TECHNISCHE UNIVERSITÄT WIEN

VIENNA UNIVERSITY OF TECHNOLOGY

DISSERTATION

Low-Complexity Real-Time Signal Processing for Wireless Communications

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Doktors der technischen Wissenschaften

eingereicht an der Technischen Universität Wien
Fakultät für Mathematik und Geoinformation

von
Dipl.-Ing. Florian Kaltenberger
Matrikelnummer: 9725857
Marxergasse 30/5, 1030 Wien

Supervisor:	Ao. Univ.-Prof. Dr.techn. Christoph Überhuber Vienna University of Technology, Austria
Examiner:	Prof. Dr.sc.tech. Bernard Fleury Aalborg University, Denmark

Wien, 20. Mai 2007

_____

Kurzfassung

Signalverarbeitungsalgorithmen moderner Mobilkommunikationssysteme stellen sehr hohe Anforderungen an die Rechenleistung der zugrunde liegenden Hardware bei eng begrenztem Platz und Leistungsverbrauch. Diese Algorithmen müssen aber in Echtzeit ausgeführt werden, da sonst Dienste wie z.B. die Sprachtelephonie gar nicht möglich wären. Diese beiden Anforderungen an Signalverarbeitungsalgorithmen—Bewältigung eines hohen Rechenaufwandes und Echtzeitfähigkeit—schließen sich aber oft gegenseitig aus. Die vorliegende Dissertation liefert Lösungen dieses Problems, indem die Komplexität von Signalverarbeitungsalgorithmen auf Kosten der Rechengenauigkeit reduziert wird. Das ist vor allem dann interessant, wenn die Genauigkeit der zugrunde liegenden Hardware oder die Anforderungen an die Rechengenauigkeit des Algorithmus limitiert sind.

Der erste Teil der Dissertation beschäftigt sich mit der Echtzeitsimulation von geometriebasierten Kanalmodellen. Derartige Kanalmodelle werden oft zur Simulation von MIMO (Multiple-Input Multiple-Output) Kanälen eingesetzt, da sie die räumliche Struktur der Mobilfunkkanäle implizit berücksichtigen. Solche Modelle erfordern allerdings eine sehr hohe Rechenleistung, da für jeden Ausbreitungspfad, für jeden Zeitpunkt sowie auch für jede Frequenz eine komplexe Exponentialfunktion ausgewertet werden muss. Auf Echtzeit-Kanalsimulatoren ist daher die Anzahl der Pfade limitiert, die simuliert werden kann.

Zeitvariante, frequenzselektive MIMO Kanäle können mathematisch als vierdimensionale Folgen beschrieben werden, wobei die einzelnen Dimensionen die Zeit, die Frequenz, den Raum am Sender und den Raum am Empfänger repräsentieren. In den meisten Mobilkommunikationssystemen sind diese Folgen bandlimitiert: In der Zeit durch die maximale Dopplerverschiebung, in der Frequenz durch die maximale Verzögerung im Kanal und im Raum durch die Aufspreizung der Ein- bzw. Ausfallswinkel.

Für eine block-basierte Kanalsimulation müssen indexlimitierte Teile dieser Folgen ausgewertet werden. Man kann zeigen, dass solche indexlimitierten Teile einer bandlimitierten Folge einen Unterraum niedriger Dimension aufspannen. Derselbe Unterraum wird auch von mehrdimensionalen DPS (discrete prolate spheroidal) Folgen aufgespannt.

Ein wichtiger Beitrag dieser Arbeit ist ein neuer Algorithmus zur approximativen Berechnung der Projektion eines geometriebasierten Kanalmodells auf diesen

Unterraum in $\mathcal{O}(1)$ Operationen. Der Approximationsfehler kann über die Unterraumdimension kontrolliert werden. Dadurch lässt sich Rechenaufwand und Genauigkeit des Kanalmodells auf die zugrunde liegende Hardware abstimmen. Auf einer Hardware mit 16 bit Fixpunktarithmetik lässt sich dadurch der Rechenaufwand im Vergleich zu einer konventionellen "Sum-of-sinusoids" Methode um mehr als eine Größenordung reduzieren.

Der zweite Teil der Dissertation geht auf die Entwicklung und die Simulation von Empfängerarchitekturen von modernen Mobilfunksystemen ein. Spezielles Augenmerk wird dabei auf Empfängerarchitekturen mit geringer Komplexität und Robustheit gegenüber Interferenz gelegt.

Zuerst untersuchen wir den Durchsatz verschiedener Empfängerarchitekturen des HSDPA (High Speed Downlink Packet Access) Systems von UMTS mittels einer MATLAB Simulationsumgebung. Das hat den Vorteil, dass Algorithmen schnell und einfach in einer höheren Programmiersprache umgesetzt werden können. Ein herkömmlicher Rake-Empfänger, ein Rake-Empfänger mit Interferenzunterdrückung und ein LMMSE (Linear Minimum Mean Square Error) Entzerrer werden untersucht. Wir zeigen, dass ein LMMSE Entzerrer auf Kosten eines höheren Rechenaufwands einen höheren Durchsatz als ein Rake-Empfänger erzielt. Der Rechenaufwand kann jedoch erheblich reduziert werden, wenn man einen iterativen Krylov-Algorithmus zum Lösen des entstehenden Gleichungssystems einsetzt.

Viele Effekte, die in einer Hardwareimplementierung auftreten, können bei einer reinen Software-Simulation nicht berücksichtigt werden. Daher zeigen wir in einem zweiten Beispiel die Entwicklung und Implementierung eines GSM-Empfängers mit Intelligenten Antennen auf einer speziellen Signalverarbeitungshardware. Der Empfänger muss ein generalisiertes Eigenwertproblem in Echtzeit lösen. Wir zeigen, wie dieses Problem numerisch in Fixpunktarithmetik gelöst werden kann. Darüber hinaus zeigen wir Ergebnisse die mit dem ARC SmartSim Kanalsimulator erzielt worden sind.

Abstract

Many signal processing algorithms in modern wireless communication systems come with a very high computational complexity. Moreover, these algorithms must be implemented in real time. Since the computational performance of digital signal processors is limited, these two requirements—dealing with arithmetic complexity and real-time constraints—are often mutually exclusive. This thesis tries to solve this problem by trading the complexity of signal processing algorithms for accuracy. This approach is feasible if the accuracy of the underlying hardware or the accuracy requirements of the algorithm's result is limited.

The first part of this thesis deals with the real-time simulation of geometry-based channel models. Geometry-based channel models implicitly model the spatial structure of the channel and are thus ideally suited for simulating MIMO (multiple-input multiple-output) systems. However, such models are computationally expensive, since for every propagation path, every time instance, and every delay or frequency bin, a complex exponential has to be evaluated. On a real-time hardware channel simulator the number of paths P that can be simulated is thus limited by the available processing power.

Time-variant, frequency-selective MIMO channels can be described as four-dimensional sequences, where the dimensions represent time, frequency, space at the receiver, and space at the transmitter. In wireless communication systems, such sequences are bandlimited by the maximum Doppler frequency in time, by the maximum delay of the channel in frequency, by the angle of arrival spread in the space at the receiver, and by the angle of departure spread at the space at the transmitter.

For block based channel simulation, index-limited parts of these four-dimensional sequences need to be evaluated. It can be shown, that such index-limited parts span a low-dimensional subspace. The same subspace is spanned by the multidimensional discrete prolate spheroidal (DPS) sequences.

An important contribution of this thesis is a new algorithm that allows to compute the projection of a geometry-based channel model onto the DPS subspace in $\mathcal{O}(1)$ arithmetic operations. By adjusting the dimension of the subspace it is possible to trade complexity for accuracy. On a 16 bit fixed-point arithmetic processor, the computational complexity can be reduced by more than one order of magnitude compared to conventional sum-of-sinusoids implementations.

The second part of this thesis addresses the development and simulation of receiver

architectures for state-of-the-art wireless communication systems. Special emphasis is put on low-complexity receiver designs and robustness of the algorithm against interference.

First, the throughput performance of different receiver architectures for the High Speed Downlink Packet Access (HSDPA) subsystem of UMTS are investigated using a MATLAB simulation environment. The advantage of this approach is that ideas can be quickly implemented in a high-level language using a broad range of toolboxes. A conventional rake receiver, a rake receiver with interference cancelation, and a receiver employing a linear minimum mean square error (LMMSE) equalizer are investigated. It is shown that a LMMSE equalizer at the receiver achieves higher throughput than a conventional rake receiver, at the cost of higher complexity. As a consequence, an iterative algorithm based on Krylov subspace projections is introduced, which approximates the LMMSE equalizer with negligible loss of performance. Computational complexity as well as storage requirements are strongly reduced.

Not all effects occurring on the final hardware, such as fixed-point issues and real-time constraints, can be taken into account in a software simulation. Therefore, in a second example, a GSM receiver employing multiple antennas is developed and implemented directly on a DSP board. The smart antenna algorithm requires the solution of an eigenvalue problem in real-time. It is shown how to solve this eigenvalue problem numerically on a fixed-point processor. Real-time simulation results obtained with the ARC SmartSim channel simulator are described.

Acknowledgements

I am grateful that I was able to participate in many joint research projects between industry, applied sciences and theoretical sciences. This thesis is the result of some of these research projects and it is to be understood that this work is the result of a team effort. This means that without the help of my research colleagues, the work would never have reached the quality presented here.

I would like to thank my supervisor Christoph Ueberhuber, who taught me the essence of what a PhD is all about and who always was open to my ideas and interests. I also thank my examiner Bernard Fleury for his interest in my work and his valuable feedback and advice.

Most of the research work of this thesis has been carried out in the research project "Math+MIMO – Future Mobile Communications Systems", a joint effort between ftw. Forschungszentrum Telekommunikation Wien, Austrian Research Centers GmbH – ARC, and Vienna University of Technology. I would like to thank all of the team members, especially Thomas Zemen, the project leader, for being my collaborator, supporter and good friend; Christoph Mecklenbäuker, for his encouragement, advice and the enjoyable extreme programming sessions; Gerhard Humer for his continuous support; Charlotte Dumard and Erwin Riegler for the good collaborations.

I would also like to thank all my colleagues at the smart antennas research group at Austrian Research Centers GmbH for the good team work and spirit: Bernhard Gmeiner, Ylber Hasani, Reinhard Kloibhofer, Samir Krzalic, Roland Lieger, Hannes Pfaffenthaler, Hermann Ritt, Florian Schupfer, Armin Selhofer, Gerhard Steinböck, and Alexander Trinkl.

I am greatful to my colleagues at ftw., Vienna University of Technology and Johannes Keppler University Linz for many interesting discussions and valuable input, especially Nicolai Czink, Klemens Freudenthaler, Franz Hlawatsch, Jürgen Lorenz, Christian Mehlführer, Troels Pedersen, Markus Rupp, and Joachim Wehinger.

Last but not least I would like to thank my friends and my family for their encouragement and for the fact that they were there for me when I needed them. My greatest thanks, however, goes to Gillian for her emotional support and love throughout the years.

This work has been funded by the Wiener Wissenschafts-, Forschungs- und Technologiefonds (WWTF), the Austrian Research Centers GmbH, and also by Kplus through the ftw. projects C3 and C9.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Scope

Mobile communications are a key technology for the 21st century. There is a never ending demand for data rate and mobility. To realize the vision of staying "optimally connected anywhere, anytime", future wireless communication systems must support high data rates for users with high mobility (traveling by car or train). Fig. 1.1 depicts mobility vs. bit rate of current (2nd and 3rd generation) and next generation (4th generation) wireless communication systems [109].

The performance of wireless communication systems is inherently limited by the wireless channel. There are two physical means to increase the data rates of current wireless communication systems without increasing the power at the transmitter: increasing the channel bandwidth [101], and using multiple transmit and receive antennas [49, 106]. When designing such systems, the frequency domain and the spatial domain of the wireless channel needs to be taken into account.

The wireless channel is characterized by multipath propagation. Signals transmitted from one transmit antenna are scattered by various objects in the environment on their way to the receive antenna. A single transmitted signal is therefore replicated and arrives at a receiver at different times, with different amplitudes, polarities, angles of arrival and possibly at different frequencies.

Multipath propagation causes time-, frequency-, and space-selective fading. Mobility of either transmitter or receiver causes different Doppler shifts in the channel and thus time selective fading. Scattering objects at different distances cause delays between the impinging signals and thus frequency selective fading. Last but not least, if multiple transmit and multiple receive antennas are used, spatial selective fading can be observed.

For the simulation of wireless communication systems, channel models are required. Especially for testing mobile radio hardware devices, real-time implementations of such channel models are needed. The channel model must be able to reproduce all properties of the wireless channel that are relevant to the performance of the system.

For example, first generation mobile communication systems are mainly narrowband systems and thus simple path loss models like the Okumura-Hata model [52]

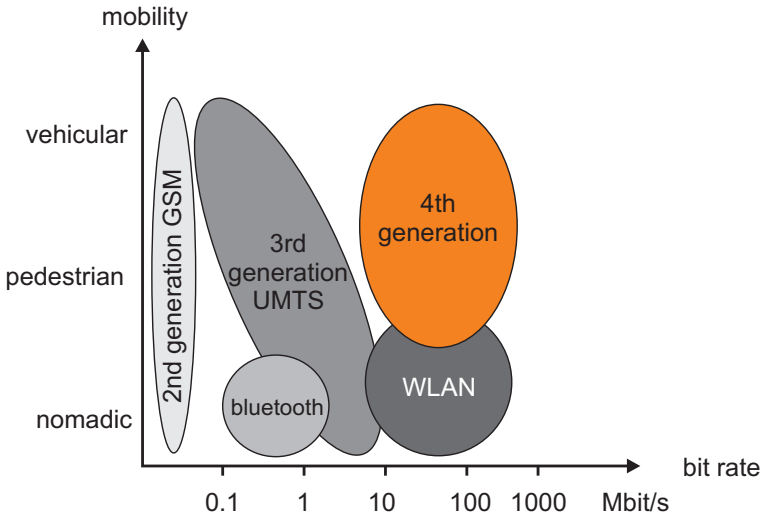

Figure 1.1: Mobility versus bit rate for existing and future mobile communications systems [109].

are sufficient. Second and third generation wireless communication systems require time-variant frequency-selective channel models. The most prominent examples are the COST 207 models [44], which were instrumental for the development of the GSM system. Fourth generation wireless communication systems employing multiple-input multiple-output (MIMO) technology require spatial channel models. In the COST 259 [34] action, a geometry-based stochastic channel model (GSCM) for the directional characteristics in macro-, micro-, and picocells was derived. Later on, this model was further refined in COST 273 [35] to include more scenarios and MIMO channels as well.

The computational complexity of the channel models increases from generation to generation. Especially geometry-based channel models are computationally intensive, since for every propagation path, every time instance, and every delay or frequency bin a complex exponential function evaluation is needed. Thus, on a real-time hardware channel simulator, like the ARC SmartSim [7], the number of paths P that can be simulated, is limited by the available processing power.

Not only the computational complexity of channel models, but also the computational complexity of receiver algorithms increases from generation to generation. On the other hand, the computational performance of digital signal processors (DSPs) does not grow at the same rate [94, 98].

To close the gap between algorithmic complexity on one hand and constraints of DSPs regarding energy, space, and computational power, low-complexity algorithms are needed. Such algorithms can be obtained, for example, by trading accuracy for efficiency. In this thesis, this concept is applied to the simulation of wireless

communication channels as well as to receiver algorithms for various communication standards. Issues like real-time constraints, numerical stability, numerical accuracy and complexity are considered. As a target architecture, the ARC SmartSim [7] development platform is used.

1.2 Synopsis

This thesis is divided in two parts. The first part deals with the low-complexity real-time simulation of wireless frequency- and time-selective MIMO communication channels. The focus is put on geometry-based channel models. The second part focuses on the development and implementation of low-complexity receiver architectures for the wireless communication standards UMTS-HSDPA and GSM.

Part I. Chapter 2 provides an introduction into characterization, modeling, and simulation of wireless communication channels.

Chapter 3 introduces multidimensional discrete prolate spheroidal (DPS) sequences. Multidimensional DPS sequences are a generalization of the DPS sequences—studied by Slepian et al. [103]—to many dimensions. However, unlike the one-dimensional case, the efficient and numerically stable calculation of multidimensional DPS sequences is not trivial and has not been treated satisfactorily in the literature. In Chapter 3 a theory of multidimensional DPS sequences based on the Kronecker product formalism is established, which allows for the efficient computation of multidimensional DPS sequences. Parts of this work have been published on [10, 19].

In **Chapter 4** the multidimensional DPS sequences are used for a subspace representation of time-variant, frequency-selective MIMO channels. The most important contribution of this chapter is an algorithm that allows to compute the projection onto the DPS subspace in $\mathcal{O}(1)$ arithmetic operations. Thus, the subspace method allows to trade accuracy for efficiency. Using a 16 bit fixed-point processor, the computational complexity can be reduced by more than an order of magnitude. Parts of this work have been published in [8–10].

In **Chapter 5** the results of the previous chapter are applied to the ARC SmartSim real-time MIMO channel simulator. For this purpose, the DPS subspace representation is applied independently to every link of the MIMO channel. It can easily be implemented as an add-on to the COST 259 GSCM of the ARC SmartSim channel simulator [7]. We show that the functionality of the new implementation is equivalent to the one of the implementation described in [7]. However, the complexity of the new method is significantly smaller for a wide range of scenarios. Parts of this work have been published in [6].

Chapter 6 gives a summary and conclusions of part one of the thesis.

Part II. The second part of the thesis starts in **Chapter 7** with an introduction into the wireless communication standards GSM and UMTS-HSDPA . The following two chapters describe specific implementations of a receiver based on these standards.

Chapter 8 evaluates the throughput performance of different receiver architectures for UMTS-HSDPA. For this purpose, we develop a simulation environment specifically tailored to HSDPA for investigating the HSDPA receiver requirements. A conventional rake receiver, a rake receiver with interference cancelation, and a receiver employing a linear minimum mean square error (LMMSE) equalizer are investigated. Last but not least, it is shown how to derive a a low-complexity implementation of the LMMSE equalizer using Krylov subspace methods. Parts of this work have been published in [1–4, 17].

Chapter 9 describes a GSM receiver employing multiple antennas. The smart antenna algorithm requires the solution of an eigenvalue problem in real-time. We show, how to solve this eigenvalue problem numerically on a fixed-point processor. An implementation and results obtained with the ARC SmartSim channel simulator are given. Parts of this work have been published in [5, 7, 16].

Finally, **Chapter 10** summarizes and concludes part two of the thesis.

Part I

Wireless Channels

2 Characterization, Modeling, and Simulation of Wireless Channels

2.1 Multipath Propagation

The mobile radio channel is a very harsh environment. Signals transmitted from one transmit antenna are reflected, refracted, diffracted, or scattered on their way to the receive antenna. Such a multipath environment causes a single transmit signal to be replicated—the equivalent of echoes in an audio environment—and arrive at a receiver at different times, with different amplitudes, polarization, angles of arrival and possibly at different frequencies (see Fig. 2.1).

The received signal power varies due to three effects: mean propagation path loss, large scale fading, and small scale fading. The mean propagation path loss is proportional to d^{-n}, where d is the distance between the transmitter (Tx) to the receiver (Rx) and the path loss exponent n varies from 2.5 to 6 depending on the terrain and foliage. Large scale fading results from blocking effects by buildings and natural features. Small scale fading results from the constructive and destructive combination of multipaths [86]. All three propagation effects can be observed in Fig. 2.2.

In this thesis we are mainly interested in the small scale effects of the channel. Small scale fading causes the channel to change over time, frequency, and space. A channel has selectivity if it varies as a function of either time, frequency, or space. The opposite of selectivity is coherence.

2.1.1 Temporal Selectivity

When the Tx, Rx, or the scatterers in the channel are moving, the received signal exhibits Doppler shifts, i. e., a pure tone of frequency ω_C spreads over a finite spectral bandwidth $[\omega_C - \omega_{\mathrm{Dmax}}, \omega_C + \omega_{\mathrm{Dmax}}]$. Assuming that the scattering objects are static, the maximum Doppler frequency ω_{Dmax} is given by the maximum speed $v_{\max}$ of the user

$$\omega_{\mathrm{Dmax}} = \frac{v_{\max}\omega_C}{c},$$

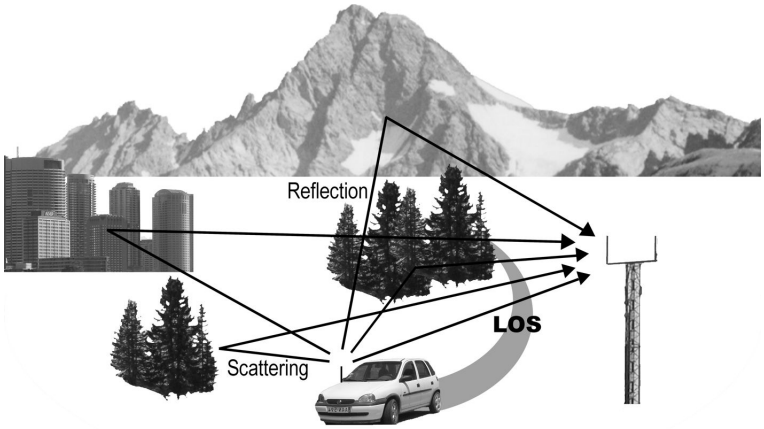

Figure 2.1: Multipath propagation in mobile radio channels. Transmitted signals are reflected and refracted on their way to the receiver.

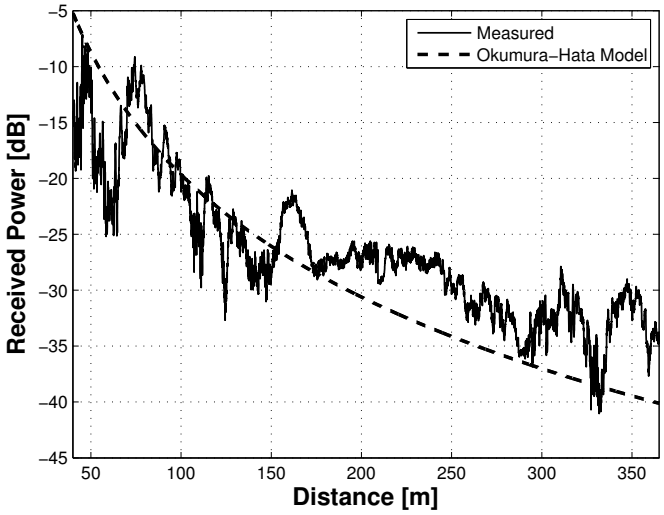

Figure 2.2: Measured received signal power vs. distance between Tx and Rx in a typical suburban environment (city of Weikendorf near Vienna [53]). The mean path loss given by the Okumura-Hata Model [52] is also given. The large scale fading as well as the small scale fading are clearly visible.

where c is the speed of light. If $\omega_{\mathrm{Dmax}} > 0$, waves impinging at the receiver with different Doppler frequencies can combine constructively and destructively causing selectivity in time. Time-selective channels are also called time-variant. Non time-selective channels are called static.

In practice, the question if a channel is time-selective does not only depend on the maximum Doppler shift ω_{Dmax}, but also on other system parameters, e. g., block-length, type of modulation, training, etc. For example, if the channel changes the phase of the incoming signal by less than $\pi/4$ within a block and BPSK modulation is used, no detection errors will occur. If in the same channel 16QAM modulation is used, symbols will be inevitably wrongly detected.

2.1.2 Frequency Selectivity

If the transmitted signal arrives at the receiver via different paths with different delays, one symbol spreads to the next symbol(s) and causes inter-symbol interference (ISI). If the maximum delay of the channel $\tau_{\max} > 0$, waves impinging at the receiver at different delays can combine constructively and destructively and cause selectivity in frequency. A non frequency-selective channel is also called frequency-flat.

In practice, the question if a channel is frequency-selective also depends on the symbol duration T_S of the system. If the maximum delay of the channel is very small compared to the symbol duration, i. e.,

$$\tau_{\max} \ll T_S,$$

ISI is negligible and the channel can be assumed to be frequency-flat.

Fig. 2.3 shows the relationship between frequency-selective, frequency-flat, time-selective and static channels.

2.1.3 Space Selectivity

If the transmitted signal leaves the transmitter via different paths that eventually reach the receiver at different directions of departure, the channel changes with the position of the transmitter. Equivalently, if the transmitted signal reaches the receiver via different paths with different directions of arrivals, the channel changes with the position of the receiver. Channels with a directional spread at the transmitter $\varphi_{\max} > 0$ as well as channels with a directional spread at the receiver $\psi_{\max} > 0$ are spatially selective.

In practice, the question if a channel is spatially selective also depends on the type of antennas used. For example, space selective fading can be observed using directional antennas or antenna arrays at the receiver and transmitter.

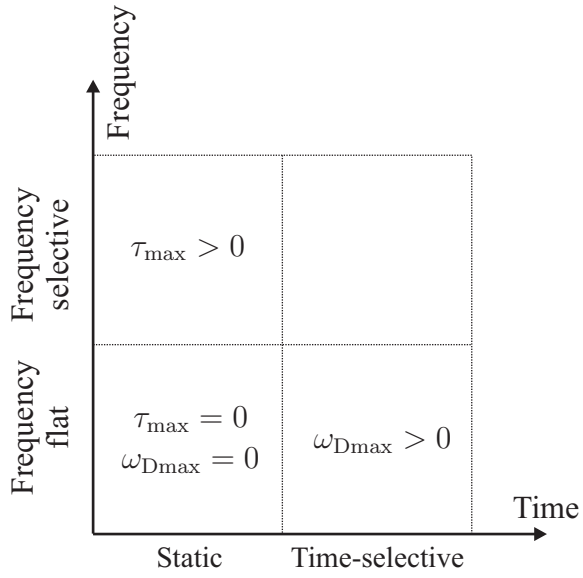

Figure 2.3: Time and frequency variation of a multipath channel. Frequency-selectivity is characterized by the maximum delay $\tau_{\max}$. Time-selectivity is characterized by the maximum Doppler shift ω_{Dmax}.

2.2 The Signal Model

In the last section, the basic effects of multipath propagation in mobile radio channels were described. In this and the following sections, a mathematical model for the mobile radio channel is presented. The model allows a stochastic characterization of the multipath effects.

We start by describing time-variant frequency-selective single-input single-output (SISO) channels first in Section 2.2.1. In Section 2.2.2 the concepts are extended to multiple-input multiple-output (MIMO) channels.

2.2.1 SISO Channels

The basic SISO system model is depicted in Fig. 2.4. It consists of a transmitter with a pulse shaping filter, a wireless channel, and a receiver with a matched filter.

In this thesis we adopt Bello's [28] approach of describing the mobile radio SISO channel $\mathcal{H}$ as a linear time-variant system. Such a system is entirely characterized by one of the four equivalent system functions, such as the time-varying transfer function, the time-variant impulse response, and the Doppler-delay spreading function.

The *time-varying transfer function* $h(t, f)$ characterizes the mobile radio channel as a function of time t and frequency f. By using a (partial) Fourier transform,

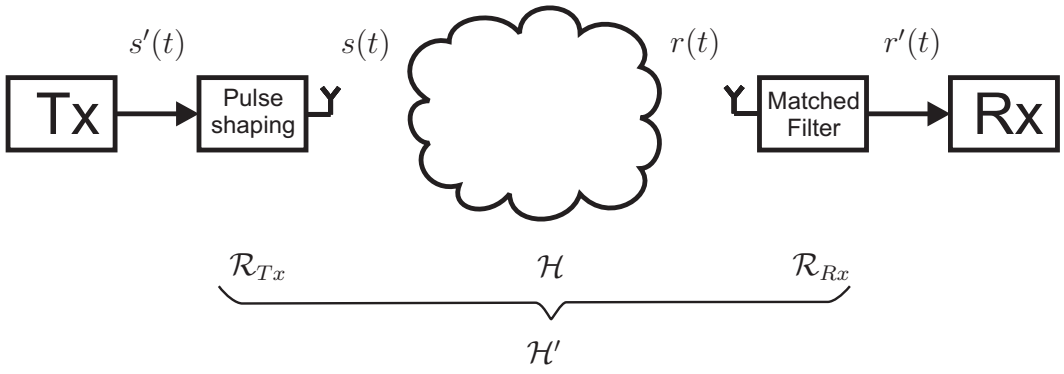

Figure 2.4: System model for SISO channels. The actions of the pulse shaping filter, the wireless radio channel including the effects of the transmit and receive antenna, and the matched filter are described by the operators $\mathcal{R}_{Tx}$, $\mathcal{H}$, and $\mathcal{R}_{Rx}$ respectively. The overall system is described by the operator $\mathcal{H}'$.

spectral domain representations of the time-varying transfer function can be obtained. The spectral domain of time t is the Doppler frequency domain, denoted by ω. The spectral domain of the frequency f is the delay domain, denoted by τ. Transformed system functions are denoted by uppercase letters $H(\cdot,\cdot)$, where the dependency variables indicate the transformed domain.

For example, the *time-variant impulse response* is given by $H(t,\tau)$ and can be obtained from the time-varying transfer function $h(t,f)$ by

$$H(t,\tau) = \int h(t,f)e^{2\pi jf\tau}\mathrm{d}\tau. \tag{2.1}$$

The *Doppler-delay spreading function* $H(\omega,\tau)$ can be obtained either from the time-variant impulse response by a partial Fourier transform with respect to t, or by a full Fourier transform of the time-varying transfer function,

$$H(\omega,\tau) = \int H(t,\tau)e^{-2\pi jt\omega}\mathrm{d}t \tag{2.2}$$

$$= \iint h(t,f)e^{-2\pi i\omega t}e^{2\pi i\tau f}\mathrm{d}t\,\mathrm{d}f. \tag{2.3}$$

The action of the channel $\mathcal{H}$ on an input signal $s(t)$ can now be written in three different but equivalent ways:

$$r(t) = \int H(t,\tau)s(t-\tau)\mathrm{d}\tau \tag{2.4}$$

$$= \iint H(\omega, \tau) s_{\omega,\tau}(t) \mathrm{d}\omega \, \mathrm{d}\tau \tag{2.5}$$

$$= \int h(t, f) S(f) e^{2\pi j f t} \mathrm{d}f, \tag{2.6}$$

where

$$s_{\omega,\tau}(t) := s(t-\tau) e^{2\pi j \omega t}$$
$$S(f) := \int s(t) e^{-2\pi j f t} \mathrm{d}t.$$

The action of the pulse shaping filter at the transmitter $\mathcal{R}_{\mathrm{Tx}}$ and the matched filter at the receiver $\mathcal{R}_{\mathrm{Rx}}$ can be described as a convolution of the input signal with the filter impulse responses $r_{\mathrm{Tx}}(\tau)$ and $r_{\mathrm{Rx}}(\tau)$ respectively.

$$s(t) = r_{\mathrm{Tx}}(t) * s'(t) := \int r_{\mathrm{Tx}}(\tau) s'(t-\tau) \mathrm{d}\tau,$$
$$r'(t) = r(t) * r_{\mathrm{Rx}}(t) := \int r_{\mathrm{Rx}}(\tau) r(t-\tau) \mathrm{d}\tau.$$

The action of the filters and the channel can also be combined to a single operator $\mathcal{H}' = \mathcal{R}_{\mathrm{Rx}} \circ \mathcal{H} \circ \mathcal{R}_{\mathrm{Tx}}$, which can be described by one of the four systems functions $h'(t, f)$, $H'(t, \tau)$, $H'(\omega, f)$, or $H'(\omega, \tau)$.

2.2.2 MIMO Channels

The basic MIMO system model is depicted in Fig. 2.5. It consists of a transmitter with a pulse shaping filter and an antenna array, a wireless channel, and a receiver with multiple antennas and a matched filter. For the moment we consider continuous arrays which are composed of an infinite number of antennas separated by infinitesimal distances (In Section 2.4, we will derive a discrete model from the continuous model). This eliminates the need to specify a priory the number of antennas and their relative positions on the antenna arrays [90]. The transmit signal $s(t, \boldsymbol{x})$ and the receive signal $r(t, \boldsymbol{y})$ therefore become functions of time and space. The positions $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors in $\mathbb{R}^3$, measured from an arbitrary but fixed reference point at the transmitter and the receiver, respectively.

The characterization of time-variant linear SISO channels can be extended to MIMO channels by incorporating the spatial domain into the channel transfer function [41,46,60,90]. Let $h(t, f, \boldsymbol{x}, \boldsymbol{y})$ denote the *time- and space-variant transfer function*. As in the SISO case, we denote transformed system functions by $H(\cdot, \cdot, \cdot, \cdot)$, where the dependency variable indicates the transformed domain. For example, $H(t, \tau, \boldsymbol{x}, \boldsymbol{y})$ denotes the *time- and space-variant impulse response*.

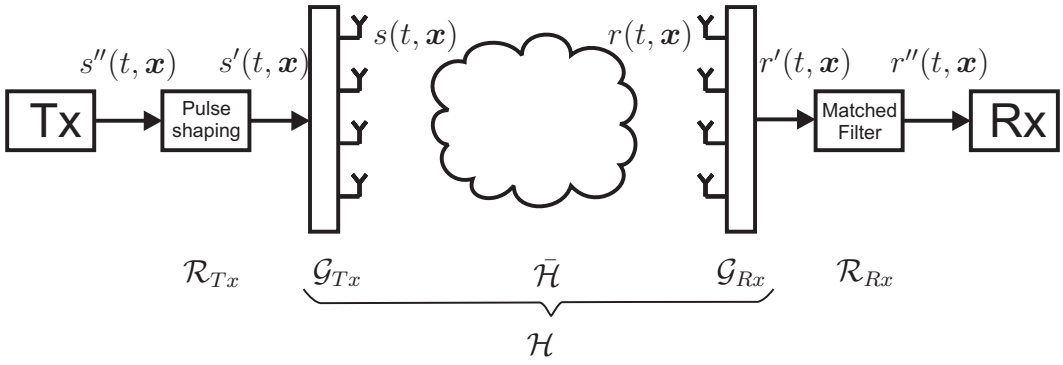

Figure 2.5: System model for MIMO channels. The actions of the pulse shaping filter, the transmit antenna array, the wireless radio channel, the receive antenna array, and the matched filter are described by the operators $\mathcal{R}_{Tx}$, $\mathcal{G}_{Tx}$, $\bar{\mathcal{H}}$, $\mathcal{G}_{Rx}$, and $\mathcal{R}_{Rx}$ respectively. The wireless radio channel including the effects of the transmit and receive antenna is described by the operator $\mathcal{H}$.

In a time- and frequency selective MIMO channel, the receive signal is related to the transmit signal by

$$\begin{aligned} r(t, \boldsymbol{y}) &= \iint H(t, \tau, \boldsymbol{x}, \boldsymbol{y}) s(t - \tau, \boldsymbol{x}) \, \mathrm{d}\tau \, \mathrm{d}\boldsymbol{x} \\ &= \iint h(t, f, \boldsymbol{x}, \boldsymbol{y}) S(f, \boldsymbol{x}) e^{2\pi j f t} \, \mathrm{d}f \, \mathrm{d}\boldsymbol{x}, \end{aligned}$$

where

$$S(f, \boldsymbol{x}) = \int s(t, \boldsymbol{x}) e^{-2\pi j f t} \, \mathrm{d}t.$$

The time- and space-variant impulse response and transfer function also include the effects of the antennas (antenna patterns and mutual coupling). A separation of the effects of the antennas from the channel is only possible in the far field of the antennas. This will be discussed next.

The spectral domains of the position vector $\boldsymbol{x}$ and $\boldsymbol{y}$ are the wavevector domains, with dependence denoted by $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$ respectively. Transformations between position and wavevector domains are obtained by a three-dimensional Fourier transform [41]. For example, the time- and space-variant transfer function can be transformed in the wavevector domain by

$$H(t, f, \boldsymbol{\zeta}, \boldsymbol{\xi}) = \iint h(t, f, \boldsymbol{x}, \boldsymbol{y}) e^{-2\pi j / \lambda \langle \boldsymbol{\zeta}, \boldsymbol{x} \rangle} e^{-2\pi j / \lambda \langle \boldsymbol{\xi}, \boldsymbol{y} \rangle} \, \mathrm{d}\boldsymbol{x} \, \mathrm{d}\boldsymbol{y},$$

where λ is the wavelength and $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \sum_{l=0}^{2} x_l y_l$ is the scalar product.

When looking at the far field of the mobile radio channel only, the wavevector spectrum is non-zero only for $\boldsymbol{\zeta}, \boldsymbol{\xi} \in \mathbb{S}_2$, the unit sphere in $\mathbb{R}_3$. In that case, the spectral domains of the position vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ can be equivalently and more intuitively represented by the angular spectra. The angular spectra are functions of the elevational and the azimuthal angle. However, to keep notation simple, we assume horizonal propagation only in this thesis and disregard the elevation angle. In case the channel is modeled as sum of randomly phased, uncorrelated homogeneous plane waves, the wavevector spectrum is equivalent to the angular spectrum [41].

We can thus define the *time-variant double-directional channel transfer function* $H(t, f, \varphi, \psi)$, which describes the complex weight of a signal leaving the transmitter at the azimuth of departure (AoD) φ and received from the azimuth of arrival (AoA) ψ at time t and frequency f. Similarly, the *Time-variant double-directional impulse response* $H(t, \tau, \varphi, \psi)$ and the *Doppler-delay double-directional spreading function* $H(\omega, \tau, \varphi, \psi)$ can be defined.

The time-variant double directional channel impulse response $H(t, \tau, \varphi, \psi)$ can now be factored in three terms describing the effect of the transmit array, the effects of the physical channel, and the effect of the receive array. Let $G_{\mathrm{Tx}}(\varphi)$ denote the complex far field antenna pattern at the transmitter and $G_{\mathrm{Rx}}(\psi)$ denote the complex far field antenna pattern at the receiver. Further let $\bar{H}(t, \tau, \varphi, \psi)$ be the time-variant double-directional impulse response of the physical channel (without the effects of the transmitter or receiver). Then

$$H(t, \tau, \varphi, \psi) = G_{\mathrm{Tx}}(\varphi)\bar{H}(t, \tau, \varphi, \psi)G_{\mathrm{Rx}}(\psi).$$

The time- and space-variant system impulse response and the time- and space-variant system transfer function can be recovered from the double-directional functions by setting

$$\boldsymbol{\zeta} = \begin{pmatrix} \cos\varphi \\ \sin\varphi \end{pmatrix}, \quad \text{and} \quad \boldsymbol{\xi} = \begin{pmatrix} \cos\psi \\ \sin\psi \end{pmatrix},$$

and writing

$$H(t, \tau, \boldsymbol{x}, \boldsymbol{y}) = \iint H(t, \tau, \varphi, \psi) e^{2\pi j/\lambda \langle \boldsymbol{\zeta}, \boldsymbol{x} \rangle} e^{2\pi j/\lambda \langle \boldsymbol{\xi}, \boldsymbol{y} \rangle} \, \mathrm{d}\varphi \, \mathrm{d}\psi,$$

$$h(t, f, \boldsymbol{x}, \boldsymbol{y}) = \iint H(t, f, \varphi, \psi) e^{2\pi j/\lambda \langle \boldsymbol{\zeta}, \boldsymbol{x} \rangle} e^{2\pi j/\lambda \langle \boldsymbol{\xi}, \boldsymbol{y} \rangle} \, \mathrm{d}\varphi \, \mathrm{d}\psi.$$

2.3 Stochastic Channel Characterization

So far, the mobile radio channel was described by deterministic functions. However, a deterministic description of the mobile radio channel is far to complex, since every scenario would have to be modeled in every detail. In this section, a stochastic

characterization of the mobile radio channel is given. For the remainder of this chapter, we assume that the channel functions defined in the last section are stochastic processes, i. e., a family of functions [83].

2.3.1 Autocorrelation Functions

The autocorrelation function (ACF) of a stochastic process, e. g., the time-variant channel transfer function $h(t)$ is given by

$$R_h(t, t') = \mathcal{E}\{h(t)h^*(t')\}.$$

The ACF provides a full description of a stochastic process if and only if (iff) the process is zero-mean Gaussian. The ACF of $h(t)$ is related to the ACF of $H(\omega)$ by a double Fourier transform [78]

$$R_H(\omega, \omega') = \iint R_h(t, t')e^{-2\pi j\omega t}e^{2\pi j\omega' t'} \,\mathrm{d}t \,\mathrm{d}t'. \tag{2.7}$$

ACFs can be defined for all channel transfer functions described in Section 2.2 including all dependencies. For example, the ACF of the space- and time-variant channel transfer function writes

$$R_h(t, t', f, f', \boldsymbol{x}, \boldsymbol{x}', \boldsymbol{y}, \boldsymbol{y}') = \mathcal{E}\{h(t, f, \boldsymbol{x}, \boldsymbol{y})h^*(t', f', \boldsymbol{x}', \boldsymbol{y}')\}.$$

A stochastic process $h(t)$ is called wide-sense stationary (WSS) iff its ACF $R_h(t, t')$ only depends on the time difference $t' - t$, i. e., $R_h(t, t') = R_h(t - t') = R_h(\Delta t)$. Including all dependencies, the channel transfer function $h(t, f, \boldsymbol{x}, \boldsymbol{y})$ is WSS if

$$R_h(t, t', f, f', \boldsymbol{x}, \boldsymbol{x}', \boldsymbol{y}, \boldsymbol{y}') = R_h(t - t', f - f', \boldsymbol{x} - \boldsymbol{x}', \boldsymbol{y} - \boldsymbol{y}').$$

The second order statistics of a WSS stochastic process do not change over time. An alternative interpretation of the WSS process $h(t)$ is obtained by investigating the ACF of its Fourier transform $H(\omega)$. Using Equation (2.7),

$$\begin{aligned} R_H(\omega, \omega') &= \iint R_h(\Delta t)e^{2\pi j\omega' t}e^{-2\pi j\omega(t+\Delta t)} \,\mathrm{d}t \,\mathrm{d}t + \Delta t \\ &= \int e^{2\pi jt(\omega'-\omega)} \,\mathrm{d}t \int R_h(\Delta t)e^{-2\pi j\omega\Delta t} \,\mathrm{d}\Delta t \\ &= \delta(\omega' - \omega)S_h(\omega), \end{aligned}$$

where $\delta(\omega' - \omega)$ is the Dirac delta function and $S_h(\omega)$ is the Doppler power spectrum of $h(t)$ (see the next subsection for a detailed discussion). Thus, in a WSS process, contributions with different Doppler shifts are uncorrelated. This fact is also called uncorrelated scattering (US). Since WSS and US are equivalent descriptions of a stochastic process in different domains, in this thesis, only the term WSS is used.

2.3.2 Power Spectra

The power spectrum of a WSS stochastic process $h(t)$ is defined by

$$S_h(\omega) = \mathcal{E}\{|H(\omega)|^2\},$$

where $H(\omega)$ is the Fourier transform of $h(t)$. By the Wiener-Khintchine Theorem [93], the power spectrum $S_h(\omega)$ is related to the ACF $R_h(\Delta t)$ by a Fourier transform:

$$S_h(\omega) = \int R_h(\Delta t) e^{-2\pi j \Delta t \omega} \, \mathrm{d}\Delta t.$$

The power spectrum $S_h(\omega)$ of a temporal process $h(t)$ is called a *Doppler spectrum*; it describes the average power received at a certain Doppler frequency ω. The power spectrum $S_h(\tau)$ of a frequency process $h(f)$ is called a *power delay spectrum*; it describes the average power received at a certain delay τ.

The power spectrum $S_h(\boldsymbol{\xi})$ of a spatial process $h(\boldsymbol{y})$ is called the *wavevector power spectrum*. When looking at the far field of the mobile radio channel only, the wavevector power spectrum simplifies to an angular power spectrum $S_h(\psi)$, which describes the average power received at a certain AoA ψ.

For the complete time- and space-variant channel transfer function $h(t, f, \boldsymbol{x}, \boldsymbol{y})$, it is possible to define a joint power spectrum that is a function of Doppler ω, delay τ, and wavevectors $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$:

$$S_h(\omega, \tau, \boldsymbol{\zeta}, \boldsymbol{\xi}) = \mathcal{E}\{|H(\omega, \tau, \boldsymbol{\zeta}, \boldsymbol{\xi})|^2\}.$$

The joint power spectrum is related to the joint ACF $R_h(\Delta t, \Delta f, \Delta \boldsymbol{x}, \Delta \boldsymbol{y})$ by the Wiener-Khintchine Theorem,

$$S_h(\omega, \tau, \boldsymbol{\zeta}, \boldsymbol{\xi}) = \iiiint R_h(\Delta t, \Delta f, \Delta \boldsymbol{x}, \Delta \boldsymbol{y}) \\ e^{-2\pi j \Delta t \omega} e^{-2\pi j \Delta f \tau} e^{-2\pi j \langle \Delta \boldsymbol{x}, \boldsymbol{\zeta} \rangle} e^{-2\pi j \langle \Delta \boldsymbol{y}, \boldsymbol{\xi} \rangle} \, \mathrm{d}\Delta t \, \mathrm{d}\Delta f \, \mathrm{d}\Delta \boldsymbol{x} \, \mathrm{d}\Delta \boldsymbol{x}. \quad (2.8)$$

Looking only at the far field, we can again describe the joint power spectrum of the mobile radio channel more intuitively by a function of Doppler ω, delay τ, AoD ϕ, and AoA ψ:

$$S_h(\omega, \tau, \varphi, \psi) = \mathcal{E}\{|H(\omega, \tau, \varphi, \psi)|^2\}.$$

Power spectra depending on three, two or one variable can be obtained from the joint power spectrum $S_h(\omega, \tau, \varphi, \psi)$ by integrating out the unwanted variable(s). For example, the *angular delay power spectrum* can be obtained by

$$S_h(\tau, \varphi, \psi) = \int S_h(\omega, \tau, \varphi, \psi) \mathrm{d}\omega.$$

A channel is said to be separable, if its joint power spectrum can be written as a product of its corresponding single power spectra, i. e., if

$$S_h(\omega, \tau, \varphi, \psi) = S_h(\omega) S_h(\tau) S_h(\varphi) S_h(\psi).$$

2.4 The Sampled Signal Model

Until now, the channel was modeled as a continuous function of time, frequency, and space. For practical purposes, like channel estimation or simulation, the channel has to be sampled in time, frequency, and space.

2.4.1 SISO channels

Due to the bandlimiting filters at the transmitter and the receiver, the wireless channel can be assumed to be bandlimited, i. e., $h(t, f) = 0$ for $|f| > B$. This allows to sample the channel also in the delay domain. Denote by $B = 1/T_S$ is the sampling rate in time and by $1/F_S$ the sampling rate in frequency. The sampled impulse response and the sampled transfer function then write

$$H_{m,n} = H(mT_S, nT_S), \tag{2.9}$$

$$h_{m,q} = h(mT_S, qF_S), \tag{2.10}$$

where n and m are the discrete time and delay indices and q is the discrete frequency index.

In order to be able to reconstruct the continuous channel from its samples, by the sampling theorem we have to choose

$$\frac{1}{T_S} \geq \frac{1}{2\omega_{\mathrm{Dmax}}} \quad \text{and} \quad \frac{1}{F_S} \geq \tau_{\max},$$

where $2\omega_{\mathrm{Dmax}}$ is the maximum Doppler bandwidth and $\tau_{\max}$ is the maximum delay of the channel. While the first condition is always fulfilled in mobile communication channels, F_S has to be chosen carefully to satisfy the latter condition. Often the sampling rate in frequency is chosen as $1/F_S = T_S Q$, where Q is the number of frequency bins within the bandwidth B. Then the second condition becomes $Q \geq \tau_{\max}/T_S$.

The input-output relation for a sampled SISO channel now writes

$$r_m = \sum_{n=0}^{N-1} H_{m,n} s_{m-n} \tag{2.11}$$

where $s_m = s(mT_S)$ and $r_m = r(mT_S)$ are the sampled input and output signals and N is the maximum support of $H_{m,n}$ with respect to n. Due to the bandlimiting filters $r_{\mathrm{Tx}}(\tau)$ and $r_{\mathrm{Rx}}(\tau)$, the support of $H_{m,n}$ is actually infinite. However, the contributions for large n will be very small, so that they can be neglected. We can assume that $N = N_C + 2N_F + 1$, where $N_C = \lceil \tau_{\max}/T_S \rceil$ and $2N_F + 1$ is the length of the filters $r_{\mathrm{Tx}}(\tau)$ and $r_{\mathrm{Rx}}(\tau)$.

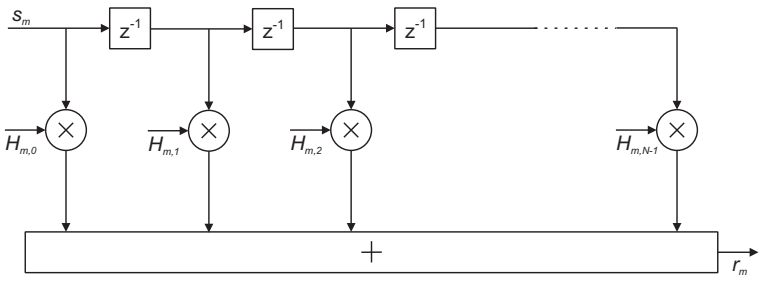

Figure 2.6: Tapped delay line implementation of a sampled SISO channel.

Equation (2.11) corresponds to modeling the channel as a tapped delay line or a linear time-variant finite impulse response (FIR) filter. An implementation of the FIR filter is depicted in Fig. 2.6.

Alternatively, the input-output relation can be written using the sampled transfer function,

$$r_m = \sum_{q=0}^{Q-1} h_{m,q} S_q e^{-2\pi j m q/Q}, \tag{2.12}$$

where $S_q = \sum_m s_m e^{2\pi j m q}$ is the discrete time Fourier transform (DTFT) of the input signal s_m. In practice however, (2.12) is implemented using the overlap-add, or overlap-save method [93] since the input signal s_m is not known a priori for all time $m \in \mathbb{Z}$.

In the overlap-add method, the input signal s_m is segmented into blocks of length $K = Q - N + 1$ and $N - 1$ zeros are appended to every block to get blocks of length Q:

$$\boldsymbol{s}^{(b)} = [\underbrace{s_{bK}, \ldots, s_{(b+1)K-1}}_{K}, \underbrace{0, \ldots, 0}_{N-1}]^T. \tag{2.13}$$

Further define

$$\boldsymbol{h}^{(b)} = [h_{bK,0}, \ldots, h_{bK,Q-1}]^T. \tag{2.14}$$

Then of every block, the discrete Fourier transform (DFT) or its fast implementation, the fast Fourier transform (FFT) is taken of every block to get

$$\boldsymbol{S}^{(b)} = \mathbf{F}_Q \boldsymbol{s}^{(b)}. \tag{2.15}$$

The b-th output block is given by

$$\boldsymbol{r}^{(b)} = \mathbf{F}_Q^{-1}(\boldsymbol{S}^{(b)} \odot \boldsymbol{h}^{(b)}), \tag{2.16}$$

where $\odot$ denotes the Hadamard (element-wise) product. The vectors $\boldsymbol{r}^{(b)}$ are finally overlapped by $N-1$ samples and added to yield

$$r_m = \sum_b r^{(b)}_{m-bK}, \tag{2.17}$$

where $r^{(b)}_k := 0$ if $k \notin \{0, \ldots, Q-1\}$.

From the derivations above it can be seen that the overlap-add method assumes that the channel $h_{m,q}$ is constant for K samples (cf. (2.14)). Therefore K must be chosen small enough, so that the assumption is not violated. On the other hand, choosing K too small renders the overlap-add method more complex than the direct implementation (2.11).

Since our main interest is in highly time-variant channels, we conclude that the overlap-add method is not feasible and thus we will stick to the direct method (2.11).

2.4.2 MIMO channels

Let $H(t, \tau, \boldsymbol{x}, \boldsymbol{y})$ denote the time- and space-variant bandlimited channel impulse response incorporating the complex antenna patterns. Accordingly, let $h(t, f, \boldsymbol{x}, \boldsymbol{y})$ denote the time- and space-variant bandlimited channel transfer function incorporating the complex antenna patterns.

Denote by $1/D_S$ the spatial sampling rate. The sampled time- and space-variant impulse response and the sampled time- and space-variant transfer function are given by

$$H_{m,n,\boldsymbol{u},\boldsymbol{v}} = H(mT_S, nT_S, \boldsymbol{u}D_S, \boldsymbol{v}D_S) \tag{2.18}$$

$$h_{m,q,\boldsymbol{u},\boldsymbol{v}} = h(mT_S, qF_S, \boldsymbol{u}D_S, \boldsymbol{v}D_S), \tag{2.19}$$

where $\boldsymbol{u} \in \mathbb{Z}^3$ and $\boldsymbol{v} \in \mathbb{Z}^3$ denote the discrete space indices at the transmitter and receiver.

When the discrete positions of the transmit and receive antennas are given and denoted by $\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N_{\mathrm{Tx}}-1}$ and $\boldsymbol{v}_0, \ldots, \boldsymbol{v}_{N_{\mathrm{Rx}}-1}$, it is convenient to collect the transmit and receive signals in vectors and the channel transfer function or impulse response in a matrix.

$$\boldsymbol{s}_m = [s(mT_S, \boldsymbol{u}_0 D_S), \ldots, s(mT_S, \boldsymbol{u}_{N_{\mathrm{Tx}}-1} D_S)]^T, \tag{2.20}$$

$$\boldsymbol{r}_m = [r(mT_S, \boldsymbol{v}_0 D_S), \ldots, r(mT_S, \boldsymbol{v}_{N_{\mathrm{Rx}}-1} D_S)]^T, \tag{2.21}$$

$$\mathbf{H}_{m,n} = \begin{bmatrix} H_{m,n,\boldsymbol{u}_0,\boldsymbol{v}_0} & \cdots & H_{m,n,\boldsymbol{u}_0,\boldsymbol{v}_{N_{\mathrm{Tx}}-1}} \\ \vdots & \ddots & \vdots \\ H_{m,n,\boldsymbol{u}_{N_{\mathrm{Rx}}-1},\boldsymbol{v}_0} & \cdots & H_{m,n,\boldsymbol{u}_{N_{\mathrm{Rx}}-1},\boldsymbol{v}_{N_{\mathrm{Tx}}-1}} \end{bmatrix}, \tag{2.22}$$

$$\mathbf{h}_{m,q} = \begin{bmatrix} h_{m,q,\boldsymbol{u}_0,\boldsymbol{v}_0} & \cdots & h_{m,q,\boldsymbol{u}_0,\boldsymbol{v}_{N_{\mathrm{Tx}}-1}} \\ \vdots & \ddots & \vdots \\ h_{m,q,\boldsymbol{u}_{N_{\mathrm{Rx}}-1},\boldsymbol{v}_0} & \cdots & h_{m,q,\boldsymbol{u}_{nRX-1},\boldsymbol{v}_{N_{\mathrm{Tx}}-1}} \end{bmatrix}. \tag{2.23}$$

The input-output relation of a sampled MIMO channel may be expressed as

$$\boldsymbol{r}_m = \sum_{n=0}^{N-1} \mathbf{H}_{m,n} \boldsymbol{s}_{m-n}. \tag{2.24}$$

Again, we can also write the input-output relation by

$$\boldsymbol{r}_m = \sum_{q=0}^{Q-1} \mathbf{h}_{m,q} \boldsymbol{S}_q e^{-2\pi jmq/Q}, \tag{2.25}$$

where $\boldsymbol{S}_q = \sum_m \boldsymbol{s}_m e^{2\pi jmq}$ is the DTFT of the input signal $\boldsymbol{s}_m$.

2.4.3 Autocorrelation Functions and Power Spectra

Autocorrelation functions and power spectra can also be defined for the discrete time functions introduced in the previous subsections.

Let $h_m = h(mT_S)$ be the sampled time-variant transfer function of a frequency flat SISO channel. The autocorrelation function of h_m is defined as

$$R_h(m, m') = \mathcal{E}\{h_m h^*_{m'}\}.$$

The process h_m is WSS, iff $R_h(m, m') = R_h(m - m') = R_h(\Delta m)$. The correlation matrix $\mathbf{R}$ of the discrete temporal process h_m evaluates the autocorrelation function of M successive samples,

$$\mathrm{R}_{m,m'} = R_h(m, m') \quad m, m' = 0, \ldots, M-1.$$

The power spectrum of a WSS process h_m is defined as

$$S_h(\nu) = \mathcal{E}\{|H(\nu)|^2\},$$

where $H(\nu)$ is the DTFT of h_m. By the Wiener-Khintchine Theorem [93], the power spectrum $S_h(\nu)$ is related to the ACF $R_h(\Delta m)$ by a DTFT:

$$S_h(\nu) = \sum_{m=-\infty}^{\infty} R_h(\Delta m) e^{-2\pi j \Delta m \nu}.$$

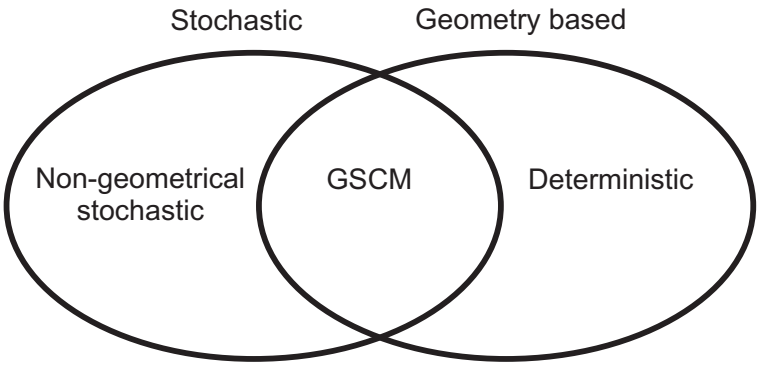

Figure 2.7: Mobile radio channel model characterization.

Autocorrelation functions and power spectra can also be defined for functions of two or more variables. For example, let $h_{r,s}$ be the time-invariant frequency-flat transfer function of a MIMO channel. The autocorrelation function of $h_{r,s}$ is given by

$$R_h(r, r', s, s') = \mathcal{E}\{h_{r,s} h^*_{r',s'}\}.$$

The correlation matrix $\mathbf{R}$ of the MIMO channel with N_{Tx} transmit antennas and N_{Rx} receive antennas is defined as

$$\mathrm{R}_{r+sN_{\mathrm{Tx}},r'+s'N_{\mathrm{Tx}}} = R_h(r, r', s, s') \quad s, s' = 0, \ldots, N_{\mathrm{Tx}} - 1 \quad r, r' = 0, \ldots, N_{\mathrm{Rx}} - 1.$$

2.5 Mobile Radio Channel Models

There exist a vast amount of different channel models for SISO as well as MIMO channels. A specific model usually serves one specific purpose. In this section, some basic principles for channel modeling are reviewed.

Mobile radio channel models can be classified in different ways. In this thesis we classify mobile radio channel models in *stochastic* and *geometry-based* models (see Fig. 2.7). Stochastic models describe the mobile radio channel by means of ACFs and power spectra introduced in Section 2.3. Geometry-based models on the other hand describe the time- and space variant transfer function of the mobile radio channel directly. The classification is based on the classification used in [23] for physical channel models. However, we use the classification in a wider sense and do not only focus on models for MIMO channels.

The classification between stochastic and geometry-based models is not always unique. In fact there exists a channel model that can be seen as both, namely the *geometry-based stochastic channel model* (GSCM). The ambiguity results from the fact, that every GSCM induces a certain stochastic model. Vice versa, for many

stochastic models, a parameter set for a GSCM can be found. The complement of the GSCM with respect to geometry-based models is called *deterministic* models. The complement of the GSCM with respect to stochastic models is called *non-geometrical stochastic models*.

2.5.1 Non-geometrical Stochastic Models

Non-geometrical stochastic models describe the mobile radio channel in a completely stochastic way by prescribing the ACF or the power spectrum of the channel without assuming an underlying geometry. In the following, we give examples of models for the Doppler power spectrum, the power delay spectrum, and the angular power spectra. We implicitly assume that the channel is WSS.

2.5.1.1 Time-selective Fading

A very popular model for the Doppler power spectrum of time-variant channels is the Clarke spectrum [33]:

$$S_h(\omega) = \begin{cases} \frac{1}{\pi\omega_{\mathrm{Dmax}}} \sqrt{1 - \left(\frac{\omega}{\omega_{\mathrm{Dmax}}}\right)^2}, & |\omega| \leq \omega_{\mathrm{Dmax}} \\ 0, & \text{otherwise} \end{cases} \tag{2.26}$$

where ω_{Dmax} is the maximum Doppler frequency. The spectrum $S_h(\omega)$ can be obtained by assuming horizontal propagation and a uniform distribution of the angles of arrival. Those assumptions are not always fulfilled in real channels and measured Doppler spectra often show a very different behavior [21, 122].

2.5.1.2 Frequency-selective Fading

Some very popular models for power delay profiles are the models developed by the COST 207 action [44], which were later standardized by the ITU for the GSM standard. The models define power delay profiles for the following scenarios:

- Typical urban (TU):

$$S_h(\tau) = \begin{cases} e^{-\tau} & \text{for } 0 < \tau < 7\mu\text{s} \\ 0 & \text{otherwise} \end{cases}$$

- Bad urban (BU):

$$S_h(\tau) = \begin{cases} e^{-\tau} & \text{for } 0 < \tau < 5\mu\text{s} \\ 0.5e^{5-\tau} & \text{for } 5 < \tau < 10\mu\text{s} \\ 0 & \text{otherwise} \end{cases}$$

- Rural area (RA):

$$S_h(\tau) = \begin{cases} e^{-9.2\tau} & \text{for } 0 < \tau < 0.7\mu\text{s} \\ 0 & \text{otherwise} \end{cases}$$

- Hilly terrain (HT):

$$S_h(\tau) = \begin{cases} e^{-3.5\tau} & \text{for } 0 < \tau < 2\mu\text{s} \\ 0.1e^{15-\tau} & \text{for } 15 < \tau < 20\mu\text{s} \\ 0 & \text{otherwise} \end{cases}$$

2.5.1.3 Space-selective Fading

Experimental investigations have revealed that in typical macrocellular environments the azimuth power spectrum at the base station can be accurately described by a truncated Laplace function

$$S_h(\psi) = ce^{-|\psi-\psi_0|/\sigma},$$

where ψ_0 is the mean AoA and σ is the angular spread [46]. Scatterers or groups of scatterers with a very large geometrical extent give rise to an azimuth power spectrum given by a truncated Gaussian function,

$$S_h(\psi) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\psi-\psi_0)^2}{2\sigma^2}}.$$

The Laplacian as well as the Gaussian distribution have the drawback that they are defined on the whole real line $\mathbb{R}$. A more natural angular distribution is the von-Mises distribution, which is defined on the unit sphere $\mathbb{S}_1$:

$$S_h(\psi) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa\cos(\psi-\psi_0)},$$

where $I_n(\cdot)$ is the modified Bessel function of the first kind and order n, $\kappa \geq 0$ is called the concentration parameter and ψ_0 is the mean AoA. The von-Mises distribution converges to a Gaussian distribution with $\sigma = 1/\kappa$ for $\kappa \to \infty$ and to a uniform distribution for $\kappa \to 0$ [46].

A model for the bi-azimuthal power spectrum $S_h(\varphi, \psi)$ can be obtained by modeling the AoA/AoD spectrum separately and assuming that $S_h(\varphi, \psi) = S_h(\varphi)S_h(\psi)$. This model is also called the Kronecker model, since the correlation matrix of the corresponding MIMO channel can be written as the Kronecker product [61],

$$\mathbf{R} = \mathbf{R}_{\text{Tx}}^T \otimes \mathbf{R}_{\text{Rx}}, \tag{2.27}$$

where $\mathbf{R}_{\text{Tx}}$ is the transmit correlation matrix and $\mathbf{R}_{\text{Rx}}$ is the receive correlation matrix. However, it has been shown that the Kronecker model underestimates the capacity of large MIMO systems [82]. A more realistic models for the bi-azimuth power spectrum is the two-dimensional von-Mises distribution [115, 116].

2.5.2 Geometry-based Models

Geometry-based models use the theory of electromagnetic wave propagation to characterize wireless channels. Like all electromagnetic phenomena, wireless transmissions obey Maxwell's laws of electrodynamics [69,114]. The wireless channel can be assumed to be an uncharged vacuum. In an uncharged vacuum, Maxwell's four equations reduce to a single equation for the electric field $\boldsymbol{E}(\boldsymbol{y}, t)$, called the vector-wave equation [95, Sec. 105]

$$\nabla^2 \boldsymbol{E}(\mathbf{y}, t) - \frac{1}{c^2} \frac{\partial^2 \boldsymbol{E}(\mathbf{y}, t)}{\partial t^2} = 0, \tag{2.28}$$

where ∇^2 is the Laplacian operator and c is the speed of light.

If the received signal comprises a narrowband signal modulated on a carrier wave with angular frequency ω_C, the solutions of (2.28) can be restricted to a time-harmonic, monochromatic waves $\boldsymbol{E}(\mathbf{y}, t) = \bar{\boldsymbol{E}}(\mathbf{y}) \exp(-j\omega_C t)$. The resulting equation is the Helmholtz equation,

$$\nabla^2 \bar{\boldsymbol{E}}(\mathbf{y}) + \frac{\omega_C^2}{c^2} \bar{\boldsymbol{E}}(\mathbf{y}) = 0. \tag{2.29}$$

At the receiver, the antenna measures the vector-valued electric field $\bar{\boldsymbol{E}}(\boldsymbol{y})$ and converts it to a scalar voltage $h(\boldsymbol{y})$, which obeys the scalar Helmholtz equation (2.29) [41]. The entity $h(\boldsymbol{y})$ can be seen as the transfer function of a narrow-band, time-invariant SIMO channel. In the following, a model for $h(\boldsymbol{y})$ will be derived based on the solutions to the Helmholtz equation.

All possible solutions to the Helmholtz equation can be written as a superposition of plane waves,

$$h(\boldsymbol{y}) = \sum_p \beta_p e^{2\pi j(\phi_p - \langle \boldsymbol{\xi}_p, \boldsymbol{y} \rangle)}, \quad \text{with } \langle \boldsymbol{\xi}_p, \boldsymbol{\xi}_p \rangle = k^2 \tag{2.30}$$

where $k = 2\pi/\lambda = 2\pi\omega_C/c$ is the free space wavenumber, β_p are the real amplitudes, ϕ_p are the real phases, and $\boldsymbol{\xi}_p$ are the wavevectors of the plane wave p [41]. The wavevectors $\boldsymbol{\xi}_p$ can take real as well as complex values. A real wavevector corresponds to a homogeneous wave and a complex wavevector to a inhomogenous wave. In [41] it is shown, that the contributions of inhomogenous waves can either be neglected or approximated by homogenous waves. The wavevector of a homogenous wave always points on the direction of propagation.

If the plane wave has been modulated with a band-limited signal, then the narrowband characterization of Equation (2.30) becomes insufficient to characterize the channel. In [41] it is shown how the solutions (2.30) can be expanded to a wideband characterization of the channel for small frequency displacements f around

the carrier frequency ω_C. The author further simplifies the expansion and shows that within a local area around the receiver, the channel transfer function can be approximated by

$$h(f, \boldsymbol{y}) = \sum_p \beta_p e^{2\pi j(\phi_p - \langle \boldsymbol{\xi}_p, \boldsymbol{y} \rangle - f\tau_p)}, \tag{2.31}$$

where τ_p is the time the p-th plane wave needs to travel from the transmitter to the receiver.

The channel transfer function (2.31) can be extended to also include small displacements $\boldsymbol{x}$ of the transmitter,

$$h(f, \boldsymbol{x}, \boldsymbol{y}) = \sum_p \beta_p e^{2\pi j(\phi_p + \langle \boldsymbol{\zeta}_p, \boldsymbol{x} \rangle - \langle \boldsymbol{\xi}_p, \boldsymbol{y} \rangle - f\tau_p)}, \tag{2.32}$$

where $\boldsymbol{\zeta}_p$ is the direction of departure of the p-th plane wave.

Last, but not least, if the transmitter is moving at speed $\boldsymbol{v}$, the channel transfer function becomes time-variant.

$$h(t, f, \boldsymbol{x}, \boldsymbol{y}) = \sum_p \beta_p e^{2\pi j(\phi_p + \langle \boldsymbol{\zeta}_p, \boldsymbol{x} \rangle - \langle \boldsymbol{\xi}_p, \boldsymbol{y} \rangle - f\tau_p + t\omega_p)}, \tag{2.33}$$

where $\omega_p = \langle \boldsymbol{\xi}_p, \boldsymbol{v} \rangle$ is the Doppler shift of the p-th plane wave.

To summarize, using the plane wave solutions to the Helmholtz equation (2.29), we have derived a description of the time- and space-variant transfer function of the channel. The plane waves in (2.33) are also called multipath components (MPC) of the channel. Each MPC is characterized by its real weight β_p, its phase ϕ_p, its direction of departure $\boldsymbol{\zeta}_p$, its direction of arrival $\boldsymbol{\xi}_p$, its delay τ_p, and its Doppler shift ω_p.

2.5.3 Geometry-based Stochastic Models

A GSCM assumes that the underlying geometry of the mobile radio channel can be described by a finite number of singular point scatterers. Those point scatterers also model effects like reflection, refraction, and diffraction. In a GSCM, the locations and the weights of the scatterers are described in a stochastic (random) fashion according to a certain probability distribution.

Looking at the far field only, and assuming plane wave propagation, the parameters of the MPCs in the time- and space-variant channel transfer function is then found using a simplified ray tracing procedure. Every ray corresponds to a MPC p of the channel and is characterized by its real weight β_p, its phase ϕ_p, its Doppler shift ω_p, its delay τ_p, its AoD φ_p, and its AoA ψ_p (see Fig. 2.8). More sophisticated models can also include parameters like elevation angles and polarization.

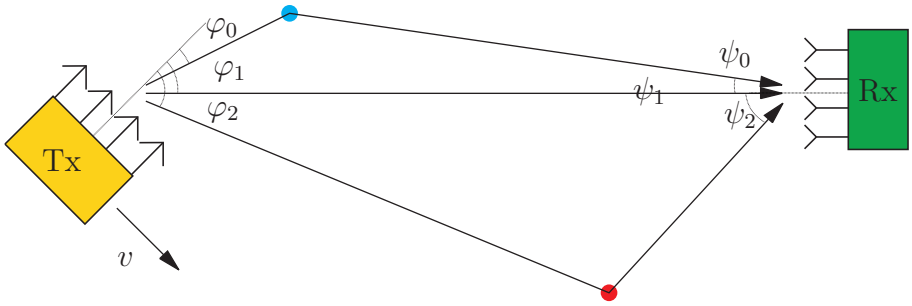

Figure 2.8: Time-variant frequency-selective double-directional multi-path propagation model for a wireless radio channel. The signals sent from the transmitter, moving at speed v, arrive at the receiver. Each MPC p has real weight β_p, phase shift ϕ_p, time delay τ_p, Doppler shift ω_p, AoD φ_p, and AoA ψ_p.

GSCMs either assume single-bounce scattering or multiple-bounce scattering. Given the distribution of the scatterers and the number of bounces, ACF and power spectra of the channel can be calculated. In that sense GSCMs are a subset of stochastic models. Compared to the non-geometrical stochastic models described in Section 2.5.1, GSCMs are more powerful since then can also model non-WSS channels and long-term evolutions of the channel.

Examples. A special GSCM for flat fading SISO channels is obtained by placing the scatterers on a circle around the receiver, moving at speed v, according to a uniform distribution (see Fig. 2.9). It can be shown that that this model is equivalent to Clarke's model presented in Section 2.5.1.1. In particular, the ACF of the fading process is the zeroth-order Bessel function of the first kind. This implies that the Doppler spectrum has the classical U-shape given by equation (2.26).

More sophisticated GSCMs are the models developed by the COST (European Co-operation in the field of Scientific and Technical research) Actions 259 "Wireless Personalized Flexible Communications" [34] and 273 "Towards Mobile Broadband Multimedia Networks" [35]. These models are also implemented on the *ARC SmartSim* real-time channel simulator [7,54] and will be explained in some more detail in Chapter 5.

Other GSCMs are the 3GPP spacial channel model used for UMTS [71] and the I-METRA models used for the IEEE 802.11n (WLAN) standard [42]. They can be seen as subsets of the COST 259/273 channel models with different parameter sets. See [23] for an overview of those models.

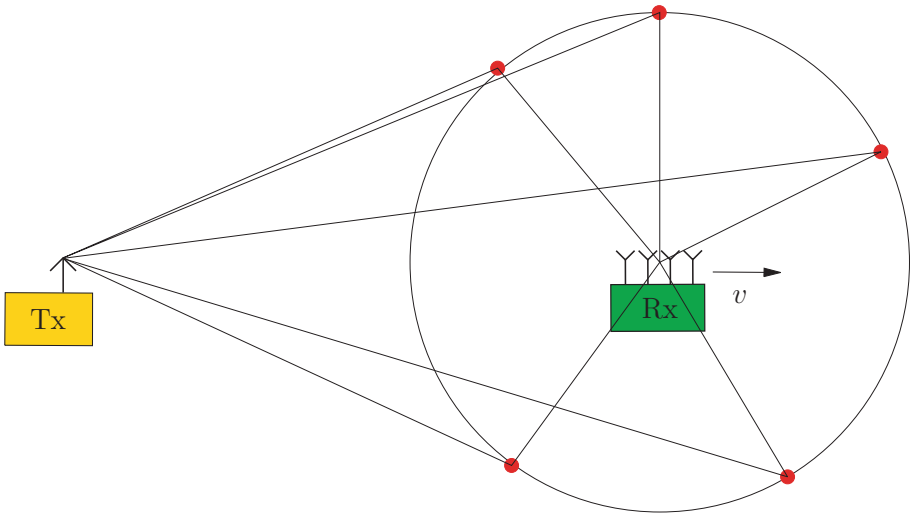

Figure 2.9: One-ring scatterer model for a wireless radio channel. The scatterers are distributed uniformly on a circle around the receiver.

2.5.4 Deterministic Channel Models

Deterministic channel models are also geometry-based models. In contrast to the GSCM, deterministic models aim at reproducing the geometry of the scenario as accurately as possible. Therefore large databases describing the scenario are required. To calculate the actual wave propagation, computer programs such as the method of moments (MoM) of the finite-difference time domain (FDTD) can be used.

For urban areas the databases usually consist of buildings which are represented as polygonal prisms. In this case the most appropriate simulation method is raytracing (RT). RT uses the theory of geometrical optics to treat reflections and transmission on plane surfaces and diffraction on rectilinear edges. Using RT, the space-and-time-variant transfer function can be written in the same way as (2.33). The only difference is that the parameters of the MPCs are now calculated by the RT tool.

2.6 Simulation of Mobile Radio Channels

When simulating wireless communication systems on a digital computer, implementations of channel models are required. In the computer simulation of mobile radio channels, issues like numerical stability, numerical accuracy and complexity need to be considered. Different hardware architectures—ranging from general purpose, floating point arithmetic workstations to specialized, fixed-point arithmetic embedded processors—may require different algorithms or at least different parameters.

We focus on the simulation of the discrete temporal fading process $h_m = h(mT_S)$.

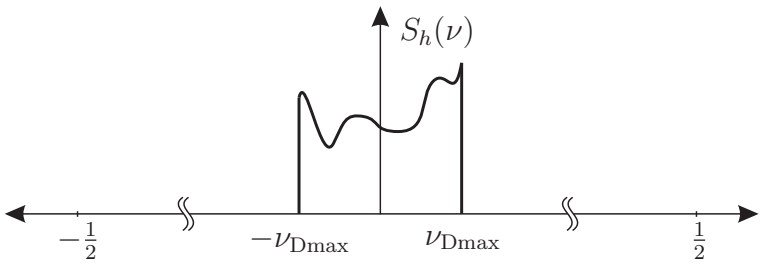

Figure 2.10: Doppler spectrum $S_h(\nu)$ of the time-variant channel $h(t)$. The maximum Doppler bandwidth $2\nu_{\mathrm{Dmax}}$ is much smaller than the available normalized channel bandwidth.

However, the methods presented in this section are fairly general and can be extended to the multidimensional case, allowing the simulation of time, frequency and spatial fading processes.

Let ν_{Dmax} denote the normalized (one-sided) Doppler bandwidth of the sampled channel transfer function h_m, i. e., the DTFT $H(\nu)$ vanishes for $|\nu| > \nu_{\mathrm{Dmax}}$. The normalized (one-sided) Doppler bandwidth is related to the Doppler bandwidth by $\nu_{\mathrm{Dmax}} = \omega_{\mathrm{Dmax}} T_S$. In typical wireless communication systems, the temporal fading process h_m is highly oversampled (see Fig. 2.10). In other words, the maximum normalized Doppler bandwidth $2\nu_{\mathrm{Dmax}}$ is much smaller than the available normalized channel bandwidth,

$$\nu_{\mathrm{Dmax}} \ll \frac{1}{2}. \tag{2.34}$$

In the following two subsections simulation methods for the channel models described in the last section are given. Correlation based simulation is described in Section 2.6.1. Geometry-based simulation is described in Section 2.6.2.

2.6.1 Correlation-based Simulation

Correlation-based simulation methods calculate realizations of the channel transfer function using its ACF. Generation of M variates $\boldsymbol{h} = [h_0, \ldots, h_{M-1}]^T$ with any correlation matrix $\mathbf{R}$ can be achieved by decomposing

$$\mathbf{R} = \mathbf{Q}\mathbf{Q}^H \tag{2.35}$$

and writing $\boldsymbol{h} = \mathbf{Q}\boldsymbol{g}$, where $\boldsymbol{g}$ is a vector of M i. i. d. Gaussian variates. The decomposition (2.35) can be calculated in $\mathcal{O}(M^2)$ operations using a Cholesky factorization. For band-limited processes however, an approximation using a singular value decomposition is required due to the ill-conditioned correlation matrix. In this case, $\mathcal{O}(M^3)$ operations are required which limits the method to small sample sizes.

The correlation matrix method can also be used for non-WSS processes. Therefore, it is very attractive for an application in the spatial domains.

If the process h_m is WSS, another possibility to generate M correlated Gaussian variates is to take a vector of M i. i. d. complex Gaussian variates, multiplying it with the sampled spectrum $S_h(n/M),\ n = 0, \ldots, M-1$ in the frequency domain and taking the inverse discrete Fourier transform (IDFT) resp. its fast implementation, the inverse fast Fourier transform (IFFT). The IDFT method has been used for the generation of correlated sequences in the time domain [117]. However, due to the small bandwidth of the process h_m, only a few samples of the spectrum $S_h(n/M),\ n = 0, \ldots, M-1$ are nonzero, which leads to numerical difficulties. Further, since all samples are generated with a IFFT operation, the storage requirements of this approach are very high and make it unattractive for the generation of a very large number of variates.

Last but not least, h_m can be generated by passing a complex white Gaussian noise sequence through a filter with the desired spectrum $S_h(\nu)$. This method has the advantage, that new elements of the sequence can be generated when needed. It is therefore very attractive for an application in the time domain. However, the drawback of this method is that due to the small bandwidth of the process, the filter is hard to implement (high complexity, numerical problems). Recently, several approaches have been presented to tackle those problems.

Baddour [26] uses an autoregressive stochastic model for the simulation of correlated Rayleigh channels. Essentially, this technique employs all-pole infinite-impulse response (IIR) filtering to shape the spectrum of uncorrelated Gaussian variates. The authors' main contribution is a procedure that overcomes the numerical difficulties in solving the Yule-Walker equations for the coefficients of the model. Therefore he uses diagonal loading of the rank deficient covariance matrix.

Selva [100] uses a Farrow interpolation filter structure that approximates a bounded bandlimited signal in a finite time interval with a given accuracy. The method uses a polynomial approximation of the signal. The coefficients of the polynomial can be computed with low complexity.

The methods presented in [119] can also be used to generate a fading process according to Clarke's model. Similar to [100], the process first has to be generated at a low sampling rate. Then, a polynomial spline interpolator implemented using a pre-filter—post-filter structure can be used to interpolate the signal at the desired rate.

2.6.2 Geometry-based Simulation

Geometry-based simulation methods assume that the channel transfer function can be written as a finite sum of MPCs (see Equation (2.33)). Geometry-based sim-

ulation is therefore the natural choice for geometry-based models, which calculate the parameters of the MPCs directly. However, they can also be used to simulate non-geometrical stochastic models. Therefore, the MPC parameters are taken as samples of the underlying distribution. The number of MPCs has to be chosen, such that the statistics of the geometry-based model match the original ones.

We focus again on the simulation of time-variant frequency-flat SISO channels. The geometry-based model for the channels transfer function $h(t)$ can be obtained by setting $f = 0$, $\boldsymbol{\xi} = \mathbf{0}$, and $\boldsymbol{\zeta} = 0$ in (2.33):

$$h(t) = \sum_{p=0}^{P-1} \beta_p e^{2\pi j(\omega_p t + \phi_p)}. \tag{2.36}$$

The sampled channel transfer function is then given by

$$h_m = \sum_{p=0}^{P-1} \beta_p e^{2\pi j(\nu_p m + \phi_p)}, \tag{2.37}$$

where $\nu_p = \omega_p T_S$ is the normalized Doppler shift of the p-th MPC. We refer to Equation (2.37) as the sum of complex exponentials (SoCE) algorithm for computing the channel transfer function h_m.

Clarke's model [33] is a special case of (2.37) and can be obtained by relating the normalized Doppler shift ν_p to the AoA ψ_p by $\nu_p = \nu_{\mathrm{Dmax}} \cos(\psi_p)$ and setting $\beta_p = 1/\sqrt{P}$. Assuming that ψ_p, and ϕ_p are mutually independent and that ψ_p and ϕ_p are uniformly distributed on the interval $[-\pi, \pi)$, it can be shown that the spectrum of the process h_m approaches $S_h(\nu)$ given by (2.26) as $P \to \infty$.

Jakes [57] proposed a simplified version of this model by setting the initial phases $\phi_p = 0$ and spacing the AoAs equidistantly, i. e.,

$$\psi_p = \frac{2\pi p + \vartheta}{P}, \quad p = 0, \ldots, P-1, \tag{2.38}$$

where ϑ is uniformly distributed on the interval $[-\pi, \pi)$. If P is a multiple of four, symmetries can be utilized and only $P/4$ sinusoids have to be evaluated [57]. This model is therefore often referenced as the sum-of-sinusoids (SoS) model. Jakes' model reduces the computational complexity of Clarke's model by a factor of four. However, the second order statistics of Jakes' simplification do not match the ones of Clarke's model [85] and Jakes' model is not wide sense stationary [91]. Attempts to improve the second order statistics while keeping the reduced complexity of Jakes' model are reported in [39, 68, 85, 91, 118, 121, 123].

For example, an enhanced version of Jakes' simplification was proposed by Zheng and Xiao [123]. By leaving the amplitudes and the initial phases as random variables,

Zheng and Xiao's models obtain statistical properties similar to those of the reference model. Yet another model has been proposed by Zajič and Stüber [118]. Compared to Zheng and Xiao's models, it uses slightly different initial phases, which leads to less correlated I and Q components.

However, due to the equidistant spacing of the AoAs, none of these models achieves all the desirable statistical properties of Clarke's reference model [19]. One of the main contributions of this thesis is an algorithm that allows to reduce the complexity of Clarke's original model by more than an order of magnitude without imposing restrictions on the AoAs.

On the number of MPCs. The number of required MPCs depends on the modeled scenario, the system bandwidth, and the number of antennas used. In this thesis we choose the number of MPCs such that the channel is Rayleigh fading (except for the line-of-sight (LOS) component).

For narrowband frequency-flat systems, approximately $P_0 = 40$ MPCs are necessary to achieve a Rayleigh fading statistic [123]. If the channel bandwidth is increased, also the number of resolvable MPCs increases. The ITU channel models [73], which are used for a bandwidth of up to 5MHz in UMTS systems, specify a power delay profile with up to six delay bins. The I-METRA channel models for the IEEE 802.11n wireless LAN standard [42] are valid for up to 40MHz and specify a power delay profile with up to 18 delay bins. This requires a total number of MPCs of up to $P_1 = 18P_0 = 720$. Diffuse scattering can also be modeled with a geometry-based channel model by increasing the number of MPCs. In theory, diffuse scattering results from the superposition of an infinite number of MPCs [37]. However, good approximations can be achieved by using a large, but finite number of MPCs [81,87]. In MIMO channels, the number of MPCs multiplies by $N_{\mathrm{Tx}}N_{\mathrm{Rx}}$, since every antenna sees every scatterer from a different AoA and AoD respectively. For a 4×4 system, the total number of MPCs can thus reach up to $P = 16P_1 = 1.2 \cdot 10^4$.

2.7 Real-Time MIMO Hardware Channel Simulation

Hardware channel simulators allow to simulate mobile radio channels in real time. They usually consist of a powerful baseband signal processing unit and radio frequency (RF) frontends for input and output. The RF frontends down-convert and digitize the RF signals into a digital baseband representation. In the baseband processing unit, the channel transfer function is calculated according to the underlying channel model and the signal is convolved with the channel. Alternatively, the channel transfer function can be calculated in advance and only played back by the channel simulator. Finally, the convolved signal is converted to an analog signal again and up-converted to RF. The benefits of hardware channel simulation

are real time and repeatable performance evaluation for any device under test. The challenges in MIMO channel simulation are the high computational complexity and the multichannel RF frontends.

Channel simulators can be characterized by two fundamental characteristics: the number of channels and the number of taps. Further characteristics include the RF frequency and bandwidth. The following subsections give a short overview of MIMO channel simulators available on the market today (see also [35]).

2.7.1 ARC SmartSim

The SmartSim[1] [7] from the Austrian Research Centers GmbH – ARC can simulate up to eight channels with up to 20 taps each. It implements the COST 259 geometry-based stochastic channel model. Alternatively, user-defined channel models can be generated in advance and played back. It has an RF frontend with a center frequency of 2.58 GHz and a 20 MHz bandwidth. More details about the ARC SmartSim can be found in Section 5.2.

2.7.2 Azimuth ACE 400NB

The ACE 400NB[2] from Azimuth is designed for bi-directional simulation of 4×4 MIMO systems (up to 32 channels) with up to 18 taps. The simulator features built-in IEEE 802.11n channel models. The input RF frequency is scalable from 2400–2500 MHz and 4900–5845 MHz and the bandwidth is 40MHz [25].

2.7.3 Elektrobit Propsim C8

The Propsim C8[3] [62] from Elektrobit Ltd. features up to 8 channels with up to 24 taps. Channel models must be precalculated and uploaded to the simulator. Elektrobit also provides tools to analyze and preprocess channel sounder measurements for the Propsim C8. The RF frequency range is 350 MHz – 6 GHz and the maximum RF bandwidth is 70 MHz.

2.7.4 Spirent SR5500

The SR5500[4] from Spirent features up to 8 channels with up to 24 taps and can be configured to simulate MIMO systems with up to 8 channels. Channel models

[1] http://www.smartsim.at
[2] http://www.azimuthsystems.com
[3] http://www.propsim.com
[4] http://www.spirentcom.com

must be precalculated and uploaded to the simulator. The SR5500 supports an RF frequency range of 400–2700 MHz and a bandwidth of 28 MHz [105].

2.8 Conclusions and Outlook

The simulation of wireless mobile communication channels is very important for the design and the testing of receiver algorithms. Channel models should be able to represent time-selective, frequency-selective, and space-selective fading appropriately. On the other hand, the model should allow for a low-complexity implementation on a DSP, so that it can be simulated in real time. This is especially important for the test of mobile radio hardware devices using hardware channel simulators.

Stochastic channel models describe the mobile radio channel by means of ACF and power spectra, which are either measured or assumed to be known. Geometry-based channel models on the other hand characterize the channel on the basis of physical wave propagation as a superposition of MPCs. The parameters of the MPCs are obtained either deterministically or from the distribution of the scatterers. The latter case leads to the GSCM, which can be seen as a subset of stochastic as well as geometry-based models. GSCM however have the advantage that they can also model non-WSS channels as well as long-term evolutions of the channel.

Simulation methods for the described models include correlation-based simulation and geometry-based simulation. It has been shown that especially for temporal processes correlation based methods lead to numerical problems due to the high oversampling of the process. Geometry-based simulation methods on the other hand, are numerically stable but come at a higher computational complexity. Many attempts to reduce the computational complexity in such models have been made. However, none of the resulting simulators is able to reproduce the desired statistics correctly.

We conclude that for the simulation of time-variant, frequency-selective MIMO channels, geometry-based channel models are perfectly suited. However, the simulation of geometry-based channel models comes with a high computational complexity, since for every propagation path, every time instance, and every frequency bin, one or more complex exponentials have to be evaluated. Therefore, on a real-time hardware channel simulator the number of paths P that can be simulated, is limited by the available processing power.

Hardware channel simulators have a limited accuracy due to their analog/digital converters and the digital signal processors. For example, the DSP of the ARC SmartSim channel simulator [7] has a fixed-point arithmetic with 16 bit precision. In such an architecture, the evaluation of the complex exponential is quite costly and is usually done by table look-up and interpolation.

In this thesis the limited accuracy of hardware channel simulators is exploited to derive a novel low-complexity implementation of geometry-based channel models. Using a subspace-based simulation method it is possible to increase the number of paths P that can be simulated on the hardware and reduce the overall computational complexity at the same time. By adjusting the dimension of the subspace, it is possible to trade accuracy for efficiency. The ARC SmartSim channel simulator is used as a reference architecture.

3 Multidimensional DPS Sequences

3.1 Introduction

In Chapter 2, channel models and simulation methods for mobile radio channels were described. It was shown in Section 2.6 that especially the simulation of time-variant flat-fading SISO channels is a non-trivial problem, due to the very small normalized bandwidth of the fading process.

A very effective way to deal with this problem is to use a subspace representation of the fading process based on discrete prolate spheroidal (DPS) sequences [103]. In [121] this subspace representation was used for the purpose of channel estimation of time-variant frequency-flat SISO channels. Other applications of DPS sequences include spectrum estimation [108] as well as approximation and prediction of bandlimited signals [19, 103].

In this chapter we firstly review DPS sequences and secondly extend them to multiple dimensions. These novel multidimensional DPS sequences will then be used in Chapter 4 for a subspace representation of time-variant frequency-selective MIMO channels.

Multidimensional DPS sequences are a generalization of the original DPS sequences—studied by Slepian et al. [103]—to many dimensions. However, unlike in the one-dimensional case, the efficient and numerically stable calculation of those sequences is not trivial and has not been treated satisfactorily in the literature. In this chapter we establish a theory of the multidimensional DPS sequences based on the Kronecker product formalism, which allows to compute the multidimensional DPS sequences for rectangular passband regions efficiently.

Related work includes the paper [102] of Slepian, where generalized prolate spheroidal functions are described. Prolate spheroidal functions are the continuous analog of the DPS sequences. However, Slepian only studies multidimensional prolate spheroidal functions with spherical passband regions and hence uses spherical coordinates.

Multidimensional DPS sequences have also been used in [40] for bandlimited extrapolation of multidimensional signals. The authors' main contribution is a definition of the essential dimension of the signal subspace in this case. The problem of calculating the sequences is not treated. In [111] the prolate spheroidal window

is calculated for the special case, where the passband region is a rhombic dodecahedron. However, they use the defining integral equation for calculation, which becomes numerically unstable for large sequence lengths.

3.2 One-dimensional DPS Sequences

A square-summable sequence $v_m \in L^2(\mathbb{Z})$, defined on the Euclidean space $\mathbb{Z}$, is bandlimited over a bounded region $W \subset \mathbb{R}$ (or W-limited for short) if its DTFT

$$V(f) = \sum_{m\in\mathbb{Z}} v_m e^{-2\pi jfm}, \quad f \in \mathbb{R} \tag{3.1}$$

vanishes outside the region W:

$$V(f) = 0 \quad f \notin W. \tag{3.2}$$

The W-limited sequence v_m can be reconstructed using the inverse DTFT

$$v_m = \int_W V(f) e^{2\pi jfm} \mathrm{d}f, \quad m \in \mathbb{Z}. \tag{3.3}$$

Bandlimited sequences cannot be index-limited at the same time except for $v_m \equiv 0$ [29]. However, we can look for sequences, whose energy concentration on the index set $I \subset \mathbb{Z}$ is maximal, i. e.,

$$\lambda = \frac{\sum_{m\in I} |v_m|^2}{\sum_{m\in\mathbb{Z}} |v_m|^2} \to \max. \tag{3.4}$$

Solutions to this problem are given by the DPS sequences [103].

Definition 3.2.1. *The one-dimensional discrete prolate spheroidal (DPS) sequences $v_m^{(d)}(W, I)$ with band-limit $W = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}]$ and concentration region $I = \{M_0, \ldots, M_0 + M - 1\}$ are defined as the real solutions of*

$$\sum_{n=M_0}^{M_0+M-1} \frac{\sin(2\pi\nu_{\mathrm{Dmax}}(m-n))}{\pi(n-m)} v_n^{(d)}(W, I) = \lambda_d(W, I) v_m^{(d)}(W, I). \tag{3.5}$$

They are sorted such that their eigenvalues $\lambda_d(W, I)$ are in descending order

$$\lambda_0(W, I) > \lambda_1(W, I) > \ldots > \lambda_{M-1}(W, I). \tag{3.6}$$

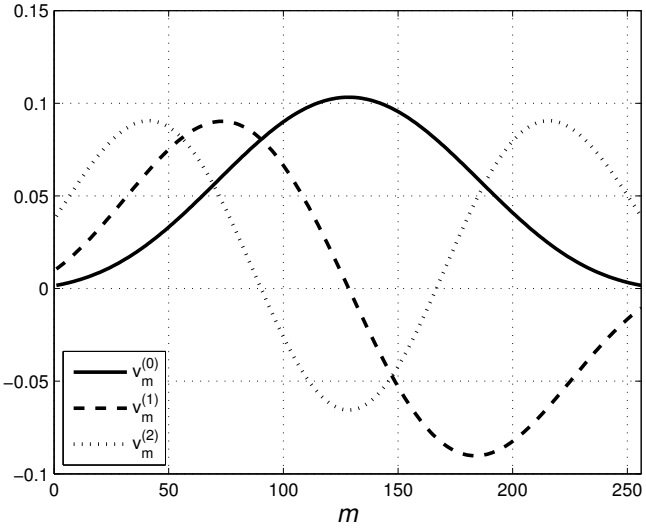

Figure 3.1: The first three one-dimensional DPS sequences $v_m^{(0)}, v_m^{(1)}$, and $v_m^{(2)}$ for $M_0 = 0$, $M = 256$, and $M\nu_{\mathrm{Dmax}} = 2$.

To ease notation, we drop the explicit dependence of $v_m^{(d)}(W, I)$ on W and I when it is clear from the context. Further, we define the DPS vector $\boldsymbol{v}^{(d)}(W, I) \in \mathbb{C}^M$ as the DPS sequence $v_m^{(d)}(W, I)$ index-limited to I.

The DPS vectors $\boldsymbol{v}^{(d)}(W, I)$ are also eigenvectors of the $M \times M$ matrix $\mathbf{K}$ with elements $\mathrm{K}_{m,n} = \sin(2\pi\nu_{\mathrm{Dmax}}(m-n))/\pi(n-m)$. The eigenvalues of this matrix decay exponentially and thus render numerical calculation difficult. Fortunately there exists a tridiagonal matrix commuting with $\mathbf{K}$, which enables fast and numerically stable calculation of DPS sequences [88, 103]. Figs. 3.1 and 3.2 illustrate one-dimensional DPS sequences and their eigenvalues, respectively.

Some properties of DPS sequences are summarized in the following theorem.

Theorem 3.2.1.

1. *The sequences $v_m^{(d)}(W, I)$ are band-limited to W.*

2. *The eigenvalue $\lambda_d(W, I)$ of the DPS sequence $v_m^{(d)}(W, I)$ denotes the energy concentration of the sequence within I*

$$\lambda_d(W, I) = \frac{\sum_{m \in I} |v_m^{(d)}(W, I)|^2}{\sum_{m \in \mathbb{Z}} |v_m^{(d)}(W, I)|^2}. \tag{3.7}$$

3. *The eigenvalues $\lambda_d(W, I)$ satisfy $1 < \lambda_i(W, I) < 0$. They are clustered around 1 for $d \leq D' - 1$, and decay exponentially for $d \geq D'$, where $D' = \lceil |W||I| \rceil + 1$.*

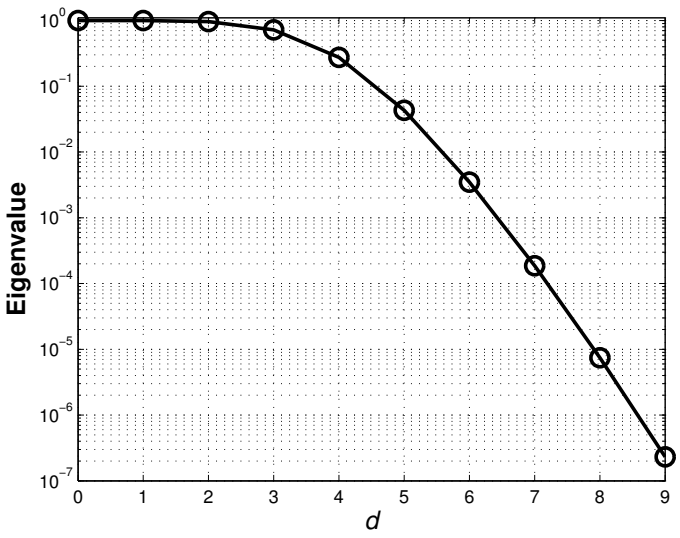

Figure 3.2: The first ten eigenvalues $\lambda_d,\ d = 0, \ldots, 9$ of the one-dimensional DPS sequences for $M_0 = 0$, $M = 256$, and $M\nu_{\mathrm{Dmax}} = 2$. The eigenvalues are clustered around 1 for $d \leq D' - 1$, and decay exponentially for $d \geq D'$, where the essential dimension of the signal subspace $D' = \lceil 2\nu_{\mathrm{Dmax}} M \rceil + 1 = 5$.

4. *The DPS sequences $v_m^{(d)}(W, I)$ are orthogonal on the index set I and on $\mathbb{Z}$.*

5. *Every band-limited sequence h_m can be decomposed uniquely as $h_m = h'_m + g_m$ where h'_m is a linear combination of DPS sequences $v_m^{(d)}(W, I)$ for some I and $g_m = 0$ for all $m \in I$.*

Proof: See Slepian et al. [103]. The proof of Property 2 is also given in the next section in the more general multidimensional setting. □

3.3 Multidimensional DPS Sequences

The extension of DPS sequences to multiple dimensions is straightforward. A square-summable multidimensional sequences $v_{\boldsymbol{m}} \in L^2(\mathbb{Z}^N)$, defined on the Euclidean space $\mathbb{Z}^N$ of N dimensions, is bandlimited over a bounded region $W \subset \mathbb{R}^N$ (or W-limited for short), if its DTFT

$$V(\boldsymbol{f}) = \sum_{\boldsymbol{m} \in \mathbb{Z}^N} v_{\boldsymbol{m}} e^{-2\pi j \langle \boldsymbol{f}, \boldsymbol{m} \rangle}, \quad \boldsymbol{f} \in \mathbb{R}^N \tag{3.8}$$

vanishes outside the region W:

$$V(\boldsymbol{f}) = 0 \quad \boldsymbol{f} \notin W. \tag{3.9}$$

A W-limited sequence $v_{\boldsymbol{m}}$ can be reconstructed using the inverse DTFT

$$v_{\boldsymbol{m}} = \int_W V(\boldsymbol{f}) e^{2\pi j \langle \boldsymbol{f}, \boldsymbol{m} \rangle} \mathrm{d}\boldsymbol{f}, \quad \boldsymbol{m} \in \mathbb{Z}^N. \tag{3.10}$$

As in the one-dimensional case we can look for sequences, whose energy concentration on the index set $I \subset \mathbb{Z}^N$ is maximal, i. e.,

$$\lambda = \frac{\sum_{\boldsymbol{m} \in I} |v_{\boldsymbol{m}}|^2}{\sum_{\boldsymbol{m} \in \mathbb{Z}^N} |v_{\boldsymbol{m}}|^2} \to \max. \tag{3.11}$$

Solutions to this problem are given by the multidimensional DPS sequences.

Definition 3.3.1. *Let $I \subset \mathbb{Z}^N$ be an N-dimensional finite index set with $L = |I|$ elements, and $W \subset (-1/2, 1/2)^N$ an N-dimensional band-limiting region.* Multidimensional discrete prolate spheroidal (DPS) sequences $v_{\boldsymbol{m}}^{(d)}(W, I)$ *are defined as the solutions of the eigenvalue problem*

$$\sum_{\boldsymbol{m'} \in I} v_{\boldsymbol{m'}}^{(d)}(W, I) K^{(W)}(\boldsymbol{m'} - \boldsymbol{m}) = \lambda_d(W, I) v_{\boldsymbol{m}}^{(d)}(W, I), \quad \boldsymbol{m} \in \mathbb{Z}^N, \tag{3.12}$$

where

$$K^{(W)}(\boldsymbol{m'} - \boldsymbol{m}) = \int_W e^{2\pi j \langle \boldsymbol{f''}, \boldsymbol{m'} - \boldsymbol{m} \rangle} \mathrm{d}\boldsymbol{f''}. \tag{3.13}$$

They are sorted such that their eigenvalues $\lambda_d(W, I)$ are in descending order

$$\lambda_0(W, I) > \lambda_1(W, I) > \ldots > \lambda_{L-1}(W, I). \tag{3.14}$$

To ease notation, we drop the explicit dependence of $v_{\boldsymbol{m}}^{(d)}(W, I)$ on W and I when it is clear from the context. Further, we define the multidimensional DPS vector $\boldsymbol{v}^{(d)}(W, I) \in \mathbb{C}^L$ as the multidimensional DPS sequence $v_{\boldsymbol{m}}^{(d)}(W, I)$ index-limited to I. In particular, if every element $\boldsymbol{m} \in I$ is indexed lexicographically, such that $I = \{\boldsymbol{m}_l,\ l = 0, 1, \ldots, L-1\}$, then

$$\boldsymbol{v}^{(d)}(W, I) = [v_{\boldsymbol{m}_0}^{(d)}(W, I), \ldots, v_{\boldsymbol{m}_{L-1}}^{(d)}(W, I)]^T. \tag{3.15}$$

Example. In the two-dimensional case $N = 2$ with bandlimiting region W and index set I given by

$$I = \{0, \ldots, M-1\} \times \{-\lfloor Q/2 \rfloor, \ldots, \lfloor Q/2 \rfloor - 1\} \tag{3.16}$$

$$W = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}] \times [0, \theta_{\max}], \tag{3.17}$$

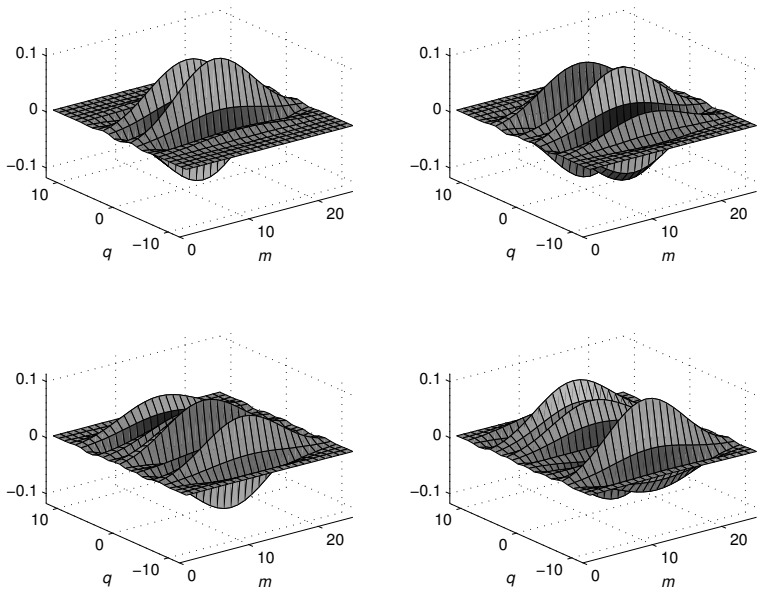

Figure 3.3: The real part of the first four two-dimensional DPS sequences $\boldsymbol{v}^{(d)}$, $d = 0, \ldots, 3$ for $M = Q = 25$, $M\nu_{\mathrm{Dmax}} = 2$, and $Q\theta_{\max} = 5$.

Equation (3.12) reduces to

$$\sum_{n=0}^{M-1} \sum_{p=-\lfloor Q/2 \rfloor}^{\lfloor Q/2 \rfloor - 1} \frac{\sin(2\pi\nu_{\mathrm{Dmax}}(m-n))}{\pi(n-m)} \frac{e^{2\pi i(p-q)\theta_{\max}} - 1}{2\pi i(p-q)} v_{n,p}^{(d)} = \lambda_d v_{m,q}^{(d)}. \tag{3.18}$$

Examples of two-dimensional DPS sequences and their eigenvalues are given in Figs. 3.3 and 3.4.

All the properties of Theorem 3.2.1 also apply to multidimensional DPS sequences [40]. The only difference is that m has to be replaced with $\boldsymbol{m}$ and $\mathbb{Z}$ with $\mathbb{Z}^N$.

Theorem 3.3.1.

1. *The sequences $v_{\boldsymbol{m}}^{(d)}(W, I)$ are bandlimited to W.*
2. *The eigenvalue λ_d of the DPS sequence $v_{\boldsymbol{m}}^{(d)}$ denotes the energy concentration of the sequence within I*
$$\lambda_d = \frac{\sum_{\boldsymbol{m} \in I} |v_{\boldsymbol{m}}^{(d)}|^2}{\sum_{\boldsymbol{m} \in \mathbb{Z}^N} |v_{\boldsymbol{m}}^{(d)}|^2}. \tag{3.19}$$
3. *The eigenvalues λ_d satisfy $1 \leq \lambda_i \leq 0$. They are clustered around 1 for $d < D'$, and decay exponentially for $d > D'$, where $D' = \lceil |W||I| \rceil + 1$.*
4. *The DPS sequences $v_{\boldsymbol{m}}^{(d)}$ are orthogonal on the index set I and on $\mathbb{Z}^N$.*

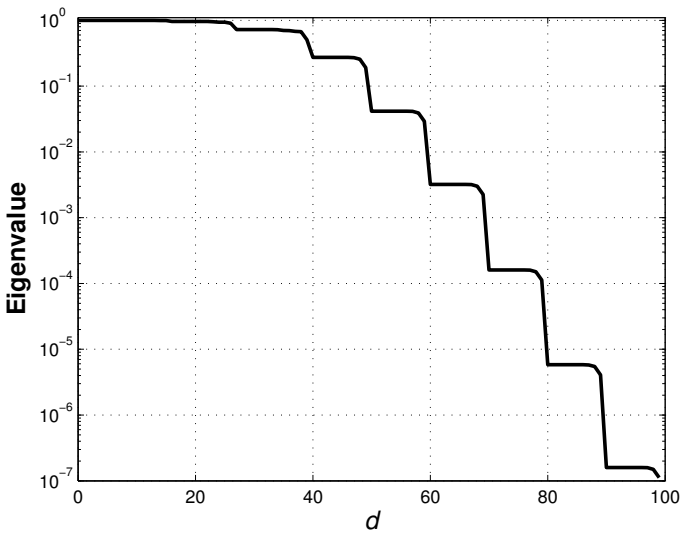

Figure 3.4: First 100 eigenvalues λ_d, $d = 0, \ldots, 99$ of two-dimensional DPS sequences for $M = Q = 25$, $M\nu_{\mathrm{Dmax}} = 2$, and $Q\theta_{\max} = 5$. The eigenvalues are clustered around 1 for $d \leq D' - 1$, and decay exponentially for $d \geq D'$, where the essential dimension of the signal subspace $D' = \lceil |W||I| \rceil + 1 = 41$.

5. *Every W-limited sequence $h_{\boldsymbol{m}}$ can be decomposed uniquely as $h_{\boldsymbol{m}} = h'_{\boldsymbol{m}} + g_{\boldsymbol{m}}$ where $h'_{\boldsymbol{m}}$ is a linear combination of DPS sequences $v^{(d)}_{\boldsymbol{m}}(W, I)$ for some I and $g_{\boldsymbol{m}} = 0$ for all $m \in I$.*

Proof: Most of the properties are proven in [40]. Property 2 is now proven explicitly.

Parseval's theorem can be used to transform the maximization problem (3.11) into

$$\frac{\int_W \int_W \sum_{\boldsymbol{m}\in I} e^{2\pi j\langle \boldsymbol{m}, \boldsymbol{f}-\boldsymbol{f}'\rangle} V(\boldsymbol{f}) V^*(\boldsymbol{f}') \mathrm{d}\boldsymbol{f} \mathrm{d}\boldsymbol{f}'}{\int_W |V(\boldsymbol{f})|^2 \mathrm{d}\boldsymbol{f}} = \lambda. \tag{3.20}$$

It can be seen that less a proportionality constant a maximizing $V(\boldsymbol{f})$ must satisfy

$$\int_W V(\boldsymbol{f}) \sum_{\boldsymbol{m}\in I} e^{2\pi j\langle \boldsymbol{m}, \boldsymbol{f}-\boldsymbol{f}'\rangle} d\boldsymbol{f} = \lambda V(\boldsymbol{f}'). \tag{3.21}$$

Alternatively this can be written as

$$\int_W K(\boldsymbol{f}, \boldsymbol{f}') V(\boldsymbol{f}) d\boldsymbol{f} = \lambda V(\boldsymbol{f}'), \tag{3.22}$$

where the kernel K is defined as

$$K(\boldsymbol{f}, \boldsymbol{f}') = \sum_{\boldsymbol{m}\in I} e^{2\pi j\langle \boldsymbol{m}, \boldsymbol{f}\rangle} e^{-2\pi j\langle \boldsymbol{m}, \boldsymbol{f}'\rangle}. \tag{3.23}$$

The integral equation (3.22) is a homogeneous Fredholm equation of the second kind with an Hermitean degenerate kernel K [92]. Thus the solutions of (3.21) can be found by writing the left hand side of (3.21) as

$$\int_W K(\boldsymbol{f}, \boldsymbol{f}')V(\boldsymbol{f})\mathrm{d}\boldsymbol{f} = \sum_{\boldsymbol{m}\in I} \alpha_{\boldsymbol{m}} e^{-2\pi j\langle \boldsymbol{f}', \boldsymbol{m}\rangle}, \tag{3.24}$$

where $\alpha_{\boldsymbol{m}} = \int_W e^{2\pi j\langle \boldsymbol{f}, \boldsymbol{m}\rangle} V(\boldsymbol{f})\mathrm{d}\boldsymbol{f}$. Substituting this into equation (3.21) results in

$$\sum_{\boldsymbol{m}\in I} \alpha_{\boldsymbol{m}} e^{-2\pi j\langle \boldsymbol{f}', \boldsymbol{m}\rangle} = \lambda V(\boldsymbol{f}'). \tag{3.25}$$

Multiplying with $e^{2\pi j\langle \boldsymbol{f}', \boldsymbol{n}\rangle}$ and integrating over $\boldsymbol{f}'$ results in

$$\sum_{\boldsymbol{m}\in I} \alpha_{\boldsymbol{m}} \int_W e^{2\pi j\langle \boldsymbol{f}', \boldsymbol{n}-\boldsymbol{m}\rangle}\mathrm{d}\boldsymbol{f}' = \lambda\alpha_{\boldsymbol{n}}. \tag{3.26}$$

This equation is the same as (3.12) and the solutions are thus the multidimensional DPS sequences $v_{\boldsymbol{m}}^{(d)}$ with eigenvalues $\lambda_d,\ d = 1, \ldots, L-1$. The eigenvalues and eigenfunctions of (3.21) are thus given by λ_d and

$$U_d(\boldsymbol{f}') = \sum_{\boldsymbol{m}\in I} v_{\boldsymbol{m}}^{(d)} e^{-2\pi j\langle \boldsymbol{f}', \boldsymbol{m}\rangle} \tag{3.27}$$

respectively. Returning to the original maximization problem (3.20), the maximum value of (3.20) is λ_0 and it is attained for $V(\boldsymbol{f}) = U_0(\boldsymbol{f})$. Thus, the corresponding sequence $v_{\boldsymbol{m}}^{(d)}$ has an energy concentration of λ_0.

Moreover, the sequence bandlimited to W orthogonal to $v_{\boldsymbol{m}}^{(0)}$ and most concentrated in I is given by $v_{\boldsymbol{m}}^{(1)}$ with concentration λ_1, etc. □

Note that the functions $V(\boldsymbol{f})$ are very similar to the DPS wave functions defined by Slepian for the one-dimensional case [103]. They only differ slightly in the definition of the kernel K in Equation (3.23) and their normalization.

3.4 Calculation of Multidimensional DPS Sequences

In this section a new way of calculating multidimensional DPS sequences is derived, provided their passband region W can be written as a Cartesian product of one-dimensional intervals.

Indexing every element $\boldsymbol{m} \in I$ lexicographically, such that $I = \{\boldsymbol{m}_l,\ l = 0, 1, \ldots, L-1\}$, we define the matrix $\mathbf{K}^{(W)}$ by

$$\mathrm{K}_{k,l}^{(W)} = K^{(W)}(\boldsymbol{m}_k - \boldsymbol{m}_l), \quad k, l = 0, \ldots, L-1, \tag{3.28}$$

where the kernel $K^{(W)}$ is given by (3.13). Let $\boldsymbol{v}^{(d)}(W,I)$ and $\lambda_d(W,I),\ d=0,\ldots,L-1$ denote the eigenvectors and eigenvalues of $\mathbf{K}^{(W)}$

$$\mathbf{K}^{(W)}\boldsymbol{v}^{(d)}(W,I)=\lambda_d(W,I)\boldsymbol{v}^{(d)}(W,I), \tag{3.29}$$

where

$$\lambda_0(W,I)\geq\lambda_1(W,I)\geq\ldots\geq\lambda_{L-1}(W,I). \tag{3.30}$$

It can be shown that the eigenvectors $\boldsymbol{v}^{(d)}(W,I)$ and the eigenvalues $\lambda_d(W,I)$ are exactly the multidimensional DPS vectors defined in (3.15) and their corresponding eigenvalues. If the DPS sequences are required for $\boldsymbol{m}\notin I$, they can be extended using equation (3.12).

The multidimensional DPS vectors can theoretically be calculated for an arbitrary passband region W directly from the eigenproblem (3.29). However, since the matrix $\mathbf{K}^{W}$ has an exponentially decaying eigenvalue distribution, this method is numerically unstable.

If W can be written as a Cartesian product of one-dimensional intervals (i. e., W is a hyper-cube),

$$W=W_0\times\ldots\times W_{N-1}, \tag{3.31}$$

where $W_i=[W_{0,i}-W_{\max,i},W_{0,i}+W_{\max,i}]$, and the index-set I is written as

$$I=I_0\times\ldots\times I_{N-1}, \tag{3.32}$$

where $I_i=\{M_{0,i},\ldots,M_{0,i}+M_i-1\}$, the defining kernel $K^{(W)}$ for the multidimensional DPS vectors evaluates to

$$\begin{aligned}K^{(W)}(\boldsymbol{u})&=\int_{W_{0,i}-W_{\max,i}}^{W_{0,i}+W_{\max,i}}\cdots\int_{W_{0,N-1}-W_{\max,N-1}}^{W_{0,N-1}+W_{\max,N-1}}e^{2\pi jf_0''u_0}\cdots e^{2\pi jf_{N-1}''u_{N-1}}\mathrm{d}f_0''\cdots\mathrm{d}f_{N-1}''\\&=\prod_{i=0}^{N-1}K^{(W_i)}(u_i),\end{aligned} \tag{3.33}$$

where $\boldsymbol{u}=[u_0,\ldots,u_{N-1}]^T\in I$. This means that the kernel $K^{(W)}$ is separable and thus the matrix $\mathbf{K}^{(W)}$ can be written as a Kronecker product

$$\mathbf{K}^{(W)}=\mathbf{K}^{(W_0)}\otimes\cdots\otimes\mathbf{K}^{(W_{N-1})}, \tag{3.34}$$

where $\mathbf{K}^{(W_i)},\ i=0,\ldots,N-1$ are the kernel matrices corresponding to the one-dimensional DPS vectors. Now let $\lambda_{d_i}(W_i,I_i)$ and $\boldsymbol{v}^{(d_i)}(W_i,I_i),\ d_i=0,\ldots,M_i-1$ denote the eigenvalues and the eigenvectors of $\mathbf{K}^{(W_i)},\ i=0,\ldots,N-1$ respectively. Then the eigenvalues of $\mathbf{K}^{(W)}$ are given by [79, Chap. 9]

$$\lambda_{\boldsymbol{d}}(W,I)=\lambda_{d_0}(W_0,I_0)\cdots\lambda_{d_{N-1}}(W_{N-1},I_{N-1}),\ d_i=0,\ldots,M_i-1,\ i=0,\ldots,N-1 \tag{3.35}$$

and the corresponding eigenvectors are given by

$$\boldsymbol{v}^{(\boldsymbol{d})}(W,I) = \boldsymbol{v}^{(d_0)}(W_0,I_0) \otimes \cdots \otimes \boldsymbol{v}^{(d_{N-1})}(W_{N-1},I_{N-1}),$$
$$d_i = 0,\ldots,M_i-1,\ i=0,\ldots,N-1. \quad (3.36)$$

The eigenvalues $\lambda_{\boldsymbol{d}}(W,I)$ and the eigenvectors $\boldsymbol{v}^{(\boldsymbol{d})}(W,I)$ are index by $\boldsymbol{d} = [d_0,\ldots,d_{N-1}]^T \in I$. The multidimensional DPS vectors $\boldsymbol{v}^{(d)}(W,I)$ are obtained by reordering the eigenvectors $\boldsymbol{v}^{(\boldsymbol{d})}(W,I)$ and eigenvalues $\lambda_{\boldsymbol{d}}(W,I)$ according to (3.30). Therefore we define the mapping $d = \sigma(\boldsymbol{d})$, such that $\lambda_d(W,I) = \lambda_{\sigma(\boldsymbol{d})}(W,I)$ is the d-th largest eigenvalue. Further define the inverse mapping $\boldsymbol{d} = \delta(d) = \sigma^{-1}(d)$, such that for a given order d of the multidimensional DPS vector $\boldsymbol{v}^{(d)}(W,I)$, the corresponding one-dimensional DPS vectors can be found. When a certain multidimensional DPS sequence of a given order d is needed, the eigenvalues $\lambda_d,\ d=0,\ldots,L-1$ have to be calculated and sorted first. Then the one-dimensional DPS sequences corresponding to $\boldsymbol{d} = \delta(d)$ can be selected.

4 Subspace Representation of Geometry-Based Channel Models

4.1 Introduction

The computer simulation of mobile radio channels is a computationally intensive problem in wireless communications. This is especially true for geometry-based channel simulation introduced in Section 2.6.2, since a large number of complex exponential functions—one for every MPC—must be evaluated and summed up. This chapter presents a novel low-complexity algorithm for geometry-based channel simulation on hardware channel simulators.

On a real-time hardware channel simulator (cf. Section 2.7) the processing power of the baseband unit limits the number of MPCs that can be calculated and hence the model accuracy. We note that the accuracy of the channel simulator is limited by the arithmetic precision of the baseband unit as well as the resolution of the analog/digital converters. On the ARC SmartSim channel simulator [7] for example, the baseband processing hardware uses 16-bit fixed-point processors and an analog/digital converter with 14-bit precision. This corresponds to a maximum achievable accuracy of $E_{\max} = 2^{-13}$.

The new simulation algorithm presented in this chapter exploits the limited numerical accuracy of hardware channel simulators by using a truncated basis expansion of the channel transfer function. The basis expansion is based on the fact that wireless fading channels are highly oversampled. Index-limited snapshots of the sampled fading process span a subspace of small dimension. The same subspace is also spanned by index-limited discrete prolate spheroidal (DPS) sequences [121]. In this chapter, we show that the projection of the channel transfer function onto the DPS subspace can be calculated approximately but very efficiently in $\mathcal{O}(1)$ operations from the MPC parameters given by the model. Furthermore, the subspace representation is independent of the number of MPCs. Thus, in the hardware simulation of wireless communication channels, the number of paths can be increased and more realistic models can be computed. By adjusting the dimension of the subspace, the approximation error can be made smaller than the numerical precision given by the hardware, allowing one to trade accuracy for efficiency. Using the multidimensional

DPS sequences developed in Chapter 3, the DPS subspace representation can also be extended to simulate time-variant wideband MIMO channel models.

One particular application of the new algorithm is the computation of Rayleigh fading processes using Clarke's [33] channel model. As noted in Section 2.6.1, many attempts to reduce the computational complexity of Clarke's model have been made. However, due to the equidistant spacing of the AoAs, none of these models achieves all the desirable statistical properties of Clarke's reference model [19]. Our new approach presented in this chapter allows us to reduce the complexity of Clarke's original model by more than an order of magnitude without imposing restrictions on the AoAs.

This rest of this chapter is organized as follows. In Section 4.2 a subspace representation of time-variant, frequency-flat, single-input single-output (SISO) channels based on DPS sequences is derived and a low-complexity algorithm for the calculation of the basis coefficients of the DPS subspace representation is given. Section 4.3 extends the concepts of the first Section to higher dimensions. In Section 4.4, the multidimensional DPS subspace representation is applied to doubly-selective MIMO channels and numerical examples are given. Section 4.5 concludes the chapter with a discussion of the results.

4.2 One-dimensional DPS Subspace Representation

4.2.1 Time-Variant Frequency-Flat SISO Channels

We start deriving the DPS subspace representation for the generic geometry-based channel model for time-variant, frequency-flat SISO channels depicted in Fig. 4.1. The sampled channel transfer function h_m is given by (2.37), which is repeated here for convenience:

$$h_m = h(mT_S) = \sum_{p=0}^{P-1} \eta_p e^{2\pi j \nu_p m}, \tag{4.1}$$

where the gain β_p and the phase shift ϕ_p are incorporated in complex weight η_p (i.e., $\eta_p = \beta_p e^{2\pi j \phi_p}$) and ν_p is the normalized Doppler shift of the p-th MPC.

We assume that the normalized Doppler shifts ν_p are bounded by the maximum normalized (one-sided) Doppler bandwidth ν_{Dmax}, which is given by the maximum speed $v_{\max}$ of the transmitter, the carrier frequency ω_C, the speed of light c, and the sampling rate $1/T_S$,

$$|\nu_p| \leq \nu_{\mathrm{Dmax}} = \frac{v_{\max}\omega_C}{c} T_S. \tag{4.2}$$

Due to (4.2), the time-variant fading process $\{h_m\}$ is bandlimited to $W = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}]$. Let $I = \{M_0, \ldots, M_0 + M - 1\}$ denote a finite index set on which

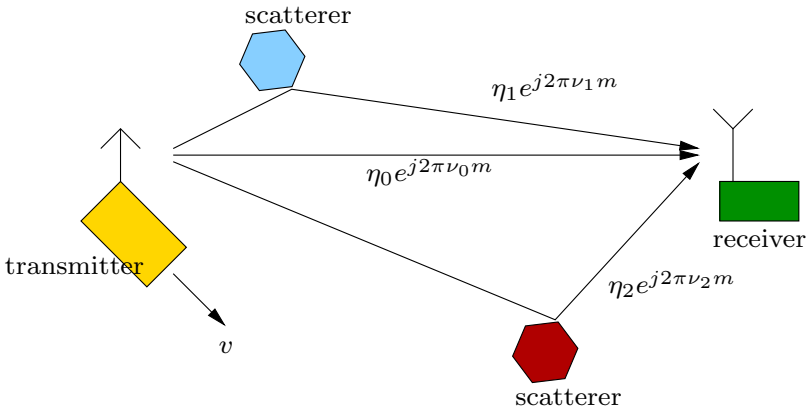

Figure 4.1: Geometry-based channel model for a time-variant, flat-fading SISO channel. The signals sent from the transmitter, moving at speed v, arrive at the receiver via different paths. Each MPC p has complex weight η_p and normalized Doppler shift ν_p [120].

we want to calculate h_m. Due to Property 5 of Theorem 3.2.1, h_m can be decomposed into $h_m = h'_m + g_m$, where h'_m is a linear combination of the DPS sequences $v_m^{(d)}(W, I)$ and $h_m = h'_m$ for all $m \in I$. Therefore, the vectors

$$\boldsymbol{h} = [h_{M_0}, h_{M_0+1}, \ldots, h_{M_0+M-1}]^T \in \mathbb{C}^M \tag{4.3}$$

obtained by index-limiting h_m to I can be represented as a linear combination of the DPS vectors

$$\boldsymbol{v}^{(d)}(W, I) = [v_{M_0}^{(d)}(W, I), v_{M_0+1}^{(d)}(W, I), \ldots, v_{M_0+M-1}^{(d)}(W, I)]^T \in \mathbb{C}^M. \tag{4.4}$$

Properties 2 and 3 of Theorem 3.2.1 show that the first $D' = \lceil 2\nu_{\mathrm{Dmax}} M \rceil + 1$ DPS sequences contain almost all of their energy in the index-set I. Therefore, the vectors $\{\boldsymbol{h}\}$ span a subspace with essential dimension [121]

$$D' = \lceil 2M\nu_{\mathrm{Dmax}} \rceil + 1. \tag{4.5}$$

Due to (2.34), the time-variant fading process is highly oversampled. Thus the maximum number of subspace dimensions M is reduced by $2\nu_{\mathrm{Dmax}} \ll 1$. In typical wireless communication systems, the essential subspace dimension D' is in the order of two to five only. This fact is exploited in the following definition.

Definition 4.2.1. *Let $\boldsymbol{h}$ be a vector obtained by index-limiting a bandlimited process with bandlimit W to the index set I. Further, collect the first D DPS vectors $\boldsymbol{v}^{(d)}(W, I)$ in the matrix*

$$\mathbf{V} = [\boldsymbol{v}^{(0)}(W, I), \ldots, \boldsymbol{v}^{(D-1)}(W, I)]. \tag{4.6}$$

The DPS subspace representation *of* $\boldsymbol{h}$ *with dimension* D *is defined by*

$$\hat{\boldsymbol{h}}^{D} = \mathbf{V}\boldsymbol{\alpha}, \tag{4.7}$$

where $\boldsymbol{\alpha}$ *is the projection of the vector* $\boldsymbol{h}$ *onto the columns of* $\mathbf{V}$

$$\boldsymbol{\alpha} = \mathbf{V}^{H}\boldsymbol{h}. \tag{4.8}$$

For the purpose of channel simulation, it is possible to use $D > D'$ DPS vectors in order to increase the numerical accuracy of the subspace representation. The subspace dimension D has to be chosen such that the bias of the subspace representation is small compared to the machine precision of the underlying simulation hardware. This is illustrated in subsection 4.2.3 with numerical examples.

In terms of complexity, the problem of computing the series (4.1) was reformulated into the problem of computing the basis coefficients $\boldsymbol{\alpha}$ of the subspace representation (4.7). If they were computed directly using (4.8), the complexity of the problem would not be reduced. In the following subsection, we derive a novel low-complexity method to calculate the basis coefficients $\boldsymbol{\alpha}$ approximately.

4.2.2 Approximate Calculation of the Basis Coefficients

In this section, an approximate method to calculate the basis coefficients $\boldsymbol{\alpha}$ in (4.7) with low complexity is presented. Until now we have only considered the time domain of the channel and assumed that the bandlimiting region W is symmetric around the origin. To make the methods in this section also applicable to the frequency domain and the spatial domains (cf. Section 4.3), we make the more general assumption that

$$W = [W_0 - W_{\max}, W_0 + W_{\max}]. \tag{4.9}$$

The projection of a single complex exponential vector

$$\boldsymbol{e}_p = [e^{2\pi j\nu_p M_0}, \ldots, e^{2\pi j\nu_p (M_0+M-1)}]^T$$

onto the basis functions $\boldsymbol{v}^{(d)}(W, I)$ can be written as a function of the Doppler shift ν_p, the bandlimit region W, and the index set I,

$$\gamma_d(\nu_p; W, I) = \sum_{m=M_0}^{M_0+M-1} v_m^{(d)}(W, I) e^{2\pi j m \nu_p}. \tag{4.10}$$

Since $\boldsymbol{h}$ can be written as

$$\boldsymbol{h} = \sum_{p=0}^{P-1} \eta_p \boldsymbol{e}_p \tag{4.11}$$

the basis coefficients $\boldsymbol{\alpha}$ (4.8) can be calculated by

$$\boldsymbol{\alpha} = \sum_{p=0}^{P-1} \eta_p \mathbf{V}^H \boldsymbol{e}_p = \sum_{p=0}^{P-1} \eta_p \boldsymbol{\gamma}_p, \tag{4.12}$$

where $\boldsymbol{\gamma}_p = [\gamma_0(\nu_p; W, I), \ldots, \gamma_{D-1}(\nu_p; W, I)]^T$ denote the basis coefficients for a single MPC.

To calculate the basis coefficients $\gamma_d(\nu_p; W, I)$, we take advantage of the DPS wave functions $U_d(f; W, I)$. For the special case $W_0 = 0$ and $M_0 = 0$ the DPS wave functions are defined in [103]. For the more general case, the DPS wave functions are defined as the eigenfunctions of

$$\int_W \frac{\sin(M\pi(\nu - \nu'))}{\sin(\pi(\nu - \nu'))} U_d(\nu'; W, I) \mathrm{d}\nu = \lambda_d(W, I) U_d(\nu; W, I), \quad \nu \in W. \tag{4.13}$$

They are normalized such that

$$\int_W |U_d(\nu; W, I)|^2 \mathrm{d}\nu = 1, \tag{4.14}$$

$$U_d(W_0; W, I) \geq 0, \quad \left.\frac{\mathrm{d}U_d(\nu; W, I)}{\mathrm{d}f}\right|_{\nu = W_0} \geq 0, \tag{4.15}$$

$$d = 0, \ldots, D-1. \tag{4.16}$$

The DPS wave functions are closely related to the DPS sequences. It can be shown that the amplitude spectrum of a DPS sequence limited to $m \in I$ is a scaled version of the associated DPS wave function (cf. (26) in [103])

$$U_d(\nu; W, I) = \epsilon_d \sum_{m=M_0}^{M_0+M-1} v_m^{(d)}(W, I) e^{-j\pi(2M_0+M-1-2m)\nu}, \tag{4.17}$$

where $\epsilon_d = 1$, if d is even, and $\epsilon_d = j$ if d is odd.

Comparing (4.10) with (4.17) shows that the basis coefficients can be calculated according to

$$\gamma_d(\nu_p; W, I) = \frac{1}{\epsilon_d} e^{j\pi(2M_0+M-1)\nu_p} U_d(\nu_p; W, I). \tag{4.18}$$

The following definition and theorem show that $U_d(\nu_p; W, I)$ can be approximately calculated from $v_m^{(d)}(W, I)$ by a simple scaling and shifting operation [9].

Definition 4.2.2. *Let $v_m^{(d)}(W, I)$ be the DPS sequences with bandlimit region $W = [W_0 - W_{\max}, W_0 + W_{\max}]$ and index set $I = \{M_0, \ldots, M_0 + M - 1\}$. Further denote by $\lambda_d(W, I)$ the corresponding eigenvalues. For $\nu_p \in W$ define the index m_p by*

$$m_p = \left\lfloor \left(1 + \frac{\nu_p - W_0}{W_{\max}}\right) \frac{M}{2} \right\rfloor,$$

Approximate DPS wave functions *are defined as*

$$\tilde{U}_d(\nu_p; W, I) := \pm e^{2\pi j(M_0+M-1+m_p)W_0} \sqrt{\frac{\lambda_d M}{2W_{\max}}} v_{m_p}^{(d)}(W, I), \tag{4.19}$$

where the sign is taken, such that the following normalization holds:

$$\tilde{U}_d(W_0; W, I) \geq 0, \quad \left.\frac{\mathrm{d}\tilde{U}_d(\nu_p; W, I)}{\mathrm{d}\nu_p}\right|_{\nu_p=W_0} \geq 0, \quad d = 0, \ldots, D-1. \tag{4.20}$$

Theorem 4.2.1. *Let $\psi_d(c, f)$ be the prolate spheroidal wave functions [104]. Let $c > 0$ be given and set*

$$M = \left\lfloor \frac{c}{\pi W_{\max}} \right\rfloor.$$

If $W_{\max} \to 0$,

$$\sqrt{W_{\max}}\tilde{U}_d(W_{\max}\nu_p; W, I) \sim \psi_d(c, \nu_p) \tag{4.21}$$

$$\sqrt{W_{\max}}U_d(W_{\max}\nu_p; W, I) \sim \psi_d(c, \nu_p). \tag{4.22}$$

In other words, both the approximate DPS wave functions as well as the DPS wave functions themselves converge to the prolate spheroidal wave functions.

Proof: For $W_0 = 0$ and $M_0 = 0$, i. e., $W' = [-W_{\max}, W_{\max}]$ and $I' = \{0, \ldots, M-1\}$ the proof is given in [103, Sec. 2.6]. The general case follows by using the two identities

$$v_m^{(d)}(W, I) = e^{2\pi j(m+M_0)W_0} v_{m+M_0}^{(d)}(W', I'), \tag{4.23}$$

$$U_d(\nu, W, I) = e^{\pi j(2M_0+M-1)(\nu-W_0)} U_d(\nu - W_0; W', I'). \tag{4.24}$$

□

The theorem suggests that the approximate DPS wave functions can be used as an approximation to the DPS wave functions. Therefore, the basis coefficients (4.18) can be calculated approximately by

$$\tilde{\gamma}_d(\nu_p; W, I) := \frac{1}{\epsilon_d} e^{j\pi(2M_0+M-1)\nu_p} \tilde{U}_d(\nu_p; W, I). \tag{4.25}$$

The theorem does not indicate the quality of the approximation. It can only be deduced that the approximation improves as the bandwidth $W_{\max}$ decreases, while the number of samples $M = \lfloor c/\pi W_{\max} \rfloor$ increases. This fact is exploited in the following definition.

Definition 4.2.3. *Let* $\boldsymbol{h}$ *be a vector obtained by index-limiting a bandlimited process of the form* (4.1) *with bandlimit* $W = [W_0 - W_{\max}, W_0 + W_{\max}]$ *to the index set* $I = \{M_0, \ldots, M_0 + M - 1\}$. *For a positive integer* r—the resolution factor—*define*

$$I_r = \{M_0, M_0 + 1, \ldots, M_0 + rM - 1\}, \quad \textit{and} \tag{4.26}$$

$$W_r = [W_0 - \frac{W_{\max}}{r}, W_0 + \frac{W_{\max}}{r}]. \tag{4.27}$$

Then, the approximate DPS subspace representation *with dimension* D *and resolution factor* r *is given by*

$$\tilde{\boldsymbol{h}}^{D,r} = \mathbf{V}\tilde{\boldsymbol{\alpha}}^r \tag{4.28}$$

whose approximate basis coefficients *are*

$$\tilde{\alpha}_d^r = \sum_{p=0}^{P-1} \eta_p \tilde{\gamma}_d(\nu_p/r, W_r, I_r). \tag{4.29}$$

Note, that the DPS sequences are required in a higher resolution only for the calculation of the approximate basis coefficients. The resulting $\tilde{\boldsymbol{h}}^{D,r}$ has the same sample rate for any choice of r.

4.2.3 Bias of the Subspace Representation

In this subsection, the square bias of the subspace representation

$$\mathrm{bias}^2_{\hat{\boldsymbol{h}}^D} = \mathcal{E}\left\{\frac{1}{M}\|\boldsymbol{h} - \hat{\boldsymbol{h}}^D\|^2\right\} \tag{4.30}$$

and the square bias of the approximate subspace representation

$$\mathrm{bias}^2_{\tilde{\boldsymbol{h}}^{D,r}} = \mathcal{E}\left\{\frac{1}{M}\|\boldsymbol{h} - \tilde{\boldsymbol{h}}^{D,r}\|^2\right\} \tag{4.31}$$

are analyzed.

For ease of notation we assume again that $W = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}]$, i. e., we set $W_0 = 0$ and $W_{\max} = \nu_{\mathrm{Dmax}}$. However, the results also hold for the general case (4.9). If the Doppler shifts $\nu_p,\ p = 0, \ldots, P-1$ are distributed independently and uniformly on W, the DPS subspace representation $\hat{\boldsymbol{h}}$ coincides with the Karhunen-Loeve transform of $\boldsymbol{h}$ [83] and it can be shown that

$$\mathrm{bias}^2_{\hat{\boldsymbol{h}}^D} = \frac{1}{M\nu_{\mathrm{Dmax}}} \sum_{d=D}^{M-1} \lambda_d(W, I). \tag{4.32}$$

Parameter	Value
Carrier frequency ω_C	2 GHz
Sample rate $1/T_\mathrm{S}$	3.84 MHz
Blocklength M	2560 samples
Mobile velocity v_max	100 km/h
Maximum Doppler f_Dmax	185 Hz
Maximum norm. Doppler ν_Dmax	4.82×10^{-5}

Table 4.1: Simulation parameters for the numerical experiments in the time domain. The carrier frequency and the sample rate resemble those of a UMTS system [55]. The blocklength is chosen to be as long as a UMTS frame.

If the Doppler shifts $\nu_p,\ p = 0, \ldots, P-1$ are not distributed uniformly, (4.32) can still be used as an approximation for the square bias [9].

For the square bias of the approximate DPS subspace representation $\tilde{\boldsymbol{h}}^{D,r}$, no analytical results are available. However, for the minimum achievable square bias, we can make the following conjecture

$$\mathrm{bias}^2_{\mathrm{min},r} = \min_D \mathrm{bias}^2_{\tilde{\boldsymbol{h}}^{D,r}} \approx \left(\frac{2\nu_\mathrm{Dmax}}{r}\right)^2. \tag{4.33}$$

The conjecture is substantiated by numerical Monte-Carlo simulations using the parameters from Table 4.1. The Doppler shifts $\nu_p,\ p = 0, \ldots, P-1$ are distributed independently and uniformly on W. The results are illustrated in Fig. 4.2. It can be seen that the square bias of the subspace representation $\mathrm{bias}^2_{\hat{\boldsymbol{h}}^D}$ decays with the subspace dimension. For $D \geq \lceil 2M\nu_\mathrm{Dmax} \rceil + 1$ this decay is even exponential. These two properties can also be seen directly from Equation (4.32) and the exponential decay of the eigenvalues $\lambda_d(W, I)$. The square bias $\mathrm{bias}^2_{\tilde{\boldsymbol{h}}^{D,r}}$ of the approximate subspace representation is similar to $\mathrm{bias}^2_{\hat{\boldsymbol{h}}^D}$ up to a certain subspace dimension. Thereafter, the square bias of the approximate subspace representation levels out at $\mathrm{bias}^2_{\mathrm{min},r} \approx (2\nu_\mathrm{Dmax}/r)^2$. Increasing the resolution factor pushes the levels further down.

Let the maximal allowable square error of the simulation be denoted by E^2_max. Then, the approximate subspace representation can be used without loss of accuracy, if D and r are chosen such that

$$\mathrm{bias}^2_{\tilde{\boldsymbol{h}}^{D,r}} \overset{!}{\leq} E^2_\mathrm{max}. \tag{4.34}$$

Good approximations for D and r can be found by

$$D = \operatorname*{argmin}_D \mathrm{bias}^2_{\hat{\boldsymbol{h}}^D} \leq E^2_\mathrm{max} \quad \text{and} \quad r = \operatorname*{argmin}_r \mathrm{bias}^2_{\mathrm{min},r} \leq E^2_\mathrm{max}. \tag{4.35}$$

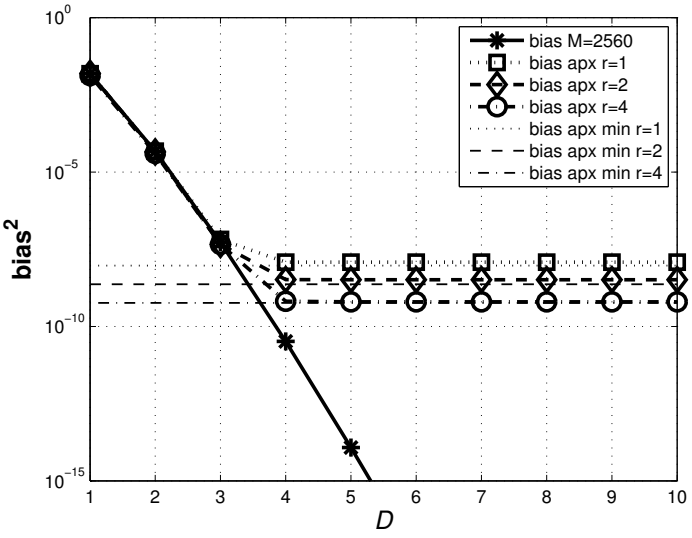

Figure 4.2: $\text{bias}^2_{\tilde{\boldsymbol{h}}_D}$ (denoted by "bias"), $\text{bias}^2_{\tilde{\boldsymbol{h}}_{D,r}}$ (denoted by "bias apx"), and $\text{bias}^2_{\text{min},r}$ (denoted by "bias apx min") for $\nu_{\text{Dmax}} = 4.82 \times 10^{-5}$ and $M = 2560$. The factor r denotes the resolution factor.

The first expression can be computed using (4.32). Using conjecture (4.33), the latter one evaluates to

$$r = \left\lfloor \frac{2\nu_{\text{Dmax}}}{E_{\text{max}}} \right\rfloor . \tag{4.36}$$

On a 14 bit fixed-point processor, the maximum achievable accuracy is $E^2_{\text{max}} = (2^{-13})^2 \approx 1.5 \cdot 10^{-8}$. For the example of Fig. 4.2, where the maximum Doppler shift $\nu_{\text{Dmax}} = 4.82 \times 10^{-5}$ and the number of samples $M = 2560$, the choice $D = 4$ and $r = 2$ makes the simulation as accurate as possible on that hardware. Depending on the application, a lower accuracy might also be sufficient.

4.2.4 Complexity and Memory Requirements

In this subsection, the computational complexity of the approximate subspace representation (4.28) is compared to the SoCE algorithm (4.1). The complexity is expressed in number of complex multiplications (CM) and evaluations of the complex exponential (CE). Additionally, we compare the number of memory access (MA) operations, which gives a better complexity comparison than the actual memory requirements.

We assume that all complex numbers are represented using their real and imaginary part. A CM thus requires four multiplication and two addition operations. As a reference implementation for a CE we use a table look-up with linear interpolation

for values between table elements [7]. This implementation needs six addition, four multiplication and two memory access operations.

Let number of operations that are needed to evaluate $\boldsymbol{h}$ and $\tilde{\boldsymbol{h}}$ be denoted by $C_{\boldsymbol{h}}$ and $C_{\tilde{\boldsymbol{h}}}$ respectively. For the SoCE algorithm, for every $m \in I = \{M_0, \ldots, M_0 + M - 1\}$ and every $p = 0, \ldots, P-1$, a CE and a CM has to be evaluated, i. e.,

$$C_{\boldsymbol{h}} = MP\,\mathrm{CE} + MP\,\mathrm{CM}. \tag{4.37}$$

For the approximate DPS subspace representation with dimension D, first the approximate basis coefficients $\tilde{\boldsymbol{\alpha}}$ have to be evaluated, requiring

$$C_{\tilde{\boldsymbol{\alpha}}} = DP(\mathrm{CE} + 2\,\mathrm{CM} + \mathrm{MA}) + DP\,\mathrm{CM} \tag{4.38}$$

operations where the first term accounts for (4.25) and the second term for (4.29). In total, for the evaluation of the approximate subspace representation (4.28),

$$C_{\tilde{\boldsymbol{h}}} = MD(\mathrm{CM} + \mathrm{MA}) + C_{\tilde{\boldsymbol{\alpha}}} \tag{4.39}$$

operations are required. For large P, the approximate DPS subspace representation reduces the number of arithmetic operations compared to the SoCE algorithm by

$$\frac{C_{\boldsymbol{h}}}{C_{\tilde{\boldsymbol{h}}}} \to \frac{M(\mathrm{CE} + \mathrm{CM})}{D(\mathrm{CE} + 3\,\mathrm{CM})}. \tag{4.40}$$

The memory requirements of the DPS subspace representation are determined by the blocklength M, the subspace dimension D and the resolution factor r. If the DPS sequences are stored with 16-bit precision,

$$\mathrm{Mem}_{\tilde{\boldsymbol{h}}} = 2rMD\,\mathrm{byte} \tag{4.41}$$

are needed.

In Fig. 4.3, $C_{\boldsymbol{h}}$ and $C_{\tilde{\boldsymbol{h}}}$ are plotted over the number of paths P for the parameters given in Table 4.1. Multiplications and additions are counted as one operation. Memory access operations are counted separately. The subspace dimension is chosen to be $D = 4$ according to the observations of the last subsection. The memory requirements for the DPS subspace representation are $\mathrm{Mem}_{\tilde{\boldsymbol{h}}} = 80\,\mathrm{kbyte}$.

It can be seen that the complexity of the approximate DPS subspace representation in terms of number of arithmetic operations as well as memory access operations increases with slope D, while the complexity of the SoCE algorithm increases with slope M. Since in the given example $D \ll M$, the approximate DPS subspace representation already enables a complexity reduction by more than one order of magnitude compared to the SoCE algorithm for $P = 30$ paths. Asymptotically, the number of arithmetic operations can be reduced by a factor of $C_{\boldsymbol{h}}/C_{\tilde{\boldsymbol{h}}} \to 465$.

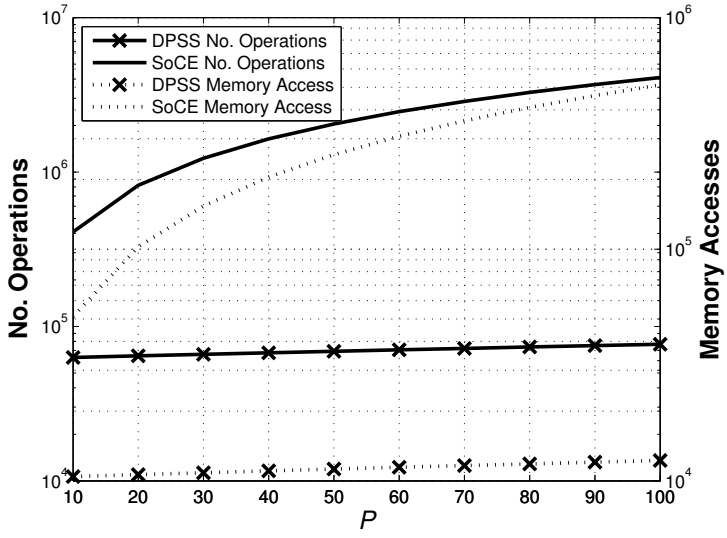

Figure 4.3: Complexity in terms of number of arithmetic operations (left abscissa) and memory access operations (right abscissa) vs. the number of MPCs P. We show results for the sum of complex exponentials algorithm (denoted by "SoCE") and the approximate subspace representation (denoted by "DPSS") using $M = 2560$, $\nu_{\mathrm{Dmax}} = 4.82 \times 10^{-5}$, and $D = 4$

4.3 Extension to Multiple Dimensions

4.3.1 Multidimensional DPS Subspace Representation

The DPS subspace representation (Definition 4.2.1) can be extended to multiple dimensions. The multidimensional equivalent of equation (4.1) is given by

$$h_{\boldsymbol{m}} = \sum_{p=0}^{P-1} \eta_p e^{j2\pi\langle \boldsymbol{f}_p, \boldsymbol{m}\rangle} \quad \text{for } \boldsymbol{m} \in I, \boldsymbol{f}_p \in W, \tag{4.42}$$

where $I \subset \mathbb{Z}^N$ is an N-dimensional finite index set and $W \subset (-1/2, 1/2)^N$ is an N-dimensional interval.

It is easily verified that (4.42) is bandlimited to W. The Fourier transform of $h_{\boldsymbol{m}}$ evaluates to

$$H(\boldsymbol{f}) = \sum_{p=0}^{P-1} \eta_p \delta(\boldsymbol{f} - \boldsymbol{f}_p) \tag{4.43}$$

where $\delta(\boldsymbol{f} - \boldsymbol{f}_p) = \delta(f_0 - f_{0,p}) \cdots \delta(f_{N-1} - f_{N-1,p})$ is the N-dimensional delta function. Since $H(\boldsymbol{f}) = 0$ for $\boldsymbol{f} \notin W$, $h_{\boldsymbol{m}}$ is bandlimited with bandlimit W.

Define by $\boldsymbol{h}$ the vector obtained by index-limiting the sequence (4.42) to the index set I and sorting the elements lexicographically. In analogy to the one-dimensional case, the subspace spanned by $\{\boldsymbol{h}\}$ is also spanned by the multidimensional DPS vectors $\boldsymbol{v}^{(d)}(W, I)$ defined in Chapter 3. Therefore, we can formulate the *multidimensional (approximate) DPS subspace representation* of $\boldsymbol{h}$.

Definition 4.3.1. *Let $\boldsymbol{h}$ be a vector obtained by index-limiting a multidimensional bandlimited process of the form* (4.42) *with bandwidth W to the index set I. Let $\boldsymbol{v}^{(d)}(W, I)$ be the multidimensional DPS vectors for the multidimensional bandlimit region W and the multidimensional index set I. Further, collect the first D DPS vectors $\boldsymbol{v}^{(d)}(W, I)$ in the matrix*

$$\mathbf{V} = [\boldsymbol{v}^{(0)}(W, I), \ldots, \boldsymbol{v}^{(D-1)}(W, I)]. \tag{4.44}$$

Then the multidimensional DPS subspace representation *of $\boldsymbol{h}$ with dimension D is defined by*

$$\hat{\boldsymbol{h}}^{D} = \mathbf{V}\boldsymbol{\alpha}, \tag{4.45}$$

where $\boldsymbol{\alpha}$ is the projection of the vector $\boldsymbol{h}$ onto the columns of $\mathbf{V}$

$$\boldsymbol{\alpha} = \mathbf{V}^{H}\boldsymbol{h}. \tag{4.46}$$

The square bias of the multidimensional DPS subspace representation is defined similarly to the one-dimensional case. If all the $f_{p,i},\ p = 0, \ldots, P-1$ are distributed uniformly on W_i, then it can be shown similarly to the one-dimensional case that

$$\mathrm{bias}^2_{\hat{\boldsymbol{h}}^D} = \frac{1}{|I||W|} \sum_{d=D}^{|I|-1} \lambda_d(W, I). \tag{4.47}$$

The subspace dimension D has to be chosen, such that the bias of the subspace representation is small compared to the machine precision $E_{\max}$ of the underlying simulation hardware, i. e.,

$$D = \operatorname*{argmin}_{D} \mathrm{bias}^2_{\hat{\boldsymbol{h}}^D} \leq E^2_{\max}. \tag{4.48}$$

Alternatively, an estimate for D can be obtained by estimating the subspace dimensions D_i of the corresponding one-dimensional DPS subspace representations and noting that $D \leq D_0 \ldots D_{N-1}$.

The following theorem shows how the multidimensional projection (4.46) can reduced to a series of one-dimensional projections.

Theorem 4.3.1. *Let $\hat{\boldsymbol{h}}^D$ be the N-dimensional DPS subspace representation of $\boldsymbol{h}$ with subspace dimension D, band-limiting region W, and index set I. If W and I can be written as Cartesian products*

$$W = W_0 \times \ldots \times W_{N-1}, \tag{4.49}$$

$$I = I_0 \times \ldots \times I_{N-1}, \tag{4.50}$$

where $W_i = [W_{0,i} - W_{\max,i}, W_{0,i} + W_{\max,i}]$, and $I_i = \{M_{0,i}, \ldots, M_{0,i} + M_i - 1\}$, then for every $d = 0, \ldots, D-1$ there exist $d_0, \ldots, d_{N-1}$ such that the N-dimensional DPS basis vectors $\boldsymbol{v}^{(d)}(W, I)$ can be written as

$$\boldsymbol{v}^{(d)}(W, I) = \boldsymbol{v}^{(d_0)}(W_0, I_0) \otimes \cdots \otimes \boldsymbol{v}^{(d_{N-1})}(W_{N-1}, I_{N-1}). \tag{4.51}$$

Further, the basis coefficients of the approximate DPS subspace representation

$$\tilde{\boldsymbol{h}}^D = \mathbf{V}\tilde{\boldsymbol{\alpha}} \tag{4.52}$$

are given by

$$\tilde{\boldsymbol{\alpha}} = \sum_{p=0}^{P-1} \eta_p \Big(\tilde{\boldsymbol{\gamma}}_p^{(0)} \otimes \cdots \otimes \tilde{\boldsymbol{\gamma}}_p^{(N-1)} \Big), \tag{4.53}$$

where $\tilde{\gamma}_{p,d}^{(i)} = \tilde{\gamma}_{d_i}(f_{p,i}, W_i, I_i)$ are the one-dimensional approximate basis coefficients defined in Equation (4.25). Additionally, resolution factors r_i can be used to improve the approximation.

Proof: For I given by (4.50), $\boldsymbol{h}$ can be written as

$$\boldsymbol{h} = \sum_{p=0}^{P-1} \eta_p \left(\boldsymbol{e}_p^{(0)} \otimes \cdots \otimes \boldsymbol{e}_p^{(N-1)} \right),$$

where $\boldsymbol{e}_p^{(i)} = [e^{2\pi j f_{p,i} M_{0,i}}, \ldots, e^{2\pi j f_{p,i}(M_{0,i}+M_i-1)}]^T$. Further, since W is given by (4.49), the results of Section 3.4 can be used and $\mathbf{V}$ can be written as

$$\mathbf{V} = \mathbf{V}_0 \diamond \cdots \diamond \mathbf{V}_{N-1},$$

where $\diamond$ denotes the column-wise Kronecker product (Khatri-Rao product). Every matrix $\mathbf{V}_i$ contains the one-dimensional DPS vectors $\boldsymbol{v}^d(W_i, I_i)$ in its columns.

Using the identity

$$(\mathbf{A}_0 \diamond \cdots \diamond \mathbf{A}_{N-1})(\boldsymbol{b}_0 \otimes \cdots \otimes \boldsymbol{b}_{N-1}) = \mathbf{A}_0\boldsymbol{b}_0 \otimes \cdots \otimes \mathbf{A}_{N-1}\boldsymbol{b}_{N-1},$$

the basis coefficients $\boldsymbol{\alpha}$ can be calculated by

$$\begin{aligned}\boldsymbol{\alpha} &= \mathbf{V}^H \boldsymbol{h} \\ &= \sum_{p=0}^{P-1} \eta_p \Big(\mathbf{V}_0^H \diamond \cdots \diamond \mathbf{V}_{N-1}^H \Big) \Big(\boldsymbol{e}_p^{(0)} \otimes \cdots \otimes \boldsymbol{e}_p^{(N-1)} \Big) \\ &= \sum_{p=0}^{P-1} \eta_p \Big(\underbrace{\mathbf{V}_0^H \boldsymbol{e}_p^{(0)}}_{=:\boldsymbol{\gamma}_p^{(0)}} \otimes \cdots \otimes \underbrace{\mathbf{V}_{N-1}^H \boldsymbol{e}_p^{(N-1)}}_{=:\boldsymbol{\gamma}_p^{(N-1)}} \Big).\end{aligned}$$

□

The theorem allows to use the methods of Section 4.2.2 to calculate the basis coefficients of the multidimensional DPS subspace representation approximately with low complexity. The resolution factors $r_i,\ i = 0, \ldots, N-1$ have to be chosen, such that the bias of the subspace representation is small compared to the machine precision $E_{\max}$ of the underlying simulation hardware. A necessary, but not a sufficient condition for this, is to use the methods of Section 4.2.3 for each dimension independently, i. e., to choose $r_i = 2W_{\max,i}/E_{\max}$. However, it has to be verified numerically that the multidimensional DPS subspace representation achieves the required numerical accuracy.

4.3.2 Complexity and Memory Requirements

In this subsection we evaluate the complexity and memory requirements of the N-dimensional SoCE algorithm and the N-dimensional approximate DPS subspace representation, given by Theorem 4.3.1. These results are a generalization to the results of Section 4.2.4. We assume that the one-dimensional DPS sequences $\boldsymbol{v}^{(d_i)}(W_i, I_i),\ i = 0, \ldots, N-1$ have been precalculated. Further we assume that $D = D_0 \cdots D_{N-1}$, where $D_i = \max d_i$ is the maximum number of one-dimensional DPS vectors in dimension i needed to construct the N-dimensional vectors up to the dimension D.

Denote the number of operations that are needed to evaluate $\boldsymbol{h}$ (4.63) and $\tilde{\boldsymbol{h}}^D$ (4.52) by $C_{\boldsymbol{h}}$ and $C_{\tilde{\boldsymbol{h}}^D}$ respectively. For the SoCE algorithm,

$$C_{\boldsymbol{h}} = |I| P (\mathrm{CE} + \mathrm{CM}). \tag{4.54}$$

For the approximate DPS subspace representation with dimension D, firstly the N-dimensional DPS basis vectors need to be calculated from the one-dimensional DPS vectors (cf. Equation 4.51), requiring

$$C_{\mathbf{V}} = (N-1)|I| D\, \mathrm{CM}. \tag{4.55}$$

Secondly, the approximate basis coefficients $\tilde{\boldsymbol{\alpha}}$ have to be evaluated according to (4.53), requiring

$$C_{\tilde{\boldsymbol{\alpha}}} = \Big(\sum_{i=0}^{N-1} |D_i|(\mathrm{CE} + \mathrm{CM} + \mathrm{MA}) + ND\,\mathrm{CM} \Big) P. \tag{4.56}$$

In total, for the evaluation of the approximate subspace representation (4.52),

$$C_{\tilde{\boldsymbol{h}}^D} = |I|D(\mathrm{CM} + \mathrm{MA}) + C_{\mathbf{V}} + C_{\tilde{\boldsymbol{\alpha}}} \tag{4.57}$$

operations are required.

Asymptotically for $P \to \infty$, the N-dimensional DPS subspace representation reduces the number of arithmetic operations compared to the SoCE algorithm by

$$\frac{C_{\boldsymbol{h}}}{C_{\tilde{\boldsymbol{h}}}} \to \frac{|I|(\mathrm{CE} + \mathrm{CM})}{\sum_{i=0}^{N-1} D_i(\mathrm{CE} + \mathrm{CM}) + ND\,\mathrm{CM}}. \tag{4.58}$$

The memory requirements of the DPS subspace representation are determined by the size of the index set I, the number of DPS vectors D_i, and the resolution factors r_i. If the DPS sequences are stored with 16 bit precision,

$$\mathrm{Mem}_{\tilde{\boldsymbol{h}}} = \sum_{i=0}^{N-1} 2r_i|I_i|D_i \,\mathrm{byte} \tag{4.59}$$

are needed.

4.4 Doubly-selective MIMO Channels

In this section, the multidimensional DPS subspace representation is applied to the simulation of time-variant, frequency-selective geometry-based MIMO channels. For simplicity we assume uniform linear arrays (ULA) with omnidirectional antennas. Thus, the double-directional transfer function (2.33) simplifies to

$$h(t, f, x, y) = \sum_{p=0}^{P-1} \eta_p e^{2\pi j \omega_p t} e^{-2\pi j \tau_p f} e^{2\pi j/\lambda \sin(\varphi_p) x} e^{-2\pi j/\lambda \sin(\psi_p) y}, \tag{4.60}$$

where t denotes time, f denotes frequency, x the position of the transmit antenna on the ULA, y the position of the receive antenna on the ULA, and λ the wavelength. Each MPC p is characterized by its complex weight η_p, which embodies the gain β_p and the phase shift ϕ_p, time delay τ_p, Doppler shift ω_p, AoD φ_p, and AoA ψ_p.

4.4.1 Observations

In this subsection we show that the sampled time-variant wideband MIMO channel transfer function is band-limited in time, frequency, and space. Denote by F_S the width of a frequency bin and by D_S the distance between antennas. The sampled channel can be described as a four-dimensional sequence $h_{m,q,r,s} = h(mT_S, qF_S, rD_S, sD_S)$, where m denotes discrete time, q denotes discrete frequency, s denotes the index of the transmit antenna, and r denotes the index of the receive antenna. Further, let $\nu_p = \omega_p T_S$ denote the normalized Doppler shift, $\theta_p = \tau_p F_S$ the normalized delay, $\zeta_p = \sin(\varphi_p) D_S/\lambda$ and $\xi_p = \sin(\psi_p) D_S/\lambda$ the normalized angles of departure and arrival, respectively. If all these indices are collected in the vectors

$$\boldsymbol{m} = [m, q, s, r]^T \quad \text{and} \tag{4.61}$$

$$\boldsymbol{f}_p = [\nu_p, -\theta_p, \zeta_p, -\xi_p]^T, \tag{4.62}$$

$h_{\boldsymbol{m}}$ can be written as

$$h_{\boldsymbol{m}} = \sum_{p=0}^{P-1} \eta_p e^{j2\pi\langle \boldsymbol{f}_p, \boldsymbol{m}\rangle}. \tag{4.63}$$

Before the multidimensional DPS subspace representation can be applied, the index-limiting region and the bandlimiting region of (4.63) need to be identified.

Index-Limitation. We assume that for hardware implementation, $h_{\boldsymbol{m}}$ is calculated block-wise for M samples in time, Q bins in frequency, N_{Tx} transmit antennas, and N_{Rx} receive antennas. Accordingly the index set is defined by

$$I = \{0, \ldots, M-1\} \times \{-\lfloor Q/2 \rfloor, \ldots, \lfloor Q/2 \rfloor - 1\} \times \{0, \ldots, N_{\mathrm{Tx}} - 1\} \times \{0, \ldots, N_{\mathrm{Rx}} - 1\}. \tag{4.64}$$

Band-Limitation. The band-limitation of $h_{\boldsymbol{m}}$ is defined by the following physical parameters of the channel.

1. The maximum normalized Doppler shift of the channel ν_{Dmax} defines the band-limitation in the time domain. It is determined by the maximum speed of the user $v_{\max}$, the carrier frequency ω_C, the speed of light c, and the sampling rate $1/T_S$, i. e.,

$$\nu_{\mathrm{Dmax}} = \frac{v_{\max}\omega_C}{c} T_S. \tag{4.65}$$

2. The maximum normalized delay of the scenario $\theta_{\max}$ defines the band-limitation in the frequency domain. It is determined by the maximum delay $\tau_{\max}$ and the sample rate $1/F_S$ in frequency

$$\theta_{\max} = \tau_{\max} F_S. \tag{4.66}$$

3. The minimum and maximum normalized AoA, $\xi_{\min}$ and $\xi_{\max}$ define the band-limitation in the spatial domain at the receiver. They are given by the minimum and maximum AoA, $\psi_{\min}$ and $\psi_{\max}$, the spatial sampling distance D_S and the wavelength λ:

$$\xi_{\min} = \sin(\psi_{\min}) D_S/\lambda, \quad \xi_{\max} = \sin(\psi_{\max}) D_S/\lambda. \tag{4.67}$$

   The band-limitation at the transmitter is given similarly by the normalized minimum and maximum normalized AoD, $\zeta_{\min}$ and $\zeta_{\max}$.

In summary it can be seen that $h_{\boldsymbol{m}}$ is bandlimited to

$$W = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}] \times [0, \theta_{\max}] \times [\zeta_{\min}, \zeta_{\max}] \times [\xi_{\min}, \xi_{\max}]. \tag{4.68}$$

Therefore, the multidimensional DPS subspace representation given in Definition 4.3.1 may be applied to $\boldsymbol{h}$, which is the vector obtained by index-limiting the sequence (4.63) to the index set I (4.64),

$$\tilde{\boldsymbol{h}}^D = \mathbf{V}\tilde{\boldsymbol{\alpha}}. \tag{4.69}$$

Further, the band-limiting region W (4.68) and the index set I (4.64) fulfill the perquisites of Theorem 4.3.1 with

$$\begin{aligned}
&W_{0,0} = 0, && W_{\max,0} = \nu_{\mathrm{Dmax}}, && M_{0,0} = 0, && M_0 = M,\\
&W_{0,1} = \frac{\theta_{\max}}{2}, && W_{\max,1} = \frac{\theta_{\max}}{2}, && M_{0,1} = -\left\lfloor \frac{Q}{2} \right\rfloor, && M_1 = Q,\\
&W_{0,2} = \frac{\zeta_{\max} + \zeta_{\min}}{2}, && W_{\max,2} = \frac{\zeta_{\max} - \zeta_{\min}}{2}, && M_{0,2} = 0, && M_2 = N_{\mathrm{Tx}},\\
&W_{0,3} = \frac{\xi_{\max} + \xi_{\min}}{2}, && W_{\max,3} = \frac{\xi_{\max} - \xi_{\min}}{2}, && M_{0,3} = 0, && M_3 = N_{\mathrm{Rx}}.
\end{aligned}$$

Thus, the the basis coefficients of the multidimensional DPS subspace representation can be calculated with low complexity.

4.4.2 Hybrid DPS Subspace Representation

In addition to the multidimensional DPS subspace representation, we also propose a hybrid DPS subspace representation, that applies a DPS subspace representation in time and frequency domains, and computes the complex exponentials in the spatial domain directly. Therefore the four-dimensional channel transfer function $h_{\boldsymbol{m}}$ (4.63) is split into $N_{\mathrm{Tx}} N_{\mathrm{Rx}}$ two-dimensional transfer functions $h^{r,s}_{\boldsymbol{m}'}$ describing the transfer function between transmit antenna r and receiver antenna s.

$$h^{r,s}_{\boldsymbol{m}'} := h_{\boldsymbol{m}',r,s} = \sum_{p=0}^{P-1} \underbrace{\eta_p e^{-j2\pi\xi_p r} e^{j2\pi\zeta_p s}}_{\eta_p^{r,s}} e^{j2\pi\langle \boldsymbol{f}'_p, \boldsymbol{m}'\rangle} \quad \text{for } \boldsymbol{m}' \in I', \boldsymbol{f}'_p \in W', \tag{4.70}$$

where the bandlimit region W' and the index set I' are given by

$$I' = \{M_0, \ldots, M_0 + M - 1\} \times \{-\lfloor Q/2 \rfloor, \ldots, \lfloor Q/2 \rfloor - 1\}, \tag{4.71}$$

$$W' = [-\nu_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}] \times [0, \theta_{\max}]. \tag{4.72}$$

Then, a the two-dimensional DPS subspace representation can be applied to each

$$\boldsymbol{h}^{r,s} = [h^{r,s}_{M_0,-Q/2}, h^{r,s}_{M_0,-Q/2+1}, \ldots, h^{r,s}_{M_0+M-1,Q/2}]^T$$

independently, resulting in

$$\tilde{\boldsymbol{h}}^{r,s,D} = \mathbf{V}\tilde{\boldsymbol{\alpha}}^{r,s}.$$

4.4.3 DPS Subspace Representation in the Time-Delay Domain

Since most channel simulators filter the signals in the time-delay domain rather than in the time-frequency domain, we need to transform the subspace representation of $\boldsymbol{h}^{r,s}$ by a partial inverse discrete Fourier transform (DFT).

Define the index set

$$J = \{0, \ldots, M - 1\} \times \{0, \ldots, N - 1\},$$

where N is the number of delay bins and for simplicity, it can be assumed that $N = Q$. Denote by $\boldsymbol{H}^{r,s}$ the vector of elements of the time-variant impulse response $H^{r,s}_{m,n}$ indexed lexicographically,

$$\boldsymbol{H}^{r,s} = [H^{r,s}_{0,0}, H^{r,s}_{0,1}, \ldots, H^{r,s}_{M-1,N-1}]^T.$$

Assuming that the channel is band-limited, $\boldsymbol{H}^{r,s}$ is related to $\boldsymbol{h}^{r,s}$ by

$$\boldsymbol{H}^{r,s} = (\mathbf{I}_M \otimes \mathbf{F}_N^{-1})\boldsymbol{h}^{r,s},$$

where $\mathbf{F}_N^{-1}$ be the inverse DFT matrix of size N. A subspace representation of $\boldsymbol{H}^{r,s}$ can be obtained by defining the basis vectors in the time-delay domain as

$$\mathbf{U} = (\mathbf{I}_M \otimes \mathbf{F}_Q^{-1})\mathbf{V}. \tag{4.73}$$

Then the approximate DPS subspace representation of $\boldsymbol{H}^{r,s}$ is given by

$$\tilde{\boldsymbol{H}}^{r,s,D} = \mathbf{U}\tilde{\boldsymbol{\alpha}}^{r,s}. \tag{4.74}$$

Parameter	Value
Width of frequency bin F_S	15 kHz
Number of frequency bins Q	256
Maximum delay $\tau_{\max}$	3.7 μs
Maximum norm. delay $\theta_{\max}$	$\approx 1/18$

Table 4.2: Simulation parameters for the frequency domain.

4.4.4 Numerical Examples

Section 4.2 demonstrated that an application of the approximate DPS subspace representation to the time-domain of wireless channels may save more than an order of magnitude in complexity. In this subsection, the multidimensional approximate DPS subspace representation is applied to an example of a time-variant frequency-selective channel as well as an example of a time-variant frequency-selective MIMO channel. A comparison of the arithmetic complexity is given. We assume a 14-bit fixed-point hardware architecture, i. e., a maximum allowable square error of $E_{\max}^2 = (2^{-13})^2 \approx 1.5 \cdot 10^{-8}$.

4.4.4.1 Time and Frequency Domain

Table 4.2 contains the simulation parameters of the numerical experiments in the frequency domain. The remaining parameters are chosen according to Table 4.1. We assume a typical urban environment with a maximum delay spread of $\tau_{\max} = 3.7\mu$s given by the ITU Pedestrian B channel model [73].

Using conjecture (4.33), the estimated values of the resolution factors are $r_0 = 2\nu_{\mathrm{Dmax}}/E_{\max} \approx 2$ and $r_1 = \theta_{\max}/E_{\max} \approx 512$ (rounded to the next power of two). The square bias

$$\mathrm{bias}^2_{\tilde{\boldsymbol{h}}^D} = \mathcal{E}\left\{\frac{1}{MQ}\|\tilde{\boldsymbol{h}}^D - \boldsymbol{h}^D\|^2\right\} \tag{4.75}$$

of the two-dimensional exact and the approximate DPS subspace representation is plotted in Fig. 4.4 against the subspace dimension D. It can be seen that $\mathrm{bias}^2_{\tilde{\boldsymbol{h}}^D} \approx E_{\max}^2$ at a subspace dimension of approximately $D = 80$. The maximum number of one-dimensional DPS vectors is $D_0 = 4$ and $D_1 = 23$.

4.4.4.2 Time, Frequency, and Spatial Domain

Table 4.3 contains the simulation parameters of the numerical experiments in the spatial domain. The remaining parameters are chosen according to Tables 4.1 and 4.2. We assume uniform linear arrays at the transmitter and the receiver with

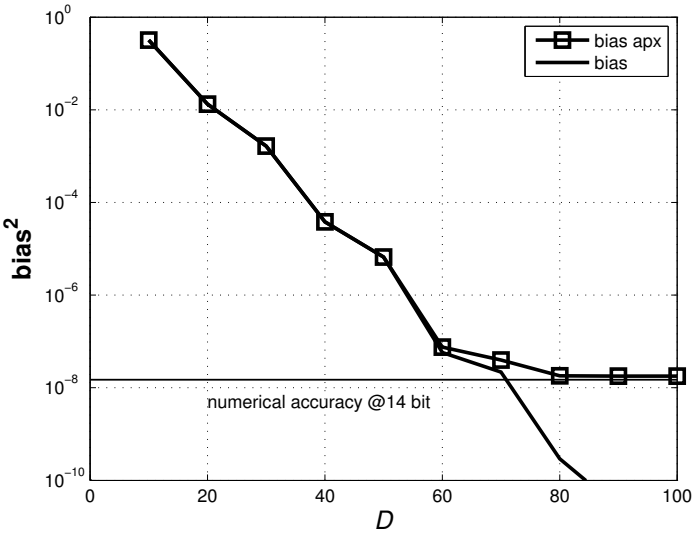

Figure 4.4: $\text{bias}^2_{\hat{\boldsymbol{h}}_D}$ (denoted by "bias") and $\text{bias}^2_{\tilde{\boldsymbol{h}}_D}$ (denoted by "bias apx") for the subspace representation in the time and frequency domain with $\nu_{\text{Dmax}} = 4.82 \cdot 10^{-5}$, $M = 2560$, $\theta_{\max} = 0.056$ and $Q = 256$. The resolution factors are fixed to $r_0 = 1$ and $r_1 = 512$. The thin horizontal line denotes the numerical accuracy of a fixed-point 14-bit processor.

Parameter	**Value**
Spacing between antennas D_S	$\lambda/2$ m
Number of Tx antennas N_{Tx}	8
Number of Rx antennas N_{Rx}	8
AoD interval $[\varphi_{\min}, \varphi_{\max}]$	$[-5°, 5°]$
AoA interval $[\psi_{\min}, \psi_{\max}]$	$[-5°, 5°]$
Normalized AoD bandwidth $\zeta_{\max} - \zeta_{\min}$	0.087
Normalized AoA bandwidth $\xi_{\max} - \xi_{\min}$	0.087

Table 4.3: Simulation parameters for the numerical experiments in the spatial domains.

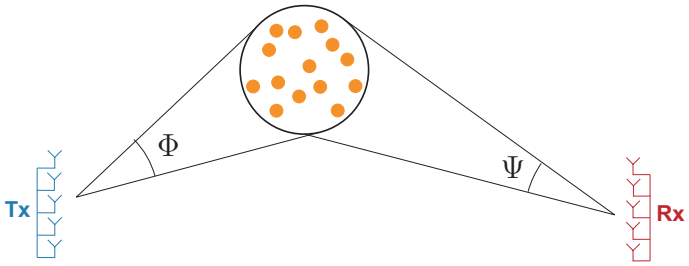

Figure 4.5: Scenario of a mobile radio channel with one cluster of scatterers. The AoD and the AoA are limited within the intervals $\Phi = [\varphi_{\min}, \varphi_{\max}]$ and $\Psi = [\psi_{\min}, \psi_{\max}]$ respectively.

spacing $D_S = \lambda/2$ and $N_{\mathrm{Tx}} = N_{\mathrm{Rx}} = 8$ antennas each. Further we assume, that there is only one cluster of scatterers in the scenario, which is not in the vicinity of the transmitter or receiver (see Fig. 4.5) and we assume no line of sight. The AoA and AoD are assumed to be limited by $[\psi_{\min}, \psi_{\max}] = [\varphi_{\min}, \varphi_{\max}] = [-5°, 5°]$, which has been observed in measurements [36].

A four-dimensional DPS subspace representation is applied to the channel (4.63) with I and W defined in Equations (4.64) and (4.68). Following the same procedure as above the estimated values for the resolution factors and the max. number of one-dimensional DPS sequences in the spatial domains are $r_2 = \zeta_{\max} - \zeta_{\min}/E_{\max} \approx 512$, $r_3 = \xi_{\max} - \xi_{\min}/E_{\max} \approx 512$ (rounded to the next power of 2), and $D_2 = D_3 = 5$.

4.5 Results and Discussion

A complexity comparison of the SoCE algorithm and the approximate DPS subspace representation for one, two, and four dimensions is given in Fig. 4.6. It was evaluated using Equations (4.54) and (4.57). Also shown is the complexity of the four-dimensional hybrid DPS subspace representation.

It can be seen that for time-variant flat-fading channels, the one-dimensional DPS subspace representation requires fewer arithmetic operations for $P > 2$ MPCs. The more number of MPCs are used, the more complexity is saved. Asymptotically, the number of arithmetic operations is reduced by $C_{\boldsymbol{h}}/C_{\tilde{\boldsymbol{h}}} \to 465$.

For time-variant frequency selective channels, the two-dimensional DPS subspace representation requires fewer arithmetic operations for $P > 30$ MPCs. However, as noted in Section 2.6.2, channel models for systems with the given parameters require $P = 400$ paths or more. For such a scenario, the DPS subspace representation also saves two orders of magnitude in complexity. Asymptotically, the number of arithmetic operations is reduced by a factor of $C_{\boldsymbol{h}}/C_{\tilde{\boldsymbol{h}}} \to 6.8 \times 10^3$ (cf. Equation (4.58)). The memory requirements are $\mathrm{Mem}_{\tilde{\boldsymbol{h}}} = 5.83\,\mathrm{Mbyte}$ (cf. Equation (4.59)).

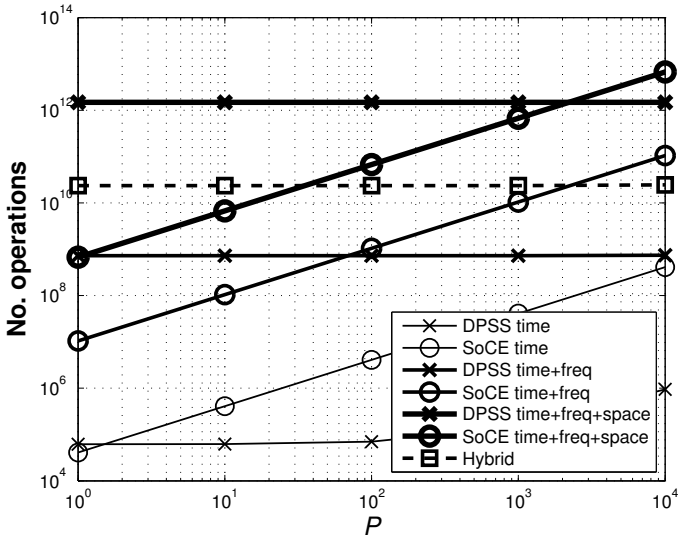

Figure 4.6: Complexity in terms of number of arithmetic operations vs. the number of MPCs P. We show results for the SoCE algorithm (denoted by "SoCE") and the approximate DPS subspace representation (denoted by "DPSS") for one, two, and four dimensions. Also shown is the complexity of the four-dimensional hybrid DPS subspace representation (denoted by "Hybrid").

For time-variant, frequency selective MIMO channels, the four-dimensional DPS subspace representation requires fewer arithmetic operations for $P > 2 \times 10^3$ MPCs. Since MIMO channels require the simulation of up to 10^4 MPCs (cf. Section 2.6.2), complexity savings are still possible. The asymptotic complexity savings are $C_{\boldsymbol{h}}/C_{\tilde{\boldsymbol{h}}} \rightarrow 1.9 \times 10^4$. However, in the region $P < 2 \times 10^3$ MPCs, the four-dimensional DPS subspace representation requires more complex operations than the corresponding SoCE algorithm. Thus, even though we choose a "best case" scenario with only one cluster, a small angular spread and a low numerical accuracy, there is hardly any additional complexity reduction if the DPS subspace representation is applied in the spatial domain.

The hybrid DPS subspace representation on the other hand exploits the savings of the DPS subspace representation in the time and frequency domain only. From Fig. 4.6 it can be seen that it has fewer arithmetic operations than the four-dimensional DPS subspace representation and the four-dimensional SoCE algorithm for $60 < P < 2 \times 10^3$ MPCs. Thus the hybrid method is preferable for channel simulations in this region. Further, this method also allows for an efficient parallelization on hardware channel simulation, which is the topic of the next chapter.

5 Low-Complexity Geometry-based MIMO Channel Simulation

5.1 Introduction

The design and optimization of modern radio communication systems requires realistic models of the radio propagation channel. The COST 259 GSCM [34, 54] is a very general MIMO channel model that can be used to simulate a wide range of scenarios. The COST 259 GSCM distinguishes macrocells (outdoor urban), microcells (outdoor city) and picocells (indoor). For the test of mobile radio hardware devices, real-time implementations of such channel models are needed. The ARC SmartSim channel simulator is a real-time hardware MIMO channel simulator that implements the COST 259 GSCM [7]. As noted in Section 2.6.2, for realistic simulation results, a large number of MPCs must be evaluated and summed up in the baseband processing unit of the simulator. On the other hand, the number of MPCs that can be simulated, is limited by the available processing power.

In Chapter 4, it was shown that the overall computational complexity of geometry-based channel models can be reduced by several orders of magnitude by applying a hybrid DPS subspace representation to the time- and space-variant transfer function of the channel. By adjusting the dimension of the subspace, it is possible to trade complexity for accuracy. Furthermore, the subspace representation is independent of the number of paths and the projection can be calculated efficiently in $\mathcal{O}(1)$ operations directly from the path parameters given by the model.

In this chapter, it is shown how the subspace based method can be incorporated in a GSCM and implemented on the ARC SmartSim channel simulator. It is shown that this implementation is equivalent to the implementation described in [7]. The complexity of the proposed method is however smaller for a wide range of scenarios.

This chapter is organized as follows. In Section 5.2, the ARC SmartSim channel simulator and its signal model are described. Section 5.3 describes the COST 259 GSCM in more detail. The implementation of the DPS subspace representation on the ARC SmartSim is treated in Section 5.4. Complexity comparisons and the memory requirements of the methods are given in Section 5.5. A summary is given in Section 5.6.

Figure 5.1: The baseband processing unit of the ARC SmartSim channel simulator.

5.2 The ARC SmartSim Channel Simulator

The hardware of the ARC SmartSim channel simulator is a modular architecture comprised of a baseband signal processing unit, an analog frontend and a radio frequency (RF) frontend. The baseband signal processing unit is a parallel architecture of DSP boards, which are described in Section 5.2.1. The analog and RF boards are described in Section 5.2.2.

5.2.1 The Baseband Processing Unit

The DSP boards basically consist of a Texas Instruments (TI) C6416 DSP and a Xilinx Virtex 2 FPGA. Fig. 5.2 depicts a block diagram of the board. The DSP has a fixed-point architecture with 16 bit precision. The FPGA provides digital baseband interfaces for input and output signals, as well as for communication over the internal backplane. The baseband signals are implemented by means of LVDS (Low Voltage Differential Signalling) and using standard "Channel Link" connectors. A maximum sample rate of 40MSPS is supported and the dynamic range is 16 bit for inputs and 32 bit for outputs. A Motorola Coldfire microprocessor is used for configuration of the board and communication with a PC. A more detailed description of the board is given in [5, 14, 15].

The MIMO channel simulation method is depicted in Fig. 5.3 for a MIMO channel with $N_{\mathrm{Tx}} = 2$ transmit antennas and $N_{\mathrm{Rx}} = 4$ receive antennas. The input signals $x^s(t),\ s = 0, \ldots, N_{\mathrm{Tx}} - 1$, are fed into the first N_{Tx} boards and then distributed to the other boards. Each DSP board computes the channel impulse response $H^{r,s}(t, \tau)$ based on the COST259 GSCM for one pair of transmit and receive antennas (r, s)

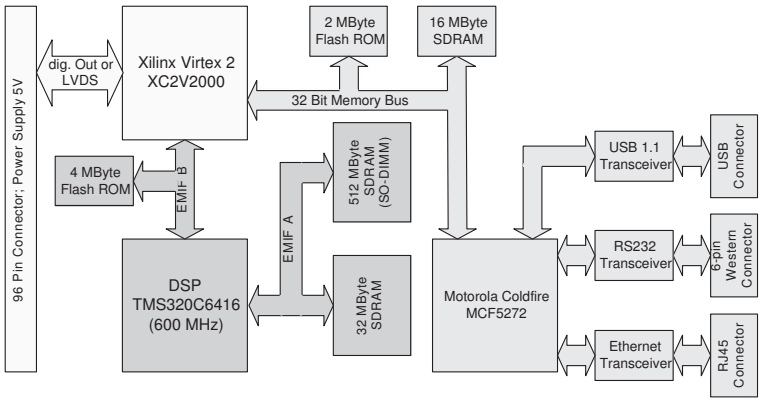

Figure 5.2: Block diagram of the DSP-board.

and its time-variant convolution with the input signal

$$y^{r,s}(t) = \int H^{r,s}(t,\tau)x^{s}(t-\tau)\mathrm{d}\tau. \tag{5.1}$$

Thus, in a $N_{\mathrm{Tx}} \times N_{\mathrm{Rx}}$ MIMO scenario, $N_{\mathrm{Tx}}N_{\mathrm{Rx}}$ DSP-boards are needed. After the time-variant convolution, the results of N_{Tx} adjacent boards are summed up, resulting in

$$y^{r}(t) = \sum_{s=0}^{N_{\mathrm{Tx}}-1} y^{r,s}(t). \tag{5.2}$$

5.2.2 Analog and RF Units

Each DSP-board can optionally be equipped with an analog frontend providing in- and outputs at an intermediate frequency (IF) of 140 MHz with a maximum bandwidth of 40 MHz and a dynamic range of 14 bit input and 16 bit output respectively. The board comprises a digitizer, which is built by two TI ADS5500, a digital to analog (D/A) converter plus a Xilinx XC2VP30 FPGA, which handles I/O interfacing, digital mixing, and controlling of the board.

The RF frontends are separated from the rest of the system and reside in an own rack identical to the rack of the platform to reduce influences on the system. Both the transmitter and the receiver are implemented on a separate board. The current version of the RF receiver is designed for WiMAX signals with a center frequency of 2.58 GHz and a bandwidth of 20 MHz. Extensions complying WLAN and UMTS standards are planned for the next version [65].

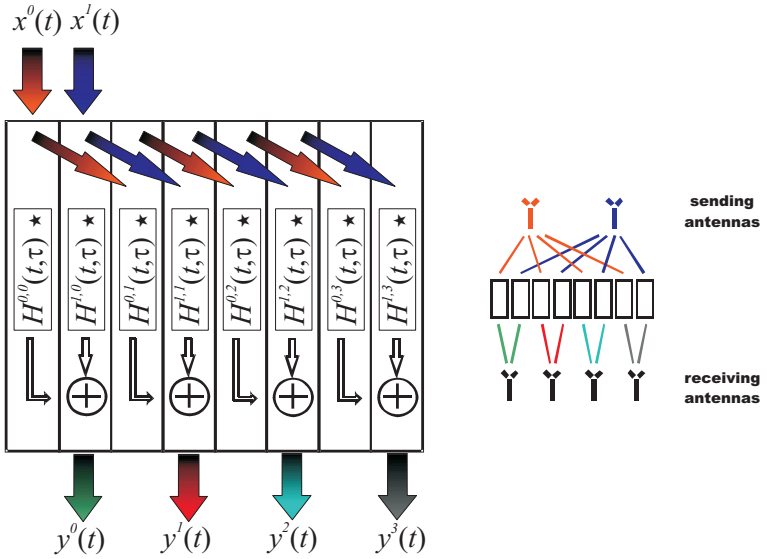

Figure 5.3: MIMO 2 → 4 scenario and the corresponding configuration of development platform.

5.3 The COST 259 Geometry-based Stochastic Channel Model

The principle of the COST 259 GSCM is shown in Fig. 5.4 [34, 54]. A base station (BS) and a mobile station (MS), each having several antennas, are placed within the simulation area and a velocity vector is assigned to the MS. According to the simulated scenario, clusters of scatterers are placed in simulation area. For example, Fig. 5.4 shows a microcell scenario with a cluster of scatterers near the BS, another one around the MS, and one far cluster. The positions of the scatterers in the clusters are calculated according the cluster's angular delay power spectrum (ADPS) at the beginning of a simulation run and are assumed to be stationary within one simulation run [78].

Assuming specular reflection at the scatterers, raytracing is used to compute the MPCs of the channel. Transmitted signals can be scattered once (dashed lines) or twice (dotted lines); cf. Fig. 5.4. The solid line denotes the line of sight path. The parameters of the MPCs can be regarded as constant for displacements of the MS within a range of a few wavelengths (except for the phase). Such a range is called local area [78].

For every MPC in the local area u, the gain $\beta'_p[u]$, the phase shift $\phi_p[u]$, the delay $\tau_p[u]$, the AoD $\varphi_p[u]$ and the AoA $\psi_p[u]$ of every MPC P are calculated. Within a

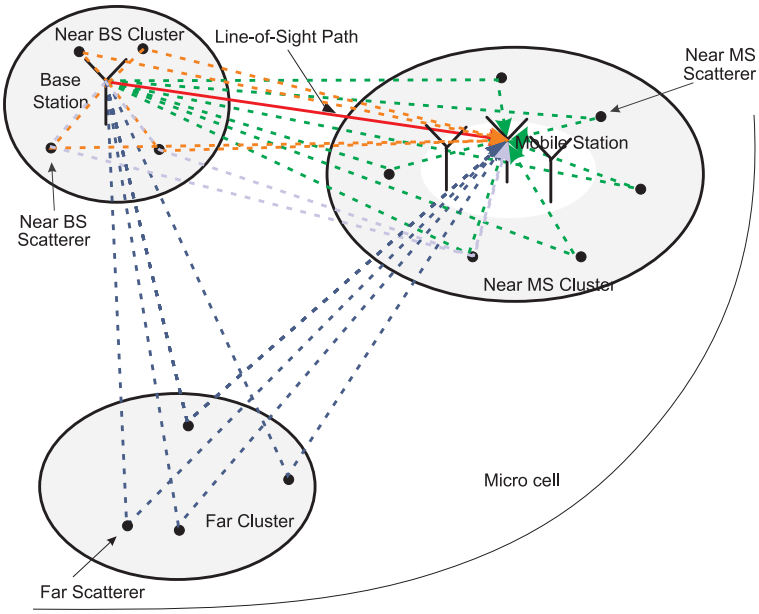

Figure 5.4: Geometrical model for signal propagation.

local area u, the space-variant impulse response is then given by

$$H(\tau, \boldsymbol{x}, \boldsymbol{y}; u) = \sum_{p=0}^{P-1} \beta_p'[u] e^{2\pi j \phi_p[u]} \delta(\tau - \tau_p[u]) e^{2\pi j/\lambda \langle \boldsymbol{\zeta}_p'[u], \boldsymbol{x} \rangle} e^{-2\pi j/\lambda \langle \boldsymbol{\xi}_p'[u], \boldsymbol{y} \rangle}, \tag{5.3}$$

where $\boldsymbol{\zeta}_p'[u]$ and $\boldsymbol{\xi}_p'[u]$ are the normalized vectors pointing in the direction of departure and direction of arrival respectively (compare Section 2.5.3).

Without loss of generality, we assume that the MS acts as the receiver and the BS as the transmitter. Denote by $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{N_{\mathrm{Tx}}-1}$ and $\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N_{\mathrm{Rx}}-1}$, the positions of the transmit and receive antennas within the local area u and assign a constant speed vector $\boldsymbol{v}$ to the MS. Further denote by $G_{\mathrm{Tx}}(\varphi)$ and $G_{\mathrm{Rx}}(\psi)$ the antenna patterns of the transmitter and receiver respectively. Then, the time-variant impulse response between transmit antenna s and receive antenna r can be written as

$$H^{r,s}(t, \tau; u) = \sum_{p=0}^{P-1} \beta_p[u] e^{2\pi j \phi_p^{r,s}[u]} \delta(\tau - \tau_p[u]) e^{-2\pi j \omega_p[u] t}, \tag{5.4}$$

where $\beta_p[u] = \beta_p'[u] G_{\mathrm{Tx}}(\varphi_p) G_{\mathrm{Rx}}(\psi_p)$ is the attenuation including antenna gains,

$$\phi_p^{r,s}[u] = e^{2\pi j/\lambda \langle \boldsymbol{\zeta}_p'[u], \boldsymbol{x}_s \rangle} e^{-2\pi j/\lambda \langle \boldsymbol{\xi}_p'[u], \boldsymbol{y}_r \rangle} \tag{5.5}$$

is the phase shift of path p between transmit antenna s and receive antenna r, and

$$\omega_p[u] = \frac{\langle \boldsymbol{\xi}_p'[u], \boldsymbol{v} \rangle}{\lambda} \tag{5.6}$$

is the Doppler shift of path p.

Assuming ideal bandlimiting filters at the transmitter and the receiver with bandwidth B, the time-variant impulse response can be sampled with rate $1/T_S$,

$$H_{m,n}^{r,s}[u] = \sum_{p=0}^{P-1} \beta_p[u] e^{j2\pi\phi_p^{r,s}[u]} \operatorname{sinc}(nT_S - \tau_p[u]) e^{j2\pi\nu_p[u]m}, \tag{5.7}$$

where $\operatorname{sinc}(x) = \frac{\sin(2\pi x B)}{\pi x}$ and $\nu_p[u] = \omega_p[u]T_S$ is the normalized Doppler shift. The sampled time-variant transfer function is obtained by transforming (5.7) into the frequency domain.

$$h_{m,q}^{r,s}[u] = \sum_{p=0}^{P-1} \beta_p[u] e^{j2\pi\phi_p^{r,s}[u]} e^{-2\pi j\theta_p[u]q} e^{j2\pi\nu_p[u]m}, \tag{5.8}$$

where $\theta_p = \tau_p/T_S$ is the normalized delay.

5.4 Real-time Implementation Aspects

In this Section, the DPS subspace representation is applied to the COST 259 GSCM. We also propose an implementation on the ARC SmartSim channel simulator.

The channel simulation is carried out in parallel—one DSP board for every pair (r, s) of Tx and Rx antennas. This is also depicted in Fig. 5.3 [5, 6]. Only at the beginning of each simulation run, the global parameters of the scenario are initialized and distributed to the DSP boards.

The real-time channel simulation is then divided into three main parts, namely the propagation module, the DPS subspace module, and the convolution module (see Fig. 5.5). The propagation module calculates the parameters of all MPCs of the channel. The DPS subspace module takes those parameters and calculates the subspace coefficients. Both modules are implemented on the DSP. The convolution module is implemented in the FPGA. It takes the input samples and convolves them with the calculated impulse response.

5.4.1 Propagation Module

The computation of the parameters of the MPCs is divided in a large scale and a small scale update to fulfill the real-time constraints.

The large scale update updates the positions of the scatterers according to the stochastic properties of the channel model and thus simulates long term fading. Since these values are varying significantly only after a MS movement of approx. ten wave lengths they are refreshed every 10 ms (corresponding to a carrier frequency of

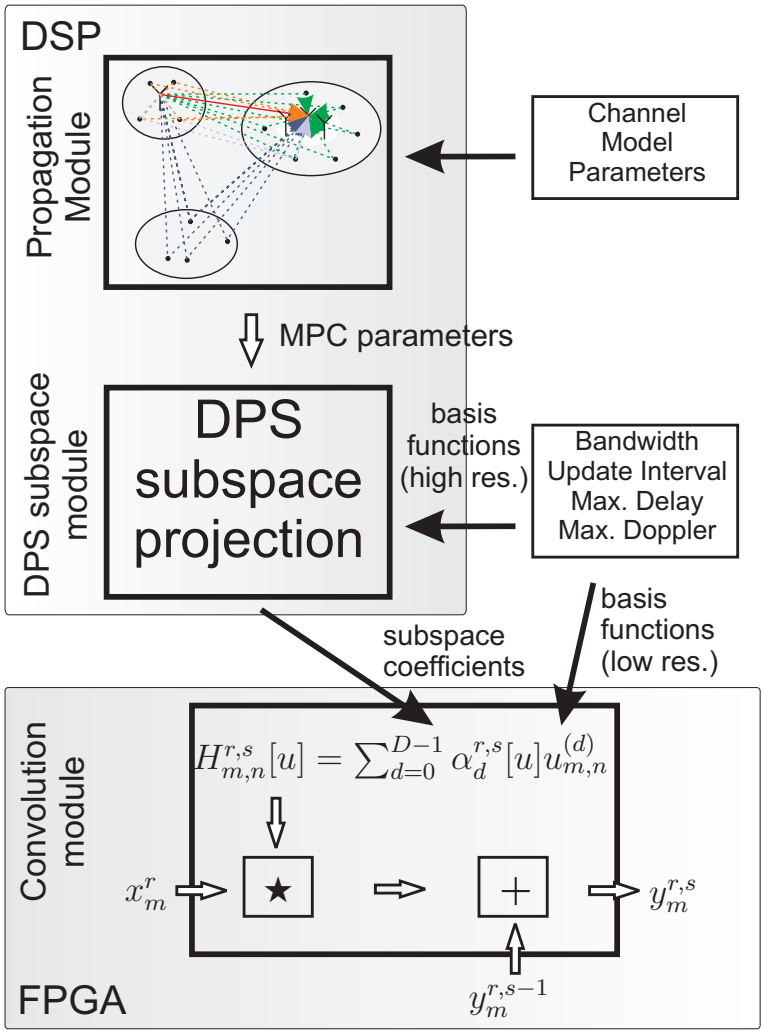

Figure 5.5: Implementation of the GSCM with the DPS subspace projection.

2 GHz and a maximum speed of 500 km/h). In the large scale update, the different attenuation values like the shadow fading, the attenuation of the scatterers and the rice factor are calculated and are linearly interpolated in between. The path lengths of the MPCs are calculated twice in this 10 ms update interval and are piecewise quadratically interpolated in between. For a detailed view on the calculations of these parameters see [34, 54, 56].

The large scale update is further divided into small scale updates of length $T = 0.5\,\text{ms}$, which corresponds to approx. half a wavelength if the MS is moving at maximum speed of 500 km/h. Within a small scale update, the MS stays within a local area. Thus, with slight abuse of notation, we will identify the local area u with the small scale update u. In the small scale update, the attenuation $\beta_p[u]$, the phase $\phi_p^{r,s}[u]$, the AoD $\varphi_p[u]$, the AoA $\psi_p[u]$, the delay $\tau_p[u]$, and the Doppler shift $\nu_p[u]$ of all MPCs $p = 0, \ldots, P-1$ are calculated.

5.4.2 DPS Subspace Module

The DPS subspace module is part of the small scale update. Let $M = T/T_S$ be the number of samples within a small scale update. Further choose $Q > N = N_C + 2N_F + 1$, where $N_C = \lfloor \tau_{\max}/T_S \rfloor$ is maximum delay of the channel in samples and N_F is the group delay of the transmit and receive filters in samples.

In a small scale update $h_{m,q}^{r,s}[u]$ needs to be calculated on the index set

$$I'_u = \underbrace{\{Mu, \ldots, Mu + M - 1\}}_{=:I_{0,u}} \times \underbrace{\{-\lfloor Q/2 \rfloor, \ldots, \lfloor Q/2 \rfloor - 1\}}_{=:I_1}. \tag{5.9}$$

Since $h_{m,n}^{r,s}[u]$ is bandlimited by

$$W' = \underbrace{[-\nu_{\text{Dmax}}, \nu_{\text{Dmax}}]}_{=:W_0} \times \underbrace{[0, \theta_{\max}]}_{=:W_1}, \tag{5.10}$$

where $\nu_{\text{Dmax}} = \|\boldsymbol{v}\| \frac{T_S}{\lambda}$ is the maximum normalized Doppler bandwidth and $\theta_{\max} = \tau_{\max}/T_S$ is the maximum normalized delay in the channel, the two-dimensional DPS subspace representation given in Definition 4.2.3 can be applied.

Denote by $\boldsymbol{h}^{r,s}[u]$ the vector of elements of $h_{m,n}^{r,s}[u]$ index-limited to I'_u and whose elements are sorted lexicographically,

$$\boldsymbol{h}^{r,s}[u] = [h_{Mu,0}^{r,s}[u], h_{Mu+1,0}^{r,s}[u], \ldots, h_{Mu+M-1,N-1}^{r,s}[u]]^T. \tag{5.11}$$

The DPS subspace module now takes the complex attenuation

$$\eta_p^{r,s}[u] = \beta_p[u] e^{j2\pi\phi_p^{r,s}[u]},$$

the normalized Doppler shift $\nu_p[u]$, and the normalized delay $\theta_p[u]$ and computes the basis coefficients $\tilde{\boldsymbol{\alpha}}^{r,s}[u]$ of the DPS subspace representation according to

$$\tilde{\boldsymbol{\alpha}}^{r,s}[u] = \sum_{p=0}^{P-1} \eta_p^{r,s}[u](\tilde{\boldsymbol{\gamma}}_p^{(0)} \otimes \tilde{\boldsymbol{\gamma}}_p^{(1)}), \tag{5.12}$$

where $\tilde{\gamma}_{p,d}^{(0)} = \tilde{\gamma}_d(\nu_p[u], W_0, I_{0,u})$ and $\tilde{\gamma}_{p,d}^{(1)} = \tilde{\gamma}_d(\theta_p[u], W_1, I_1)$ are defined in equation (4.25). For ease of notation, the resolution factors r_0 and r_1 are not included. However, the DPS sequences must be available in the DSP in their maximum resolution.

The DPS subspace representation now writes

$$\tilde{\boldsymbol{h}}^{D,r,s}[u] = \mathbf{V}(W, I)\tilde{\boldsymbol{\alpha}}^{r,s}[u], \tag{5.13}$$

where

$$\mathbf{V}(W, I) = [\boldsymbol{v}^{(0)}(W, I), \ldots, \boldsymbol{v}^{(D-1)}(W, I)] \tag{5.14}$$

is the matrix containing the two-dimensional DPS vectors.

5.4.3 Convolutional Module

The convolution module resides in the FPGA and has the task of computing the time-variant impulse response of the channel and convolve it with the input signal. Therefore, the DPS subspace representation (5.13) needs to be transformed from the time-frequency domain back into the time-delay domain.

The approximate DPS subspace representation of

$$\boldsymbol{H}^{r,s} = [H_{0,0}^{r,s}, H_{0,1}^{r,s}, \ldots, H_{M-1,N-1}^{r,s}]^T.$$

is given by

$$\tilde{\boldsymbol{H}}^{D,r,s}[u] = \mathbf{U}(W, I)\tilde{\boldsymbol{\alpha}}^{r,s}[u], \tag{5.15}$$

where the matrix $\mathbf{U}(W, I)$ has been defined in Section 4.4.3 and can be precalculated and stored in the memory of the FPGA. Compared to the DPS subspace module, in the convolution module the DPS vectors only need to be stored in low resolution.

Finally, the sampled input signal $x_m^s = x^s(mT_S)$ is convolved with the impulse response $\tilde{H}_{m,n}^{r,s}[u]$

$$y_m^{r,s} = \sum_{n=0}^{N-1} \tilde{H}_{m,n}^{r,s}[u]x_{m-n}^s. \tag{5.16}$$

After the convolution, the results of N_{Tx} adjacent boards are summed up, resulting in

$$y_m^r = \sum_{s=0}^{N_{\mathrm{Tx}}-1} y_m^{r,s}. \tag{5.17}$$

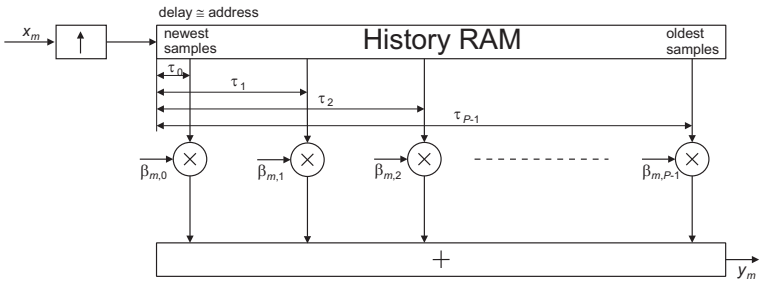

Figure 5.6: Convolution with the History RAM

Equation (5.16) can be implemented using a tapped delay line (TDL) depicted in Fig. 2.6. This implementation is different to the one used in [7], where the input signal is highly oversampled with oversampling factor O_T and written to a "History RAM" (see Fig. 5.6). From there, for each path, the correctly delayed input samples are multiplied with their complex attenuation $\beta_{m,p} = \beta_p(mT_S)e^{2\pi j\phi_p^{r,s}(mT_S)}$ and added up to yield one output sample. Assuming that the oversampling filter has an ideal rectangular spectrum with bandwidth B, it can be shown that the proposed implementation is equivalent to the History RAM implementation.

5.5 Complexity Comparison and Memory Requirements

In this section, the complexities and the memory requirements of three different implementations for the channel simulation are reported. The first two implementations use a tapped delay line, where the tap weights are either generated using DPS sequences (5.15) or where the tap weights are calculated using (5.7). They are compared to the implementation using the History RAM [7]. Further, the memory requirements of the new method are given.

The implementation is divided between the DSP and the FPGA of the DSP board. The complexity and memory requirements of the DPS subspace projection, which resides in the DSP, was already studied in Chapter 4. In this section, only the complexity of the part of the DPS subspace representation that is implemented in the FPGA is considered.

5.5.1 Complexity Comparison

The complexity is measured using number of complex multiplications (CM) and number of evaluations of a complex exponential (CE). We count the number of operations for a single time step m of length T_S.

Parameter	Value
Carrier frequency ω_C	2×10^9 Hz
Bandwidth B	3.84×10^6 Hz
Sample rate $1/T_\mathrm{S}$	15.36×10^6 Hz
Update interval T	0.5×10^{-3} sec
Blocklength $M = T/T_S$	7680 samples
Delay bins Q	256 bins
Filter group delay N_F	5
OSF History RAM O_T	8

Table 5.1: Simulation parameters for the numerical experiments.

The complexity of the History RAM (HR) implementation is

$$C_\mathrm{HR} = (O_T(2N_F + 1))\mathrm{CM} + P(\mathrm{CM} + \mathrm{CE}), \tag{5.18}$$

where the first term accounts for the oversampling filter with oversampling factor O_T and the second term accounts for the number of paths in the HR.

The complexity of the TDL implementation, where the tap weights are calculated using (5.7) is

$$C_\mathrm{TDL} = N\,\mathrm{CM} + P(\mathrm{CM} + \mathrm{CE}), \tag{5.19}$$

where the first term accounts for the implementation of the tapped delay line (5.16) and the second term accounts for calculating the time-variant impulse response (5.7).

The complexity of the TDL implementation, where the tap weights are calculated using DPS sequences (5.15) is

$$C_\mathrm{DPSS} = N\,\mathrm{CM} + D\,\mathrm{CM}, \tag{5.20}$$

where the first term accounts for the implementation of the tapped delay line (5.16) and and the second term accounts for calculating the time-variant impulse response (5.15).

For the complexity comparison, the parameters of Table 5.1 are used. The maximum normalized Doppler shift is fixed to $\nu_\mathrm{Dmax} = 1.2044 \times 10^{-5}$, which corresponds to a user velocity of 100 km/h. In Fig. 5.7, the minimum complexity regions of the different implementations are plotted in dependence of the delay of the channel in samples N_C and the number of paths P. The subspace dimension D has been calculated using (4.34).

It can be seen that the tapped delay line implementation performs best for small delay spreads and small number of paths. If the delay spread is large, but the number of paths is small, i. e., the impulse response of the channel is sparse, then the

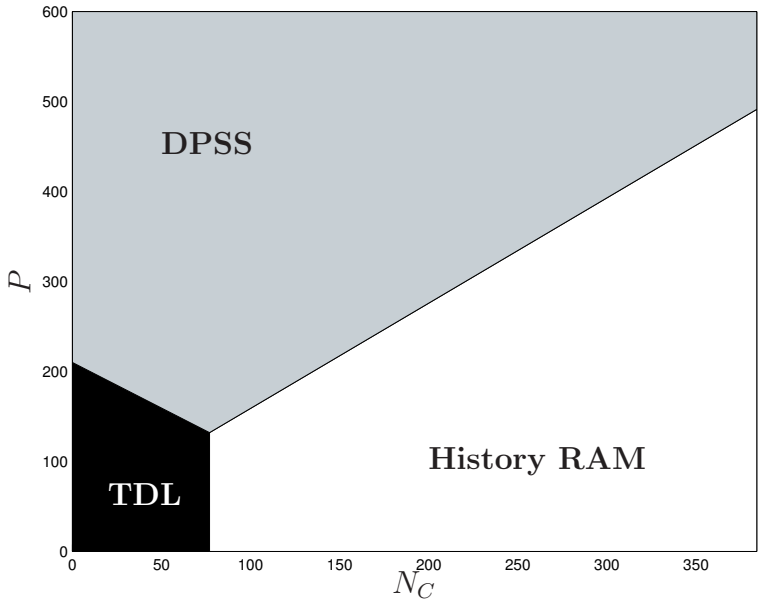

Figure 5.7: Minimum complexity regions of the implementation of the convolution in dependence of the number of paths P and maximum delay of the channel N_C (in samples).

implementation using the History RAM is favorable. In all other cases, i. e., when the number of paths is large, then the implementation using the DPS sequences should be preferred.

5.5.2 Memory Requirements

The special choice of the bandlimit region (5.10) and the time-concentration regions (5.9) as Cartesian products enables efficient storage of the two-dimensional DPS sequences, because they can be calculated as a Kronecker product of one-dimensional sequences. Therefore, to store $\mathbf{U}$, only MD_0+QD_1 elements need to be stored, where D_0 and D_1 are the maximum number of one-dimensional DPS vectors needed to construct the two-dimensional vectors. D_0 and D_1 can be calculated using equation (4.34). The results are summarized in Table 5.2 using the parameters of Table 5.1. For comparison, the History RAM implementation of [7] needs only 8 kByte of memory.

	$\nu_{\mathrm{Dmax}}(\times 10^{-4})$				
N_C	0.1204	0.2409	0.3613	0.4818	0.6022
64	95	99	103	107	110
128	159	163	167	171	174
192	223	227	231	235	238
256	287	291	295	299	302
320	351	355	359	363	366
384	415	419	423	427	430

Table 5.2: Memory requirements for the two-dimensional DPS sequences in kByte in dependence of the maximum normalized Doppler shift ν_{Dmax} and maximum number of delay taps N_C. Every complex sample is allocated 32 bits.

5.6 Summary

We have presented a low-complexity implementation of the COST 259 GSCM, which is especially suited for real-time channel simulation. The method allows to simulate scenarios with a very high number of propagation paths. Compared to other implementations the newly proposed has less complexity for most scenarios. On the other hand, the memory requirements are higher.

6 Conclusions

We have introduced a low-complexity algorithm for the computer simulation of geometry-based MIMO channel models. This algorithm exploits the low-dimensional subspace spanned by multidimensional DPS sequences. By adjusting the dimension of the subspace, it is possible to trade computational complexity for accuracy. Thus the algorithm is ideally suited for fixed-point hardware architectures with limited precision.

We demonstrated that the complexity reduction depends mainly on the normalized bandwidth of the underlying fading process in time, frequency and space. If the bandwidth is very small compared to the sampling rate, the essential subspace dimension of the process is small and the complexity can be reduced substantially. In the time domain, the maximum Doppler bandwidth of the fading process is much smaller than the system sampling rate. Compared with the SoCE algorithm, our new algorithm reduces the complexity by more than one order of magnitude on 14-bit hardware.

The bandwidth of a frequency-selective fading process is given by the maximum delay in the channel, which is a factor of five to ten smaller than the sampling rate in frequency. Therefore, the DPS subspace representation also reduces the computational complexity when applied in the frequency domain. To achieve a satisfactory numerical accuracy, the resolution factor in the approximation of the basis coefficients needs to be large, resulting in high memory requirements. On the other hand, it was shown that the number of memory access operations is small. Since this figure has more influence on the run-time of the algorithm, the approximate DPS subspace representation is preferable over the SoCE algorithm for a frequency-selective fading-process.

The bandwidth of the fading process in the spatial domain is determined by the angular spread of the channel, which is almost as large as the spatial sampling rate for most scenarios in wireless communications. Therefore applying the DPS subspace representation in the spatial domain does not achieve any additional complexity reduction for the scenarios of interest.

As a consequence, a hybrid implementation has been proposed, which applies the DPS subspace representation both in the time and frequency domain and evaluates the spatial fading process directly. This method also enables a natural parallelization of the algorithm, which allows an implementation on the ARC SmartSim channel

simulator. The newly proposed implementation is especially favorable for scenarios with a high number of propagation paths.

Part II

Receiver Architectures

7 Wireless Communication Systems

7.1 Introduction

The performance of wireless communication systems is inherently limited by the wireless communication channel. The understanding of the mobile radio channel and the underlying electromagnetic wave propagation, as well as realistic channel models are therefore of utmost importance for the design of wireless communication systems [30].

In the first part of this thesis the wireless communication channel for time-variant, frequency-selective, multiple-input multiple-output (MIMO) system was studied. In this part of this thesis receiver architectures for state-of-the-art wireless communication systems are developed and their performance is investigated using (*i*) a software simulation approach and (*ii*) a rapid prototyping approach. Special emphasis is put on low-complexity receiver designs and robustness of the algorithm against interference.

The wireless communications systems design starts in general with an algorithm design stage, followed by extensive software simulations (in, e. g., MATLAB) of the proposed algorithms. However, many effects cannot be taken into account in the software simulation that can occur on the final hardware, such as fixed-point issues and real-time constraints. Rapid prototyping enables early elimination of fundamental errors made during the first design stages of wireless systems, drastically cutting costs and time-to-market [96]. With the ARC SmartSim development platform, rapid prototyping of such algorithms, on a hardware likely to be found in the final product is possible. Moreover, the ARC SmartSim development platform also provides a real-time hardware channel simulator allowing unlimited and reproducible evaluation of the prototype.

This part of the thesis is organized as follows. The rest of this chapter gives an introduction to the Universal Mobile Telecommunications Standard (UMTS) with special emphasis on the High Speed Downlink Packet Access (HSDPA) and the Global System for Mobile Communications (GSM).

In **Chapter 8**, different receiver architectures for UMTS-HSDPA are described. To asses the throughput performance of these receivers, we have developed a MATLAB simulation environment specifically tailored to HSDPA. A conventional rake

	IEEE802.11a	**GSM**	**UMTS-FDD**
Frequency range	5.5G Hz	900, 1800, 1900 MHz	2 GHz
Channel bandwidth	20 MHz	200 kHz	5 MHz
Access mode	FDMA/TDMA	FDMA/TDMA	Direct sequence (DS) CDMA
Duplex mode	Half duplex	FDD	FDD
Users per carrier frequency	-	8	-
Modulation	OFDM with subcarrier modulation BPSK/QPSK/ 16QAM/64QAM	GMSK	QPSK
Error correction code	Convolutional	CRC, convolutional	CRC, convolutional, turbo
Bit (chip) rate	6/9/12/18/24/ 36/48/54 Mbps	270.833 kbps	3.840 Mchip/s
Number of bits (chips)/burst (slot)	52 modulated symbols per OFDM symbol	156.25	2560
Frame duration	Packets of several 100 μs	4.615 ms	10 ms
Number of bursts (slots)/frame	Variable	8	15
Burst (slot) duration	1 OFDM symbol of 3.3μs + 0.8 μs guard time	0.577 ms	0.667 ms
Maximum cell radius	Some 10 m	36 km (10 km)	Few km
Spreading sequences	-	-	User specific OVSF codes, call specific scrambling
Spreading factor	1023 or 10230	-	2^k $(k = 2, 3, \ldots, 8)$, 512 for downlink only
Bit (chip) pulseshaping	-	Gauss (BT = 0.3)	Root-raised cosine, filter roll-off factor 0.22
Net datarate	Up to 25 Mbps	13 kbps	8 kbps to 2 Mbps
Evolutionary concepts	IEEE802.11n	GPRS, HSCSD, EDGE	HSDPA
Comparable systems	HiperLAN/2	IS-136, PDC	UMTS-TDD, Cdma2000

Table 7.1: Parameters of selected air interfaces [58].

receiver, a rake receiver with interference cancelation and a receiver employing an linear minimum mean square error (LMMSE) equalizer are investigated. Last but not least, a low-complexity implementation of the LMMSE equalizer using Krylov subspace methods is derived.

In **Chapter 9**, a receiver for GSM employing multiple antennas is developed and its implementation on the ARC SmartSim development platform is described. The smart antenna algorithm requires the solution of an eigenvalue problem in real-time. The main contribution of this chapter is the numerical solution of this eigenvalue problem on a fixed-point processor. Bit error rate (BER) results obtained with the ARC SmartSim channel emulator are given.

Finally, **Chapter 10** gives a summary and conclusions of part two of the thesis.

7.2 The UMTS-HSDPA Standard

The first release (Release 99) of the Universal Mobile Telecommunication Systems (UMTS) standard was published in 1999 by the 3rd Generation Partnership Project (3GPP) [20, 73, 75–77]. UMTS is a mobile communication standard of the 3rd generation offering multimedia services like video telephony, enhanced data services, etc.

The UMTS air interface uses wideband code division multiple access (WCDMA) with a bandwidth of 3.84 MHz. There is a standard for time divison duplex (TDD) and frequency division duplex (FDD), which are quite different in some aspects. In this thesis, we will only focus on the downlink of the FDD standard. WCDMA allows a frequency reuse factor of $r = 1$, i. e., every cell can use the same frequency band [55]. The most important parameters of the UMTS air interface can be seen in Table 7.1.

UMTS features moderately data rates up to 384 kbps for packet data services. Efficient, fast, and flexible assignment of radio resources to packet users is highly desirable from both a user's and an operator's viewpoint. UMTS Release 5 addresses future packet services over links with fluctuating link quality via its High Speed Downlink Packet Access (HSDPA) sub-system [74–77].

HSDPA introduces three physical channels into UMTS Release 5 (see Fig. 7.1). The High Speed Physical Downlink Shared Channel (HS-PDSCH) is the data channel which is shared by all HSDPA users of a single cell in the time and code domain. The HS-PDSCH consists of up to 15 subchannels corresponding to 15 Walsh-Hadamard channelization codes with Spreading Factor (SF) 16. The Transmission Time Interval (TTI) is 2 ms, which is called a subframe. A fast scheduler which resides in the basestation (Node-B) is responsible for selecting packets to be transmitted in each subframe. Up to four High Speed Shared Control Channels (HS-SCCH) inform the selected users on the used Modulation and Coding Scheme (MCS), the current Hybrid Automatic Repeat Request (HARQ) process, and the Redundancy and Constellation Version (RV) of the retransmission.

An uplink signalling channel, the so-called High Speed Dedicated Physical Control Channel (HS-DPCCH), is assigned to each HSDPA user. The HS-DPCCH carries Acknowledgement messages (ACKs), respectively Negative Acknowledgement messages (NACKs), as well as the Channel Quality Indicator (CQI). The ACK/NACK and the CQI are transmitted no later than 7.5 slots after the corresponding subframe was transmitted.

Three closely coupled procedures govern the performance of the HSDPA, namely: Adaptive Modulation and Coding (AMC), fast Hybrid Automatic Repeat reQuest (HARQ), and fast packet scheduling.

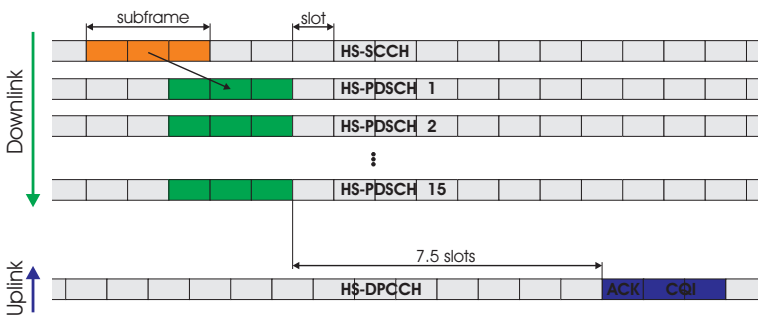

Figure 7.1: HSDPA channels and frame structure.

7.2.1 Adaptive Modulation and Coding

Instead of compensating the varying downlink radio conditions on the HS-PDSCH by means of fast power control at a fixed data rate, the data rate is adjusted depending on measured channel quality in each subframe [84]. The data rate adjustment is achieved by puncturing and repetition ("rate matching") of the rate 1/3 turbo-coded data stream and by selecting either QPSK modulation or 16QAM.

7.2.2 Fast HARQ

In HSDPA, it is foreseen that the user equipment (UE) stores the data from previous transmissions to enable joint decoding of retransmissions with incremental redundancy (IR). The IR versions are generated by rate matching and constellation rearrangement in case of 16QAM. The UE needs internal memory to store the original data packet which is combined with the retransmitted packet [51]. This technique increases the probability of successful decoding for retransmissions significantly. Retransmissions are requested until the data are correctly decoded or a maximum number of attempts is exceeded [47].

7.2.3 Fast Scheduling

A key component of HSDPA is the packet scheduler, which is located in the Node-B. For each TTI the scheduling algorithm controls the allocation of channelization codes on the HS-PDSCH to the users. The scheduling policy itself is not standardized in UMTS. Various trade-offs between simplicity, total throughput, and user fairness can be implemented.

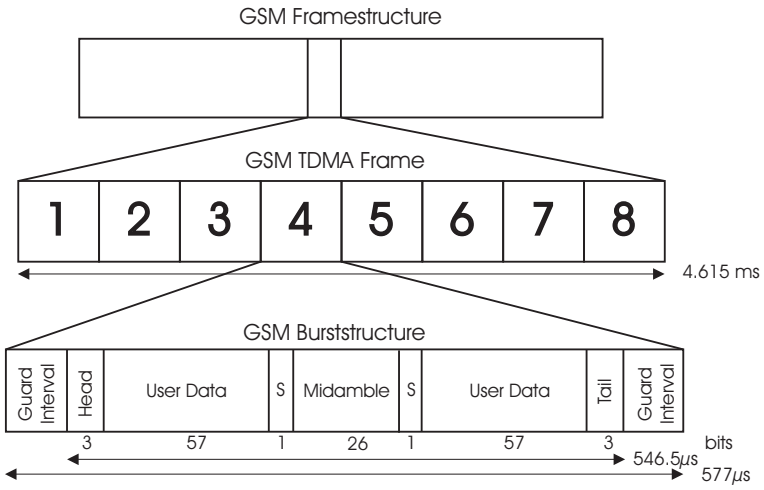

Figure 7.2: Burst structure of GSM

7.3 The GSM Standard

The first standard for the Global System for Mobile Communications (GSM) was published in 1990. It was the first standard for fully digital mobile networks (2nd generation networks). With over 2 billion subscribers worldwide (November 2006) it is the most successful mobile communication standard worldwide. The most important parameters of the GSM air interface are given in Table 7.1.

GSM is a cellular system using a mixture of time division multiplexing (TDMA) and frequency division multiplexing (FDMA). Uplink and downlink operate in different frequency bands (frequency division duplex - FDD). Each frequency band is divided into several channels of 200 kHz bandwidth. In order to reduce the interference in the system, neighboring cells cannot use the same channels. Therefore, a frequency reuse factor $r > 1$ is necessary.

GSM uses Gaussian minimum shift keying (GMSK) to modulate the data. The frame structure of GSM is depicted in Fig. 7.2. One frame is 4.615 ms long and consists of eight bursts. One burst is 577 μs long, where 546.5 μs contain data and the rest comprises the guard interval. There exist different kind of bursts (synchronization bursts, frequency correction bursts, etc.). The most common one is the normal burst, which is also shown in Fig. 7.2. The normal burst consists of 148 bits, where 114 bits are used for data, 26 bits for training (midamble). The rest are predefined signalling bits.

The bits of the midamble are predefined training sequence codes (TSCs) and can be used for channel estimation at the receiver. There exist eight different TSCs in the GSM standard [43] that could be used to distinguish different users. However, in current GSM systems, the same TSC is used for all users in the same cell.

8 Throughput Maximizing Receiver Architectures for UMTS-HSDPA

8.1 Introduction

In third generation wireless communication systems, enhanced data rates are required to enable multimedia services like video telephony and web browsing. In UMTS this is achieved by using WCDMA with a bandwidth of 3.84 MHz. For certain channels, however, the performance of this system is severely limited by multiple access interference (MAI). This is especially true for the HSDPA subsystem of UMTS.

MAI arises in WCDMA systems from different sources. The first source of MAI stems from the fact that the orthogonal variable spreading factor (OVSF) codes used to distinguish the different channels in UMTS are not orthogonal to time-shifted versions of themselves. Therefore in a frequency selective channel the different channel cannot be separated without error. Further interference is generated by the synchronization channel (SCH). Since the SCH must be detected before all other channels it is neither spreaded nor scrambled. Thus it is not orthogonal to the other channels and creates additional interference. Last but not least, interference is also generated from neighboring cells due to the frequency reuse factor of $r = 1$ and the fact that the scrambling sequences of different cells are not 100 % orthogonal.

In this chapter, several receiver architectures that try to alleviate MAI are investigated. The signal model is defined in Section 8.2. In Section 8.3 the basic rake receiver for UMTS HSDPA is explained and enhancements using interference cancelation are presented. In Section 8.4, a linear minimum mean square error (LMMSE) equalizer is used instead of a rake receiver. Section 8.5 introduces an iterative algorithm based on Krylov subspace projections, approximating the LMMSE equalizer with negligible loss of performance for the receiver. Section 8.6 describes the simulation environment developed for UMTS HSDPA. Sections 8.7 and 8.8 present throughput simulation results for the rake and the LMMSE receiver. Finally conclusions are given in Section 8.9.

Most of this work has been presented in [1–4].

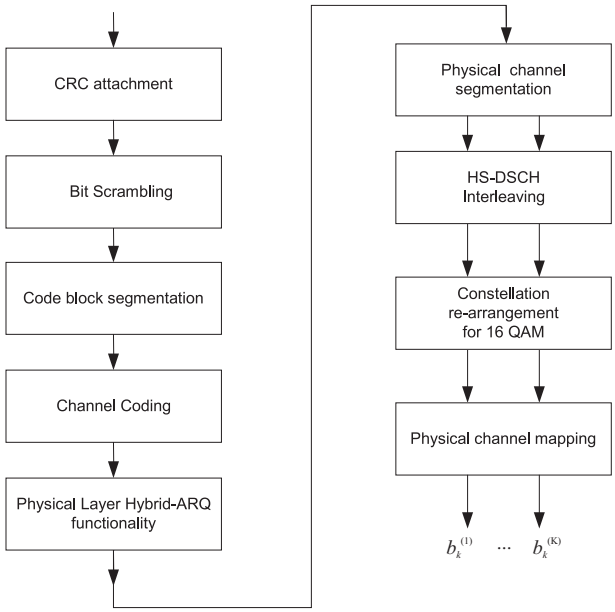

Figure 8.1: Coding chain for HS-PDSCH [75].

8.2 Signal Model

Our signal model for UMTS HSDPA consists of the high speed physical downlink shared channel (HS-PDSCH), the primary common pilot channel (P-CPICH), the primary synchronization channel (P-SCH) and the primary common control physical channel (P-CCPCH).

The coding chain for the HS-PDSCH is depicted in Fig. 8.1. Data is taken form the packed scheduler, coded and segmented into K HS-PDSCHs. The transport block size determines the rate matching parameters in the HARQ block and thus the effective coding rate. For a detailed description of the coding chain see [55, 75]. The coded bits $b_k^{(i)}$ are mapped onto the data symbols $d_l^{(i)}$ by either QPSK or 16QAM modulation. There are $M = 480$ symbols in a subframe per data channel i.

The primary common pilot channel P-CPICH consists of 30 predefined symbols p_l to enable channel estimation at the receiver. The data $d_l^{(i)}$ symbols, the pilot symbols p_l, and the symbols q_l of the primary common control physical channel P-CCPCH are spreaded by Walsh-Hadamard sequences $c_m^{(i,\mathrm{SF})}$, which are also called OVSF codes. The spreading factor of the pilot channel as well as the primary common control physical channel is $\mathrm{SF}_p = \mathrm{SF}_q = 256$ and the one of the data channel $\mathrm{SF}_d = 16$. The P-CCPCH is time shared with the synchronization channel SCH. The SCH is neither spreaded nor scrambled. It consists of 256 chips s_m taken from a Golay-sequence. The remaining 2304 chips per slot are used by the P-CCPCH.

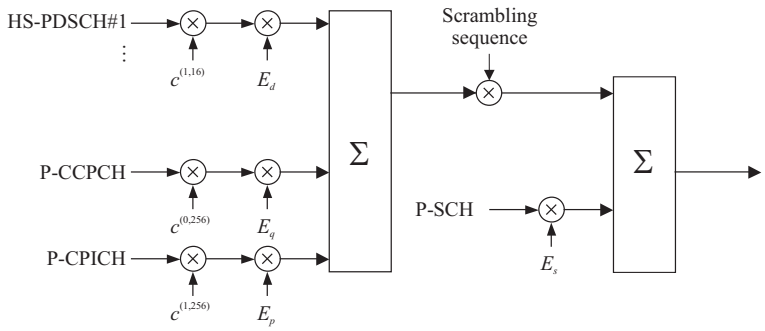

Figure 8.2: Combining of downlink physical channels [77].

After spreading and scrambling, the transmit signal is expressed as (see Fig. 8.2):

$$t_m = a_{m \bmod 38400} \times \Bigg\{ \sqrt{E_d} \sum_{i=1}^{K} c^{(i,\mathrm{SF}_d)}_{m \bmod \mathrm{SF}_d} \, d^{(i)}_{\lfloor m/\mathrm{SF}_d \rfloor} + \sqrt{E_q} \, c^{(0,\mathrm{SF}_q)}_{m \bmod \mathrm{SF}_q} \, q_{\lfloor m/\mathrm{SF}_q \rfloor} + \sqrt{E_p} \, c^{(1,\mathrm{SF}_p)}_{m \bmod \mathrm{SF}_p} \, p_{\lfloor m/\mathrm{SF}_p \rfloor} + \mathrm{OCNS} \Bigg\} + \sqrt{E_s} \, s_m. \tag{8.1}$$

The sequence a_n denotes the scrambling code that is uniquely assigned to a particular base station. It is a Gold-sequence that is terminated after 38400 chips. The five terms in (8.1) make up the N HSDPA data channels, the primary common control physical channel, the pilot signal, the parts that resemble the orthogonal channel noise simulator (OCNS), and the synchronization signal. The OCNS accounts for other transport channels active in the cell. The weighting factors $\sqrt{E_d}$, $\sqrt{E_q}$, $\sqrt{E_p}$, and $\sqrt{E_s}$ are calculated from the relative power ratios of the channels to the total transmit power spectral density (i. e., E_c/I_{or}), whereas $I_{or} = 1$.

The transmit symbol stream t_m is then oversampled and filtered using a root-raised cosine (RRC) filter with roll-off factor $\alpha = 0.22$. At the receiver, a second root-raised cosine filter is applied as a matched filter and downsampling is performed. The equivalent downsampled baseband received signal is given by

$$r_m = \sum_{n=0}^{N-1} H_{m,n} t_{m-n} + v_m, \tag{8.2}$$

where t_{m-n} is the transmitted signal delayed with n samples, $H_{m,n}$ are the downsampled and filtered channel coefficients of the time-variant frequency selective channel, N denotes the delay spread in samples and v_m is i.i.d. zero-mean additive white Gaussian noise with variance σ_v^2. The channel is normalized to unit gain

$$\mathcal{E} \left\{ \sum_{n=0}^{N-1} |H_{m,n}|^2 \right\} = 1. \tag{8.3}$$

8.3 Rake Receiver with Interference Cancelation

A receiver for UMTS HSDPA consists of two parts. The receiver front end comprises a matched filter, downsampling and descrambling (correlation). The second part of the receiver is responsible for the despreading, demodulation and decoding of the received signal. It basically has to invert the channel coding chain from Fig. 8.1.

Due to the wideband nature of the radio propagation channel in a WCDMA system, the signal energy may arrive at the receiver across different time instants. These echoes create interference at the receiver and thus degrade its performance. On the other hand, by using multiple, time-shifted correlation receivers, those echoes can be identified and combined. This kind of receiver is called a rake receiver. Each finger of the rake is allocated to those delay positions on which significant energy arrives. For each rake finger, channel estimation is performed, so that the signals from the different rake fingers can be combined coherently [55].

However, the rake receiver is not able to eliminate all the interference caused by the multipath propagation of the channel. Therefore, in this section we propose to use an additional Interference Cancelation (IC) unit. Fig. 8.3 shows a block diagram of the proposed receiver. The receiver further employs an IR buffer for combining of (previously erroneous) retransmissions. Not shown in the figure is the RRC chip-matched filter and downsampling.

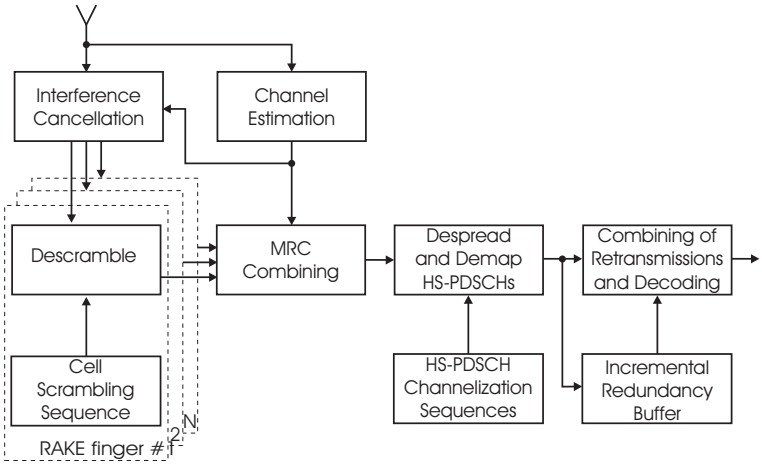

Figure 8.3: Rake receiver with SCH/P-CPICH cancelation.

Channel estimation is carried out with the help of the P-CPICH by means of a symbol-level least squares estimator. The channel is assumed to be constant within one subframe. In case of a frequency selective channel, the channel coefficients are estimated for each delay separately. The delays are assumed to be known at the receiver, although a path-searcher can easily be implemented.

The IC unit respreads the known P-SCH and P-CPICH chip streams with the estimated channel impulse response and subtracts them from the received chip stream. The delayed and descrambled signals from the rake fingers are combined using Maximum Ratio Combing (MRC).

The UE then despreads and demodulates its assigned HS-PDSCH channels using the channelization sequences and modulation index signalled in the HS-SCCH. The demodulator returns log-likelihood ratios (LLRs) of the bits. The LLR of a bipolar bit b is defined as $\ln\left(\frac{p(b=+1)}{p(b=-1)}\right)$, where $p(\cdot)$ denotes probability and $\ln(\cdot)$ is the natural logarithm.

The LLR values are sent through an inverse rate matching block, which sets punctured bits to 0 and combines (i. e., adds up) repeated bits with bits previously stored in the IR buffer. The result is stored again in the IR buffer and sent to the Turbo decoder. The Turbo decoder uses the max-log-MAP algorithm [27, 110], where after every iteration a cyclic redundancy check (CRC) is performed. If the CRC succeeds or if a maximum of 8 iterations has been reached, the algorithm is terminated. Thus, computational power can be saved.

8.4 LMMSE Equalizer

A rake receiver and interference cancelation are not the only means to improve the performance of an UMTS HSDPA receiver. In this section, a receiver architecture using a chip-rate linear minimum mean square error (LMMSE) equalizer is presented. The LMMSE criterion can be mathematically formulated as minimizing the cost-function

$$J(\boldsymbol{f}) = \mathcal{E}\left\{\left|\boldsymbol{f}^H \boldsymbol{r}_m - t_{m-n}\right|^2\right\}, \tag{8.4}$$

where $\boldsymbol{f} = [f_0, \ldots, f_{N_E-1}]^T$ is the vector of the N_E equalizer coefficients, t_{m-n} is a by n samples delayed version of the transmitted chip stream and $\boldsymbol{r}_m = [r_m, \ldots, r_{m-N_E+1}]^T$ is a column vector of the last N_E received (chip-spaced) samples.

If the channel is assumed to be constant within one subframe, i. e., $H_{m,n} = H_n$ for $m \in T_{\text{subframe}}$, a solution minimizing (8.4) can be found in [99]

$$\boldsymbol{f} = \sigma_s^2 \left(\sigma_s^2 \, \mathbf{H}\mathbf{H}^H + \sigma_v^2 \mathbf{I}\right)^{-1} \mathbf{H}\boldsymbol{e}_n, \tag{8.5}$$

where σ_s^2/σ_v^2 is the signal to noise ratio (SNR), $\mathbf{I}$ stands for the identity matrix, and $\boldsymbol{e}_n$ is a unit vector with a one a the n-th position. $\mathbf{H}$ is the channel matrix with Toeplitz structure and size $N_E \times (N + N_E - 1)$, denoting the convolution with the

channel impulse response

$$\mathbf{H} = \begin{bmatrix} H_0 & \cdots & H_{N-1} & & 0 \\ & \ddots & \ddots & \ddots & \\ 0 & & H_0 & \cdots & H_{N-1} \end{bmatrix} \tag{8.6}$$

The matrix $\mathbf{H}$ has to be found through channel estimation, which is again performed with a symbol-level least squares estimator.

The receiver structure is shown in Fig. 8.4. Like the rake receiver in the previous subsection, the LMMSE receiver also performs demodulation and decoding using an IR buffer.

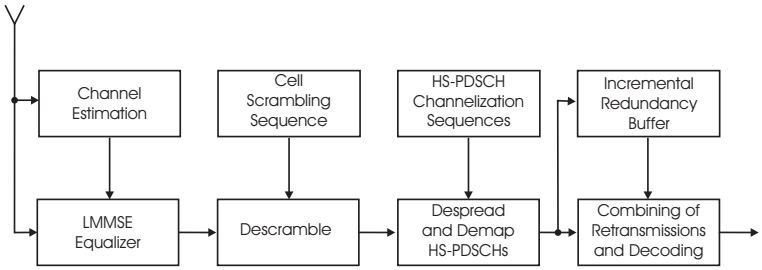

Figure 8.4: HSDPA receiver with LMMSE equalizer and IR buffer.

Complexity. The LMMSE equalizer requires the computation of the $N_E \times N_E$ matrix

$$\mathbf{A} = \sigma_s^2 \, \mathbf{H}\mathbf{H}^H + \sigma_v^2 \mathbf{I}. \tag{8.7}$$

Given the Toeplitz structure of $\mathbf{H}$, this corresponds to about $N(N+1)/2$ CM. Further, we have to solve the system of linear equations $\mathbf{A}\boldsymbol{f} = \boldsymbol{b}$, where $\boldsymbol{b} = \mathbf{H}\boldsymbol{e}_\tau$. It can be seen that $\mathbf{A}$ has also Toeplitz structure, and therefore the solution can be obtained using $2.25N_E^2 + 7.25N_E$ CM [63]. This leads to a total computational complexity

$$C_{\text{LMMSE}} \approx 0.5N(N+1) + N_E(1.25N_E + 7.25) \text{ CM}. \tag{8.8}$$

8.5 LMMSE Equalizer based on Krylov Subspace Methods

It was seen in the last section that the solution of the system of linear equations requires most complexity in the calculation of the LMMSE filter coefficients. In this section we describe the Krylov subspace methods [97] to solve the system of linear equations in the LMMSE equalizer. The Krylov subspace method is an iterative

algorithm. Controlling the number of iterations, it is possible to trade accuracy for efficiency.

We consider the linear system $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$, where $\mathbf{A}$ is a known invertible matrix of size $Q \times Q$ and $\boldsymbol{b}$ a known vector of length Q. Krylov subspace based algorithms [97] iteratively compute an approximation of $\boldsymbol{x}$ starting from an initial guess $\boldsymbol{x}_0$ and using projections on Krylov subspaces. We describe in this section the special case when $\mathbf{A}$ is hermitian.

8.5.1 Krylov Subspace Projection Method

The Cayley-Hamilton theorem states that there is a minimum polynomial $\mathcal{R}$ of degree $R \leq Q$ such that $\mathcal{R}(\mathbf{A}) = \mathbf{0}$ and $\mathbf{I}_Q, \mathbf{A}, \ldots, \mathbf{A}^{R-1}$ are linearly independent, which we can rewrite as $\mathbf{A}^{-1} = \sum_{r=1}^{R} a_r \mathbf{A}^{r-1}$. The coefficients $a_1, \cdots, a_R \in \mathbb{C}$ are defined by $\mathcal{R}$. Our aim is to approximate $\mathbf{A}^{-1}$ as a linear combination of the first s terms, where $s \ll R$

$$\mathbf{A}^{-1} \approx \sum_{r=1}^{s} a_r \mathbf{A}^{r-1} \,. \tag{8.9}$$

We consider an initial guess $\boldsymbol{x}_0$ for $\boldsymbol{x}$ and define the initial error $\boldsymbol{r}_0 = \boldsymbol{x} - \boldsymbol{x}_0$ and $\tilde{\boldsymbol{b}} = \boldsymbol{b} - \mathbf{A}\boldsymbol{x}_0 = \mathbf{A}\boldsymbol{r}_0$. Thus $\boldsymbol{x} = \boldsymbol{x}_0 + \mathbf{A}^{-1}\tilde{\boldsymbol{b}}$ and using (8.9) we write

$$\boldsymbol{x} \approx \boldsymbol{x}_0 + \sum_{r=1}^{s} a_r \mathbf{A}^{r-1}\tilde{\boldsymbol{b}} = \boldsymbol{x}_s \,. \tag{8.10}$$

$\boldsymbol{x}_s$ is the approximation of $\boldsymbol{x}$ at the step s and $\boldsymbol{y}_s = \boldsymbol{x}_s - \boldsymbol{x}_0$ is the residual vector. We also define the error at step s by $\boldsymbol{r}_s = \boldsymbol{x} - \boldsymbol{x}_s = \boldsymbol{r}_0 - \boldsymbol{y}_s$. Note that if $s = R$ the approximation (8.10) becomes an equality and $\boldsymbol{r}_s = 0$ and $\boldsymbol{y}_s = \boldsymbol{r}_0$.

The residual vector $\boldsymbol{y}_s$ is computed as element of the Krylov subspace of $\mathbf{A}$ and $\tilde{\boldsymbol{b}}$ with dimension s defined by $\mathcal{K}_s = \text{span}\left\{\tilde{\boldsymbol{b}}, \mathbf{A}\tilde{\boldsymbol{b}}, \ldots, \mathbf{A}^{s-1}\tilde{\boldsymbol{b}}\right\}$. The error $\boldsymbol{r}_s$ is assumed to be uncorrelated to $\mathcal{K}_s$. In other words, $\boldsymbol{x}_0$ is projected onto $\mathcal{K}_s$ such that $\mathbf{A}\boldsymbol{r}_s \perp \mathcal{K}_s$.

8.5.2 The Krylov Subspace Based Algorithm

We write $\boldsymbol{y}_s$ as element of $\mathcal{K}_s$ as $\boldsymbol{y}_s = \mathbf{V}_s\boldsymbol{z}_s$, where $\boldsymbol{z}_s \in \mathbb{C}^s$ and $\mathbf{V}_s = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_s]$ is an orthonormal basis of $\mathcal{K}_s$. $\mathbf{V}_s$ is iteratively obtained by applying the Gram-Schmidt orthonormalization to the basis $\boldsymbol{B}_s = [\mathbf{V}_{s-1}, \mathbf{A}^{s-1}\tilde{\boldsymbol{b}}]$. The condition $\mathbf{A}\boldsymbol{r}_s \perp \mathcal{K}_s$ becomes

$$\mathbf{V}_s^H \mathbf{A}\boldsymbol{r}_s = 0 \quad \Leftrightarrow \quad \mathbf{V}_s^H \tilde{\boldsymbol{b}} = \mathbf{V}_s^H \mathbf{A}\mathbf{V}_s\boldsymbol{z}_s \,. \tag{8.11}$$

Furthermore, the vectors $\boldsymbol{v}_i$ for $i \in \{1, \ldots, s\}$ are such that $\mathbf{A}\boldsymbol{v}_i \in \mathcal{K}_{i+1}$. Thus $\boldsymbol{v}_l^H \mathbf{A}\boldsymbol{v}_i = 0$ if $l > i + 1$ and $\mathbf{T}_s = \mathbf{V}_s^H \mathbf{A}\mathbf{V}_s$ is an upper Hessenberg matrix. $\mathbf{A}$ being

symmetric, $\mathbf{T}_s$ is consequently tridiagonal symmetric. We denote its elements on the main diagonal as $\alpha_i \in \mathbb{R}$ and on the secondary diagonals as $\beta_i \in (0; +\infty)$. We also know that $\tilde{\boldsymbol{b}} = \|\tilde{\boldsymbol{b}}\|\boldsymbol{v}_1$.

Inserting these results into (8.11), we obtain $\boldsymbol{z}_s = \mathbf{T}_s^{-1}\|\tilde{\boldsymbol{b}}\|\boldsymbol{e}_1$, where $\boldsymbol{e}_1 = [1, 0, \ldots, 0]^T$ has length s. We see that $\boldsymbol{z}_s$ is proportional to $\boldsymbol{c}_{\text{first}}^{(s)}$, first column of $\mathbf{T}_s^{-1}$. To compute $\boldsymbol{c}_{\text{first}}^{(s)}$, we apply the matrix inversion lemma for partitioned matrices [79] to the iterative relation

$$\mathbf{T}_s = \begin{bmatrix} \mathbf{T}_{s-1} & \beta_s \tilde{\boldsymbol{e}}_{s-1} \\ \beta_s \tilde{\boldsymbol{e}}_{s-1}^T & \alpha_s \end{bmatrix},$$

where $\tilde{\boldsymbol{e}}_{s-1} = [0, \ldots, 0, 1]^T$ has length $s-1$. This gives the following set of iterative equations

$$\begin{aligned} \boldsymbol{c}_{\text{first}}^{(s)} &= \begin{bmatrix} \boldsymbol{c}_{\text{first}}^{(s-1)} \\ 0 \end{bmatrix} + \gamma_s^{-1} c_{\text{last},1}^{(s-1)*} \begin{bmatrix} \beta_s^2 \boldsymbol{c}_{\text{last}}^{(s-1)} \\ -\beta_s \end{bmatrix} \\ \boldsymbol{c}_{\text{last}}^{(s)} &= \gamma_s^{-1} \begin{bmatrix} -\beta_s \boldsymbol{c}_{\text{last}}^{(s-1)} \\ 1 \end{bmatrix}, \end{aligned} \tag{8.12}$$

where $\boldsymbol{c}_{\text{last}}^{(s)}$ is the last column of $\mathbf{T}_s^{-1}$, and $\gamma_s = \alpha_s - \beta_s^2 c_{\text{last},s-1}^{(s-1)}$ is a scalar. Finally, we obtain our approximation at step s with $\boldsymbol{x} \approx \boldsymbol{x}_s = \boldsymbol{x}_0 + \|\tilde{\boldsymbol{b}}\|\mathbf{V}_s \boldsymbol{c}_{\text{first}}^{(s)}$. The final step S is referred as the number of iterations in the algorithm or as the dimension of the Krylov subspace where we project $\boldsymbol{x}_0$. The corresponding algorithm is summarized in Table 8.1.

8.5.3 Complexity Comparison

Using the Krylov based algorithm, two matrix-vector products in $\mathbf{H}(\mathbf{H}^H \boldsymbol{v}_s) + \sigma^2 \boldsymbol{v}_s$ ($N(N + N_E)$ CM) and two inner products to compute α and β ($2N_E$ CM) are required at every step s. The total computational complexity after S iterations is then

$$C_{\text{Krylov}} \approx S(N_E N + 2N_E + N^2)\text{CM}. \tag{8.13}$$

Furthermore, using the Krylov subspace method allows storage savings: instead of storing $\mathbf{A}$, only $\boldsymbol{v}_s$ for $s \in \{1, \ldots, S\}$ is stored.

The computational complexity as well as the equalizer performance increase with N_E, thus a tradeoff needs to be found. A reasonable choice is $N_E \approx 3N$ [50], where the delay spread N is given by the channel model used.

1	input $\mathbf{A}$, $\boldsymbol{b}$, $\boldsymbol{x}_0$, S
2	$\tilde{\boldsymbol{b}} = \boldsymbol{b} - \mathbf{A}\boldsymbol{x}_0$
3	$\boldsymbol{v}_1 = \tilde{\boldsymbol{b}}/\|\tilde{\boldsymbol{b}}\|$
4	$\boldsymbol{u} = \mathbf{A}\boldsymbol{v}_1$
5	$\alpha = \boldsymbol{v}_1^H \boldsymbol{u}$
6	$\boldsymbol{c}_{\text{first}} = \boldsymbol{c}_{\text{last}} = 1/\alpha$
7	$\boldsymbol{w} = \boldsymbol{u} - \alpha \boldsymbol{v}_1$
8	for $s = 2, \ldots, S$
9	$\quad \beta = \|\boldsymbol{w}\|$
10	$\quad \boldsymbol{v}_s = \boldsymbol{w}/\beta$
11	$\quad \boldsymbol{u} = \mathbf{A}\boldsymbol{v}_s$
12	$\quad \alpha = \boldsymbol{v}_s^H \boldsymbol{u}$
13	$\quad \gamma = \alpha - \beta^2 c_{\text{last},s-1}$
14	$\quad \boldsymbol{c}_{\text{first}}, \boldsymbol{c}_{\text{last}}$ using eq. (8.12)
15	$\quad \boldsymbol{w} = \boldsymbol{u} - \alpha \boldsymbol{v}_s - \beta \boldsymbol{v}_{s-1}$
16	end
17	$\mathbf{V}_S = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_S]$
18	output $\boldsymbol{x}_S = \|\tilde{\boldsymbol{b}}\| \mathbf{V}_S \boldsymbol{c}_{\text{first}} + \boldsymbol{x}_0$

Table 8.1: Krylov subspace based algorithm for a Hermitean matrix.

8.5.4 Choice of Parameters

The error $\boldsymbol{r}_S$ resulting from the Krylov subspace method is bounded by [59]

$$\|\boldsymbol{r}_S\|_A \leq 2\|\boldsymbol{r}_0\|_A \left(\frac{\sqrt{k_A} - 1}{\sqrt{k_A} + 1} \right)^S ,$$

where $k_A > 1$ is the condition number of $\mathbf{A}$ (ratio of largest and smallest eigenvalues). Convergence is thus assured, but the convergence speed depends strongly on the matrix $\mathbf{A}$ and on the initial guess $\boldsymbol{x}_0$. It is necessary to appropriately choose the parameters S and $\boldsymbol{x}_0$.

If no information on $\boldsymbol{f}$ is available at the receiver, we choose $\boldsymbol{x}_0 = [0, \ldots, 0]^T$. However, the LMMSE equalizer (8.5) depends only on the channel estimate $\mathbf{H}$. When $\mathbf{H}$ is varying slowly, the equalizers from one subframe to another are assumed to be strongly correlated. Thus a more suitable choice for initial guess is $\boldsymbol{x}_0^{(sub+1)} = \boldsymbol{f}^{(sub)}$, where $^{(sub)}$ denotes the subframe index. We consider these two approaches, called Standard and Adaptive Krylov respectively.

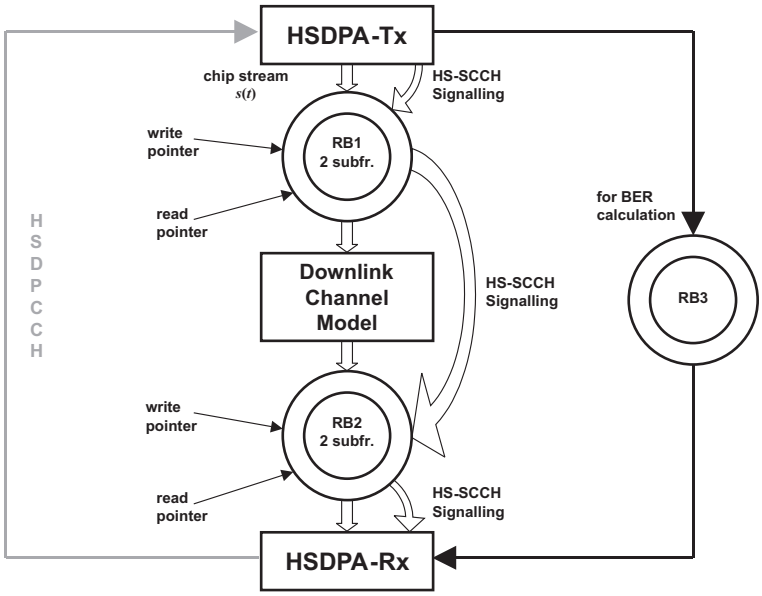

Figure 8.5: HSDPA simulator architecture with three ringbuffers (RB).

8.6 Simulator Description

The HSDPA link-level baseband simulator is implemented in the MATLAB language. The simulator consists of the Node-B transmit side, the downlink channel model, the HSDPA receiver, and three ringbuffers (RB), as illustrated in Fig. 8.5. The simulator is modular such that the transmitter, channel, and receiver can easily be changed by other implementations.

For each subframe the HSDPA transmitter generates the HS-PDSCHs for the scheduled users, the P-SCH, the P-CCPCH and the P-CPICH according to [75]. In our current implementation the Paging Indicator Channel (PICH), the Dedicated Physical Channel (DPCH) and the HS-SCCHs are not transmitted. The power allocated to the active interfering channels is re-assigned to the OCNS transmission, as suggested in [74]. The OCNS is described in further detail below. The HS-SCCHs are signalled error-free to the receiver, see Fig. 8.5.

Data streams for the HS-PDSCHs are generated randomly. The packet size is either defined by the signalled CQI and the UE capability class or by the testcases given in [73]. The information bits for each user are encoded according to [75]. The coding chain includes CRC attachment, channel coding (Turbo Coding), HARQ functionality, and interleaving.

All channels except for the SCH are modulated, spreaded by orthogonal Walsh-Hadamard sequences and subsequently scrambled by the cell-specific Gold sequence. All channels are then weighted and added to a single chip stream according to [77].

The weighting factors are calculated from the relative power ratios of the channels to the total transmit power spectral density (E_c/I_{or}). To achieve a total transmit power spectral density of $I_{or} = 1$ (0 dB), further channels are generated by the OCNS and are added to the chip stream [73]. Those channels have a fixed spreading factor of SF = 128 and are used to simulate the users or control signals on the other orthogonal channels of a downlink.

We assume a frequency selective, time-variant channel model where each tap of the impulse response has a Jakes Doppler spectrum with a maximum Doppler frequency given by the UE speed. The channel is implemented as a time-variant finite impulse response filter with sample spaced taps using the MATLAB `filter` function. We use an oversampling factor of two and RRC pulse shaping filters at the transmitter and receiver. The sample spaced filter coefficients are generated from the International Telecommunications Union (ITU) channel models [73] by a sinc interpolation.

For correct implementation of the filtering operation induced by the downlink channel model, at least two consecutive subframes are generated at the HSDPA transmitter and buffered in RB1 (see Fig. 8.5). This approach is required for modelling the interference caused by the RRC filter at the transmitter, the delay spread of the channel, and the chip-matched filter at the receiver. After matched filtering the subframe is buffered in RB2 before it is read out by the receiver. The receiver processes the subframe including the spill-overs to adjacent subframes. In parallel, the code blocks of each subframe can be stored in RB3 for calculating the BER from the simulations if this is wanted.

For throughput simulations only the information bits excluding CRC bits of acknowledged data blocks (CRC does not fail) are summed up.

8.7 Rake Numerical Experiments

Throughput simulations for an HSDPA receiver with "UE Capability 6" [72] were carried out for QPSK and 16QAM. At the transmitter, the fixed reference channel H-Set 3, as defined in [73] is generated (see Table 8.2).

The relative power ratios of the simulated physical channels to the total transmit power spectral density (E_c/I_{or}) are given in Table 8.3 and are set in compliance to the HSDPA test cases [73]. The E_c/I_{or} of the HS-PDSCH is varied.

No pathloss is assumed ($\hat{I}_{or} = I_{or}$) and the interference from other cells as well as the noise is modeled as additive white Gaussian noise (AWGN) with variance $\sigma^2 = I_{oc}$. We have used $\hat{I}_{or}/I_{oc} \in \{10, 15\}$ dB for QPSK and for 16QAM simulations.

Throughput simulations are carried out for two propagation channels with various cancelation modes.

Parameter	Value	
Modulation	QPSK	16QAM
Nominal Avg. Inf. Bit Rate [kbps]	1601	2332
Inter-TTI Distance	1	1
No. of HARQ Processes	6	6
Coding Rate	0.67	0.61
No. of Physical Channel Codes	5	4

Table 8.2: Fixed Reference Channel H-Set 3.

Parameter	Value
UE capability class	6
Combining	soft
RV coding sequence	$\{0, 2, 5, 6\}$ for QPSK $\{6, 2, 1, 5\}$ for 16QAM
P-CPICH E_c/I_{or}	-10 dB
SCH E_c/I_{or}	-12 dB
P-CCPCH E_c/I_{or}	-12 dB
OCNS	on
$\hat{I}_{or}/I_{oc}$	$\{10, 15\}$ dB
Delay estimation	perfect
Channel coefficient estimation	least squares
Turbo decoding	max-log-MAP - 8 iterations

Table 8.3: HSDPA simulation parameters.

8.7.1 AWGN Channel

We assume an AWGN channel with zero-mean and variance σ_v^2. The simulation is performed with and without cancelation of the SCH channel. The cancelation of any other channel is obsolete due to the orthogonality of the spreading codes. Results for the AWGN channel are shown in Fig. 8.6 for QPSK and in Fig. 8.7 for 16QAM.

8.7.2 Frequency Selective Rayleigh Fading Channel

For simulations with a frequency selective Rayleigh fading channel, the ITU Pedestrian-B channel model is used [73]. This channel model defines a power delay profile with 6 paths at specified delays. The mobile speed is $3\,\text{km/h} \approx 0.83\,\text{m/s}$.

For collecting the entire received energy, the number of rake fingers is set to the number of propagation paths. The simulation setup was the same as in the AWGN case and various IC modes have been tested, namely no IC, IC of the SCH, IC of the P-CPICH and IC of SCH and P-CPICH. The results are shown in Fig. 8.8 for

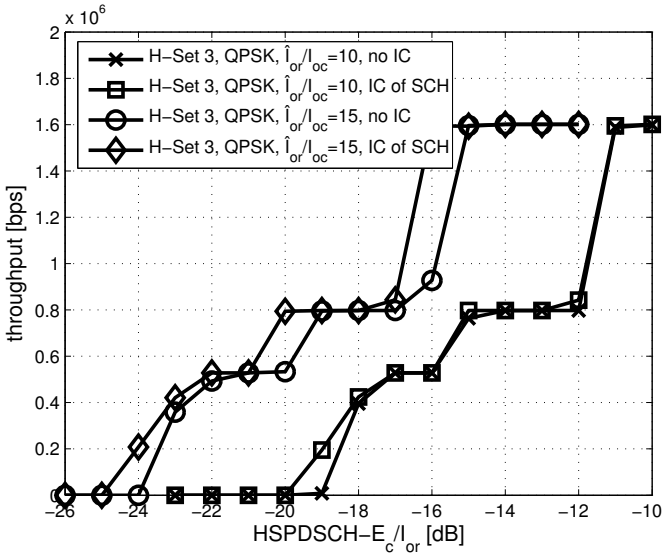

Figure 8.6: QPSK throughput results for AWGN channel for $\hat{I}_{or}/I_{oc} \in \{10, 15\}$ dB with and without IC of SCH.

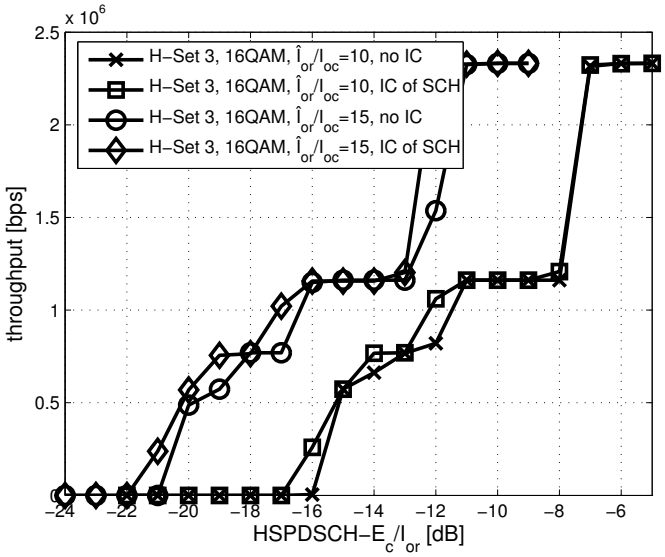

Figure 8.7: 16QAM throughput results for AWGN channel for $\hat{I}_{or}/I_{oc} \in \{10, 15\}$ dB with and without IC of SCH.

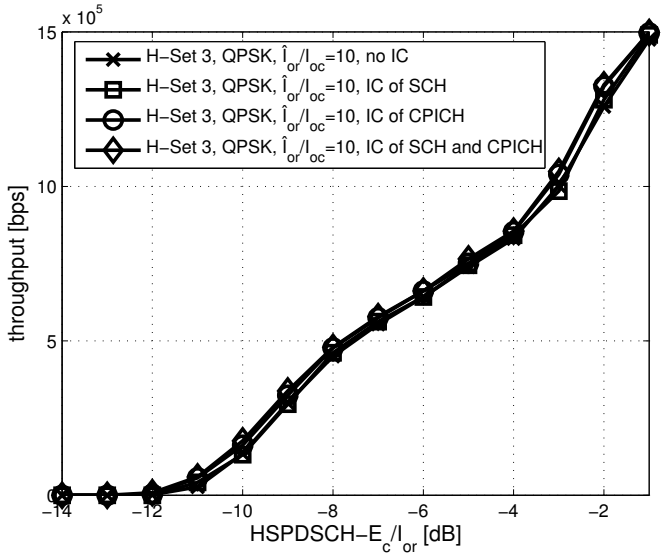

Figure 8.8: QPSK throughput results for ITU Pedestrian-B channel for $\hat{I}_{or}/I_{oc} = 10\,\mathrm{dB}$ with several cancellation modes.

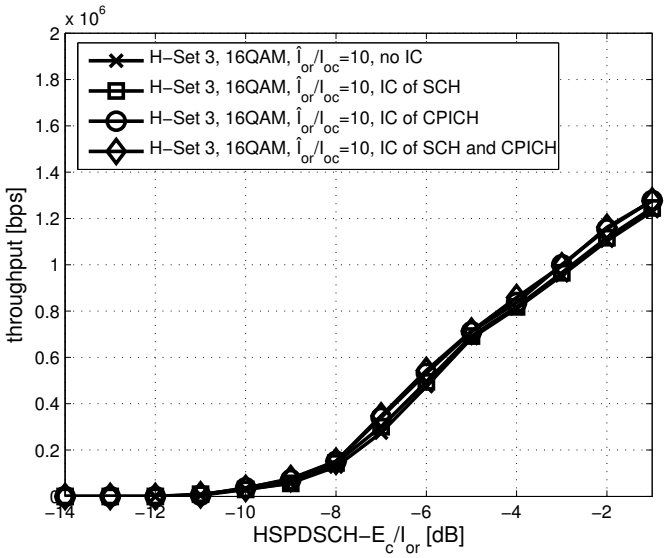

Figure 8.9: 16QAM throughput results for ITU Pedestrian-B channel for $\hat{I}_{or}/I_{oc} = 10\,\mathrm{dB}$ with several cancellation modes.

QPSK and in Fig. 8.9 for 16QAM, respectively. Throughput results for simulations with $\hat{I}_{or}/I_{oc} = 15\,\text{dB}$ are not shown here because they are very similar to the case of $\hat{I}_{or}/I_{oc} = 10\,\text{dB}$ except for a horizontal shift.

8.7.3 Discussion of Results

The rake receiver front-end of a future HSDPA receiver suffers from a significant error floor due to interference from other channels. Especially the SCH causes high interference, because it is neither spreaded nor scrambled and thus not orthogonal to the other codes.

The AWGN simulations for QPSK and 16QAM show that cancelation of the SCH increases the throughput on the HS-PDSCH. This increase is most significant at the transition from the power level, where one transmission is sufficient for successful decoding and that power level, where a retransmission is needed. For example, for the QPSK reference channel at a $\hat{I}_{or}/I_{oc} = 15\,\text{dB}$ this takes place at an E_c/I_{or} of $-16\,\text{dB}$.

The achieved throughput for 16QAM on AWGN is very close to the results for the "3.6 Mbps" channel in [80]. If IC is used then a performance gain of 1 dB is observed.

In the case of a frequency selective channel, interference also stems from multipath propagation and the fact that time-shifted versions of channelization codes are not orthogonal anymore. This interference, however, is not only caused by signalling channels like SCH and CPICH since canceling of those channels does not increase the performance significantly. More performance gain can be achieved by a receiver structure which cancels all interfering channels as proposed in [67]. It is also worth noting that in case of a frequency selective channel, the 16QAM reference channel has less throughput than the QPSK reference channel. This confirms the fact that 16QAM was not intended for usage in channels with large delay spread.

8.8 LMMSE Numerical Experiments

The setup for the LMMSE numerical experiments is the same as for the rake numerical experiments described in the previous section. The only additional parameter is the equalizer length N_E, which was set to $N_E = 48$.

Throughput simulations are carried out for the frequency-selective Rayleigh fading channels ITU Pedestrian A (PA3), B (PB3) and Vehicular A (VA30 and VA120) [73]. The UE speed is 3 km/h, 30 km/h and 120 km/h for PB3/PA3, VA30 and VA120 respectively. The parameters are summarized in Table 8.4.

Figs. 8.10 and 8.11 show the throughput for the frequency selective channels PA3

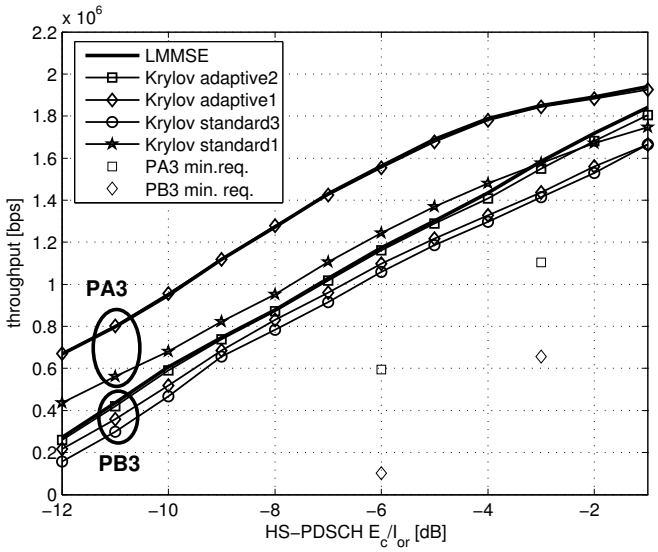

Figure 8.10: Throughput for ITU-PA3 and PB3 models with 16QAM modulation and OCNS, using Standard or Adaptive Krylov subspace method with $S \in \{1, 2, 3\}$ iterations. We show also the minimum requirements for the two models.

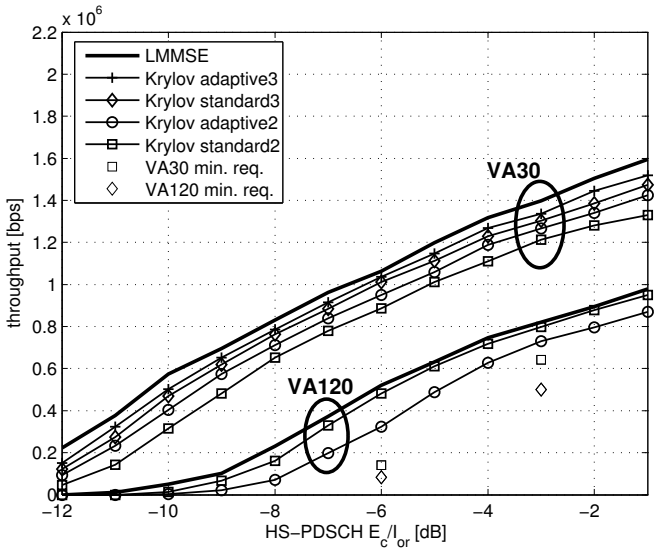

Figure 8.11: Throughput for ITU-VA30 and VA120 models with 16QAM modulation and OCNS, using Standard or Adaptive Krylov subspace method with $S \in \{1, 2, 3\}$ iterations. We show also the minimum requirements for the two models.

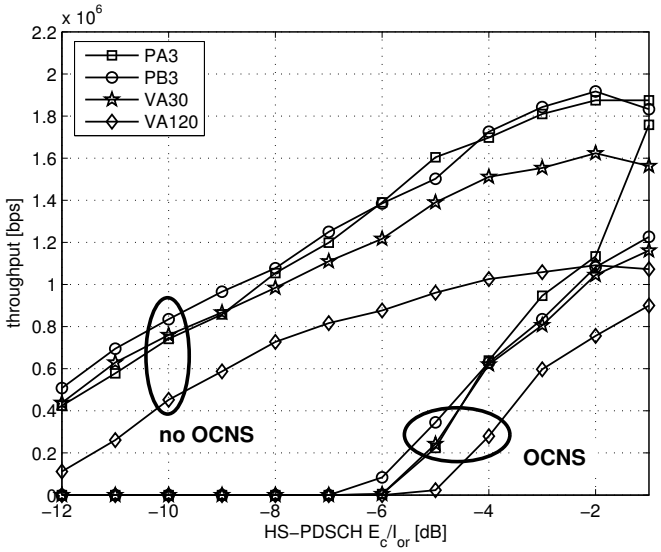

Figure 8.12: Throughput for the ITU-PA3, PB3, VA30 and VA120 models with 16-QAM modulation, using the LMS equalizer. We show both results with and without OCNS channel.

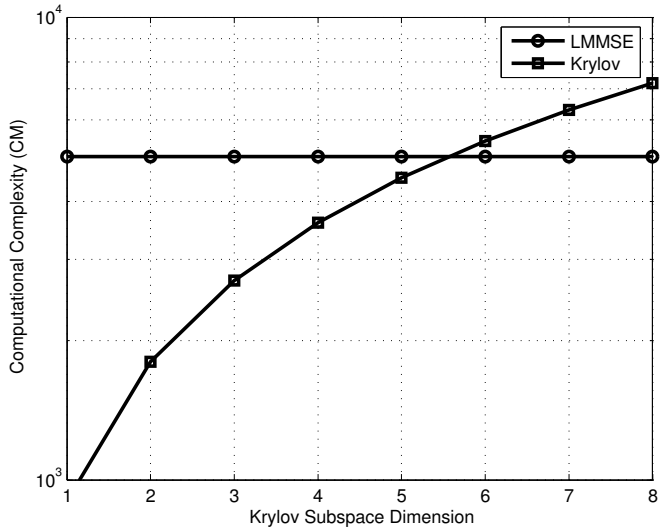

Figure 8.13: Computational Complexity for the exact LMMSE equalizer and the Krylov based equalizer, function of the Krylov subspace dimension S.

Relative Delay [ns]	**Relative Mean Power [dB]**
ITU Pedestrian A, Speed 3 km/h (PA3)	
0	0
110	-9.7
190	-19.2
410	-22.8
ITU Pedestrian B, Speed 3 km/h (PB3)	
0	0
200	-0.9
800	-4.9
1200	-8.0
2300	-7.8
3700	-23.9
ITU vehicular A, Speed 30 km/h, 120 km/h (VA30, VA120)	
0	0
310	-1.0
710	-9.0
1090	-10.0
1730	-15.0
2510	-20.0

Table 8.4: ITU channel models for UMTS [73]

and PB3 and for VA30 and VA120 respectively. The symbols in Figs. 8.10, 8.11 and 8.12 at -3 dB and -6 dB show the minimum requirements given by the UMTS standard [73]. We compare the throughput for the exact LMMSE equalizer, and the Standard and Adaptive Krylov equalizers with varying subspace dimension. For a slow varying channel (PA3 and PB3), the Adaptive equalizer converges faster to the LMMSE performance than the Standard one. While increasing the UE speed, the Standard equalizer performance gets closer to the Adaptive one (VA30) until it outperforms it (VA120). These results show that, when the channel changes slowly, the LMMSE equalizer will show small variations and thus time coherence from one subframe to another can be exploited. However, when the UE is moving fast, no such information is beneficial. For the investigated channels, a Krylov subspace dimension $S \leq 3$ is sufficient to attain the LMMSE throughput.

A comparison between the computational complexities of the exact LMMSE equalizer and the Krylov based LMMSE equalizer is shown in Fig. 8.13 for $N_E = 48$ and $N = 15$ (Vehicular A channels in our simulations). The approximate number of CM is shown versus the Krylov subspace dimension (or iteration number) S. It can be seen that the computational complexity is reduced by about a half order of magnitude.

In Fig. 8.12 we show for comparison the throughput obtained using a Least Mean

Square (LMS) equalizer as implemented in [50]. The interference caused by the OCNS degrades the performance of the LMS equalizer, which then performs a lot worse than an LMMSE equalizer.

8.9 Conclusions

In this chapter several receiver structures for UMTS HSDPA have been proposed. Throughput simulations were carried out using a specifically developed simulation environment for HSDPA in UMTS Release 5 and beyond.

Throughput simulations for the rake receiver using an AWGN channel indicate that the interference generated by the SCH is not negligible. Canceling the SCH interference at the mobile side leads to an improvement of approx. 1 dB. Therefore, it is worthwhile to implement IC for the SCH in future mobile receivers for HSDPA.

The throughput results for the rake receiver using the Pedestrian-B channel show little sensitivity to the SCH and CPICH interference. We conclude that the SCH and CPICH are not the primary source of interference in frequency selective channels when a rake receiver is used.

Therefore an LMMSE equalizer for UMTS HSDPA was introduced. The LMMSE equalizer corrects for the distortion brought about by the channel and (partly) restores orthogonality. Simulation results indicate that the LMMSE equalizer shows superior throughput than a conventional rake receiver. The gain in E_c/I_{or} is approx. 1.6 dB for QPSK and 2 dB for 16QAM.

However, this increase in throughput comes at the cost of more complexity since for the LMMSE a $N_E \times N_E$ matrix inverse has to be computed every subframe. Therefore, a low-cost approximation of an LMMSE equalizer for HSDPA using the Krylov subspace method has been introduced. This equalizer was tested with frequency selective fading channels (ITU-PA3, PB3, VA30, VA120) and turned out to perform as well as the exact LMMSE equalizer, while the computational complexity and the storage requirements of the algorithm are strongly reduced. Exploiting the temporal coherence of the channel in two consecutive subframes allows further computational complexity reduction.

9 Smart Antenna Receiver Architectures for GSM

9.1 Introduction

Despite the advances in 3rd generation mobile communication systems like UMTS, the GSM standard is still the most popular. Therefore there is a wide interest in increasing system capacity in current GSM networks.

This can be done, for example by using a small frequency reuse factor. This way the number of users in a cell and thus the system capacity can be increased. However, a tight frequency reuse introduces co-channel interference from neighboring cells. To combat co-channel interference, smart antennas and multi-user detection can be used [24, 45, 64, 89].

The major problem with such space-time algorithms is their high computational complexity. In this chapter, a smart antenna receiver algorithm for the GSM mobile standard is presented. The algorithm was originally published in [70]. The algorithm has further been optimized and implemented on the *ARC SmartSim* development platform [7].

The chapter is organized as follows. The signal model is presented in Section 9.2. In Section 9.3, the smart antenna algorithm is shortly reviewed. It is followed by the matched filter and Viterbi decoder described in Section 9.4. In Section 9.5 the most important issues of the implementation are discussed. Section 9.6 presents results from simulations with the ARC SmartSim development platform. Conclusions are drawn in Section 9.7.

9.2 Signal Model

Consider a GSM SIMO system in the uplink with one transmit antenna and N_{Rx} receive antennas. In such a scenario, we can exploit the array gain as well as the diversity gain at the receiver to achieve a higher signal to noise ratio (SNR).

GSM uses Gaussian minimum shift keying (GMSK) to modulate the data. GMSK is a continuous phase modulation (CPM) with Gaussian pulses. This modulation introduces intersymbol interference (ISI) of several symbols. According to [22, 66], the

sampled baseband representation of a GMSK modulated signal at the r-th receive antenna can be approximated by a linear representation

$$x'_{m,r} = \sum_{n=0}^{N-1} H'_{n,r} s'_{m-n} + u'_{m,r}, \tag{9.1}$$

where $s'_m = j^m s_m$ and $s_m \in \{-1, 1\}$ are the bipolar bits to be transmitted, $H'_{n,r}$ is the effective sampled time-invariant impulse response of the r-th channel of length N, and $u'_{m,r}$ is the noise-plus-interference process of the r-th channel. $H'_{n,r}$ includes the linear approximation of the GMSK signal and the distortion of the channel.

The N_{Rx} complex valued equations in (9.1) can be transformed into $2N_{\mathrm{Rx}}$ real-valued equations by performing a "derotation", i. e., a multiplication by j^{-m}:

$$x_{m,r} = \sum_{n=0}^{N-1} H_{n,r} s_{m-n} + u_{m,r}, \tag{9.2}$$

where we have defined for $r = 0, \ldots, N_{\mathrm{Rx}} - 1$

$$\begin{aligned} x_{m,2r} &= \Re\{j^{-m} x'_{m,r}\} & x_{m,2r+1} &= \Im\{j^{-m} x'_{m,r}\}, \\ H_{n,2r} &= \Re\{j^{-n} H'_{n,r}\} & H_{n,2r+1} &= \Im\{j^{-n} H'_{n,r}\}, \\ u_{m,2r} &= \Re\{j^{-m} u'_{m,r}\} & u_{m,2r+1} &= \Im\{j^{-m} u'_{m,r}\}. \end{aligned}$$

Taking samples $m = N-1, \ldots, M-1$, where M is the blocklength, (9.2) can be written in matrix notation

$$\mathbf{X} = \mathbf{S}\mathbf{H} + \mathbf{U}, \tag{9.3}$$

where the elements of the matrices $\mathbf{X}$, $\mathbf{H}$, and $\mathbf{U}$ are given by

$$\begin{aligned} \mathrm{X}_{m,r} &= x_{m,r}, & m &= N-1, \ldots, M-1, \\ \mathrm{H}_{n,r} &= H_{n,r}, & n &= 0, \ldots, N-1, \\ \mathrm{U}_{m,r} &= u_{m,r}, & r &= 0, \ldots, 2N_{\mathrm{Rx}} - 1, \end{aligned}$$

and

$$\mathbf{S} = \begin{pmatrix} s_{N-1} & s_{N-2} & \cdots & s_0 \\ s_N & s_{N-1} & \cdots & s_1 \\ \vdots & \vdots & \ddots & \vdots \\ s_{2N-2} & s_{2N-3} & \cdots & s_{N-1} \\ \vdots & \vdots & \vdots & \vdots \\ s_{M-1} & s_{M-2} & \cdots & s_{M-N} \end{pmatrix} \in \mathbb{C}^{M-N+1 \times N}.$$

After reception, the $2N_{\mathrm{Rx}}$ signals are combined using the complex weight vector $\boldsymbol{\omega} \in \mathbb{C}^{2N_{\mathrm{Rx}}}$

$$\boldsymbol{r} = \mathbf{X}\boldsymbol{\omega} = \mathbf{S}\boldsymbol{h} + \boldsymbol{u}, \tag{9.4}$$

where $\boldsymbol{h} = \mathbf{H}\boldsymbol{\omega}$ is the impulse response after the combiner and $\boldsymbol{u} = \mathbf{U}\boldsymbol{\omega}$.

9.3 The Matched Desired Impulse Response Algorithm

The MDIR (Matched Desired Impulse Response) algorithm [64] consists of a block for joint channel and weight estimation, a narrowband combiner and a Viterbi decoder (see Fig. 9.1). Although suboptimal, the MDIR with narrowband combiner results in a good compromise between reduction in complexity and performance. The algorithm uses the midamble part of one GSM burst and the minimum mean squared error (MMSE) criterion to jointly optimize the spatial weighting coefficients $\boldsymbol{w}$ together with the desired impulse response $\boldsymbol{h}$ of the equivalent channel after combining.

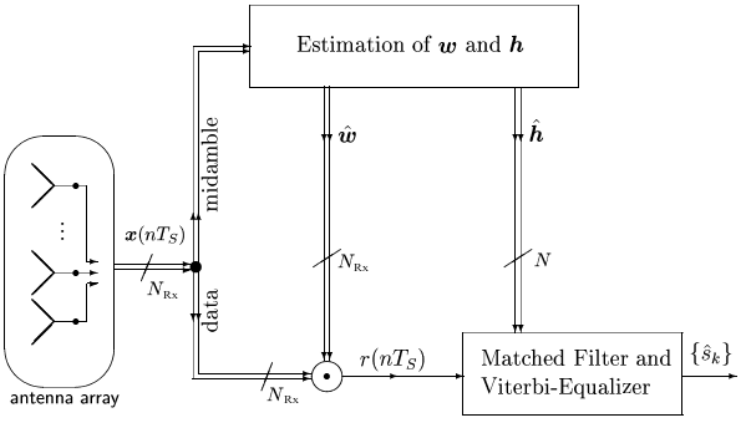

Figure 9.1: MDIR receiver with narrowband combining for N_{Rx} antennas. The derotation at the receiver frontend is not shown. The MDIR architecture operates on $2N_{\mathrm{Rx}}$ equivalent real-valued input channels [70].

From equation (9.4) it can be seen that the cost function that needs to be minimized can be written as

$$\varepsilon^2 = \varepsilon^2(\boldsymbol{w}, \boldsymbol{h}) = \|\mathbf{X}\boldsymbol{w} - \mathbf{D}\boldsymbol{h}\|^2, \tag{9.5}$$

where, $\mathbf{X}$ is the derotated, real-valued observation matrix of the midamble portion of the received data, and $\mathbf{D}$ is the Toeplitz matrix containing the bipolar $\{-1, 1\}$ symbols of the GSM training sequence code,

$$\mathbf{D} = \begin{pmatrix} d_{N-1} & d_{N-2} & \cdots & d_0 \\ d_N & d_{N-1} & \cdots & d_1 \\ \vdots & \vdots & \ddots & \vdots \\ d_{2N-2} & d_{2N-3} & \cdots & d_{N-1} \\ \vdots & \vdots & \vdots & \vdots \\ d_{M-1} & d_{M-2} & \cdots & d_{M-N} \end{pmatrix}$$

Note that the optimization of (9.5) over all $\boldsymbol{w}, \boldsymbol{h}$ is aimed at keeping the multipath content of the signal low so it can be used by the Viterbi decoder. A constraint has to be imposed in order to avoid the trivial solution. It is chosen to constrain the desired energy at the output of the spatial combiner [64]:

$$E = \|\mathbf{D}\hat{\mathbf{G}}\boldsymbol{w}\|^2 = \boldsymbol{w}^T\hat{\mathbf{G}}^T\mathbf{D}^T\mathbf{D}\hat{\mathbf{G}}\boldsymbol{w}, \tag{9.6}$$

where $\hat{\mathbf{G}}$ contains the least-squares estimates of the $2N_{\mathrm{Rx}}$ parallel diversity channel impulse responses

$$\hat{\mathbf{G}} = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{X}. \tag{9.7}$$

Introducing a Lagrangian multiplier λ, we arrive at the following cost function of the unconstrained optimization problem

$$f(\boldsymbol{w}, \boldsymbol{h}, \lambda) = \varepsilon^2(\boldsymbol{w}, \boldsymbol{h}) - \lambda(\boldsymbol{w}^T\hat{\mathbf{G}}^T\mathbf{D}^T\mathbf{D}\hat{\mathbf{G}}\boldsymbol{w} - E). \tag{9.8}$$

The solution for $\boldsymbol{h}$ is given by

$$\boldsymbol{h} = \hat{\mathbf{G}}\boldsymbol{w}. \tag{9.9}$$

Assuming that the training sequence and the noise-plus-interference are uncorrelated processes, the solution for $\boldsymbol{\omega}$ is given by the generalized eigenvector associated to the minimum generalized eigenvalue of [70]

$$\hat{\mathbf{R}}\boldsymbol{w} = \lambda\hat{\mathbf{G}}^T\mathbf{D}^T\mathbf{D}\hat{\mathbf{G}}\boldsymbol{w}, \tag{9.10}$$

where

$$\hat{\mathbf{R}} = (\mathbf{X} - \mathbf{D}\hat{\mathbf{G}})^H(\mathbf{X} - \mathbf{D}\hat{\mathbf{G}}) \approx \mathbf{X}^T\mathbf{X} - \hat{\mathbf{G}}^T\mathbf{D}^T\mathbf{D}\hat{\mathbf{G}} \tag{9.11}$$

is the noise-plus-interference covariance matrix.

The output of the combiner is then feed to a matched filter and a scalar Viterbi decoder [22], which eliminates the ISI and detects the data.

9.4 Matched Filter and Viterbi Decoder

9.4.1 Matched Filter

After having calculated the weight vector $\boldsymbol{\omega}$ and the impulse response of the combined signal $\boldsymbol{h}$, the received signal $\boldsymbol{x}_m = [x_{m,0}, \ldots, x_{m,N_{\mathrm{Rx}}-1}]^T$ writes

$$r_m = \boldsymbol{\omega}^H \boldsymbol{x}_m = \sum_{n=0}^{N-1} h_n s_{m-n} + v_m. \tag{9.12}$$

A whitening matched filter is applied to increase the SNR and to reduce the number of states in the Viterbi decoder [22]. The whitening filter coefficients are given by h^*_{-n}. The resulting signal writes

$$r'_m = \sum_{n=0}^{L-1} \eta_n s_{m-n} + v'_m, \tag{9.13}$$

where $\eta_m = \sum_n h_m h^*_{n-m}$ is the overall impulse response of length L seen by the Viterbi decoder.

9.4.2 Viterbi Decoder

The goal of the decoder is to estimate the transmitted symbols s_m from the received sequence r'_m. This can be formulated as a maximum likelihood estimation problem. Let $\boldsymbol{s}$ and $\boldsymbol{r}'$ denote the vector of input and output symbols respectively. Further denote by $P(\boldsymbol{s}|\boldsymbol{r}')$ the probability that $\boldsymbol{s}$ was sent when $\boldsymbol{r}'$ was received. The the task of the maximum likelihood estimator is to find

$$\hat{\boldsymbol{s}} = \operatorname{argmax} P(\boldsymbol{s}|\boldsymbol{r}'). \tag{9.14}$$

The problem (9.14) can be solved using the Viterbi algorithm [48,112,113]. Every channel of the form (9.13) of length L can be interpreted as a discrete-time finite-state Markov process with 2^L states. A Markov process can be represented as a state diagram, where the nodes are the states and the edges represent transitions from one state to another. In the case of our channel (9.13), the states are all possible sequences of symbols s_k of length L, $S = (s_L, s_{L-1}, \ldots, s_0)$.

The state diagram is now written in the form of a trellis diagram, where each node corresponds to a state at a specific time and each branch represents a transition to some new state at the next time instance (see Fig. 9.2). Every possible sequence of states corresponds to a unique path in the trellis.

Let $S(m)$ be any state at the m-th successive node level. We shall remove the subscripts until necessary. To construct the Viterbi algorithm, we first use the fact

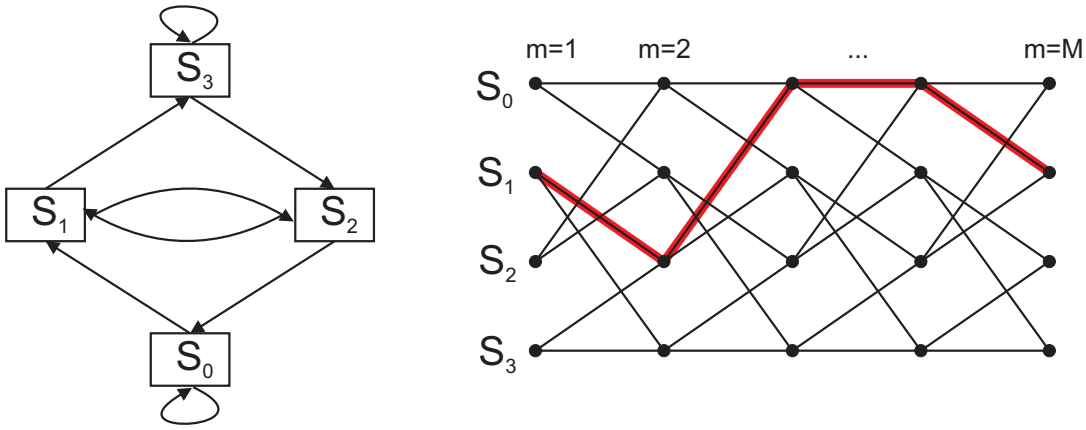

Figure 9.2: State diagram of a four state Markov process and corresponding trellis diagram.

that the transitions between states $S(m-1) \rightarrow S(m)$ in the trellis are mutually independent. For any given path from the origin $m = 1$ to an arbitrary node $(m = M)$ and the states $S(0), S(1), \ldots, S(M)$, we define the path metric by

$$\Lambda = \sum_{m=1}^{M} |r'_m - y(S(m))|^2. \tag{9.15}$$

where

$$y(S(m)) = \sum_{n=0}^{L-1} \eta_n s_{m-n}. \tag{9.16}$$

In order to maximize the likelihood function $P(\boldsymbol{s}|\boldsymbol{r}')$ (9.14), (9.15) needs to be minimized. To do so it is sufficient to minimize the sum over the first $M-1$ terms for each state $S(M-1)$ at the $(M-1)$th node level and then minimize the sum of this and the M-th term over all states $S(M)$. The Viterbi algorithm can thus be summarized as follows.

Algorithm 9.4.1 (Viterbi Algorithm).

for $m = 1$ *to* M ***do***

for *all states* $S_i(m)$ ***do***

Calculate the branch metrics

$$\Lambda(S_j(m-1), S_i(m)) = |r'_m - y(S_i(m))|^2$$

for all states $S_j(m-1)$ *leading to* $S_i(m)$

Calculate the state metrics

$$\Sigma(S_i(m)) = \min_{S_j(k-1)} \{\Sigma(S_i(m-1)) + \Lambda(S_j(m-1), S_i(m))\}$$

and store the path $S_j(m-1) \rightarrow S_i(m)$ *that achieves this minimum as the* survivor path

Due to the head and tail bit of a GSM burst, the initial and the final state are predefined. Therefore, at the last node level M only one branch trough the trellis will survive. This path then gives the most likely transmitted sent sequence.

9.5 Real-Time Implementation Aspects

The MDIR receiver with narrowband combiner, matched filter and Viterbi decoder (see Fig. 9.1) has been fully implemented on the DSP board of the ARC SmartSim development platform [7]. This section describes some of the implementation details that need to be taken care of on a fixed-point hardware.

The most crucial point of the MDIR algorithm is solving the eigenvalue problem (9.10). Since the calculation of $\hat{\mathbf{R}}$ can lead to numerical problems (its entries can be very small), (9.10) was slightly modified to

$$\mathbf{X}^H\mathbf{X}\boldsymbol{w} = \tilde{\lambda}\hat{\mathbf{G}}^T\mathbf{D}^T\mathbf{D}\hat{\mathbf{G}}\boldsymbol{w}. \tag{9.17}$$

This modification leaves the eigenvectors and the order of the eigenvalues unchanged.

For a given midamble, the pseudoinverse $(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}$ of the midamble matrix $\mathbf{D}$ needed for the calculation of $\hat{\mathbf{G}}$ can be precalculated. Care needs to be taken that no overflow occurs in the calculation of the matrices in (9.17). It was found out that a scaling of the matrix $\mathbf{X}$ by

$$\alpha \approx \frac{1}{\|\mathbf{X}\|_F} \tag{9.18}$$

rounded to the next power of two, avoids overflows.

The matrix $\mathbf{X}^H\mathbf{X}$ is positive definite and symmetric, hence all eigenvalues are positive and real. Since only the eigenvector with the minimum eigenvalue is needed, i. e., the eigenvalue closest to zero, the inverse power method [38] with shift zero is the most attractive method to solve the generalized eigenvalue problem (9.17). The algorithm can be summarized as follows

Algorithm 9.5.1. *Given a positive definite matrix* $\mathbf{A}$*, a general matrix* $\mathbf{B}$*, and an initial vector* $\boldsymbol{v}^{(0)}$*, the following algorithm computes the generalized eigenvector associated with the minimum generalized eigenvalue of* $\lambda\mathbf{A}\boldsymbol{x} = \mathbf{B}\boldsymbol{x}$*.*

for $k = 0, 1, \ldots$ ***do***
  Solve $\mathbf{A}\boldsymbol{w} = \mathbf{B}\boldsymbol{v}^{(k-1)}$ *for* $\boldsymbol{w}$
  $\boldsymbol{v}^{(k)} = \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$

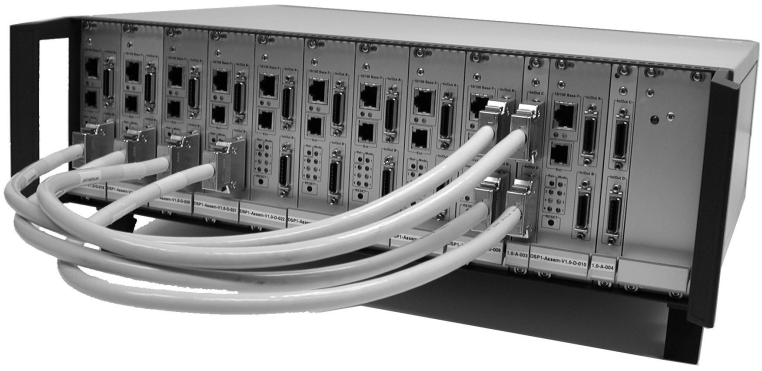

Figure 9.3: The baseband development platform in practice

For the solution of linear systems of equations in Algorithm 9.5.1 a Schur-Cholesky factorization of the matrix $\mathbf{A}$ can be used.

$$\mathbf{A} = \mathbf{L}\mathbf{L}^H, \tag{9.19}$$

where $\mathbf{L}$ is a nonsingular lower tridiagonal matrix with positive diagonal entries. Then step one of Algorithm 9.5.1 can be solved efficiently using backsubstitution.

A good trade-off between precision and complexity was achieved by carrying out the Cholesky factorization in 32-bit precision, while doing the iteration of the inverse power method in 16-bit precision. It was found out that the algorithm converges sufficiently after three iterations.

For the Viterbi decoder described in Section 9.4 an implementation using $L = 2^4 = 16$ states was used [107].

9.6 Simulation and Results

The performance of the smart antenna receiver was evaluated using the real-time ARC SmartSim development platform in the digital baseband domain. The parameters of the COST 259 GSCM (cf. Section 5.3) are chosen according to the macrocellular general typical urban (GTU) environment [34]. The simulations were carried out with a uniform linear antenna array with 4 antennas and a spacing of $\lambda/2$ which corresponds to aprrox. 0.15 m at a carrier frequency of 1.845 GHz. The simulation parameters are summarized in Table 9.1 and the simulation setup is shown in Fig. 9.3. The four leftmost boards are used for the simulation of the SIMO channel and the board second to right is used as a receiver. The other boards are idle. Clearly visible are the four "ChannelLink" LVDS baseband cables.

We have carried out three different simulations which are presented in the following subsections. The first one compares the receiver performance measured in

Parameter	Value
Channel Model	COST 259 GTU
Cell size	1000m
No. of Base Stations	1
No. of BS antennas	1
No. of mobile stations	1
No. of MS antennas	4
MS antenna pattern	Linear array omnidirectional antennas $\lambda/2$ spacing
Carrier frequency	1.845 GHz
Symbol Rate	275 kHz

Table 9.1: GSM simulation parameters

uncoded BER to the path loss of the received signal. The second and the third compare the receiver performance when varying the number of antennas and the signal to noise ratio.

9.6.1 BER vs. Pathloss

In this simulation the attenuation of the received signal was compared to the BER of the receiver. Therefore the MS was set to move at 1 m/s on the x-axis of the cell starting at a distance of 200 m going to 1000 m. The BS power was set to 40 dBm and the noise level was set at a constant level of -100 dBm. In the lower part of Fig. 9.4 the received power is displayed, whereas in the top part of Fig. 9.4 the BER of the receiver using the MDIR algorithm with 2 antennas is displayed. It can be verified that when the receive power falls below the -100 dBm level, the BER of the receiver goes up. In cases with enough receive power left, the algorithm performs well.

9.6.2 Comparison of BER for Different SNRs

In this simulation run the MS was set to circle around the BS at distance of 500 m with a speed of 17.45 m/s, so that it needs exactly 3 min for one round. The simulation was carried out with various signal to noise ratios.

In Fig. 9.5 the BER of the smart antenna receiver using the MDIR algorithm with two antennas is plotted over the time. The different lines correspond to simulation runs with different SNRs. It can be seen that the SNR affects the receiver performance as expected.

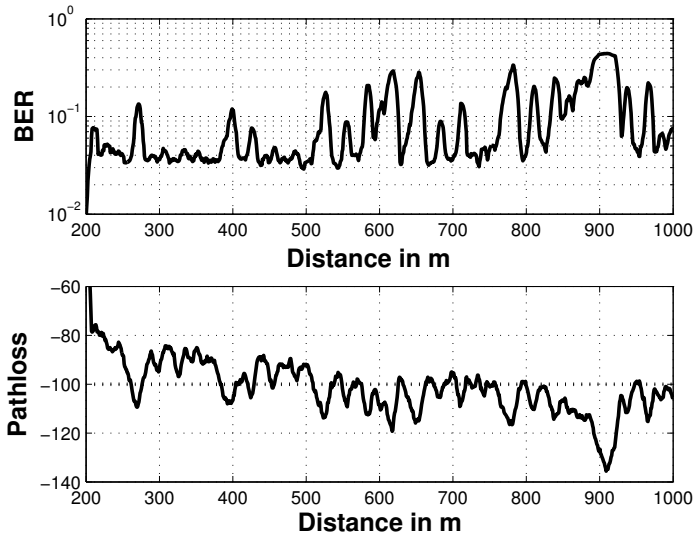

Figure 9.4: Top: BER of the receiver using the MDIR algorithm with 2 antennas; Bottom: Received power at the MS and noise level (red dotted line).

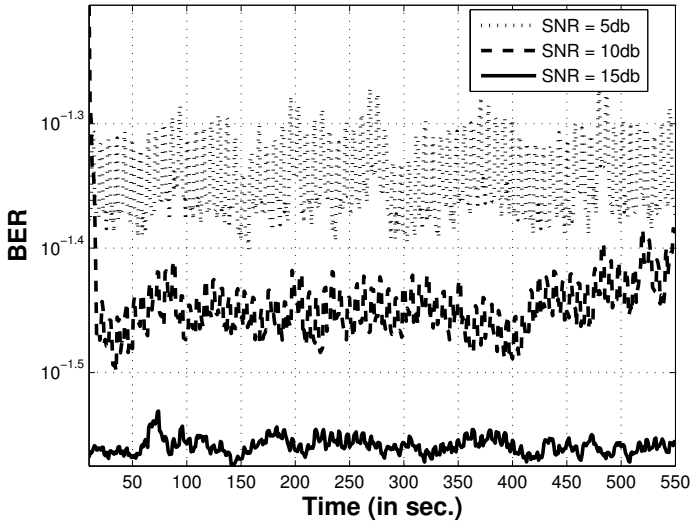

Figure 9.5: Comparison of BER for the MDIR algorithm for different SNRs.

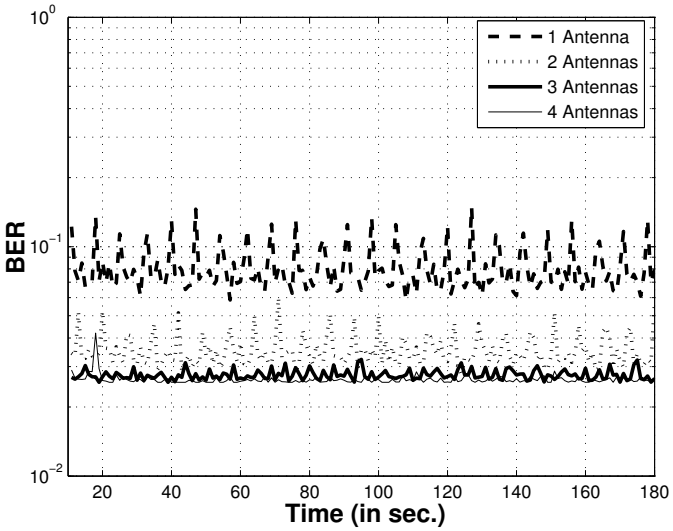

Figure 9.6: Comparison of BER for the MDIR algorithm for different number of antennas.

9.6.3 Comparison of BER for Different No. of Antennas

In the last simulation the receiver performance was investigated using different number of antennas. The simulation parameters were the same as above, except that the SNR was fixed to 15 dB. It can be seen in Fig. 9.6 that the second antenna increases the performance by a factor of approx. 6, the third antenna by a factor of 7 and the fourth one by a factor of almost 8. It can be seen that the variance of the BER decreases with an increasing number of antennas.

9.7 Conclusions

A smart antenna receiver for the GSM mobile standard was presented and implemented on the receiver unit of the ARC SmartSim development platform. The calculation of the receiver weights requires the solution of a generalized eigenvalue problem. A numerically stable, iterative algorithm was presented, which is able to solve the problem with sufficient accuracy in only three iterations.

Real-time simulations were carried out using the ARC SmartSim Channel Emulator in the digital baseband domain. The results show that (*i*) the channel emulator is working correctly in real-time and provides realistic results, and that (*ii*) using smart antennas in a GSM receiver provides superior performance to the single antenna case.

10 Summary and Conclusions

The second part of the thesis presented different receiver architectures for the state-of-the-art wireless communication systems UMTS-HSDPA and GSM. Special emphasis was put on low-complexity receiver design and robustness of the algorithms against interference.

Interference limits the throughput of the High Speed Downlink Packet Access (HSDPA) sub-system of UMTS. It was shown that a linear minimum mean square error (LMMSE) equalizer at the receiver achieves higher throughput than a conventional rake receiver, at the cost of higher complexity. Therefore, an iterative algorithm based on Krylov subspace projections was introduced that approximates the LMMSE equalizer with negligible loss of performance. Computational complexity as well as storage requirements are strongly reduced.

The capacity of wireless communication systems is also limited by the channel and the interference created by other users. For GSM, a smart antenna receiver was presented that enables to cancel interference and maximize the SNR at the receiver by using a joint space-time optimization. The receiver was implemented on the receiver unit of the ARC SmartSim development platform. A numerically stable, iterative algorithm was presented that solves the generalized eigenvalue problem with sufficient accuracy in only three iterations.

The development of the receiver architectures for UMTS and GSM exemplify two different design stages in the wireless communication system design. While the UMTS receiver was completely implemented and evaluated in MATLAB, the GSM receiver was implemented on a DSP board and evaluated in real time using a channel simulator. The first approach has the advantage that ideas can quickly be implemented in a high level language using a broad range of toolboxes.

However, many effects cannot be taken into account in the software simulation, which occur on the final hardware, such as fixed-point issues and real-time constraints. The development of the GSM receiver showed that such issues are not trivial to deal with and should be taken into consideration as early as possible in the design stage. On the other hand, debugging and verification of hardware implementations can be quite cumbersome.

Rapid prototyping tries to combine these two approaches and to exploit the benefits of each of them. One key element in rapid prototyping is to have only one code, also called "golden code" [96]. This code can either be mapped onto the simulation

environment like MATLAB or SIMULINK, or onto the prototype hardware.

This approach has, for example, been taken in [31] and [12]. There the "Golden Code" is written in GENERICC, which is C language enriched with predefined keywords for interfacing. A set of tools maps GENERICC algorithms into either SIMULINK S-functions for simulation purposes, into code to run on a TI C64 processor [12] or into C code that can be mapped via automatic tools into VHDL and subsequently into FPGAs [31]. This approach assists rapid prototyping and code co-verification massively. Another advantage is that wordlength effects can be investigated easily by only changing the types of the variables used in the GENERICC files.

A List of Abbreviations

Abbreviation	Description
ACF	Autocorrelation Function
ADPS	Angular Delay Power Spectrum
AMC	Adaptive Modulation and Coding
AoA	Azimuth of Arrival
AoD	Azimuth of Departure
ARC	Austrian Research Centers
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BS	Base Station
CDMA	Code Division Multiple Access
CE	Complex Exponential
CM	Complex Multiplication
COST	European Co-operation in the Field of Scientific and Technical Research
CPM	Continuous Phase Modulation
CQI	Channel Quality Indicator
CRC	Cyclic Redundancy Check
dB	deci-Bel
DFT	Discrete Fourier Transform
DPS	Discrete Prolate Spheroidal
DSP	Digital Signal Processing
DTFT	Discrete Time Fourier Transform
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FPGA	Field Programmable Gate Array
GMSK	Gaussian Minimum Shift Keying
GCM	Geometry-based Channel Model

GSCM	Geometry-based Stochastic Channel Model
GSM	Global System for Mobile Communications
GTU	General Typical Urban
HARQ	Hybrid Automatic Repeat Request
HSDPA	High Speed Downlink Packet Access
IC	Interference Cancelation
IF	Intermediate Frequency
IIR	Infinite Impulse Response
IR	Incremental Redundancy
ISI	Intersymbol Interference
ITU	International Telecommunications Union
LLR	Log-likelihood Ratio
LMMSE	Linear Minimum Mean Square Error
LMS	Least Mean Square
LOS	Line of Sight
LVDS	Low Voltage Differential Signalling
MA	Memory Access
MAI	Multiple Access Interference
MCS	Modulation and Coding Scheme
MDIR	Matched Desired Impulse Response
MIMO	Multiple-Input Multiple-Output
MIPS	Mega Instructions per Second
MISO	Multiple-Input Single-Output
MMSE	Minimum Mean Square Error
MPC	Multipath component
MRC	Maximum Ratio Combining
MS	Mobile Station
MSPS	Mega Samples per Second
OCNS	Orthogonal Code Noise Simulator
OVSF	Orthogonal Variable Spreading Factor
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RB	Ringbuffer
RF	Radio Frequency
RRC	Root-Raised Cosine
RV	Redundancy Version
Rx	Receiver
SF	Spreading Factor

SIMO	Single-Input Multiple-Output
SISO	Single-Input Single-Output
SNR	Signal-to-Noise Ratio
SoCE	Sum of Complex Exponentials
SoS	Sum of Sinusoids
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TSC	Training Sequence Code
TTI	Transmission Time Interval
Tx	Transmitter
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
US	Uncorrelated Scattering
UWB	Ultra Wide Band
VHDL	VHSIC Hardware Description Language
VHSIC	Very-High-Speed Integrated Circuit
WCDMA	Wideband Code Division Multiple Access
WLAN	Wireless Local Area Network
WSS	Wide Sense Stationary

B Notation

Symbol	Description
$\mathbb{Z}, \mathbb{R}, \mathbb{C}$	set of integers, real and complex numbers
$f(t)$	function of a continuous variable $f : \mathbb{R} \to \mathbb{R}$
f_m	function of a discrete variable/sequence $f : \mathbb{Z} \to \mathbb{R}$
$\boldsymbol{f}_m$	vector-valued function of a discrete variable/sequence $f : \mathbb{Z} \to \mathbb{R}^M$
$f_{\boldsymbol{m}}$	N-dimensional sequence $f : \mathbb{Z}^N \to \mathbb{R}$
$L^p(X)$	$\{f : X \to \mathbb{R}, \lvert f\rvert^p$ Lebesque-integrable$\}$
$\boldsymbol{a}$	column vector
a_i	ith element of $\boldsymbol{a}$
$\mathbf{A}$	matrix
$\mathrm{A}_{k,l}$	k, lth element of $\mathbf{A}$
$\mathbf{A}_{P\times Q}$	upper left part of $\mathbf{A}$ with dimension $P \times Q$
$\mathbf{A}^T$	transpose of $\mathbf{A}$
$\mathbf{A}^H$	conjugate transpose of $\mathbf{A}$
$\mathrm{diag}(\boldsymbol{a})$	diagonal matrix with entries a_i
$\mathbf{I}_Q$	$Q \times Q$ identity matrix
$\mathbf{F}_Q$	$Q \times Q$ unitary Fourier matrix
$\mathbf{1}_Q$	$Q \times 1$ column vector with all ones
$\mathbf{0}_Q$	$Q \times 1$ column vector with all zeros
a^*	complex conjugate of a
$\lfloor a \rfloor$	largest integer, lower or equal than $a \in \mathbb{R}$
$\lceil a \rceil$	smallest integer, greater or equal than $a \in \mathbb{R}$
$\lvert a\rvert$	absolute value of a
$\Re a, \Im a$	real imaginary part of a complex variable a
$\langle \boldsymbol{a}, \boldsymbol{b} \rangle$	Scalar product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$
$\lVert \boldsymbol{a} \rVert$	ℓ^2 norm of vector $\boldsymbol{a}$
$\lVert \boldsymbol{a} \rVert_{\mathbf{A}}$	$\mathbf{A}$ norm of vector $\boldsymbol{a}$ $(= \sqrt{\boldsymbol{a}^H \mathbf{A} \boldsymbol{a}})$
$\lVert \mathbf{A} \rVert_{\mathrm{F}}$	Frobenius norm of matrix $\mathbf{A}$
$\mathrm{vec}(\mathbf{A})$	stacks all columns of matrix $\mathbf{A}$ in a single vector
j	$\sqrt{-1}$
δ_{kl}	1 for $k = l$, 0 otherwise

$\|X\|$	number of elements in X, if X is a discrete index set area of X, if X is a continuous region
$\times$	Cartesian product
$\otimes$	Kronecker product
$\diamond$	Khatri-Rao product (columnwise Kronecker product)
$\odot$	Hadamard product (element-wise product)
$\star$	matrix convolution
$*$	convolution
$\mathcal{E}\{X\}$	expectation of a random variable X

C List of Symbols

Symbol	Definition
$t, f, \boldsymbol{x}, \boldsymbol{y}$	Time, frequency, antenna location at transmitter, and antenna location at receiver
$h(t, f, \boldsymbol{x}, \boldsymbol{y})$	Time- and space-variant channel transfer function
$\omega, \tau, \varphi, \psi$	Doppler frequency, delay time, azimuth of departure, azimuth of arrival
$H(\omega, \tau, \varphi, \psi)$	Doppler-delay double-directional channel spreading function
$G_{\mathrm{Tx}}(\varphi), G_{\mathrm{Rx}}(\psi)$	Complex far-field antenna pattern at the Tx/Rx
$r_{\mathrm{Tx}}(\tau), r_{\mathrm{Rx}}(\tau)$	Impulse response of the transmit/receive filter
$R_h(\Delta t, \Delta f, \Delta \boldsymbol{x}, \Delta \boldsymbol{y})$	Autocorrelation function of a WSS channel
$S_h(\omega, \tau, \varphi, \psi)$	Power spectrum of a WSS channel
T_S, F_S, D_S	Duration of a sample, width of a frequency bin, and spacing between antennas
m, n, q, s, r	Discrete time index, delay time index, frequency index, antenna index at transmitter, antenna index at receiver
$h_{m,q,r,s}$	Sampled channel transfer function
$H_{m,n,r,s}$	Sampled channel impulse response
M, N, Q	Number of samples in time, delay time, and frequency
$N_{\mathrm{Tx}}, N_{\mathrm{Rx}}$	Number of transmit antennas and receive antennas
N_C	Delay spread of the channel in samples
N_F	Group delay of the Tx and Rx filters in samples
N_E	Number of equalizer coefficients
$\mathbf{R}$	Correlation matrix
$\boldsymbol{h}$	Vector of index-limited transfer function
P	Number of MPCs
β_p	Real path weight of the p-th MPC
ϕ_p	Phase rotation of the p-th MPC
η_p	Complex path weight of the p-th MPC
ω_p, ν_p	Doppler shift and normalized Doppler shift of the p-th MPC

$\omega_{\mathrm{Dmax}}, \nu_{\mathrm{Dmax}}$	Maximum Doppler shift, maximum normalized Doppler shift
τ_p, θ_p	Delay and normalized delay of the p-th MPC
$\tau_{\max}, \theta_{\max}$	Maximum delay, maximum normalized delay
φ_p, ζ_p	AoD and normalized AoD of the p-th MPC
$\varphi_{\max}, \varphi_{\min}$	Maximum and minimum AoD
$\zeta_{\max}, \zeta_{\min}$	Maximum and minimum normalized AoD
ψ_p, ξ_p	AoA and normalized AoA of the p-th MPC
$\psi_{\max}, \psi_{\min}$	Maximum and minimum AoD
$\xi_{\max}, \xi_{\min}$	Maximum and minimum normalized AoD
ω_C, c	Carrier frequency, speed of light
$v_{\max}$	Maximum velocity of user
W	Band-limiting region
I	Index set
$v_m^{(d)}(W, I)$	d-th one-dimensional DPS sequence
$v_{\boldsymbol{m}}^{(d)}(W, I)$	d-th multidimensional DPS sequence
$\boldsymbol{v}(d)(W, I)$	One-dimensional or multidimensional DPS vector
$\lambda_d(W, I)$	Eigenvalue of d-th DPS sequence
D, D'	Subspace dimension and essential subspace dimension
$\alpha_d, \tilde{\alpha}_d$	d-th basis coefficient and approximate basis coefficient of DPS subspace representation of $\boldsymbol{h}$
$\gamma_{p,d}, \tilde{\gamma}_{p,d}$	d-th basis coefficient and approximate basis coefficient of DPS subspace representation of the p-th MPC
$U_d(\nu), \tilde{U}_d(\nu)$	DPS wave function and approximate DPS wave function
$E_{\max}^2$	Maximum squared accuracy of hardware

Bibliography

Own Publications

[1] C. Dumard, F. Kaltenberger, and K. Freudenthaler, "Low-cost LMMSE equalizer based on Krylov subspace methods for HSDPA," *IEEE Transactions on Wireless Communications*, 2007, to be published.

[2] K. Freudenthaler, F. Kaltenberger, S. Geirhofer, S. Paul, J. Berkman, J. Wehinger, C. Mecklenbräuker, and A. Springer, "Throughput analysis for a UMTS high speed downlink packet access LMMSE equalizer," in *Proc. IST Mobile and Wireless Communications Summit*, Dresden, Germany, June 2005.

[3] K. Freudenthaler, F. Kaltenberger, S. Paul, C. Mecklenbräuker, M. Huemer, and A. Springer, "Cancellation of interference from synchronization and pilot channels on high speed downlink shared channel in UMTS," in *Proc. European Wireless Conference (EWC)*, Nicosia, Cyprus, June 2005, pp. 498–503.

[4] F. Kaltenberger, K. Freudenthaler, S. Paul, J. Wehinger, C. Mecklenbräuker, and A. Springer, "Throughput enhancment by cancellation of synchronization and pilot channel for UMTS high speed downlink packet access," in *Proc. 6th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, New York, USA, June 2005, pp. 580–584.

[5] F. Kaltenberger, G. Humer, and G. Pfeiffer, "MIMO/smart antenna development platform," in *Workshop on Software Radios (WSR04)*, Karlsruhe, Germany, March 2004, pp. 117–121.

[6] F. Kaltenberger, G. Steinböck, G. Humer, and T. Zemen, "Low-complexity geometry-based MIMO channel emulation," in *Proc. European Conference on Antennas and Propagation (EuCAP 2006)*, Nice, France, November 2006, invited.

[7] F. Kaltenberger, G. Steinböck, R. Kloibhofer, R. Lieger, and G. Humer, "A multi-band development platform for rapid prototyping of MIMO systems," in *Proc. ITG Workshop on Smart Antennas*, Duisburg, Germany, April 2005.

[8] F. Kaltenberger, T. Zemen, and C. W. Ueberhuber, "Low complexity doubly selective channel simulation using multidimensional discrete prolate spheroidal sequences," in *Proc. IST Mobile and Wireless Communications Summit*, Myconos, Greece, June 2006.

[9] ——, "Low complexity simulation of wireless channels using discrete prolate spheroidal sequences," in *Proc. MATHMOD Conference*, Vienna, Austria, February 2006.

[10] ——, "Low-complexity geometry-based MIMO channel simulation," *EURASIP Journal on Advances in Signal Processing*, 2007.

[11] F. Kaltenberger and S. Yang, "MIMO-algorithms for wireless LAN," Wireless Congress, Munich, Germany, 9-10 Nov. 2005.

[12] C. Mehlführer, M. Rupp, F. Kaltenberger, and G. Humer, "A scalable rapid prototyping system for real-time MIMO transmissions," in *Proc. IEE Conference on DSP Enabled Radio*, Univ of Southampton, UK, 19-20 Sept. 2005, pp. 1–5.

[13] C. Mehlführer, M. Rupp, F. Kaltenberger, and G. Humer, "Low-complexity MIMO channel simulation by reducing the number of paths," in *Proc. Workshop on Smart Antennas (WSA 2007)*, Vienna, Austria, February 26–27 2007.

[14] G. Meindl-Pfeiffer, R. Kloibhofer, F. Kaltenberger, and G.Humer, "Development platform for MIMO software defined radio," in *Proc. Embedded World Conference*, Nuernberg, Germany, Feb. 2005, pp. 243–251.

[15] G. Meindl-Pfeiffer, R. Kloibhofer, F. Kaltenberger, and G. Humer, "Multistandard development platform for MIMO software defined radio," in *Proc. EUSIPCO*, Antalya, Turkey, Sept. 2005.

[16] G. Steinböck, F. Kaltenberger, and R. Lieger, "MIMO development platform," COST 273, Duisburg, Germany, Tech. Rep. TD (04)173, Sept. 2004.

[17] J. Wehinger, K. Freudenthaler, F. Kaltenberger, and J. Berkmann, "Influence of SNR estimation on HARQ combining in UMTS-HSDPA," in *Proc. Global Mobile Congress (GMC)*, Chongqing, China, Oct. 2005, pp. 349–354.

[18] J. Wehinger, C. F. Mecklenbräuker, S. Paul, and F. Kaltenberger, "Two-stage space-time receiver for UMTS frequency division duplex," in *ITG Workshop on Smart Antennas*, Munich, Germany, March 2004, pp. 231–234.

[19] T. Zemen, C. Mecklenbräuker, F. Kaltenberger, and B. Fleury, "Minimum-energy band-limited time-variant channel prediction with dynamic subspace selection," *IEEE Transactions on Signal Processing*, 2007, to be published.

Publications from Other Authors

[20] 3GPP, "Third Generation Partnership Project (3GPP)." [Online]. Available: http://www.3gpp.org/

[21] G. Acosta, K. Tokuda, and M. Ingram, "Measured joint doppler-delay power profiles for vehicle-to-vehicle communications at 2.4 ghz," in *Proc. IEEE GLOBECOM*, 2004.

[22] N. Al-Dhahir and G. Saulnier, "A high-performance reduced complexity GMSK demodulator," *IEEE Trans. Comm.*, vol. 46, no. 11, pp. 1409–1412, 1998.

[23] P. Almers, E. Bonek, A. Burr, N. Czink, M. Debbah, V. Degli-Esposti, H. Hofstetter, P. Kyösti, D. Laurenson, G. Matz, A. F. Molisch, C. Oestges, and H. Özcelik, "Survey of channel and radio propagation models for wireless MIMO systems," *EURASIP Journal on Wireless Communications and Networking*, 2007.

[24] H. Arthaber, A. Molisch, E. Bonek, and J. Hammerschmidt, "Diversity techniques and spatial preprocessing for existing GSM mobile terminals," in *International Conference on Telecommunications, ICT-2000*, vol. 2, May 2000, pp. 1045–1050.

[25] Azimuth Systems, Inc., "ACE 400NB MIMO channel emulator," Product Brief, 2006. [Online]. Available: http://www.azimuthsystems.com/files/public/PB_Ace400nb_final.pdf

[26] K. Baddour and N. Beaulieu, "Autoregressive modeling for fading channel simulation," *IEEE Transactions on Wireless Communications*, vol. 4, no. 4, pp. 1650–1662, July 2005.

[27] G. Bauch and V. Franz, "A comparison of soft-in/soft-out algorithms for "turbo-detection"," in *Proc. Int. Conf. on Telecommunications (ICT)*, Porto Carras, Greece, June 1998, pp. 259–262.

[28] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Comm. Syst.*, vol. CS-11, no. 4, pp. 360–393, December 1963.

[29] M. Benedicks, "On Fourier transforms of functions supported on sets of finite Lebesgue measure," *J. Math. Anal. Appl.*, vol. 106, pp. 180–183, 1985.

[30] E. Bonek, M. Herdin, W. Weichselberger, and H. Ozcelik, "MIMO - study propagation first!" in *Proc. 3rd Signal Processing and Information Technology (ISSPIT)*, Dec. 2003, pp. 150–153.

[31] G. Brandmayr, G. Humer, and M. Rupp, "Automatic co-verification of FPGA designs in simulink," in *Proc. of the Model Based Design Conference*, Munich, Germany, Jun 8th–9th 2005.

[32] S. Cherry, "Edholm's law of bandwidth," *IEEE Spectrum*, vol. 41, no. 7, pp. 58–60, July 2004.

[33] R. Clarke, "A statistical theory of mobile-radio reception," *The Bell Systems Technical Journal*, pp. 957–1000, Jul.–Aug. 1968.

[34] L. M. Correia, Ed., *Wireless Flexible Personalised Communications*. Wiley, 2001.

[35] ——, *Mobile Broadband Multimedia Networks Techniques, Models and Tools for 4G*. Elsevier, 2006.

[36] N. Czink, E. Bonek, X. Yin, and B. Fleury, "Cluster angular spreads in a MIMO indoor propagation environment," in *Proc. of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, Berlin, Germany, September 2005, pp. 1–5.

[37] V. Degli-Esposti, F. Fuschini, E. M. Vitucci, and G. Falciasecca, "Measurement and modelling of scattering from buildings," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 1, pp. 143–153, Jan. 2007.

[38] J. W. Demmel, *Applied numerical linear algebra*. SIAM, 1997.

[39] P. Dent, G. E. Bottomley, and T. Croft, "Jakes fading model revisited," *IEE Electronic Letters*, vol. 29, no. 13, pp. 1162–1163, June 1993.

[40] S. Dharanipragada and K. S. Arun, "Bandlimited extrapolation using time-bandwidth dimension," *IEEE Transactions on Signal Processing*, vol. 45, no. 12, pp. 2951–2966, December 1997.

[41] G. D. Durgin, "Theory of stochastic local area channel modelling for wireless communications," Ph.D. dissertation, Virginia Polytechnic Institute and State University, 2000.

[42] V. Erceg, L. Schumacher, P. Kyritsi, A. Molisch, D. S. Baum, A. Y. Gorokhov, C. Oestges, Q. Li, K. Yu, N. Tal, B. Dijkstra, A. Jagannatham, C. Lanzl, V. J. Rhodes, J. Medbo, D. Michelson, and M. Webster, "TGn channel models," IEEE P802.11 Wireless LANs, Tech. Rep., May 2004.

[43] "Digital cellular telecommunications system (phase 2+); multiplexing and multiple access on the radio path, GSM 05.02 version 8.5.0 release 1999," European Standard (Telecommunications Series), Draft ETSI EN 300 908 V8.5.0, ETSI, July 2000.

[44] M. Failli, "Digital land mobile radio communications," COST 207 Final Report, Tech. Rep., 1989.

[45] S. Fanfoni, D. Giancola, U. Girola, S. Parolari, A. Picciriello, and U. Spagnolini, "Space-time processing for co-channel interference rejection and channel estimation in GSM/DCS systems," in *ISSE'98, International Symposium on Signals, Systems, and Electronics*, Pisa, Italy, 1998, pp. 152–155, (ITALTEL R & S, Politecnico di Milano). [Online]. Available: http://www.elet.polimi.it/

[46] B. H. Fleury, "First- and second-order characterization of direction dispersion and space selectivity in the radio channel," *IEEE Transactions on Information Theory*, vol. 46, no. 6, pp. 2027–2044, Sept. 2000.

[47] I. Forkel and H. Klenner, "High speed downlink packet access (HSDPA) - A means of increasing downlink capacity in WCDMA cellular networks?" in *Proc. of the 5th European Wireless Conference 2004*, Barcelona, Spain, Feb. 2004.

[48] G. Forney Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, vol. IT-18, pp. 363–378, 1967.

[49] G. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, vol. 6, pp. 311–335, 1998.

[50] S. Geirhofer, C. Mehlführer, and M. Rupp, "Design and real-time measurement of HSDPA equalizers," in *Proc. 6th International Workshop on Signal Processing Advances for Wireless Communications (SPAWC)*, New York City, USA, June 2005.

[51] B. Haberland, S. Bloch, and V. Braun, "3G evolution towards high speed downlink packet access," *Alcatel Telecommunications Review*, no. 4, 2003.

[52] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Transactions on Vehicular Technology*, vol. 29, no. 3, pp. 317–325, Aug. 1980.

[53] H. Hofstetter, C. Mecklenbräuker, R. Müller, H. Anegg, H. Kunczier, E. Bonek, I. Viering, and A. Molisch, "Description of wireless MIMO measurements at 2GHz in selected environments," COST-273, Lisbon, Portugal, Temporary Document TD(02)135, September 18-19 2002.

[54] H. Hofstetter and G. Steinböck, "A geometry based stochastic channel model for MIMO systems," in *Proceedings of the ITG Workshop on Smart Antennas*, Munich, Germany, March 18th–19th 2004.

[55] H. Holma and A. Tskala, Eds., *WCDMA for UMTS*, 2nd ed. Wiley, 2002.

[56] G. Humer, R. Kloibhofer, G. Pfeiffer, and G. Steinböck, "COST 259 channel model implementation for real-time simulation," COST 273, Prague, Chech Republic, Tech. Rep. TD(03)202, Sept. 2003.

[57] W. Jakes, *Microwave Mobile Communications*. Wiley, 1974.

[58] F. K. Jondral, "Software-defined radio—basics and evolution to cognitive radio," *EURASIP Journal on Wireless Communications and Networking*, no. 3, pp. 275–283, 2005.

[59] T. Kailath and A. H. Sayed, Eds., *Fast Reliable Algorithms for Matrices with Structure*. SIAM, 1999.

[60] R. Kattenbach, "Statistical modeling of small-scale fading in directional radio channels," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 3, pp. 584–592, April 2002.

[61] J. P. Kermoal, L. Schumacher, K. I. Pedersen, P. E. Mogensen, and F. Frederiksen, "A stochastic MIMO radio channel model with experimental validation," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 6, pp. 1211–1226, Aug. 2002.

[62] J. Kolu and T. Jamsa, "A real-time simulator for MIMO radio channels," in *Proc. 5th Intl. Symposium on Wireless Personal Multimedia Communications*, vol. 2, Oct. 2002, pp. 568–572.

[63] H. Krishna and S. D. Morgera, "The Levinson recurrence and fast algorithms for solving Toeplitz systems of linear equations," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 6, pp. 839–848, June 1987.

[64] M. Lagunas, J. Vidal, and A. Pérez-Neira, "Joint array combining and MLSE for single-user receivers in multipath gaussian multiuser channels," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 11, pp. 2252–2259, Nov. 2000.

[65] R. Langwieser, M. Fischer, A. L. Scholtz, M. Rupp, and G. Humer, "Rapid prototyping for RF-transmitters and receivers," in *Proc. Fifth IASTED International Conference on Communication Systems and Networks (CSN 2006)*, Palma de Mallorca, Spain, August 2006.

[66] P. Laurent, "Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP)," *IEEE Transactions on Communications*, vol. 34, no. 2, pp. 150–160, Feb. 1986.

[67] K. M. Lee, J. M. Oh, K. Y. Kwak, and H. Y. Joung, "Throughput performance employing multipath interference canceller in high speed downlink packet access," in *1st IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, Seoul, Korea, Jan. 2004.

[68] Y. X. Li and X. Huang, "The simulation of independent Rayleigh faders," *IEEE Transactions on Communications*, vol. 50, no. 9, pp. 1503–1514, Sep. 2002.

[69] J. C. Maxwell, "A dynamical theory of the electromagnetic field," *Philosophical Transactions of the Royal Society of London*, vol. 155, pp. 459–512, 1865.

[70] C. Mecklenbräuker, R. Müller, A. Pérez-Neira, and M. Lenger, "On simplified space-time receiver structures for GSM," in *Proc. 4th European Personal Mobile Communications Conference (EPMCC)*, Wien, Feb. 2001.

[71] Members of 3GPP, "Technical specification group radio access network; spatial channel model for multiple input multiple output (MIMO) simulations (release 6) (3GPP TR 25.996 version 6.1.0)," 3GPP, Tech. Rep., Sept. 2003.

[72] ——, "Technical specification group radio access network; UE radio access capabilities definition (3GPP TS 25.306 version 6.3.0)," 3GPP, Tech. Rep., Mar. 2004.

[73] ——, "Technical specification group radio access network; User Equipment (UE) radio transmission and reception (FDD) (3GPP TS 25.101 version 6.4.0)," 3GPP, Tech. Rep., Mar. 2004.

[74] ——, "Technical specification group radio access network; high speed downlink packet access: UE radio transmission and reception (FDD) (3GPP TS 25.890 version 1.0.0)," 3GPP, Tech. Rep., May 2005.

[75] ——, "Technical specification group radio access network; multiplexing and channel coding (FDD) (3GPP TS 25.212 version 6.5.0)," 3GPP, Tech. Rep., June 2005.

[76] ——, "Technical specification group radio access network; physical channel and mapping of transport channels onto physical channels (FDD) (3GPP TS 25.211 version 6.5.0)," 3GPP, Tech. Rep., June 2005.

[77] ——, "Technical specification group radio access network; spreading and modulation (FDD) (3GPP TS 25.213 version 6.3.0)," 3GPP, Tech. Rep., June 2005.

[78] A. F. Molisch, H. Asplund, R. Heddergott, M. Steinbauer, and T. Zwick, "The COST259 directional channel model-part I: Overview and methodology," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3421–3433, Dec. 2006.

[79] T. K. Moon and W. Stirling, *Mathematical Methods and Algorithms*. Prentice Hall, 2000.

[80] Motorola, "HS-DSCH simulation results," 3GPP, Tech. Rep. R4-020615, Apr. 2002.

[81] O. Norklit and J. B. Andersen, "Diffuse channel model and experimental results for array antennas in mobile environments," *IEEE Transactions on Antennas and Propagation*, vol. 46, no. 6, pp. 834–840, June 1998.

[82] H. Ozcelik, M. Herdin, W. Weichselberger, J. Wallace, and E. Bonek, "Deficiencies of 'kronecker' MIMO radio channel model," *IEE Electronic Letters*, vol. 39, no. 16, pp. 1209–1210, August 2003.

[83] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 3rd ed. McGraw-Hill, 1991.

[84] S. Parkvall, E. Englund, P. Malm, T. Hedberg, M. Persson, and J. Peisa, "WCDMA evolved — High-speed packet-data services," *Ericsson Review*, no. 2, pp. 56–65, 2003.

[85] M. Pätzold and F. Laue, "Statistical properties of Jakes' fading channel simulator," in *Proc. IEEE VTC-Spring*, May 1998, pp. 712–718.

[86] A. Paulraj, D. Gore, and R. Nabar, *Introduction to Space-Time Wireless Communications*. Cambridge University Press, 2003.

[87] T. Pedersen and B. H. Fleury, "A realistic radio channel model based on stochastic propagation graphs," in *Proc. MATHMOD Conference*, Vienna, Austria, February 2006.

[88] D. B. Percival and A. T. Walden, *Spectral Analysis for Physical Applications*. Cambridge University Press, 1963.

[89] F. Pipon, P. Vila, P. Chevalier, and J. Monot, "Joint spatial and temporal equalization for channels with ISI and CCI: Theoretical and experimental results for a base station reception," in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC'97*. Paris: IEEE, Apr. 1997. [Online]. Available: http://www.tsi.enst.fr/~spawc

[90] A. S. Y. Poon, R. W. Brodersen, and D. N. C. Tse, "Degrees of freedom in multiple-antenna channels: A signal space approach," *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 523–536, February 2005.

[91] M. F. Pop and N. C. Beaulieu, "Limitations of sum-of-sinusoids fading channel simulators," *IEEE Transactions on Communications*, vol. 49, no. 4, pp. 699–708, Apr. 2001.

[92] D. Porter and D. S. Stirling, *Integral equations*. Cambridge University Press, 1990.

[93] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing Principles, Algorithms, and Applications*. Prentice Hall, 1996.

[94] J. M. Rabaey, "Silicon architectures for wireless systems," Tutorial, Hotchips 01, August 2001. [Online]. Available: http://bwrc.eecs.berkeley.edu/People/Faculty/jan/presentations/hotchips1.pdf

[95] V. Rojansky, *Electromagnetic Fields and Waves*. Courier Dover Publications, 1980.

[96] M. Rupp, A. Burg, and E. Beck, "Rapid prototyping for wireless designs: The five-ones approach," *Signal Processing*, vol. 83, pp. 1427–1444, 2003.

[97] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. SIAM, 2003.

[98] K. Sarrigeorgidis and J. M. Rabaey, "Massively parallel wireless reconfigurable processor architecture and programming," in *Proc. Reconfigurable Architectures Workshop (RAW)*, Nice, France, April 2003.

[99] P. Schniter and A. R. Margetts, "Adaptive Chip-Rate Equalization of Downlink Multirate Wideband CDMA," in *Proc. 36th Asilomar Conf. on Signals, Systems and Computers*, vol. 2, Nov. 2002, pp. 1228–1232.

[100] J. Selva, "Interpolation of bounded bandlimited signals and applications," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4244–4260, Nov. 2006.

[101] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–432, July 1948.

[102] D. Slepian, "Prolate spheroidal wave functions, Fourier analysis, and uncertainty - IV: Extensions to many dimensions; generalized prolate spheroidal functions," *The Bell System Technical Journal*, vol. 43, no. 6, pp. 3009–3058, November 1964.

[103] ——, "Prolate spheroidal wave functions, Fourier analysis, and uncertainty - V: The discrete case," *The Bell System Technical Journal*, vol. 57, no. 5, pp. 1371–1430, May-June 1978.

[104] D. Slepian and H. O. Pollak, "Prolate spheroidal wave functions, Fourier analysis, and uncertainty - I," *The Bell System Technical Journal*, vol. 40, no. 1, pp. 43–64, January 1961.

[105] Spirent Communications, Inc., "SR5500 wireless channel emulator," Data Sheet, 2006. [Online]. Available: http://www.spirentcom.com/documents/4247.pdf

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

[107] "Benchmark for TMS 320 C6000 series," Texas Instruments, Inc. 2003. [Online]. Available: http://www.ti.com/sc/docs/products/dsp/c6000/62bench.htm

[108] D. J. Thomson, "Spectrum estimation and harmonic analysis," in *Proceedings of the IEEE*, vol. 70, no. 9, September 1982, pp. 1055–1096.

[109] "3G/UMTS evolution: towards a new generation of broadband mobile services," UMTS Forum, Tech. Rep., Dec. 2006.

[110] M. Valenti and J. Sun, "The UMTS turbo code and an efficient decoder implementation suitable for software-defined radios," *International Journal of Wireless Information Networks*, vol. 8, no. 4, Oct. 2001.

[111] D. V. D. Ville, W. Phillips, and I. Lemahieu, "On the n-dimensional extension of the discrete prolate spheroidal window," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 89–91, March 2002.

[112] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. IT-13, pp. 260–269, 1967.

[113] ——, "A personal history of the Viterbi algorithm," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 120–142, July 2006.

[114] E. E. Weisstein, "Maxwell equations," June 2006. [Online]. Available: http://scienceworld.wolfram.com/physics/MaxwellEquations.html

[115] X. Yin, T. Pedersen, N. Czink, and B. Fleury, "Parametric characterization and estimation of bi-azimuth and delay dispersion of individual path components," in *Proceedings EuCAP 2006*, Nice, France, Nov. 2006.

[116] ——, "Parametric characterization and estimation of bi-azimuth dispersion of path components," in *Proc. SPAWC 2006*, Cannes, France, June 2006.

[117] D. J. Young and N. C. Beaulieu, "The generation of correlated Rayleigh random variates by inverse fourier transform," *IEEE Transactions on Communications*, vol. 48, no. 7, pp. 1114–1127, Jul. 2000.

[118] A. G. Zajić and G. L. Stüber, "Efficient simulation of Rayleigh fading with enhanced de-correlation properties," *IEEE Transactions on Wireless Communications*, vol. 5, no. 7, pp. 1866–1875, July 2006.

[119] Y. V. Zakharov, T. C. Tozer, and J. F. Adlard, "Polynomial spline-approximation of Clarke's model," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1198–1208, May 2004.

[120] T. Zemen, "OFDM multi-user communication over time-variant channels," Ph.D. dissertation, Vienna University of Technology, Vienna, Austria, July 2004.

[121] T. Zemen and C. F. Mecklenbräuker, "Time-variant channel estimation using discrete prolate spheroidal sequences," *IEEE Transactions on Signal Processing*, vol. 53, no. 9, pp. 3597–3607, September 2005.

[122] X. Zhao, J. Kivinen, P. Vainikainen, and K. Skog, "Characterization of Doppler spectra for mobile communications at 5.3 GHz," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 1, pp. 14–23, January 2003.

[123] Y. R. Zheng and C. Xiao, "Simulation models with correct statistical properties for Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 51, no. 6, pp. 920–928, June 2003.

Curriculum Vitae

Florian Kaltenberger was born in Vienna, Austria in 1978. He received his Diploma degree (Dipl.-Ing.) in Technical Mathematics (with distinction) from the Vienna University of Technology in 2002. During the summer of 2001 he held an internship position with British Telecom, BT Exact Technologies in Ipswich, UK, where he was working on mobile video conferencing applications. After his studies he started as a research and teaching assistant at the Vienna University of Technology, Institute for Analysis and Scientific Computing, working on distributed signal processing algorithms. In 2003 he joined the wireless communications group at the Austrian Research Centers GmbH, where he is currently working on the development of low-complexity smart antenna and MIMO algorithms as well as on the ARC SmartSim real-time hardware channel simulator. His research interests include signal processing for wireless communication, MIMO communication systems, receiver design and implementation, MIMO channel modeling and simulation, and hardware implementation issues. He is member of the International Association for the Exchange of Students for Technical Experience (IAESTE) and the Institute of Electrical and Electronics Engineers (IEEE).