 kHz. Also for each realization of *B*, 10000 channels are simulated rather than 1000, while other parameters are the same as in Table 3.3. The maximum percentage is 6.21% when α_1/α_2 = 10.

Table 3.1: Distribution of K_1, simulation case 1

	K_1								
α_1/α_2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0	0	0	0	0	0	0	1	99999
2	0	0	0	0	0	0	0	0	100000
3	50109	1	0	0	0	0	0	0	49890
4	99998	2	0	0	0	0	0	0	0
5	99993	7	0	0	0	0	0	0	0
6	99989	10	1	0	0	0	0	0	0
7	99964	30	6	0	0	0	0	0	0
8	99957	36	6	1	0	0	0	0	0
9	99930	56	13	1	0	0	0	0	0
10	99928	56	11	2	3	0	0	0	0

Table 3.2: Distribution of K_1, simulation case 2

	K_1								
α_1/α_2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	0	0	0	0	0	0	0	0	100000
2	0	0	0	0	0	0	0	0	100000
3	36191	7	0	1	0	0	0	10	63791
4	95747	91	17	1	1	1	4	24	4114
5	97931	184	51	10	7	3	6	31	1777
6	97310	317	101	42	22	11	12	55	2130
7	99003	483	195	81	22	27	10	21	158
8	98743	589	249	87	33	28	11	6	254
9	98748	706	332	125	33	11	4	1	40
10	98726	761	332	123	33	19	6	0	0

Table 3.3: Distribution of K_1, simulation case 3

	K_1				
α_1/α_2	**0**	**1**	**2**	**3**	**4**
1	1	0	2	25	99972
2	203	53	66	263	99415
3	36590	611	320	780	61699
4	80768	1184	619	1022	16407
5	89365	1567	661	791	7616
6	88486	2363	1193	1086	6872
7	88515	2805	1464	967	6249
8	89646	2594	1665	866	5229
9	93030	3279	1876	826	989
10	91130	3231	1818	686	3135

Table 3.4: Distribution of K_1, simulation case 4

	K_1				
α_1/α_2	**0**	**1**	**2**	**3**	**4**
1	1111	4	26	25	998834
2	2302	34	57	261	997346
3	6042	205	347	1087	992319
4	8797	432	921	2324	987526
5	13284	661	1614	2536	981905
6	11378	753	2108	3613	982148
7	6429	701	1895	2840	988135
8	15778	1410	4247	4792	973773
9	14065	1134	3057	4425	977319
10	19074	2421	4940	5151	968414

To ensure the optimal division among the K+1 choices, the following procedure should be adopted: first, estimate $\mathbf{H}_{F_1}$ and $\mathbf{H}_{F_2}$; next, for each division, obtain $\mathbf{H}_1, \mathbf{H}_2$ from $\mathbf{H}_{F_1}, \mathbf{H}_{F_2}$ and compute the corresponding total transmitting power; finally, select the optimal division and assign the services accordingly. This procedure should be repeated periodically to track the current channel conditions. However, from Tables 3.1-3.4 we can see that the simulations suggest that for most cases (~95%) it is a "winner-take-all" situation. Thus, if the complexity of the above procedure is too high for a given system implementation, rather than considering all possible divisions, we can simply compare $K_1 = 0$ with $K_1 = K$. However, in this case the system may suffer from capacity loss.

3.3.3 Division with power constraints on frequency bands

From the simulations with two frequency bands and a specific set of parameters, the results suggest a "winner-take-all" solution in most cases. We also presented a procedure for choosing the optimal division among the available choices; however, this may not be an attractive procedure given its complexity. From the power efficient point of view, it may be reasonable to simply put all the antennas into one frequency band, i.e. whichever one requires a lower total transmitting power for all of the required services.

However, recall that we are using cognitive radio and there may be strict regulations on the power level that can be put into a given frequency band in order to avoid interfering with the licensed or primary users. This could lead to the need to divide the antennas

among the available frequency bands. A simple asymptotic analysis can be performed using equation (1.2). Suppose that there are two frequency bands F_1 and F_2 having the same bandwidth, with power constraints of P_1 and P_2, respectively. Assuming unit noise variance and asymptotically high SNR, the total capacity that can be achieved subject to K_1 (the number of antenna pairs put onto F_1) is up to:

$$C = K_1 \log(\frac{P_1}{K_1}) + (K - K_1)\log(\frac{P_2}{K - K_1}) + O(1) \tag{3.6}$$

In order to maximize C according to K_1, we have:

$$\frac{\partial C}{\partial K_1} = \log P_1 - \log P_2 - \log K_1 + \log(K - K_1) = 0 \tag{3.7}$$

This leads to

$$\frac{P_1}{P_2} = \frac{K_1}{K - K_1} \tag{3.8}$$

This suggests that the number of antenna pairs that should put into the two frequency bands is proportional to the corresponding power constraints.

For example, ignoring the term $O(1)$ and assuming P_1 =30, P_2 = 50, K = 8, we obtain the curve shown in Fig. 3.2, which suggests that we should put three out of eight antenna pairs onto F_1.

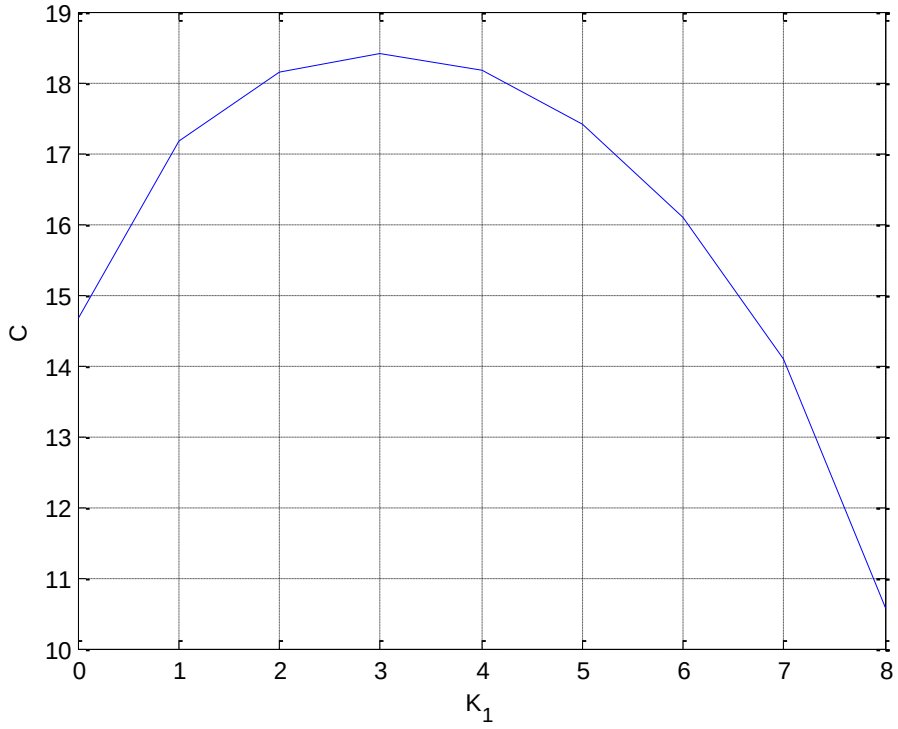


Fig. 3.2: C versus K_1 for two frequency bands with power constraints

3.4 Reconfigurable MIMO Transceiver Design

From the above discussion, we have seen that in some cases it may be more favorable to only use one frequency band (i.e., running as a single MIMO system), while in other cases (especially when there is power constraint on certain frequency bands) division into two subsystems is more desirable. This means that a flexible MIMO transceiver will be needed which can be reconfigured so that it can operate as either one system or as two subsystems. In this section we will describe a reconfigurable 4×4 16-QAM MIMO transceiver design based on the TCD. It can be reconfigured as one 4×4 system or as two 2×2 subsystems.

Fig. 3.3 shows the structure of a single TCD MIMO system. Suppose that the channel is slowly time-varying, so the TCD channel decomposition computation can be implemented in software. The outputs of this step are two matrices, **F** and **W**, where **F** is to be used at the precoder, and where the columns of **W**, which are denoted as $\mathbf{w}_i$ (i = 1, 2, …, L), are the nulling vectors to be used at the receiver.

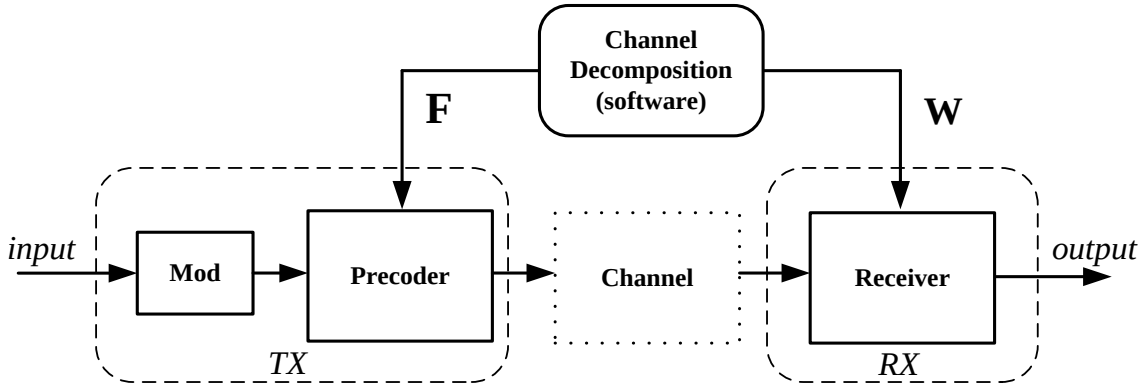


Fig. 3.3: Single TCD system structure

3.4.1 Precoder at the transmitter

The transmitter precodes the symbol vector x with F, generating the transmitted signal **s** = **Fx**, as depicted in Fig. 3.4.

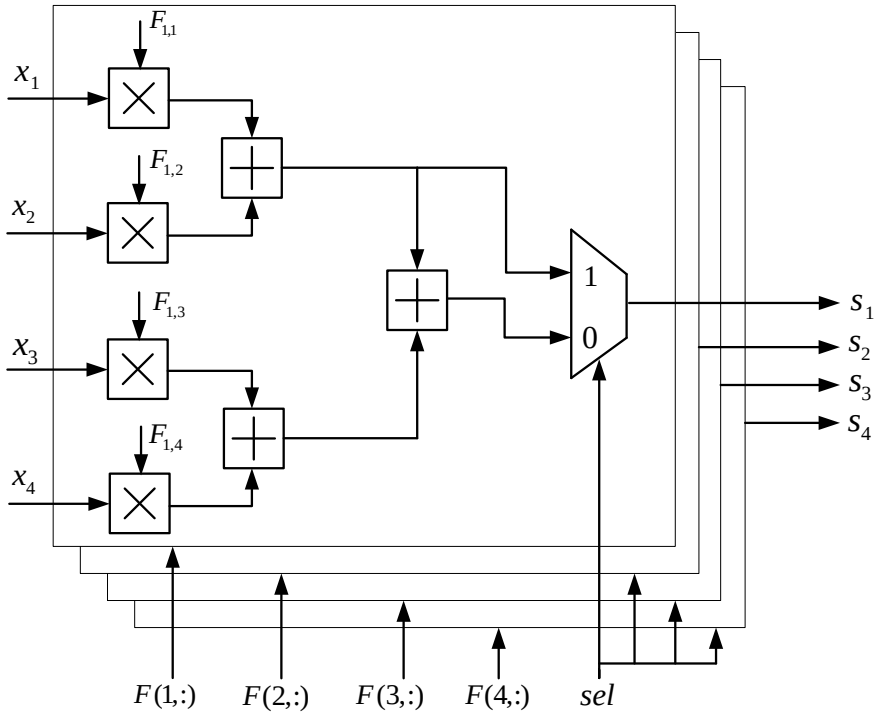


Fig. 3.4: Reconfigurable precoder structure of 4×4 MIMO

As can be seen, there are four layers in the design, each generating a precoded symbol. The *sel* signal decides whether the design is operating as a single 4×4 system or as two 2×2 subsystems. When *sel* is 0, it operates as a 4×4 system, performing a matrix multiplication where the 4×1 vector x is multiplied by the 4×4 matrix **F**. When *sel* is 1, it operates as two 2×2 subsystems, performing two matrix multiplications where 2×1 symbol vectors are multiplied by 2×2 precoding matrices. In this latter case, x_1 and x_2 are the symbols from the first subsystem and x_3 and x_4 are the symbols from the second subsystem.

We use 16-QAM modulation in our design, so every four bits are grouped together to form a symbol, where two of them represent the real part and the other two represent the imaginary part. This allows a simplification of the multiplier design in this block.

Using Gray coding, 00 indicates -3, 01 represents -1, 11 is +1 and 10 is +3. We notice that if the 0^{th} bit is 0, the absolute value is 3; otherwise, the absolute value is 1. Also, if the 1^{st} bit is 0, then the value is negative; otherwise, it is positive. Thus the multiplier here can be designed as shown in Fig. 3.5, which is simpler than a general multiplier.

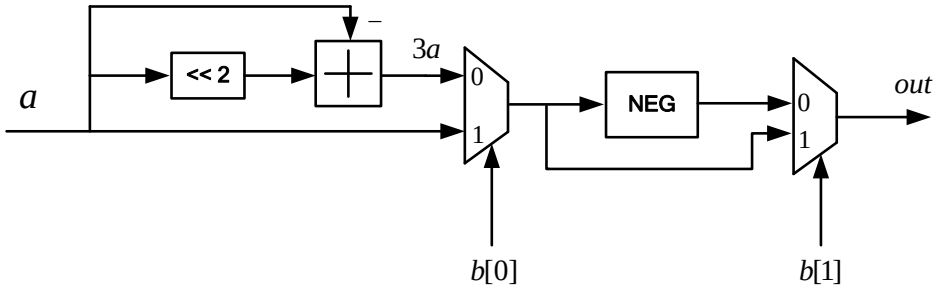


Fig. 3.5: Multiplier in precoder

3.4.2 Receiver

After passing through the channel, the received signal is $\mathbf{y} = \mathbf{HFx} + \mathbf{z}$. As mentioned at the beginning of this section, $\mathbf{w}_i$ $(i = 1, 2, ..., L)$ are the nulling vectors input to the MMSE-VBLAST decoder. We perform a sequential nulling and canceling process in the decoder which is very similar to the one used in UCD (see Section 2. 3).

This process can be summarized as follows:

$$\begin{aligned} &\text{for } i = L:-1:1 \\ &\hat{x}_i = C\left[\mathbf{y}[L-i] \times \mathbf{w}_i{}'\right] \\ &\mathbf{y}[L-i+1] = \mathbf{y}[L-i] - \tilde{\mathbf{H}}_{:,i} \times \hat{x}_i \\ &\text{end} \end{aligned} \tag{3.9}$$

Based on this process, the reconfigurable MIMO receiver design is shown in Fig. 3.6. When *sel* is 0, it performs uninterrupted 3-stage sequential nulling and canceling in

order to decode all four symbols from the received signal vector $\mathbf{y}$. When *sel* is 1, the received signal of the first and the second subsystems will be $\mathbf{y}$ and $\mathbf{y}'$, respectively. In this case, the second canceling stage is not used since now the two subsystems are decoding independently.

Note that we need to use standard, full-width multipliers for multiplying $\mathbf{y}$ by $\mathbf{w}_i$,. We have used AccelDSP [39] to design these multipliers, which can generate optimized structures from Matlab code targeting the specific FPGA device. For multiplying an estimated symbol with a column of $\tilde{\mathbf{H}}$, however, essentially the same multiplier as in the precoder design is used.

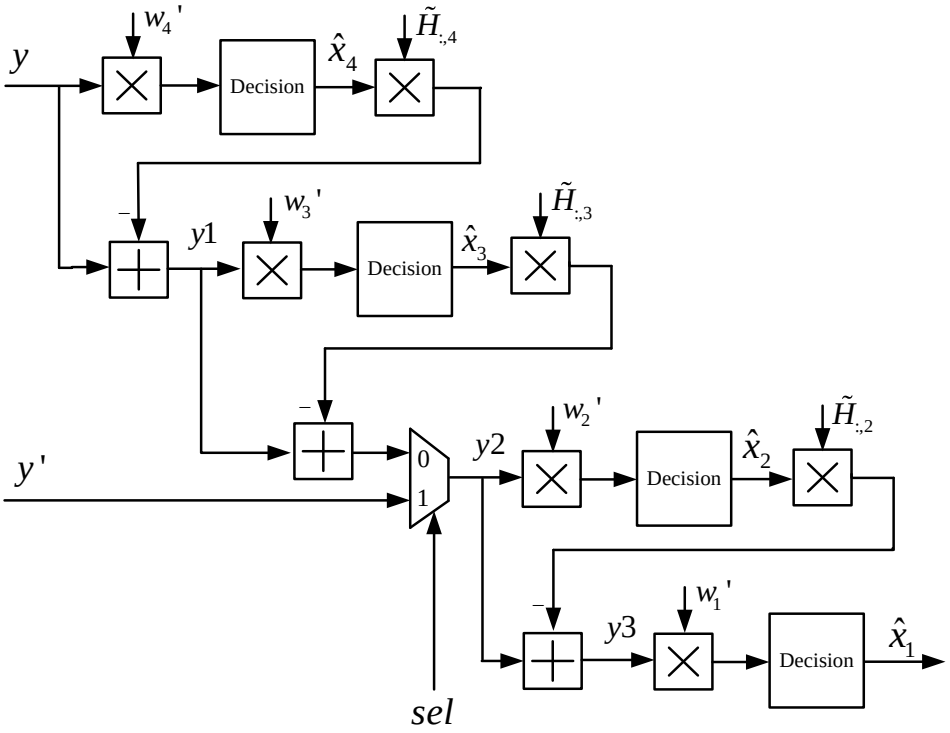


Fig. 3.6: Reconfigurable receiver structure of 4 ×4 MIMO

3.4.3 Results

The Verilog design has been synthesized and mapped onto a Xilinx xc4vlx200ff1513-11 Virtex-4 FPGA.

We use Matlab to calculate the matrices **F** and **W**, which are then read into the design. We assume that all the channel taps are zero mean, unit variance, complex Gaussian random variables. Through Matlab fixed-point simulations, we have found that a total bit-width of 15 with a 10-bit fractional part gave almost the same performance as floating-point, while any further reduction leads to noticeable performance degradation. Thus, we have used this bit-width throughout our design.

The area results are presented in Table 3.5. We can see that the reconfigurable designs are almost the same in area as the non-configurable designs. This is due to the fact that only a small amount of control logic is needed to make them reconfigurable. This indicates that the TCD scheme is well-suited for use in a flexible transceiver setting.

The speed results are listed in Table 3.6. Using the relation $\text{Throughput} = f \times \log_2 \text{M} \times K$, where M = 16 is the constellation size, we can calculate the throughput as being approximately 1.6 Gbps.

Several previous MIMO receiver designs have been based on Sphere Decoding [31] or Fixed Sphere Decoding [32] algorithms. Our design requires about the same area (in terms of the number of FPGA slices), including both the precoder and the receiver, as in those designs while doubling the throughput. However, since our design is a closed-loop one, some additional amount of overhead for the feedback of CSI will also be required.

Table 3.5: FPGA Area Results

TX/RX	Design	Area (slices)	Area increase (%)
Precoder	Non-configurable	2288	4.20
	Reconfigurable	2384	
Receiver	Non-configurable	2868 + 48 multipliers (about 9156 equivalent)	3.45
	Reconfigurable	2967 + 48 multipliers (about 9255 equivalent)	

Table 3.6: FPGA Speed results

	Max Frequency
Precoder	113MHz
Receiver	100.4MHz

3.5 Conclusions

We have combined cognitive radio and MIMO techniques to form a flexible system for supporting disparate services in a scenario where spectrum resources are scarce. Given different available frequency bands, we have shown that it is sometimes necessary to divide the system into separate subsystems, particularly when some of these bands are subject to power constraints. We then can apply the TCD to each subsystem to accommodate the required services with minimum power. We have presented a

reconfigurable 16-QAM MIMO transceiver design based on the TCD which can operate as either one 4×4 system or as two 2×2 subsystems. This transceiver has approximately the same area as its non-configurable counterpart, and it can achieve a maximum throughput of 1.6 Gbps.

Chapter 4

Fast MIMO MMSE-FDE

In the previous chapters we focused on MIMO systems under ISI-free channels only. However, this assumption is likely to be invalid nowadays since the high throughput of the system is usually larger than the channel coherence bandwidth, which leads to ISI. In recent years, single-carrier (SC) communication for ISI channels has enjoyed a revived interest, whose principle advantage over the multi-carrier scheme is lower peak-to-average power ratio (PAPR). In this chapter, we propose a new fast iterative algorithm to obtain the optimal finite impulse response (FIR) minimum-mean-square-error decision feedback equalizer (MMSE-DFE) for multiple-input multiple-output (MIMO) SC communication systems over ISI channels based on the QR decomposition of an augmented Toeplitz channel matrix. The proposed algorithm is not only computationally efficient but also very frugal in memory usage. The algorithm applies to MIMO systems with any number of transmitting and receiving antennas, either balanced or unbalanced. The iterative feature of the proposed algorithm enables a flexible choice of N_f, the length of the feedforward filter (FFF), which translates to controllable tradeoff between complexity and performance for real implementation. We also propose a conversion from

our time-domain equalization (TDE) to hybrid DFE (HDFE) for lower complexity when N_f becomes large. Complexity analysis suggests significant improvement over the prior art on TDE.

4.1 Introduction

As discussed in Chapter 1, when the MIMO system throughput is larger than the channel coherence bandwidth, there will be inter-symbol interference, i.e. ISI. In order to mitigate the ISI effect, MIMO has been often paired with multi-carrier transmitting schemes such as orthogonal frequency division multiplexing (OFDM). However, recently single carrier solutions have been receiving increasing attention due to several advantages compared to OFDM, including reduced peak-to-average power ratio (PAPR), possible lower transmitter complexity, and more robustness against carrier frequency offset [40] [41]. In particular, the 3rd generation partnership project (3GPP) has adopted single carrier frequency division multiple access (SC-FDMA) [42] [43] [44] [45] as the solution for the long term evolution (LTE) uplink transmission.

Compared to single carrier frequency-domain equalization (SC-FDE) for MIMO systems [46] [47] [48], not as much effort has been made recently for the MIMO TDE counterpart. One of the reasons why the FDE has drawn more attention than TDE is that the complexity of the FDE is lower than TDE under channels whose impulse responses span a large number of symbol intervals. However, in the small-cell scenarios such as the femtocell environment [49] [50], the channel length is typically much shorter, thus TDE may become more viable.

Among the available MIMO TDE methods, MMSE-DFE offers a good tradeoff between performance and complexity. MIMO MMSE-DFE has been investigated in the literature, including [51] [52] [53] [54]. In particular, [52] is a pioneering work which not only derives the mathematical formula of the MIMO FIR MMSE-DFE, but also provides a fast algorithm based on the generalized Schur algorithm [55]. Limited work has been done on further improving the algorithm complexity over [52]. For example, [56] reduces the complexity by formulating the MIMO MMSE-DFE as a linear estimation problem. However, it suffers from lower post-processing SNR as it only feeds back decisions from previous times but not the decisions of other users/streams at the current time.

In this chapter, we aim to achieve the same FIR MMSE-DFE as that in [52] but with much reduced computational complexity. We first prove that the optimal FIR coefficient matrices can actually be obtained through QR decomposition of an augmented channel matrix. By exploiting the Toeplitz structure of the augmented channel matrix, we propose a fast algorithm to obtain the desired sub-matrices in the QR decomposition. We then give a unifying iterative method to realize the fast algorithm which applies to all the three cases where the number of transmitting antennas is equal to, smaller than, or larger than the number of receiving ones. The proposed iterative algorithm leads to a significant reduction in both computational complexity and memory usage. The complexity analysis suggests remarkable improvement over the prior art in [52]. Note that the QR decomposition representation of MMSE-DFE has been proposed for the MIMO flat-fading channel [20] and for the SISO ISI channel [58] [59]. However, no similar method has been formulated for the case of the MIMO ISI channel.

Moreover, during the iteration process, we can monitor the achievable system throughput as the FFF length N_f of the MMSE-DFE increases. By relating the post-processing SNR of the MMSE-DFE to the Shannon capacity of the underlying MIMO ISI channel, we can determine when to terminate the iterative procedure to achieve a desired tradeoff between complexity and performance.

Furthermore, if in a large cell the required performance demands too large a value of N_f (i.e., too high complexity) for TDE, we propose a conversion from TDE to hybrid DFE (HDFE), where the feed-forward part of the equalizer is performed in frequency domain. This flexibility is very desirable in the system implementation of the heterogeneous network where different types of cells, e.g., femtocells versus macrocells, have very different channel length and thus require different FIR lengths.

The remainder of this chapter is organized as follows: Section 4.2 starts from the MIMO ISI data model and presents the derivation of FIR MIMO MMSE-DFE based on the QR decomposition, then shows that the DFE approaches the channel capacity as N_f increases towards infinity. In Section 4.3, the fast algorithm is described, and its iterative implementation is investigated. A conversion from TDE to HDFE is also proposed. Section 4.4 gives the complexity analysis and performance comparison results, while Section 4.5 provides a numerical example and some simulation results. The chapter is concluded in Section 4.6.

4.2 Derivation

In this section, we present a derivation of the FIR MIMO MMSE-DFE, including the FFF and FBF coefficient matrices, based on a QR decomposition. It is then shown that the DFE approaches the channel capacity as the FFF length N_f increases. We begin with the data model of a MIMO ISI channel.

4.2.1 Data Model

Consider a MIMO communication system with M_t transmit and M_r receive antennas in an ISI channel with v multipath taps. Assuming symbol-spaced output, the discrete-time input-output relationship at symbol time t can be modeled as

$$\mathbf{y}_t = \sum_{k=0}^{v} \mathbf{H}_k \mathbf{x}_{t-k} + \mathbf{z}_t \tag{4.1}$$

where the information symbol sequence $\{\mathbf{x}_t\}$ is assumed to be an i.i.d. M_t-dimensional vector random process with zero mean and unit variance, whose covariance matrix $E[\mathbf{x}\mathbf{x}^*] = \mathbf{I}_{M_t}$, where $(\bullet)^*$ is the complex-conjugate transpose. The received sequence $\{\mathbf{y}_t\}$ is an M_r-dimensional vector random process. Channel coefficient matrices $\{\mathbf{H}_k\}$ are of size $M_r \times M_t$, whose $(m, n)^{\text{th}}$ element indicate the channel impulse response between the m^{th} receive antenna and the n^{th} transmit antenna. The additive noise term $\{\mathbf{z}_t\}$ is a stationary M_r-dimensional vector Gaussian process that is white in both the spatial and frequency domain, i.e. with auto-correlation function $E[\mathbf{z}_t \mathbf{z}_{t-n}^*] = \delta_n \sigma_z^2 \mathbf{I}_{M_r}$, where δ_n is the Kronecker delta.

For a block of N_f output symbols, the input-output relationship (4.1) can be rewritten in block matrix form as follows:

$$\begin{pmatrix} \mathbf{y}_{t+N_f-1} \\ \vdots \\ \mathbf{y}_{t+1} \\ \mathbf{y}_t \end{pmatrix} = \begin{pmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_\nu & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_\nu & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_\nu \end{pmatrix} \times \begin{pmatrix} \mathbf{x}_{t+N_f-1} \\ \vdots \\ \mathbf{x}_t \\ \vdots \\ \mathbf{x}_{t-\nu} \end{pmatrix} + \begin{pmatrix} \mathbf{z}_{t+N_f-1} \\ \vdots \\ \mathbf{z}_{t+1} \\ \mathbf{z}_t \end{pmatrix} \tag{4.2}$$

We can write the above equation in the compact form:

$$\overline{\mathbf{y}} = \overline{\mathbf{H}}\overline{\mathbf{x}} + \overline{\mathbf{z}} \tag{4.3}$$

At the receiver end, we apply a finite-length MMSE-DFE to combat ISI and recover the information vector sequences $\{\mathbf{x}_t\}$, as shown in Fig. 4.1. Suppose the FIR MMSE-DFE consists of a feed-forward filter (FFF) with N_f taps $\left\{\mathbf{W}_i^* \in C^{M_t \times M_r}\right\}, i = 0, 1, ..., N_f - 1$ and a feed-backward filter (FBF) with N_b+1 taps $\left\{\mathbf{B}_j \in C^{M_t \times M_t}\right\}$, $j = 0, 1, ..., N_b$. We assume that within the vector $\mathbf{x}_t$, we detect the entries sequentially from the M_t^{th} entry to the 1st entry, so that $\mathbf{B}_0$ should be strictly upper triangular. The optimal N_b should be ν since $\mathbf{x}_t$ is only subject to the interference from the preceding ν symbols. We will use $N_b = \nu$ in the remainder of the chapter.

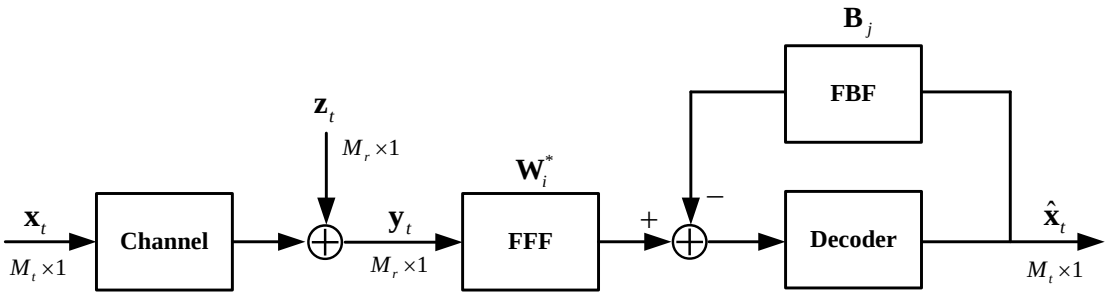


Fig. 4.1: Block diagram of MIMO MMSE-DFE

Denote

$$\bar{\mathbf{W}} = [\mathbf{W}_{N_f-1}^*, ..., \mathbf{W}_0^*]^*, \bar{\mathbf{B}} = [\mathbf{0}_{M_t \times (N_f-1)M_t}, \mathbf{I} + \mathbf{B}_0, \mathbf{B}_1, ..., \mathbf{B}_\nu] \tag{4.4}$$

With the simplifying assumption that all the previous symbols are detected and demodulated correctly, the error vector at time *t* is then

$$\mathbf{e}_t = \hat{\mathbf{x}}_t - \mathbf{x}_t = \bar{\mathbf{W}}^* \bar{\mathbf{y}} - \bar{\mathbf{B}} \bar{\mathbf{x}} \tag{4.5}$$

The MSE matrix is defined as the auto-correlation matrix of the error vector

$$\mathbf{E} = E[\mathbf{e}_t \mathbf{e}_t^*] \tag{4.6}$$

For MIMO MMSE-DFE, the error matrix is "compressed" through optimizing the FFF and FBF coefficient matrices. In the following, we will describe how to obtain the optimal FFF and FBF through QR decomposition. derivation below was originally sketched in [60].

4.2.2 Optimal FFF and FBF based on QR Decomposition

Combining (4.3) and (4.5), we can write (4.6) as

$$\begin{aligned}\mathbf{E} &= E\left[(\bar{\mathbf{W}}^*\bar{\mathbf{y}} - \bar{\mathbf{B}}\bar{\mathbf{x}})(\bar{\mathbf{W}}^*\bar{\mathbf{y}} - \bar{\mathbf{B}}\bar{\mathbf{x}})^*\right] \\ &= \left(\bar{\mathbf{W}}^*\bar{\mathbf{H}} - \bar{\mathbf{B}}\right)\left(\bar{\mathbf{W}}^*\bar{\mathbf{H}} - \bar{\mathbf{B}}\right)^* + \bar{\mathbf{W}}^*\mathbf{R}_z\bar{\mathbf{W}} \\ &= \left(\bar{\mathbf{W}} - \left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\bar{\mathbf{B}}^*\right)^*\left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right) \\ &\quad \cdot\left(\bar{\mathbf{W}} - \left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\bar{\mathbf{B}}^*\right) + \\ &\quad \bar{\mathbf{B}}\left(\mathbf{I} - \bar{\mathbf{H}}^*\left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\right)\bar{\mathbf{B}}^*\end{aligned} \tag{4.7}$$

where $\mathbf{R}_z = E[\bar{\mathbf{z}}\bar{\mathbf{z}}^*] = \sigma_z^2\mathbf{I}_{N_f M_r \times N_f M_r}$. We have

$$\mathbf{E} \geq \bar{\mathbf{B}}\left(\mathbf{I} - \bar{\mathbf{H}}^*\left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\right)\bar{\mathbf{B}}^* \tag{4.8}$$

where the equality holds if and only if

$$\bar{\mathbf{W}} = \left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\bar{\mathbf{B}}^*. \tag{4.9}$$

With such a feed-forward filter matrix, the MSE matrix becomes

$$\begin{aligned}\mathbf{E} &= \bar{\mathbf{B}}\left(\mathbf{I} - \bar{\mathbf{H}}^*\left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\right)\bar{\mathbf{B}}^* \\ &= \bar{\mathbf{B}}\left(\mathbf{I} + \bar{\mathbf{H}}^*\mathbf{R}_z^{-1}\bar{\mathbf{H}}\right)^{-1}\bar{\mathbf{B}}^*\end{aligned} \tag{4.10}$$

where we have used the matrix inversion lemma.

Consider the Cholesky factorization

$$\mathbf{I} + \bar{\mathbf{H}}^*\mathbf{R}_z^{-1}\bar{\mathbf{H}} = \bar{\mathbf{L}}^*\bar{\mathbf{L}} \tag{4.11}$$

where $\bar{\mathbf{L}}$ is an upper triangular matrix with positive diagonal elements. Then

$$\mathbf{E} = \bar{\mathbf{B}}\bar{\mathbf{L}}^{-1}\bar{\mathbf{L}}^{-*}\bar{\mathbf{B}}^{*}. \tag{4.12}$$

Since $\bar{\mathbf{L}}^{-1}$ is upper triangular, it is easy to see that

$$\bar{\mathbf{B}}\bar{\mathbf{L}}^{-1} = \left[\mathbf{0}_{M_t\times(N_f-1)M_t}, \left(\mathbf{I}+\mathbf{B}_0\right)\mathbf{L}_{N_f}, \mathbf{M}_1, \mathbf{M}_2, ..., \mathbf{M}_\nu\right] \tag{4.13}$$

where $\mathbf{L}_{N_f} \in C^{M_t\times M_t}$ is the N_f^{th} diagonal block of $\bar{\mathbf{L}}^{-1}$, and {$\mathbf{M}_1$, $\mathbf{M}_2$, …, $\mathbf{M}_\nu$} are the matrix results generated by multiplying $\bar{\mathbf{B}}$ with the last *v* block columns of $\bar{\mathbf{L}}^{-1}$.

Now consider the QR decomposition of the following augmented matrix

$$\begin{bmatrix}\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\\ \mathbf{I}\end{bmatrix} = \bar{\mathbf{Q}}\bar{\mathbf{R}}$$

$$= \left(\begin{array}{ccccccc} \square & \cdots & \mathbf{Q}_{N_f-1} & \cdots & \square & \cdots & \square \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \square & \cdots & \mathbf{Q}_1 & \cdots & \square & \cdots & \square \\ \square & \cdots & \mathbf{Q}_0 & \cdots & \square & \cdots & \square \\ \hline \square & \cdots & \square & \cdots & \square & \cdots & \square \\ & \ddots & \vdots & \ddots & & & \vdots \\ & & \square & & \ddots & & \vdots \\ & & & \ddots & & \ddots & \vdots \\ & & & & \square & \cdots & \square \end{array}\right) \left(\begin{array}{ccccccc} \square & \square & \cdots & \square & & & \\ & \square & \cdots & \square & \square & & \\ & & \ddots & \vdots & \vdots & \ddots & \\ & & & \mathbf{R}_0 & \mathbf{R}_1 & \cdots & \mathbf{R}_\nu \\ & & & & \ddots & \ddots & \vdots \\ & & & & & \square & \square \\ & & & & & & \square \end{array}\right) \tag{4.14}$$

where $\bar{\mathbf{Q}}$ is a square matrix of size $N_fM_r + (N_f + \nu)M_t$, and $\bar{\mathbf{R}}$ is of size $(N_fM_r + (N_f + \nu)M_t)(N_f + \nu)M_t$. The block matrix elements (denoted by squares) in $\bar{\mathbf{Q}}_{1:N_fM_r,\,1:(N_f+\nu)M_t}$ are of size $M_r \times M_t$, while those in $\bar{\mathbf{Q}}_{N_fM_r:end,\,1:(N_f+\nu)M_t}$ and in $\bar{\mathbf{R}}$ are of size $M_t \times M_t$, and those in $\bar{\mathbf{Q}}_{:,\,(N_f+\nu)M_t:end}$ are of size $M_r \times M_r$. $\left\{\mathbf{Q}_0, \mathbf{Q}_1, ..., \mathbf{Q}_{N_f-1}\right\}$ are within the N_f^{th} block column of $\bar{\mathbf{Q}}$, while $\left\{\mathbf{R}_0, \mathbf{R}_1, ..., \mathbf{R}_\nu\right\}$ are within the N_f^{th} block row of $\bar{\mathbf{R}}$.

Now consider

$$\begin{bmatrix} \mathbf{R}_z^{-1/2}\bar{\mathbf{H}} \\ \mathbf{I} \end{bmatrix}^* \begin{bmatrix} \mathbf{R}_z^{-1/2}\bar{\mathbf{H}} \\ \mathbf{I} \end{bmatrix} = \bar{\mathbf{R}}^*\bar{\mathbf{Q}}^*\bar{\mathbf{Q}}\bar{\mathbf{R}} = \bar{\mathbf{R}}^*\bar{\mathbf{R}} \tag{4.15}$$

Comparing with (4.11), we have $\bar{\mathbf{L}} = \bar{\mathbf{R}}$, which indicates that $\mathbf{L}_{N_f}^{-1} = \mathbf{R}_0^{-1}$. So, combining (4.12) and (4.13) we have

$$\mathbf{E} \geq (\mathbf{I} + \mathbf{B}_0)\mathbf{R}_0^{-1}\mathbf{R}_0^{-*}(\mathbf{I} + \mathbf{B}_0)^* \tag{4.16}$$

where the equality holds if and only if the $\{\mathbf{M}_i\}$ in (4.13) are zero matrices.

Recall that $\mathbf{B}_0$ is strictly upper triangular, so that $(\mathbf{I} + \mathbf{B}_0)\mathbf{R}_0^{-1}$ is an upper triangular matrix with diagonal $\mathbf{D}_{\mathbf{R}_0}^{-1}$, where $\mathbf{D}_{\mathbf{R}_0}$ stands for the diagonal matrix that shares the same diagonal elements with $\mathbf{R}_0$. Hence

$$[\mathbf{E}]_{ii} \geq [\mathbf{R}_0]_{ii}^{-2} \tag{4.17}$$

where $[\bullet]_{ii}$ indicates the i^{th} diagonal element, $i = 1, \ldots, M_t$.

On the other hand, observe that if we choose

$$\bar{\mathbf{B}} = \mathbf{A} \cdot [\mathbf{0}_{M_t \times (N_f - 1)M_r}, \mathbf{R}_0, \mathbf{R}_1, ..., \mathbf{R}_\nu] \tag{4.18}$$

where $\mathbf{A}$ is an arbitrary non-zero matrix of size $M_t \times M_t$, then $\{\mathbf{M}_i\}$ indeed are zero matrices since $\bar{\mathbf{L}}\bar{\mathbf{L}}^{-1} = \mathbf{I}$. Moreover, $\mathbf{A}$ has to be $\mathbf{D}_{\mathbf{R}_0}^{-1}$ since $\mathbf{A}\mathbf{R}_0 = \mathbf{I} + \mathbf{B}_0$. Then we have

$$\mathbf{E} = \mathbf{D}_{\mathbf{R}_0}^{-1}\mathbf{R}_0\mathbf{R}_0^{-1}\mathbf{R}_0^{-*}\left(\mathbf{D}_{\mathbf{R}_0}^{-1}\mathbf{R}_0\right)^* = \mathbf{D}_{\mathbf{R}_0}^{-2} \tag{4.19}$$

which achieves the equality of (4.17), and the MSE's of the substreams are minimized. Thus, the optimal FBF matrices are

$$\mathbf{B}_i = \begin{cases} \mathbf{D}_{\mathbf{R}_0}^{-1}\mathbf{R}_0 - \mathbf{I}, & i = 0 \\ \mathbf{D}_{\mathbf{R}_0}^{-1}\mathbf{R}_i, & 1 \le i \le \nu \end{cases} \tag{4.20}$$

Given $\bar{\mathbf{B}}$, we can solve (4.9) as

$$\begin{aligned} \bar{\mathbf{W}} &= \left(\bar{\mathbf{H}}\bar{\mathbf{H}}^* + \mathbf{R}_z\right)^{-1}\bar{\mathbf{H}}\bar{\mathbf{B}}^* \\ &= \mathbf{R}_z^{-1/2}\cdot\left(\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\bar{\mathbf{H}}^*\mathbf{R}_z^{-1/2} + \mathbf{I}\right)^{-1}\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\cdot\bar{\mathbf{B}}^* \\ &= \mathbf{R}_z^{-1/2}\cdot\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\left(\bar{\mathbf{H}}\mathbf{R}_z^{-1}\bar{\mathbf{H}}^* + \mathbf{I}\right)^{-1}\cdot\bar{\mathbf{B}}^* \\ &= \mathbf{R}_z^{-1/2}\cdot\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\left(\bar{\mathbf{R}}^*\bar{\mathbf{R}}\right)^{-1}\bar{\mathbf{R}}^*\mathbf{T} \\ &= \mathbf{R}_z^{-1/2}\cdot\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\bar{\mathbf{R}}^{-1}\cdot\mathbf{T} \end{aligned} \tag{4.21}$$

where $\mathbf{T} = [\mathbf{0}_{M_t\times(N_f-1)M_t}, \mathbf{D}_{\mathbf{R}_0}^{-1}, \mathbf{0}_{M_t\times\nu M_t}]$. From (4.14), we have $\mathbf{R}_z^{-1/2}\bar{\mathbf{H}} = \bar{\mathbf{Q}}_{(1:N_fM_r,:)}\bar{\mathbf{R}}$, so $\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}\bar{\mathbf{R}}^{-1} = \bar{\mathbf{Q}}_{(1:N_fM_r,:)}$. Therefore,

$$\bar{\mathbf{W}} = \mathbf{R}_z^{-1/2}\bar{\mathbf{Q}}_{(1:N_fM_r,:)}\mathbf{T} \tag{4.22}$$

so the optimal FFF matrices are

$$\mathbf{W}_i = \mathbf{R}_z^{-1/2}\mathbf{Q}_i\mathbf{D}_{\mathbf{R}_0}^{-1}, \quad 0 \le i \le N_f - 1. \tag{4.23}$$

Now we have obtained the optimal FFF and FBF coefficient matrices (4.20) and (4.23) based on the QR decomposition (4.14), which provides an algorithm for computing (4.20) and (4.23), and which is even faster than the algorithm developed in [52]. Before deriving the fast algorithm in Section 4.3, we conclude this section by giving insights into the fact that DFE achieves the underlying capacity of the ISI channel as the FFF length N_f increases to infinity.

4.2.3 Approaching ISI Channel Capacity

According to (4.17), by using the optimal FFF and FBF, the minimum MSE for the i^{th} substream will be $\left[\mathbf{R}_0\right]_{ii}^{-2}$, where $i = 1, \ldots, M_t$. Thus, the unbiased SINR [61] for the i^{th} substream is

$$\rho_i = \left[\mathbf{R}_0\right]_{ii}^2 - 1 \tag{4.24}$$

The total throughput is then

$$C_1 = \log \prod_{i=1}^{M_t} (1+\rho_i) = \log \prod_{i=1}^{M_t} \left[\mathbf{R}_0\right]_{ii}^2 \tag{4.25}$$

Now consider the capacity of the MIMO ISI channel. Applying the D-transform to the channel matrix taps gives

$$\mathcal{H}(D) = \sum_{k=0}^{v} \mathbf{H}_k D^k \tag{4.26}$$

Note that when $D = e^{j2\pi f}$, (4.26) becomes the discrete-time Fourier transform of the channel matrix sequence. Define

$$\mathbf{S}(D) \equiv \mathbf{I} + \frac{\mathcal{H}^*(D^{-1})\mathrm{H}(D)}{\sigma_z^2} \tag{4.27}$$

From the matrix spectral factorization theorem [62], the positive definite matrix $\mathbf{S}(D)$ can be factorized as

$$\mathbf{S}(D) = \mathrm{T}^*(D^{-1})\mathrm{T}(D) \tag{4.28}$$

where $\mathrm{T}(D) \equiv \sum_{i=0}^{\infty} \mathbf{T}_i D^i$ and $\mathbf{T}_0$ is an upper triangular matrix with positive diagonal elements. We have [63]

$$\log \prod_i [\mathbf{T}_0]_{ii}^2 = \int_0^1 \log \left| \mathbf{S}(e^{j2\pi f}) \right| df \tag{4.29}$$

where the right-hand side is just the channel capacity

$$C = \int_0^1 \log \left| \mathbf{I} + \frac{\mathcal{H}^*(e^{-j2\pi f})\mathcal{H}(e^{j2\pi f})}{\sigma_z^2} \right| df \tag{4.30}$$

Therefore, as long as

$$\lim_{N_f \to \infty} \prod_{i=1}^{M_t} [\mathbf{R}_0]_{ii}^2 = \prod_i [\mathbf{T}_0]_{ii}^2 \tag{4.31}$$

the capacity achieved by the optimal MMSE-DFE will approach the MIMO ISI channel capacity as N_f becomes large.

Recall that from (4.11) and (4.15) we have $\bar{\mathbf{L}} = \bar{\mathbf{R}}$, thus the Cholesky factorization (4.11) can also be rewritten as

$$\bar{\mathbf{S}} \equiv \mathbf{I} + \frac{\bar{\mathbf{H}}^* \bar{\mathbf{H}}}{\sigma_z^2} = \bar{\mathbf{R}}^* \bar{\mathbf{R}} \tag{4.32}$$

According to [64], as $\bar{\mathbf{S}}$ converges to a stationary limit as its dimensionality becomes infinite ($N_f \to \infty$), $\bar{\mathbf{R}}$ converges to an upper triangular Toeplitz matrix whose matrix elements in a block row correspond to the coefficients of $\mathrm{T}(D)$. This means that $\mathbf{R}_0$ equals to $\mathbf{T}_0$ and (4.31) holds. Thus, the capacity achieved by the optimal MMSE-DFE approaches the MIMO ISI channel capacity as N_f increases.

4.3 Fast Algorithm

In the previous derivation, we can see that the computationally most complicated step is to perform the QR decomposition in (4.14). However, by exploiting the Toeplitz structure of the augmented matrix, we can obtain a fast algorithm to efficiently obtain the desired block column/row in $\bar{\mathbf{Q}}/\bar{\mathbf{R}}$, thus facilitating the computation of the optimal FFF and FBF coefficient matrices. In Section 4.3.1, we first give an $M_r = M_t$ example to illustrate the fast algorithm. Then, we extend the algorithm to general M_r/M_t cases in Section 4.3.2. In Section 4.3.3, we briefly summarize Givens rotations and Householder reflections as candidate ways of eliminating certain elements in a matrix and propose an iterative method for realizing the fast algorithm which uses either of these two approaches. In Section 4.3.4, we discuss the flexible selection of N_f brought about by the iterative method, and the conversion to hybrid DFE when it becomes favorable.

4.3.1 Derivation of the Fast Algorithm: An $M_r = M_t$ Example

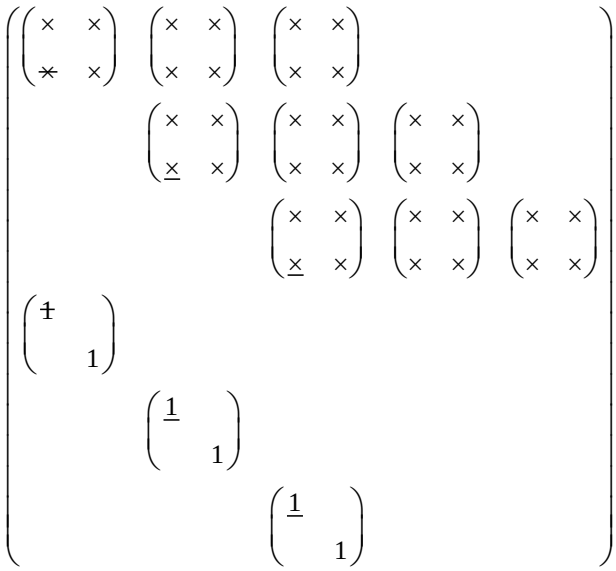
Suppose we have a 2×2 MIMO system, and let $N_f = 3$, $v = 2$. Then the augmented matrix in (4.14) will look like

$$\begin{bmatrix} \mathbf{R}_z^{-1/2}\bar{\mathbf{H}} \\ \mathbf{I} \end{bmatrix} = \begin{pmatrix} \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & & & \\ & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & & \\ & & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} & \begin{pmatrix} \times & \times \\ \times & \times \end{pmatrix} \\ \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} & & & & & \\ & \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} & & & & \\ & & \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} & & & \\ & & & \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} & & \\ & & & & \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} & \\ & & & & & \begin{pmatrix} 1 & \\ & 1 \end{pmatrix} \end{pmatrix} \tag{4.33}$$

The QR decomposition of the above matrix can be achieved by eliminating all the elements below the diagonal, column by column. We first look at how to obtain the $\{\mathbf{R}_i\}$ in (4.14).

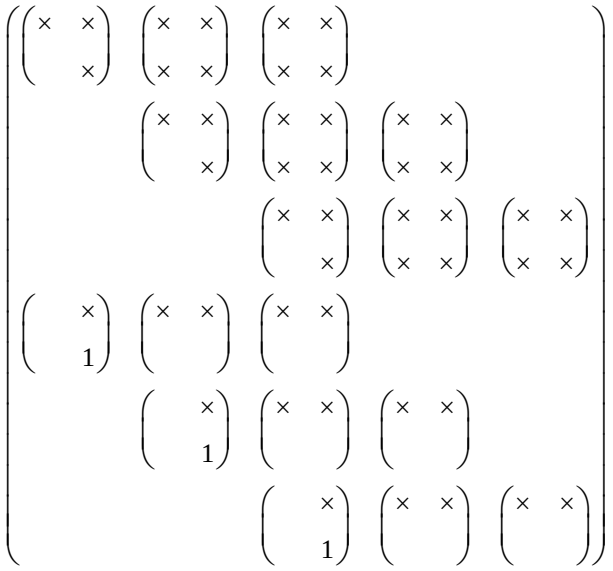
Since the desired matrices are within the N_f^{th} block row of $\bar{\mathbf{R}}$, we only need to proceed until the $N_f M_t^{\text{th}}$ column, which is the 6^{th} column in this example. This indicates that the last 4 rows will remain unchanged. Thus, we will only include the first 12 rows in the following steps.

Start with the first column, we need to eliminate two elements (shown with strikethroughs) in (4.34). This would require two Givens rotations or one Householder reflection.



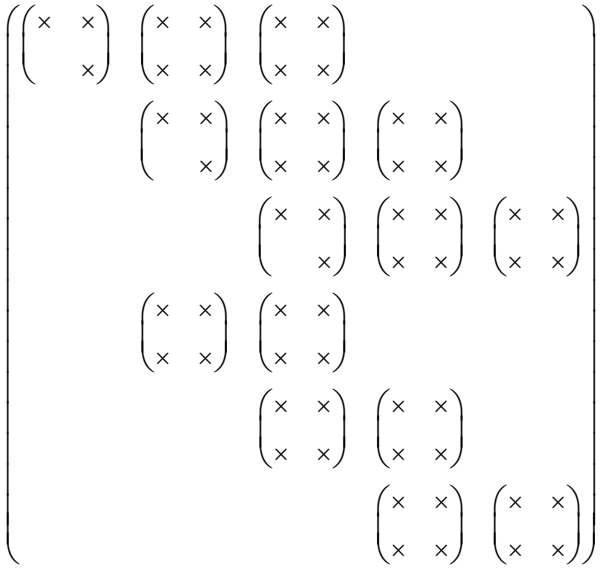
(4.34)

Observe that, due to the Toeplitz structure of the matrix, the underscored elements can be eliminated using the same transformation on the $2^{nd}/5^{th}$ and $3^{rd}/6^{th}$ block row pairs. Thus, we obtain



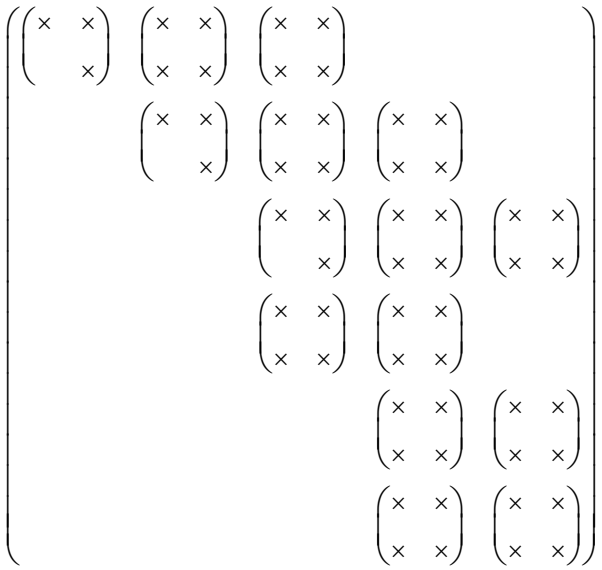
(4.35)

Next, we deal with the second column in a similar way, which leads to



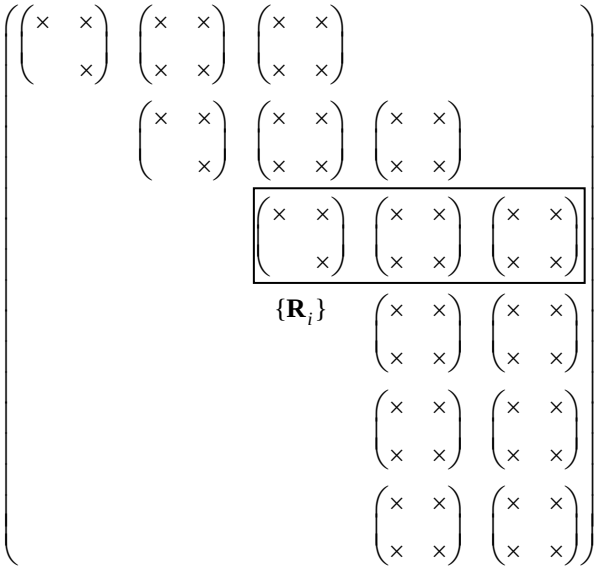
(4.36)

Continue in this manner, except that now we do the same transformation on the $2^{nd}/4^{th}$ and $3^{rd}/5^{th}$ block row pairs. The result is



(4.37)

Next, a final transformation on the $3^{rd}/4^{th}$ block row will give the desired $\{\mathbf{R}_i\}$ as boxed in (4.38).



(4.38)

For any matrix **H** with QR decomposition **H** = **QR**, when triangularizing **H** by left multiplying by transforming matrices $\mathbf{G}_N \ldots \mathbf{G}_1 \mathbf{H} = \mathbf{R}$, one also simultaneously obtains the unitary matrix $\mathbf{Q}^* = \mathbf{G}_N \ldots \mathbf{G}_1$. Thus, the same transformation above could be used on an identity matrix to obtain the $\{\mathbf{Q}_i\}$ in (4.14).

From the above example, we can see that the fast algorithm largely reduces the number of operations compared to the full QR decomposition. We only need to process the first $N_f M_t$ columns of the augmented matrix, and in each column we just need to eliminate M_r elements. A detailed complexity analysis will be presented in Section 4.

4.3.2 Extension to $M_r \neq M_t$ MIMO Systems

Our fast algorithm can also be applied to $M_r \neq M_t$ MIMO systems. For the $M_r > M_t$ case, the extension is straightforward; however, for the $M_r < M_t$ case, we need to slightly modify the original augmented matrix in order to continue utilizing the Toeplitz structure. We will explain both cases in this section.

We begin with the $M_r > M_t$ case using the same example as above except for $M_r = 3$. As in the $M_r = M_t$ case, we still only need to proceed until the $N_f M_t^{\text{th}}$ (6^{th}) column. Since the process is very similar, we omit the detailed explanation and summarize it in Fig. 4.2.

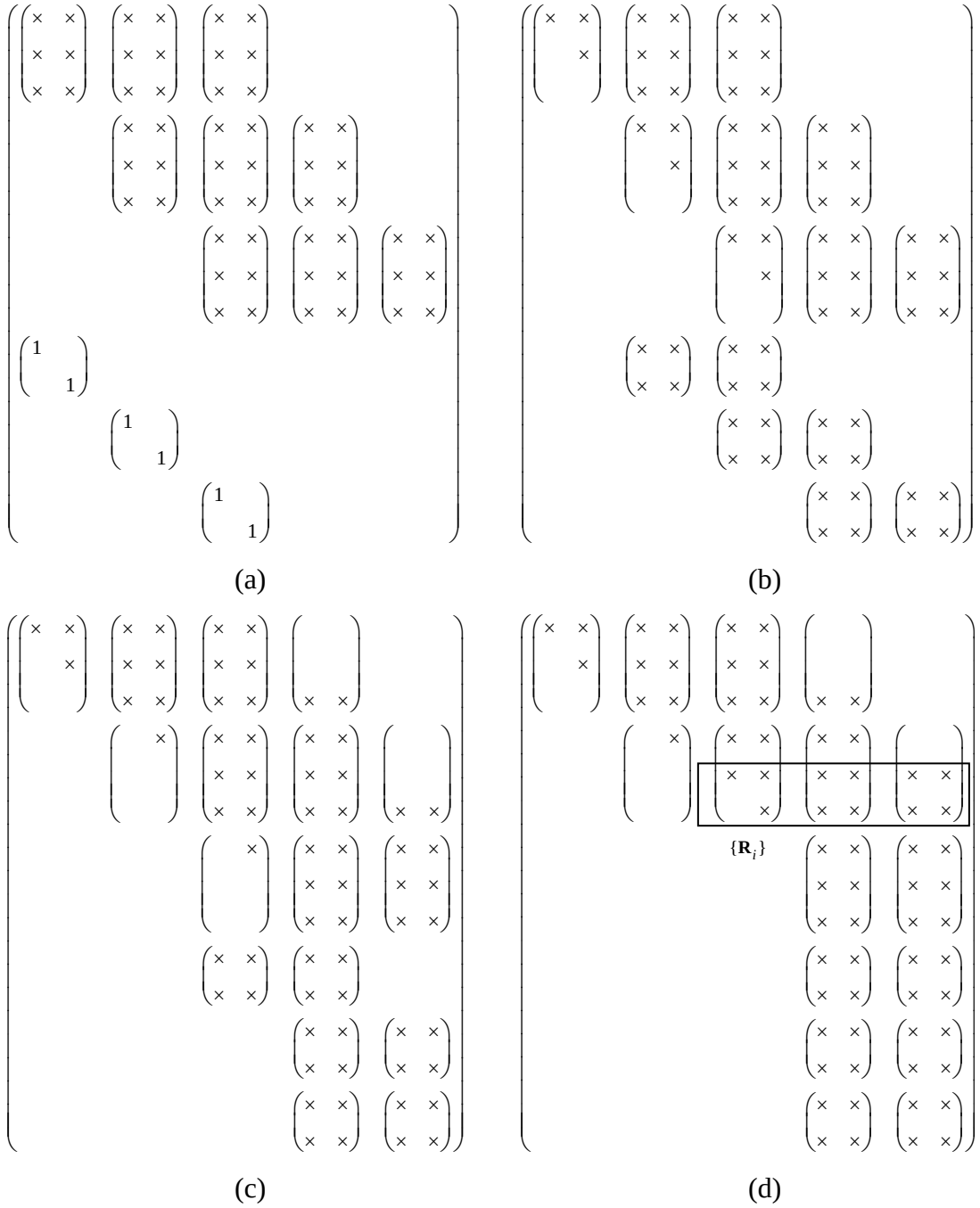


Fig. 4.2: Fast algorithm for 2 × 3 MIMO, $N_f = 3$, $v = 2$.

For the $M_r < M_t$ case, however, the fast algorithm can not be readily applied. The Toeplitz structure of the augmented matrix can not be fully utilized since not all of the diagonal elements in the first $N_f M_t$ columns belong to $\mathbf{R}_z^{-1/2}\bar{\mathbf{H}}$. However, the following observation overcomes this problem.

Suppose we already have the QR decomposition in (4.14) for an $M_r < M_t$ MIMO system. We can then pad each $\mathbf{H}_i$ with an $(M_t - M_r)$ by M_t zero matrix so that it becomes a square matrix. Then, the resulting zero-padded augmented matrix has the following QR decomposition, where $\tilde{\mathbf{H}}_i = \mathbf{H}_i / \sigma_z$.

$$\begin{pmatrix} \tilde{\mathbf{H}}_0 & \tilde{\mathbf{H}}_1 & \cdots & \tilde{\mathbf{H}}_\nu & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{H}}_0 & \tilde{\mathbf{H}}_1 & \cdots & \tilde{\mathbf{H}}_\nu & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \tilde{\mathbf{H}}_0 & \tilde{\mathbf{H}}_1 & \cdots & \tilde{\mathbf{H}}_\nu \\ \mathbf{0} & \mathbf{0} & \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{0} \\ & & & \mathbf{I}_{(N_f+\nu)M_t \times (N_f+\nu)M_t} & & & & \end{pmatrix} = \tilde{\mathbf{Q}}\tilde{\mathbf{R}}$$

$$= \begin{pmatrix} \square & \cdots & \tilde{\mathbf{Q}}_{N_f-1} & \cdots & \square & \cdots & \square \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \square & \cdots & \tilde{\mathbf{Q}}_1 & \cdots & \square & \cdots & \square \\ \square & \cdots & \tilde{\mathbf{Q}}_0 & \cdots & \square & \cdots & \square \\ \square & \cdots & \square & \cdots & \square & \cdots & \square \\ & \ddots & \vdots & \ddots & & & \vdots \\ & & \square & & \ddots & & \vdots \\ & & & \ddots & & \ddots & \vdots \\ & & & & \square & \cdots & \square \end{pmatrix} \begin{pmatrix} \square & \square & \cdots & \square & & & \\ & \square & \cdots & \square & \square & & \\ & & \ddots & \vdots & \vdots & \ddots & \\ & & & \mathbf{R}_0 & \mathbf{R}_1 & \cdots & \mathbf{R}_\nu \\ & & & & \ddots & \ddots & \vdots \\ & & & & & \square & \square \\ & & & & & & \square \end{pmatrix} \tag{4.39}$$

The relationship below can be easily verified:

$$\tilde{\mathbf{Q}}_i = \begin{bmatrix} \mathbf{Q}_i \\ \mathbf{0}_{(M_t - M_r)\times M_t} \end{bmatrix}$$
$$\tilde{\mathbf{R}} = \begin{bmatrix} \bar{\mathbf{R}} \\ \mathbf{0}_{N_f(M_t - M_r)\times(N_f+\nu)M_t} \end{bmatrix} \tag{4.40}$$

which means that the $\{\mathbf{R}_i\}$ remain unchanged. Moreover, the first $(N_f + \nu)M_t$ columns of $\tilde{\mathbf{Q}}$ are the same except for the padded zeros. The last $N_f M_r$ columns are changed since $\tilde{\mathbf{Q}}$ needs to remain unitary. Therefore, we are now able to use the fast algorithm again on the zero-padded augmented matrix and obtain the desired matrices for calculating the optimal FBF and FFF.

4.3.3 Iterative Implementation of the Fast Algorithm

As described in the previous subsections, our goal of the fast algorithm is to efficiently eliminate the elements below the first $N_f M_t$ diagonal elements of the augmented matrix in (4.14). For eliminating elements (i.e. introducing zeros) in a matrix, several methods could be used [65], with Givens rotations and Householder reflections being the two most common ones. We will give a brief summary of these two methods below, followed by a generalized step-by-step iterative method for realizing the fast algorithm.

Suppose we have a matrix $\mathbf{A}$ of size $M \times N$ and vector $\boldsymbol{u}$ is one of its columns, and suppose that we want to make all but the first element in $\boldsymbol{u}$ zero. We can use a series of Givens rotations to do this:

$$
\begin{aligned}
&\text{for } i = 2{:}N \\
&\quad \mathbf{G} = \begin{pmatrix} u_1^* & u_n^* \\ -u_n & u_1 \end{pmatrix} / \sqrt{|u_1|^2 + |u_n|^2}; \\
&\quad \mathbf{A}_{[1\,n],:} = \mathbf{G} \times \mathbf{A}_{[1\,n],:}; \\
&\text{end}
\end{aligned}
\tag{4.41}
$$

To eliminate each of the elements, we need to generate a Givens rotation matrix **G** and multiply **G** with **A**. Generating **G** requires a square root operation.

Alternatively, we can use Householder reflections. Recall that given a Householder vector $\boldsymbol{v}$ of size $N \times 1$, the corresponding Householder matrix **P** is calculated by

$$
\mathbf{P} = \mathbf{I} - \frac{2}{v^* v} v v^*. \tag{4.42}
$$

A vector multiplied by **P** is reflected in the hyperplane span $\{\boldsymbol{v}\}^{\perp}$. We can then force all required elements in $\boldsymbol{u}$ to become zero at once as follows:

$$
\begin{aligned}
&v = u; \\
&v_1 = u_1 - \text{norm}(u); \\
&\beta = \frac{2}{v^* v}; \\
&w = \beta \mathbf{A}^* v; \\
&\mathbf{A} = \mathbf{P}\mathbf{A} = (\mathbf{I} - \beta v v^*)\mathbf{A} = \mathbf{A} - v w^*;
\end{aligned}
\tag{4.43}
$$

where v_1 is the first element in v. Note that only one square root operation is needed in the process [65].

When implementing the fast algorithm, we don't have to operate on the whole augmented matrix in (4.14) thanks to its Toeplitz structure. We only need to use two block rows (see (4.34)-(4.38) and Fig. 4.2), one in the upper part (containing the channel information) and one in the lower part (containing an identity matrix). By updating the rows iteratively $N_f M_t$ times, we can obtain the desired matrices for the optimal FFF and

FBF. In this way we are able to save considerable memory storage space, while also enabling a flexible control of the SNR achieved, as demonstrated in the next subsection. We present this iterative method as Algorithm I (see next page) for general M_r/M_t cases. Note that while Givens rotations have been used in the algorithm, either Givens rotations or Householder reflections could be used in the realization. Also note that the result matrices are in **R** and **Q**.

Algorithm I. Iterative Method for Fast Algorithm

$\mathbf{B} = [\mathbf{H}_0\ \mathbf{H}_1 \cdots \mathbf{H}_\nu]./\sigma_z$; $\mathbf{C} = [\mathbf{I}_{M_t}\ \mathbf{0}_{M_t \times \nu M_t}]$;

$\mathbf{W} = [\mathbf{I}_{\max(M_r, M_t)\times M_r}\ \mathbf{0}_{\max(M_r, M_t)\times M_r(N_f-1)}]$; $\mathbf{D} = \mathbf{0}_{\max(M_r, M_t)\times M_r N_f}$;

$\mathbf{R} = \mathbf{0}_{M_t \times M_t(\nu+1)}$; $\mathbf{Q} = \mathbf{0}_{M_t \times M_r N_f}$;

givens = inline('[conj(*x*) conj(*y*); -*y x*]/norm([*x y*])', '*x*', '*y*');

for i = 1:$N_f M_t$

$\mathbf{T}$ = [$\mathbf{B}$; $\mathbf{C}$]; $\mathbf{S}$ = [$\mathbf{W}$; $\mathbf{D}$];

index = find($\mathbf{T}$(:,1));

if index(1) ≠ 1

index = [1; index];

end

for j = index(2:end)'

$\mathbf{G}$ = givens($\mathbf{T}_{1,1}$, $\mathbf{T}_{j,1}$);

$\mathbf{T}_{[1\ j],:} = \mathbf{G} \times \mathbf{T}_{[1\ j],:}$; $\mathbf{S}_{[1\ j],:} = \mathbf{G} \times \mathbf{S}_{[1\ j],:}$;

end

if $i > (N_f - 1)M_t$

$\mathbf{R}_{i-(N_f-1)M_t,\ i-(N_f-1)M_t:M_t(\nu+1)} = \mathbf{T}_{1,\ 1:M_t(\nu+1)-i+(N_f-1)M_t+1}$;

$\mathbf{Q}_{i-(N_f-1)M_t,:} = \mathbf{S}_{1,:}$;

end

$\mathbf{B} = \mathbf{T}_{2:M_r,\ 2:\text{end}}$;

if mod(i, M_t) = 1

$\mathbf{B} = [\mathbf{B}\ \mathbf{0}_{(M_r-1)\times M_t}; \mathbf{0}_{1\times(M_t-1)}\ \mathbf{T}_{1,:}]$;

else

$\mathbf{B} = [\mathbf{B}; \mathbf{0}_{1\times(M_t-1)}\ \mathbf{T}_{1,1:\text{end}-M_t}]$;

end

$\mathbf{C} = [\mathbf{T}_{M_r+1:\text{end},\ 2:\text{end}}\ \mathbf{0}_{M_t\times(\text{size}(\mathbf{B},2)-\text{size}(\mathbf{T},2)+1)}]$;

$\mathbf{W} = [\mathbf{S}_{2:M_r,:}; \mathbf{0}_{1\times M_r}\ \mathbf{S}_{1,1:\text{end}-M_r}]$; $\mathbf{D} = \mathbf{S}_{M_r+1:\text{end},:}$;

end

4.3.4 Dynamic Selection of N_f and Conversion to Hybrid DFE

As shown in Section 4.2.3, as N_f increases towards infinity, the achievable throughput will monotonically approach the MIMO ISI channel capacity. However, there is a tradeoff: higher throughput (or larger N_f) comes with higher complexity. Given a target SINR, it is desirable to find the minimum N_f satisfying the requirement. The existing algorithms assume some fixed value of N_f beforehand, and thus offer no flexibility regarding various channel environments. However, thanks to the iterative nature of Algorithm I, we can gradually increase the length N_f of the FFF and terminate the algorithm when the required SINR/throughput has been achieved.

More specifically, the algorithm keeps track of the current performance of the MMSE-DFE by multiplying the diagonal elements of $\mathbf{R}_0$ at each iterative step. By comparing (4.25) with the required throughput and/or channel capacity in (4.30), the algorithm keeps track of the amount of throughput the system can currently achieve against the channel capacity and terminates if the ratio reaches some threshold. This allows us to obtain the desired performance/complexity tradeoff, and thereby offers flexibility over the traditional methods in which the value of N_f is preset.

Moreover, we can further explore this flexibility to switch between TDE and FDE when necessary. One of the main issues preventing TDE from being applied in high-speed communication systems is its complexity in channels whose impulse response spans a large number of symbol intervals. For each symbol time, the feedforward equalization in TDE takes O(N_f) flops, while SC-FDE requires only O(log(N)) flops, where N is the FFT size. Hence, SC-FDE enjoys lower complexity than TDE in long-span multipath channels.

However, this advantage may be reversed in the emerging technology of small cell communications, e.g., femtocells, where the channel span is no more than the duration of a few symbols. Hence, it would be desirable to adaptively switch between TDE and FDE based on the complexity criteria. This flexibility is very desirable in heterogeneous networks where the channel span varies significantly between femtocells, picocells, and macrocells. With the ability to monitor the system performance/complexity during each iterative step as N_f increases, this is indeed possible.

Regarding SC-FDE under long channels, [66] emphasizes that only with a nonlinear equalizer at the receiver can SC-FDE achieve satisfactory performance. One main category of nonlinear SC-FDE is the so-called hybrid DFE (HDFE), where the FFF operates in the frequency domain on blocks of received signal, while the FBF operates in the time domain [66]. As the equalization complexity of the proposed TDE becomes too expensive with the increase of N_f, a natural yet interesting approach can be adopted: simply convert the time domain FFF to the frequency domain to form a HDFE. The block diagram of the converted receiver is shown in Fig. 4.3.

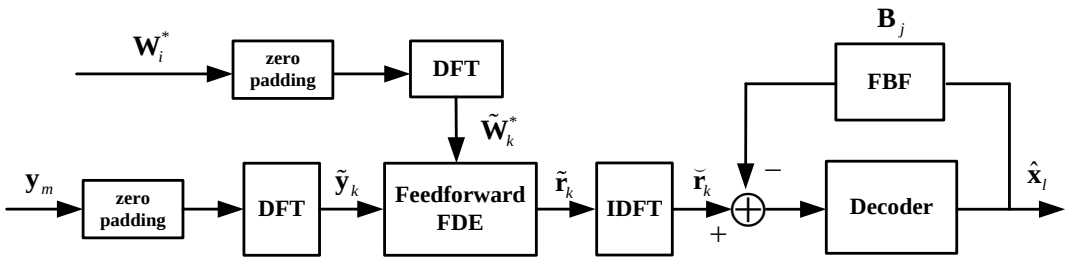


Fig. 4.3: Block diagram of converted hybrid DFE receiver

At the transmitter, suppose in each block we have L information symbol vectors $\{\mathbf{x}_l \in C^{M_t \times 1}\}$, $l = 0, 1, \ldots, L-1$ appended by v zero vectors to avoid ISI, making the total block length $(L + v)$. At the receiver, with the time domain FFF, the convolution of the received block $\{\mathbf{y}_m \in C^{M_r \times 1}\}$, $m = 0, 1, \ldots, L - 1 + v$ with N_f FFF coefficients $\left\{\mathbf{W}_i^* \in C^{M_t \times M_r}\right\}$, $i = 0, 1, \ldots, N_f - 1$ generates a length $(L + v + N_f - 1)$ result $\{\mathbf{r}_p \in C^{M_t \times 1}\}$, $p = 0, 1, \ldots, L + v + N_f - 2$. Observe that the first $(N_f - 1)$ and the last v elements of $\{\mathbf{r}_p\}$ are not utilized, and that only the middle L quantities $\{\mathbf{r}_p\}$, $p = N_f, \ldots, L + N_f - 1$ are used in estimating $\{\mathbf{x}_l\}$. Let us append $(N_f - 1 - v)$ zero vectors to $\{\mathbf{y}_m\}$ and perform a length $(L + N_f - 1)$ DFT, denoting the result as $\{\tilde{\mathbf{y}}_k \in C^{M_r \times 1}\}$, $k = 0, 1, \ldots, L + N_f - 2$. We also perform a length $(L + N_f - 1)$ DFT on $\left\{\mathbf{W}_i^*\right\}$ and denote the result as $\{\tilde{\mathbf{W}}_k^* \in C^{M_t \times M_r}\}$, $k = 0, 1, \ldots, L + N_f - 2$. The feedforward FDE generates the frequency domain equalized result $\{\tilde{\mathbf{r}}_k \in C^{M_t \times 1}\}$, $k = 0, 1, \ldots, L + N_f - 2$ by

$$\tilde{\mathbf{r}}_k = \tilde{\mathbf{W}}_k^* \tilde{\mathbf{y}}_k \tag{4.44}$$

The result is then passed through an IDFT and we obtain the time domain outcome $\{\breve{\mathbf{r}}_k \in C^{M_t \times 1}\}$, $k = 0, 1, \ldots, L + N_f - 2$, whose last L elements are the same as the useful part of the time domain FFF result $\{\mathbf{r}_p\}$, $p = N_f, \ldots, L + N_f - 1$. The following FBF is the same as the one in the original time domain MMSE-DFE. A detailed analysis of when to switch from TDE to FDE will be presented in Section 4.4.2.

4.4 Complexity Analysis and Performance Comparison

In this section, we first analyze the complexity of the proposed fast algorithm in terms of the number of complex multiplications. Then, we will describe when the switch to HDFE should be made as N_f increases. We compare the proposed TDE/HDFE complexity/performance with other TDE/HDFE methods in the last subsection.

4.4.1 Complexity of the Proposed Fast Algorithm

From Algorithm I it can be seen that we need to iterate $N_f M_t$ times in order to obtain the desired matrices **R** and **Q**. In each iteration, we need to zero out all but the first element in the first column of matrix **T**. The computational complexity in this process depends on 1) the number of nonzero elements in this column, which determines the number of rows we need to operate on; and 2) the number of elements in these rows, which determines the number of multiplications required. In our fast algorithm, the number of nonzero elements in the first column of **T** is always $M_r + 1$, which can also be observed from (4.34)-(4.38) and Fig. 4.2. This means that we need to eliminate M_r elements (i.e., generate M_r Givens rotation matrices). The number of columns in **T** is bounded by $(\nu + 2)M_t - 1$, which can be deduced from Algorithm I. Meanwhile, we also apply the same transformation on **S** in order to obtain **Q**. The number of columns in **S** stays the same as $N_f M_r$. Since one Givens rotation requires four complex multiplications, the total number of complex multiplications involved in computing **R** and **Q** is bounded by

$$4N_f M_t M_r((\nu+2)M_t - 1 + N_f M_r) \tag{4.45}$$

Note that the cost of generating the Givens rotation matrices is not considered here, whose main complexity lies in the square root operation for computing the norm in (4.42). Since we need to generate $N_f M_t M_r$ Givens rotation matrices, the same number of square root operations will be required. We also need to use (4.20) and (4.23) to calculate the final FBF and FFF coefficient matrices after we obtain **R** and **Q**; however, that cost is small compared to (4.40).

Alternatively, if Householder reflections are used, we still need to iterate the loop in Algorithm I $N_f M_t$ times. According to (4.43), we can see that we also need one square root operation here for the norm calculation, which will not be included in the formula given below. Since $\boldsymbol{u}$ now contains the $M_r + 1$ nonzero elements in the first column of **T**, $\boldsymbol{v}$ is also of size $M_r + 1$. Thus, calculating β requires $2(M_r + 1)$ real multiplications. Computing $\boldsymbol{w}$ requires at most $(M_r + 1)((\nu + 2)M_t - 1)$ complex multiplications and $(\nu + 2)M_t - 1$ real-times-complex multiplications for **T**, while for **S** it is $N_f M_r(M_r + 1)$ complex multiplications and $N_f M_r$ real-times-complex multiplications. Finally, in the last step of (4.43) we need at most another $(M_r + 1)((\nu + 2)M_t - 1)$ complex multiplications for **T** and $N_f M_r(M_r + 1)$ complex multiplications for **S**. Assuming that one complex multiplication has the complexity of four real multiplications, the total number of complex multiplications needed for the Householder reflections method is bounded by

$$N_f M_t((2M_r + 2.5)((\nu + 2)M_t - 1 + N_f M_r) + 0.5(M_r + 1)) \tag{4.46}$$

Comparing (4.46) with (4.45), it can be seen that using Householder reflection brings a significant reduction in the number of complex multiplications. For large numbers

of antennas and long channel responses, the savings will be by about a factor of 2. Thus, in the following we will use (4.46) as the complexity of our algorithm.

4.4.2 Switching Between TDE/HDFE

While increasing N_f to achieve the desired SINR, we can monitor the total complexity of the equalizer, including both coefficient calculation and equalization. For TDE, the complexity for coefficient calculation is given by (4.46), and the equalization complexity per $\mathbf{x}_l$ is

$$N_f M_t M_r + \nu M_t^2 + \frac{M_t(M_t - 1)}{2} \tag{4.47}$$

For the converted HDFE, suppose that an FFT is used at the receiver and let N be the FFT size. Select L so that $N = L + N_f - 1$. The cost of converting the FFF to the frequency domain is

$$M_t M_r \frac{N}{2} \log_2 N \tag{4.48}$$

The equalization complexity per $\mathbf{x}_l$ is

$$\frac{1}{N - N_f + 1}\left((M_t + M_r)\frac{N}{2}\log_2 N + NM_tM_r\right) + \nu M_t^2 + \frac{M_t(M_t - 1)}{2} \tag{4.49}$$

If the equalizer coefficients are updated every block, then as (4.47) becomes larger than the sum of (4.48)/L and (4.49), we can advantageously switch from TDE to HDFE.

4.4.3 Comparison with other TDE/HDFE approaches

In this subsection we will compare our work with previously proposed methods. For fast calculation of the time domain MIMO MMSE-DFE matrix coefficients, the algorithm in [52] reduced the complexity of computing the FBF matrices from $O[M_t^3(N_f+\nu)^3]$ (using a classical Gaussian elimination technique) to $O[M_t^2(N_f+\nu)^2]$; however, computing the FFF matrices via back-substitution needs $O[M_t^2 M_r(N_f+\nu)^2 N_f]$ complex multiplications. Thus, the total complexity will be an order of magnitude higher than in our algorithm.

The authors of [56] proposed a fast algorithm that requires fewer operations than our algorithm (see Table 4.1). However, that approach has an inherit limitation in that its $\mathbf{B}_0$ in (4.13) is constrained to be a zero matrix, while the algorithm of [52] and our algorithm both allow triangular $\mathbf{B}_0$ matrices, which gives better performance. This is due to the fact that such triangular matrices allow the current decisions from lower indexed streams/users to be used by higher indexed streams/users.

MIMO hybrid DFE was first proposed in [67], in which the FFF is optimized and performed in the frequency domain, while the FBF remains in time domain. Another method, FDE with noise prediction (FDE-NP) [68], is shown to have the same performance as [67], where both are optimal in the MMSE sense. Furthermore, in [68] FDE-NP is enhanced with successive interference cancellation (FDE-NP-SIC). The performance gap between FDE-NP and FDE-NP-SIC is similar to the TDE cases with zero vs. triangular $\mathbf{B}_0$ matrices.

Table 4.1: Complexity Comparison

Method	**Coefficients calculation**	**Equalization (per x_l)**
Our TD-DFE	$N_f M_t((2M_r + 2.5)$ $((\nu + 2)M_t - 1 + N_f$ $M_r) + 0.5(M_r + 1))$	$N_f M_t M_r + \nu M_t^2$ $+ \frac{M_t(M_t - 1)}{2}$
Our HDFE	$N_f M_t((2M_r + 2.5)$ $((\nu + 2)M_t - 1 + N_f$ $M_r) + 0.5(M_r + 1))$ $+ M_t M_r \frac{N}{2} \log_2 N$	$((M_t + M_r)\frac{N}{2} \log_2 N$ $+ NM_t M_r) \frac{1}{N - N_f + 1}$ $+ \nu M_t^2 + \frac{M_t(M_t - 1)}{2}$
Fast TD-DFE [52]	$O(M_t^2(N_f + \nu)^2) +$ $O(M_t^2 M_r (N_f + \nu)^2 N_f)$	$N_f M_t M_r + \nu M_t^2$ $+ \frac{M_t(M_t - 1)}{2}$
FDE-NP-SIC [68]	$N(O(M_t^3) + O(M_r^3)) +$ $BO(M_t^3) + NM_t M_r^2 +$ $(2B^2 + 3B - 3)M_t^3 + B\frac{M_t^3 - M_t^2}{2}$ $+ ((N-1)B + 2NM_r)M_t^2$	$\frac{(M_t + M_r)}{2} \log_2 N +$ $M_t M_r + BM_t^2 + \frac{M_t(M_t - 1)}{2}$
Fast TD-DFE with $\mathbf{B}_0 = \mathbf{0}$ [56]	$1.5M_t M_r^2 N_f (N_f + 1) +$ $2M_t^2 M_r + M_t M_r +$ $\min(2\nu M_t M_r + 2\nu(2M_t +$ $M_r)\log_2(2\nu), N_f \nu M_t M_r / 2)$	$N_f M_t M_r + \nu M_t^2$
FD-DFE [67]	$N(O(M_t^3) + O(M_r^3)) +$ $BO(M_t^3) + (2B^2 + 3B - 3)M_t^3$ $+ ((N-1)B + NM_r)M_t M_r +$ $((N-1)(M_r + 1)B + 2NM_r)M_t^2$	$\frac{(M_t + M_r)}{2} \log_2 N +$ $M_t M_r + BM_t^2$
FDE-NP [68]	$N(O(M_t^3) + O(M_r^3)) +$ $BO(M_t^3) + NM_t M_r^2 +$ $(2B^2 + 3B - 3)M_t^3 +$ $((N-1)B + 2NM_r)M_t^2$	$\frac{(M_t + M_r)}{2} \log_2 N +$ $M_t M_r + BM_t^2$

The coefficients calculation and equalization complexities of these methods in terms of complex multiplications are listed in Table 4.1. Recall that M_t and M_r are the number of transmit and receive antennas, N is the FFT size, v is the number of channel multipath taps, N_f is the length of FFF, and B is the length of the feedback filters in the FD methods. Here we assume $B = v$ for fair comparison. The first half of Table 4.1 contains the methods using SIC.

Fig. 4.4 compares the performance of the above methods. The resulting total MMSE is the sum of the MMSEs of all substreams, averaged over 10000 channel realizations where $\sigma_z^2 = 0.1$. With N_f increasing, the total MMSE of TDE with $\mathbf{B}_0 = \mathbf{0}$ [56] approaches HDFE without SIC (FD-DFE [67] and FDE-NP [68]), while the total MMSE of TDE with triangular $\mathbf{B}_0$ (fast TD-DFE [52] and our algorithm) merges with FDE-NP-SIC [68]. We can also clearly observe the performance gap between SIC and non-SIC methods.

Among the SIC methods, the lower complexity of the proposed approach compared to [52] has already been pointed out, while still sharing the same level of performance. The comparison between FDE-NP-SIC and our algorithm is not as straightforward, since no explicit complexity formula is given for FDE-NP-SIC. The authors do give an example in [68] where FDE-NP-SIC needs 6.9×10^3 complex multiplications with $M_t = 2$, $M_r = 2$, $N = 64$ and $B = 2$ ($v = 2$). According to Fig. 4.4, the proposed TDE method shows very close performance with that of FDE-NP-SIC when $N_f = 5$, which requires only about 3×10^3 complex multiplications. The best choice under a certain circumstance depends on the specific parameter values, especially N_f and N.

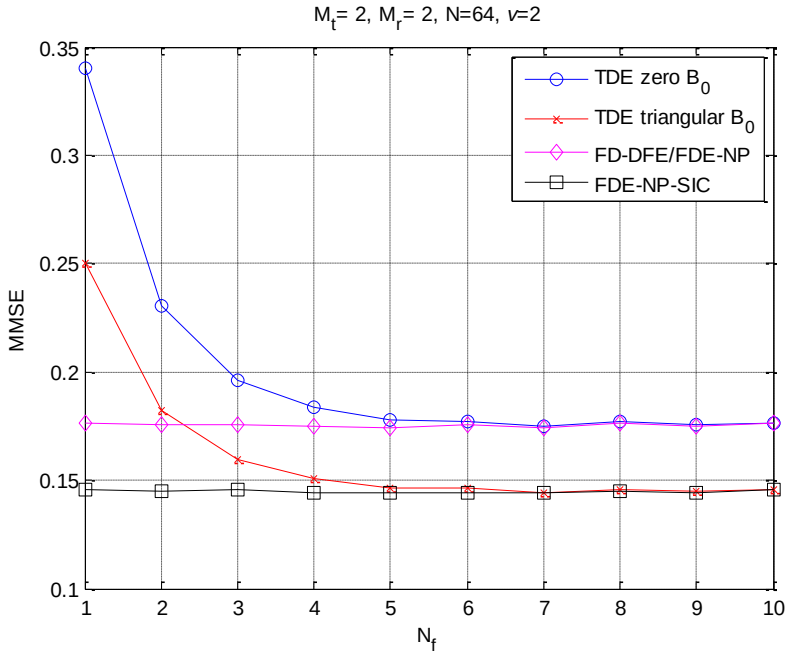


Fig. 4.4: MMSE comparison of different methods

4.5 Numerical Example and Simulations

In this section, we give a numerical example to illustrate the computation of the optimal time domain FFF and FBF matrix coefficients and to show how the increase of N_f affects the system performance and complexity, for both TDE and HDFE.

Consider a 2 × 2 MIMO system. Suppose v = 1, σ_z^2 = 0.1, N = 64 and N_f will be increased from 1 to 10. We adopt a Rayleigh fading channel where the channel taps experience exponential decay. One realization of the channel taps is:

$$\mathbf{H}_0 = \begin{pmatrix} -0.9196 - 0.0506i & -1.0526 - 0.4910i \\ -0.4278 - 1.7074i & 0.3949 - 0.9839i \end{pmatrix}$$

$$\mathbf{H}_1 = \begin{pmatrix} -0.1190 + 0.1414i & -0.3810 + 0.0631i \\ -0.5548 + 0.2567i & -0.4231 - 0.0435i \end{pmatrix}$$

The channel capacity calculated from (4.30) is 7.6439 bps/Hz. Let $N_f = 1$. Then for TDE we will have

$$\mathbf{R}_0 = \begin{pmatrix} 6.3612 & 3.9359 + 2.3478i \\ 0 & 2.1739 \end{pmatrix}$$

$$\mathbf{B}_0 + \mathbf{I} = \begin{pmatrix} 1 & 0.6187 + 0.3691i \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{B}_1 = \begin{pmatrix} -0.0244 - 0.2949i & 0.1489 - 0.1930i \\ 0.1810 + 0.1061i & 0.3413 + 0.0396i \end{pmatrix}$$

$$\mathbf{W}_0 = \begin{pmatrix} -0.2273 + 0.0125i & -1.0627 + 0.2544i \\ -0.1057 + 0.4219i & 0.0624 - 0.4876i \end{pmatrix}$$

According to (4.25), the achievable throughput for the current MMSE-DFE is 6.5751 bps/Hz, which is 86% of the channel capacity. Increasing N_f to 10, and using Householder reflections, from (4.25) and (4.46)-(4.49) we have Table 4.2 which shows the system performance in terms of C_1 and complexity per $\mathbf{x}_l$ for both TDE and HDFE. Suppose the number of $\mathbf{x}_l$ per block is $N - N_f + 1$ and that the channel is updated every block. Thus, using the quantities specified in (4.46)-(4.49), the complexity per $\mathbf{x}_l$ for TDE is given by (4.46)/($N - N_f + 1$) + (4.47), while for HDFE it is given by ((4.46) + (4.48))/($N - N_f + 1$) + (4.49).

Table 4.2: Exapmle Performance/Compexity Tradeoff

N_f	**1**	**2**	**3**	**4**	**5**
C_1(bps/Hz)	6.5751	7.5792	7.6366	7.6424	7.6430
TDE	10.4688	16.8095	24.0645	32.2787	41.5000
HDFE	34.4688	37.2540	40.9677	45.6557	51.3667
N_f	**6**	**7**	**8**	**9**	**10**
C_1(bps/Hz)	7.6431	7.6431	7.6431	7.6431	7.6431
TDE	51.7797	63.1724	75.7368	89.5357	104.6364
HDFE	58.1525	66.0690	75.1754	85.5357	97.2182

Observe that, while further increasing N_f beyond $N_f = 6$ doesn't offer any throughput improvement under the current precision, the complexity increases significantly. The complexity of TDE becomes larger than HDFE when $N_f = 8$.

Since $N_f = 2$ already achieves 99% of the channel capacity with reasonable complexity, we can consider this as a good tradeoff in practical applications. It's obvious that we should choose TDE due to its much lower complexity per $\mathbf{x}_l$ (less than half that of HDFE). The optimal FFF and FBF are

$$\mathbf{B}_0 + \mathbf{I} = \begin{pmatrix} 1 & 0.6214 + 0.3652i \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{B}_1 = \begin{pmatrix} -0.0242 - 0.2925i & 0.1477 - 0.1915i \\ 0.1663 + 0.1100i & 0.3211 + 0.0377i \end{pmatrix}$$

$$\mathbf{W}_0 = \begin{pmatrix} -0.2254 + 0.0124i & -1.0242 + 0.2536i \\ -0.1049 + 0.4185i & 0.0765 - 0.4782i \end{pmatrix}$$

$$\mathbf{W}_1 = \begin{pmatrix} -0.0118 + 0.0137i & -0.1192 - 0.0133i \\ -0.0105 - 0.0108i & 0.0243 - 0.0749i \end{pmatrix}$$

Simulation results for a 4×4 MIMO system with different channel lengths are given in Fig. 4.5 and 4.6 below. Fig. 4.5 shows the ratio of the achievable throughput versus channel capacity with the increase in N_f. Fig. 4.6 shows the complexity per $\mathbf{x}_l$ of TDE and HDFE, with FFT size $N = 128$.

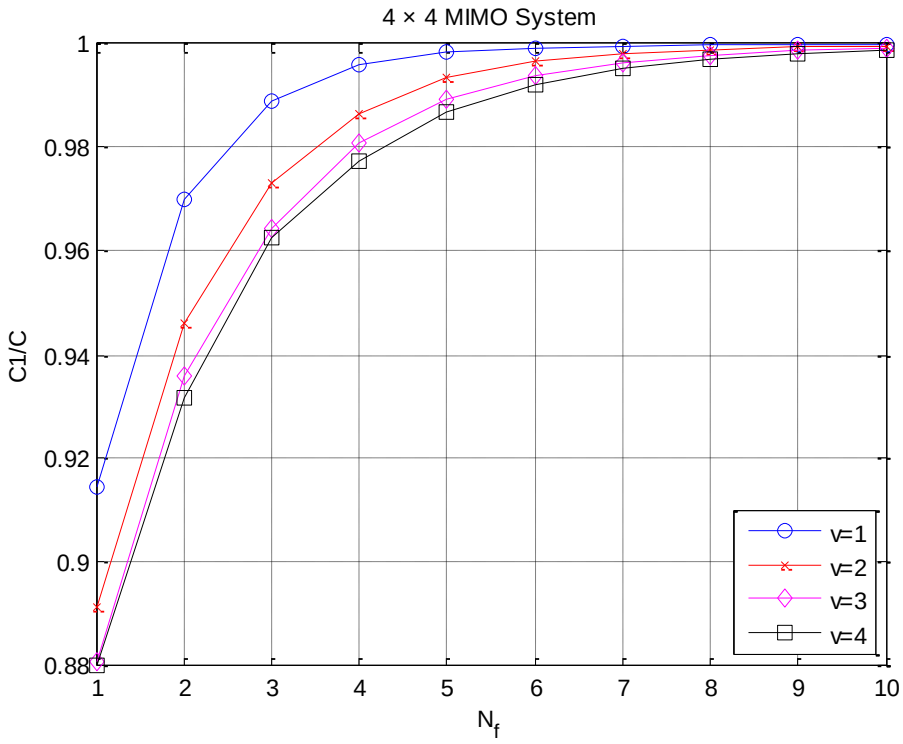


Fig. 4.5: Capacity achieved for different N_f

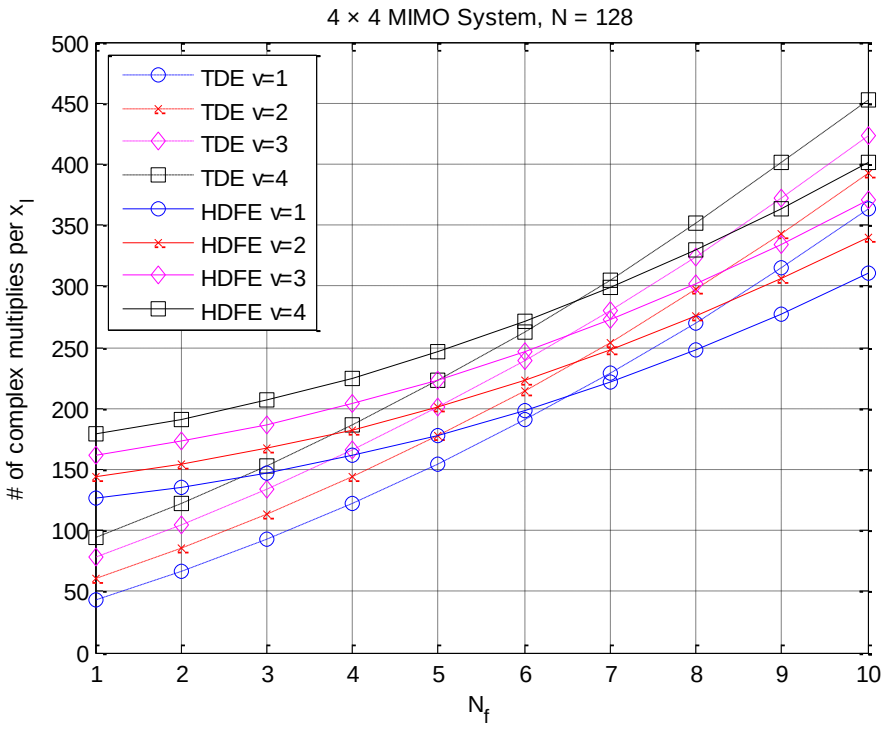


Fig. 4.6: Complexity per $\mathbf{x}_l$ for different N_f (TDE v.s. HDFE)

4.6 Conclusions

In this chapter, using the QR decomposition of an augmented Toeplitz channel matrix, we have proposed a fast iterative algorithm to obtain the optimal FIR MIMO MMSE-DFE coefficients, which offers the lowest complexity among competing approaches without sacrificing performance. Furthermore, the iterative nature of the proposed algorithm enables a flexible choice of N_f which enables a controllable performance/complexity tradeoff to be made. When the required performance demands a large N_f, which indicates high complexity for the proposed TDE, an alternative HDFE approach can be used. Comparisons with the complexities of existing TDE/HDFE methods confirm the computational efficiency of our procedure.

Chapter 5

Conclusions

By deploying multiple antennas at both the transmitting and receiving end, MIMO systems are able to support much higher throughput and/or reliability than their SISO counterparts. Multiple data streams can be sent from the transmitting antennas at the same time to exploit the available multiplexing gain. At the receiver, each antenna collects a mixture of channel-distorted transmitted data streams and noise. Decoupling these data streams can be viewed from a channel decomposition perspective; by decomposing the MIMO channel matrix into the product of several matrices in specific ways, one is able to obtain parallel independent subchannels by using these matrices to perform signal processing at the transmitter and/or receiver.

In this thesis, we first presented MIMO transceiver designs in ISI-free channels based on three recently introduced channel decomposition methods, namely the GMD, the UCD and the TCD. Traditional MIMO transceiver designs have extensively used the SVD, where the gains of the decomposed subchannels are determined by the singular values of the MIMO channel matrix. Although it can achieve the MIMO channel capacity when combined with "water-filling" power allocation, one has no control over these subchannel gains, which can have vastly different values from each other. This requires careful bit-loading, which not only increases the system complexity but it is also results in capacity

loss due to the finite constellation sizes. If the same constellation is used for all subchannels, however, there will be an inevitably tradeoff between throughput and BER performance.

The GMD solves this dilemma by decomposing the MIMO channel into multiple identical subchannels. However, it suffers from capacity loss at lower SNR due to its utilization of the ZF-VBLAST algorithm. The UCD improves on this aspect by incorporating MMSE-VBLAST instead, which is information lossless. In Chapter 2, we presented the hardware MIMO transceiver designs based on these two channel decomposition methods. The FPGA results suggest that these two methods actually have comparable hardware complexities. Thus, the UCD has been found to be a more appealing approach due to its superior performance.

Future work on these transceiver designs can include hardware implementation of the channel decomposition calculations themselves which are currently assumed to be performed in software. Transceiver designs based on limited CSI feedback can also be investigated.

In Chapter 3, we discussed the utilization of the TCD in cognitive radio systems. Suppose we need to transmit several required services with different QoS constraints, which translates to having different data rates. Using the TCD scheme, we can accommodate these services in a single MIMO system using the minimum power. If there are multiple available frequency bands obtained from spectrum sensing, however, we showed that it might be desirable to split the MIMO system into smaller subsystems, especially if there are power constraints on these frequency bands. This leads to a flexible

MIMO transceiver design based on the TCD which can perform either as a large MIMO system or as multiple smaller subsystems, depending on the requirements at any given time. The presented 4×4 reconfigurable design has approximately the same area as its non-configurable counterpart. This suggests that this type of design approach may provide a promising solution for future cognitive radio systems, where various QoS-constrained services need to be accommodated using the power-limited, time-dependent and possibly non-contiguous spectrum resources. Future work in this area can more fully explore these issues.

Chapters 2 and 3 were mainly focused on hardware transceiver realizations of the recently introduced channel decomposition methods under an ISI-free MIMO channel. However, with the increasingly high throughput in current communication systems, the ISI-free assumption becomes invalid when the system throughput is larger than the channel bandwidth. This means that aside from separating the data streams from a spatial perspective, the system also needs to deal with the interference from previously transmitted information. OFDM is a widely used multi-carrier approach to mitigate the ISI effect; however, it leads to high PAPR which is not desirable in mobile devices. Single carrier frequency domain equalization has become an uplink substitute for OFDM which avoids this problem, but its performance can only be comparable to that of OFDM if hybrid DFE (HDFE) is used, in which the feedforward portion is performed in the frequency domain and the feedback portion is performed in time domain.

In Chapter 4, we proposed a fast time domain MMSE-DFE algorithm which can be converted to HDFE when the ISI span becomes so long that the complexity of TDE

becomes higher than that of HDFE. This algorithm is also based on using a MIMO channel decomposition. Specifically, it uses the QR decomposition of an augmented channel matrix. It gives the lowest complexity among the currently available time domain MMSE-DFE algorithms offering the same performance. It also has comparable or lower complexity than presently used hybrid DFE algorithms. We proposed an iterative method to realize this algorithm, which allows us to dynamically control the number of iterations according to the required performance. Its flexibility also makes it suitable for use in heterogeneous networks in which different channel lengths are present.

Future work in this area may include the optimized design for a multi-user scenario and the implementation of a combined system using both the proposed fast algorithm and the channel decomposition methods of Chapters 2 and 3. Specifically, the use of a Graphics Processing Unit (GPU) may serve as an effective platform for realizing such a system due to the inherit parallelism in many of the matrix computations.

Bibliography

[1] A.J. Paulraj, D.A. Gore, R.U. Nabar, H. Bolcskei, "An overview of MIMO communications - a key to gigabit wireless," *Proc. of IEEE* vol. 92, no. 2, pp. 198 – 218, February 2004.

[2] D. Gesbert, M. Shafi, D.-S. Shiu; P.J. Smith, A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE J. Select Area Comm.*, vol. 21, issue 3, pp. 281-302, April 2003.

[3] S.N. Diggavi, N. Al-Dhahir, A. Stamoulis, A.R. Calderbank, "Great expectations: the value of spatial diversity in wireless networks," *Proc. of IEEE* vol. 92, no. 2, pp. 219-270, February 2004.

[4] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, vol. 10, no. 6, pp. 585-595, 1999.

[5] S. M. Alamouti, "A simple trasmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451-1458, October 1998.

[6] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communications: Performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744-765, March 1998.

[7] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Transactions on Information Theory*, vol. 45, pp. 1456-1467, July 1999.

[8] V. Tarokh, A. Naguib, N. Seshadri, and A. R. Calderbank, “Combined array processing and space-time coding,” *IEEE Transactions on Information Theory*, vol. 47, pp. 199-207, Feb. 1999.

[9] G. J. Foschini, Jr., “Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas,” *Bell Labs Tech. Journal*, vol. 1, pp. 41-59, Autumn 1996.

[10] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, “Symplified processing for high spectral efficiency wireless communication employing multiple-element arrays,” *Wireless Personal Communicaitons*, vol. 6, pp. 311-335, March 1999.

[11] M. Sellathurai and S. Haykin, “Turbo-BLAST for wireless communicaitons: Theory and experiments,” *IEEE Transactions on Signal Processing*, vol. 50, pp. 2538-2545, October 2002.

[12] G. J. Foschini, D. Chizhik, M. J. Gans, C. Papadias, and R. A. Valenzuela, “Analysis and performance of some basic space time architectures,” *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 303-320, April 2003.

[13] S. L. Ariyavisitakul, “Turbo space-time processing to improve wireless channel capacity,” *IEEE Transactions on Communications*, vol. 48, pp. 1347-1358, August 2000.

[14] L. Zheng and D. Tse, “Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Transactions on Information Theory*, vol. 49, pp. 1073-1096, May 2003.

[15] D. Palomar, J. Cioffi, and M. Lagunas, "Joint Tx-Rx beamforming design for multicarrier MIMO channels: a unified framework for convex optimization," *IEEE Transactions on Signal Processing*, vol. 51, no.9, pp. 2381-2401, Sep. 2003.

[16] http://www.mathworks.com/matlabcentral/fileexchange/12733-ofdm-water-filling-algorithm

[17] G. G. Rayleigh and J. M. Cioffi, "Spatial-temporal coding for wireless communication," *IEEE Transactions on Communications*, vol. 46, pp. 357-366, March 1998.

[18] A. Scaglione, G. B. Giannakis, and S. Barbarossa, "Filterbank transceiver optimizing information rate in bock transmissions over dispersive channels," *IEEE Transactions on Information Theory*, vol. 45, pp. 1019-1032, April 1999.

[19] Y. Jiang, J. Li, and W. Hager, "Joint transceiver design for MIMO communications using geometric mean decomposition," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3791-3803, Oct. 2005.

[20] Y. Jiang, J. Li, and W. Hager, "Uniform channel decomposition for MIMO communications," *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4283-4294, Nov. 2005.

[21] Y. Jiang and J. Li, "Tunable Channel Decomposition for MIMO Communications Using Channel State Information," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4405-4418, Nov. 2006.

[22] Y. Jiang, W. Hager, and J. Li, "The geometric mean decomposition," *Linear Algebra and Its Applications*, vol. 396, pp. 373-384, Feb. 2005.

[23] Y. Jiang, W. Hager, and J. Li, "The generalized triangular decomposition," *Mathematics of Computation*, vol. 77, no. 262, pp. 1037-1056, Apr. 2008.

[24] N. Al-Dhahir and J. M. Cioffi, "MMSE decision feedback equalizers: Finite-length results," *IEEE Trans. Inform. Theory*, vol. 41, pp. 961–976, July 1995.

[25] J. A. C. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come," *IEEE Trans. Commun*., vol. 28, pp. 5–14, May 1990.

[26] H. G. Myung, J. Lim and D. J. Goodman, "Single carrier FDMA for uplink wireless transmission," *IEEE Vehicular Technology Magazine*, Sep. 2006, pp. 30-38.

[27] BDFT 2010

[28] Xilinx Corporation, "System Generator for DSP User Guide," http://www.xilinx.com/support/sw_manuals/sysgen_user.pdf

[29] J. Wang and B. Daneshrad, "A comparative study of MIMO detection algorithms for wideband spatial multiplexing systems," in *Proc. IEEE WCNC Conf.*, vol. 1, pp. 408-413, 13-17 Mar., 2005.

[30] X. Huang, C. Liang, and J. Ma, "System Architecture and Implementation of MIMO Sphere Decoders on FPGA", *IEEE Transactions on VLSI Systems*, vol. 16, pp. 188-197, Feb. 2008.

[31] C. Dick, K. Amiri, J. R. Cavallaro, and R. Rao, "Design and architecture of spatial multiplexing MIMO decoders for FPGAs", *Asilomar Conference on Signals, Systems and Computers*, pp. 160-164, Oct. 2008.

[32] M. S. Khairy, M. M. Abdallah and S. E. –D. Habib, "Efficient FPGA prototyping of fixed sphere decoder for MIMO systems", *3rd International Design and Test Workshop*, pp. 177-181, Dec. 2008.

[33] S. Haykin, "Cognitive Radio: Brain-empowered Wireless Communications," *IEEE Journal on Selected Areas of Communications*, vol. 23, nr. 2, pp. 201-220, Feb. 2005.

[34] J. Mitola, "The software radio architecture," *IEEE Communications Magazine*, vol.33, no. 5, pp. 26 - 38, May 1995.

[35] J. Mitola, "Software radios - survey, critical evaluation and future directions," *National Telesystems Conference*, pp. 13/15 - 13/23, 1992.

[36] Asad A. Abidi, "The path to the software-defined radio receiver," *IEEE Journal of Solid-State Circuits*, vol. 42. no. 5, pp. 954 - 966, May 2007.

[37] R. Cowen-Hirsch, D. Shrum, B. Davis, D. Stewart, K. Kontson, "Software radio: evolution or revolution in spectrum management," *Proceedings, 21st Century Military Communications Conference (MILCOM)*, vol.1, pp. 8 – 14, 2000.

[38] P. J. Smith, L. M. Garth, and M. Shafi, "Performance analysis of multiple-input multiple-output singular value decomposition transceivers during fading and other cell interference," *IET Microwaves, Antennas & Propagation*, vol. 1, issue 6, pp. 1111 – 1119, December 2007.

[39] *AccelDSP Synthesis Tool User Guide*, Xilinx, Inc., Dec., 2009.

[40] Z. Wang, X. Ma and G. B. Giannakis, "OFDM or Single-Carrier Block Transmissions?" *IEEE Trans. Commun.*, vol. 52, no. 3, pp. 380-394, 2004.

[41] J. Tubbax, L. Van der Perre, M. Engels, H. De Man and M. Moonen, "OFDM versus single carrier: a realistic multi-antenna comparision," *EURASIP Journal on Applied Signal Processing*, 2004:9, pp. 1275-1287.

[42] H. G. Myung, J. Lim and D. J. Goodman, "Single carrier FDMA for uplink wireless transmission," *IEEE Vehicular Technology Magazine*, Sep. 2006, pp. 30-38.

[43] F. Pancaldi, G. M. Vitetta, R. Kalbasi, N. Al-Dhahir, M. Uysal and H. Mheidat, "Single-carrier frequency domain equalization," *IEEE Signal Processing Magazine*, Sep. 2008, pp. 37-56.

[44] C. Ciochina and H. Sari, "A review of OFDMA and single-carrier FDMA," *2010 European Wireless Conference*, pp. 706-710.

[45] D. Sinanović, G. Šišul and B. Modlic, "Comparison of BER characteristics of OFDM and SC-FDMA in frequency selective channels," *18th International Conference on Systems, Signals and Image Processing*, pp. 1-4, 2011.

[46] M. Mendicute, J. Altuna, V. Atxa and J. M. Zabalegui, "Performance comparision of OFDM and FDE single-carrier modulation for spatial multiplexing MIMO systems," *IEEE 5th Workshop on Signal Processing Advances in Wireless Communications*, pp. 532-535, 2004.

[47] Y. Mikaye, K. Kobayashi, K. Komatsu, S. Tanifuji, H. Oguma, N. Izuka, S. Kameda, N. Suematsu, T. Takagi and K. Tsubouchi, "Hybrid single-carrier and multi-carrier system: widening uplink coverage with optimally selecting SDM or joint FDE/antenna diversity," *14th International Symposium on Wireless Personal Multimedia Communications*, pp. 1-5, 2011.

[48] L. Liu, Y. Guan, Y. Zhang and D. Shao, “Design and implementation of single-carrier MIMO transmission with frequency domain equalization,” *5th International Conference on Computer Science and Education*, pp. 56-60, 2010.

[49] V. Chandrasekhar, J. Andrews and A. Gatherer, “Femtocell networks: a survey,” *IEEE Communications Magazine*, vol. 46, issue 9, pp. 59-67, Sep. 2008.

[50] S. Brueck, "Heterogeneous networks in LTE-Advanced," *8th Int. Symposium on Wireless Communication Systems*, pp. 171-175, Nov. 2011

[51] J. Yang and S. Roy, “Joint transmitter and receiver optimizaiton for multiple-input-multiple-output systems with decision feedback,” *IEEE Trans. Inform. Theory*, vol. 40, pp. 1334-1347, Sept. 1994.

[52] N. Al-Dhahir and A. H. Sayed, “The finite length multi-input multi-output MMSE-DFE,” *IEEE Trans. Signal Process*., vol. 48, no. 10, pp. 2921-2936, Oct. 2000.

[53] C. Tidestav, A. Ahlen and M. Sternad, “Realizable MIMO decision feedback equalizers: structure and design,” *IEEE Trans. Signal Process*., vol. 49, no. 1, pp. 121-133, Jan. 2001.

[54] R. F. H. Fischer, “MMSE DFE for high-rate MIMO transmission over channels with ISI,” *5th IEE Int. Conf. on 3G Mobile Communication Technologies*, 2004, pp. 83-87.

[55] H. Lev-Ari and T. Kailath, “Triangular factorization of structured Hermitian matrices,” *Oper. Theory Adv. Appl.*, I. Gohberg, Ed., 1995, vol 18, pp. 301-324.

[56] R. Merched and N. Yousef, “Fast techniques for computing finite-length MIMO MMSE decision feedback equalizers,” *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 701-711, Feb. 2006.

[57] Y. Jiang, J. Li and W. Hagger, “Uniform Channel Decomposition for MIMO Communications,” *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4283-4294, Nov. 2005.

[58] N. Al-Dhahir and A. H. Sayed, “CORDIC-based MMSE-DFE coefficient computation,” *Digital Signal Processing*, vol. 9, no. 3, pp. 178-194, 1999.

[59] B. Yang, “An improved fast algorithm for computing the MMSE decision-feedback equalizer,” Int. J. Electron. Commun., No. 6, pp. 339-345, 1999.

[60] Y. Jiang and M. K. Varanasi, "Extended Uniform Channel Decomposition for MIMO Communications with Intersymbol Interference", Proceedings of the 41st Asilomar Conference on Signals, Systems and Computers, pp. 1549-1553, Nov. 2007.

[61] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu and G. D. Forney Jr., “MMSE decision-feedback equalizers and coding—part I: equalization results,” *IEEE Trans. Commun.*, vol. 43, no. 10, pp. 2582-2594, 1995.

[62] N. Wiener and P. Masani, “The prediction theory of multivariate stochastic processes, II,” *Acta Math.*, vol. 99, no. 1, pp. 93–137, 1958.

[63] A. Duel-Hallen, “Equalizers for multiple input/multiple output channels and PAM systems with cyclostationary input sequences,” *IEEE J. Select Area Comm.*, vol. 10, issue 3, pp. 630-639, 1992.

[64] N. Al-Dhahir and J. M. Cioffi, "MMSE decision-feedback equalizers: finite-length results," *IEEE. Trans. Inform. Theory*, vol. 41, no. 4, pp. 961-975, July 1995.

[65] G. H. Golub and C. F. Van Loan, *Matrix computations*, 3rd ed. Johns Hopkins, 1996.

[66] 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*, vol. 98, no. 1, Jan. 2010.

[67] J. Tubbax, L. Van der Perre, S. Donnay and M. Engels, "Single-carrier communication using decision-feedback equalization for multiple antennas", *IEEE International Conference on Communications*, vol. 4, pp. 2321-2325, 2003.

[68] Y. Zhu and K. Letaief, "Single-carrier frequency-domain equalization with noise prediction for MIMO systems," *IEEE Trans. Commun.*, vol. 55, no. 5, pp. 1063-1076, May 2007.