Abstract:
This deliverable assesses different link-level candidate technologies for the WINNER air interface.

Keyword list: Physical layer, channel coding, modulation, demodulation, channel estimation, synchronization, iterative methods

Disclaimer:
Executive Summary

In order to achieve the goals set by the WINNER vision, the WINNER air interface has to be designed such that it adapts efficiently to the constraints set by environment conditions as well as economical considerations. The content of this document aims at providing an in-depth analysis of what the authors consider to be the best candidate technologies for what have been identified as potential key WINNER physical layer modes. The performance of different link layer techniques is compared and their suitability to specific channel conditions is discussed. Since all main candidate modulation techniques can be classified into two general areas, serial and parallel modulation (or single and multi carrier techniques), the structure of this document has been adapted to reflect this fact: one chapter is exclusively dedicated to each of the two techniques before simulation results for the different WINNER scenarios are presented in a separate chapter.

The obtained results clearly indicate that a very high efficiency can be achieved at link level. Iterative techniques are a very promising path towards approaching the fundamental performance limits at the physical layer (i.e., minimum channel estimation overhead, error rate versus SNR performance reasonably close to the bounds defined by channel capacity). Imperfections in the analog front-end, such as non-linear power amplifier, phase noise in oscillators, and I/Q-imbalance, will be a significant issue as wireless systems go to higher carrier frequencies and implementations to lower supply voltages (cf. also results in [WIND2.7]). Overcoming these limitations in the RF front-end requires either very costly RF components, or sophisticated signal processing that either avoids or corrects the resulting deteriorations in performance. Finally, interference becomes a critical issue for systems with a frequency reuse close to, or equal to one. While inter-cell operation has not been in the focus of this report, the results presented in the synchronization chapter illustrate that inter-cell timing synchronization can be achieved in OFDM based systems, which is one contribution to avoiding serious UL/DL interference problems for single frequency networks.
## Authors List

<table>
<thead>
<tr>
<th>Partner</th>
<th>Name</th>
<th>Phone / Fax / e-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>CU</td>
<td>David Falconer</td>
<td>Phone: +1 613 520 5722</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +1 613 520 5727</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:ddf@sce.carleton.ca">ddf@sce.carleton.ca</a></td>
</tr>
<tr>
<td></td>
<td>Benjamin Ng</td>
<td>e-mail: <a href="mailto:ngkoon@sce.carleton.ca">ngkoon@sce.carleton.ca</a></td>
</tr>
<tr>
<td></td>
<td>Chan Tong Lam</td>
<td>e-mail: <a href="mailto:lame@sangam.sce.carleton.ca">lame@sangam.sce.carleton.ca</a></td>
</tr>
<tr>
<td></td>
<td>Maryam Sabbaghiann</td>
<td>e-mail: <a href="mailto:msabbagh@sangam.sce.carleton.ca">msabbagh@sangam.sce.carleton.ca</a></td>
</tr>
<tr>
<td>DoCoMo</td>
<td>Gunther Auer</td>
<td>Phone: +49 89 5682 4219</td>
</tr>
<tr>
<td></td>
<td>Jérôme Bonnet</td>
<td>Fax: +49 89 5682 4301</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:auer@docomolab-euro.com">auer@docomolab-euro.com</a></td>
</tr>
<tr>
<td>EAB</td>
<td>Göran Klang</td>
<td>Phone: +46 8 404 4794</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +46 8 585 31480</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:goran.n.klang@ericsson.com">goran.n.klang@ericsson.com</a></td>
</tr>
<tr>
<td>FT</td>
<td>Jean-Philippe Javaudin</td>
<td>Phone: +332 99 12 45 95</td>
</tr>
<tr>
<td></td>
<td>Marie-Hélène Hamon</td>
<td>Fax: +332 99 12 40 98</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:jeanphilippe.javaudin@francetelecom.com">jeanphilippe.javaudin@francetelecom.com</a></td>
</tr>
<tr>
<td>MOT</td>
<td>Markus Muck</td>
<td>Phone: +33 1 6935 2573</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +33 1 6935 7701</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:Markus.Muck@motorola.com">Markus.Muck@motorola.com</a></td>
</tr>
<tr>
<td>NOKCH</td>
<td>Yong Teng</td>
<td>Phone: +86 10 65392828 2753</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +86 10 84210576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:yong.teng@nokia.com">yong.teng@nokia.com</a></td>
</tr>
<tr>
<td>PUT</td>
<td>Krzysztof Wesolowski</td>
<td>Phone: +48 61 665 2741</td>
</tr>
<tr>
<td></td>
<td>Zbigniew Długaszewski</td>
<td>Fax: +48 61 665 2572</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:wesolows@et.put.poznan.pl">wesolows@et.put.poznan.pl</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone: +48 61 665 3916</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +48 61 665 2572</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:zdlugasz@et.put.poznan.pl">zdlugasz@et.put.poznan.pl</a></td>
</tr>
<tr>
<td>SEUK</td>
<td>Thierry Lestable</td>
<td>Phone: +44.1784.428600 Ext.720</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fax: +44.1784.428624</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e-mail: <a href="mailto:thierry.lestable@samsung.com">thierry.lestable@samsung.com</a></td>
</tr>
<tr>
<td>Organization</td>
<td>Name</td>
<td>Phone</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>WINNER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SM</td>
<td>Elena Costa</td>
<td>+49 89 636 44812</td>
</tr>
<tr>
<td>SM/TUHH</td>
<td>Rainer Grünheid</td>
<td>+49 40 42878 2166</td>
</tr>
<tr>
<td>SM/UU</td>
<td>Stephan Stiglmayr</td>
<td>+49 731 50-31531</td>
</tr>
<tr>
<td>TUD</td>
<td>Ernesto Zimmermann</td>
<td>+49 351 463 34958</td>
</tr>
<tr>
<td>UOULU</td>
<td>Kimmo Kansanen</td>
<td>+358 8 553 2833</td>
</tr>
<tr>
<td></td>
<td>Jouko Leinonen</td>
<td>+358 8 553 7642</td>
</tr>
<tr>
<td></td>
<td>Zexian Li</td>
<td>+358 8 553 2877</td>
</tr>
<tr>
<td></td>
<td>Nenad Veselinovic</td>
<td>+358 8 553 2864</td>
</tr>
</tbody>
</table>
Table of Contents

1. Introduction ....................................................................................................................................... 9

2. Initial assumptions ............................................................................................................................... 10

3. Physical Layer Parameter Selection ............................................................................................... 11
   3.1 Overview of WINNER Modes ......................................................................................................... 11
   3.2 Constraints ......................................................................................................................................... 12
   3.2.1 Choice of link-layer parameters for OFDM and serial modulation with block transmission
   and cyclic prefix ....................................................................................................................................... 12
   3.2.2 Parameter Sets for WINNER scenarios ....................................................................................... 14

4. Serial Transmission ............................................................................................................................. 18
   4.1 Approaches for generating OFDM and serial modulated signals: frequency domain approach .... 18
   4.2 Approaches for generating OFDM and serial modulated signals: time domain approach ........ 18
   4.3 Similarities, differences and compatibilities between serial modulation- and multicarrier
   transmission ........................................................................................................................................... 19
   4.4 Coding .............................................................................................................................................. 19
   4.5 Modulation ........................................................................................................................................ 20
   4.6 Detection .......................................................................................................................................... 20
   4.6.1 Frequency domain equalization .................................................................................................. 20
   4.7 Estimation ......................................................................................................................................... 20
   4.8 Synchronization ............................................................................................................................... 21
   4.9 Iterative Techniques .......................................................................................................................... 22
   4.9.1 Turbo frequency domain equalization with perfect channel state information (CSI) .......... 22
   4.9.2 Turbo frequency domain equalization with channel estimation ............................................... 26
   4.10 Pre-distortion .................................................................................................................................. 28
   4.10.1 Power backoff required for typical solid state HPA’s ............................................................... 28
   4.10.2 Predistortion: methods to reduce instantaneous power variations ......................................... 29
   4.10.3 HPA predistorter ....................................................................................................................... 29

5. Multi Carrier Transmission ............................................................................................................... 32
   5.1 Coding .............................................................................................................................................. 32
   5.1.1 Methodology .................................................................................................................................. 33
   5.1.2 Performance Evaluation ............................................................................................................. 34
   5.1.3 Complexity Assessment ............................................................................................................. 39
   5.1.4 Decoding Complexity-Performance Trade-Off ......................................................................... 43
   5.1.5 Conclusions and Further Work ................................................................................................ 45
   5.2 Modulation ....................................................................................................................................... 46
   5.2.1 Specificities of advanced multi carrier schemes ....................................................................... 46
   5.3 Estimation ......................................................................................................................................... 67
   5.3.1 Pilot aided channel estimation (PACE) ....................................................................................... 67
   5.3.2 Cyclic channel estimation for high Doppler ............................................................................. 69
   5.4 Synchronization ............................................................................................................................... 73
   5.4.1 OFDM based synchronization on the downlink ........................................................................ 73
   5.4.2 Synchronization in OFDM based single cell and cellular networks ........................................ 74
   5.4.3 Synchronization for MIMO-OFDM ............................................................................................ 85
   5.5 Iterative Techniques .......................................................................................................................... 94
   5.5.1 Iterative Channel Estimation .................................................................................................... 94
   5.5.2 Iterative Interference Suppression for Pseudo-Random-Postfix OFDM based Channel
   Estimation ............................................................................................................................................... 107
   5.6 Pre-distortion .................................................................................................................................. 110

6. Evaluation of proposed techniques .................................................................................................. 114
   6.1 Wide area cellular, uplink (SM) ....................................................................................................... 114
   6.1.1 Linear equalization with perfect channel state information .................................................... 114
6.1.2 Linear equalization: coding and modulation evaluation.................................................. 115
6.1.3 Linear equalization with training sequences ................................................................... 118
6.1.4 Effect of frequency offset .............................................................................................. 120
6.1.5 IFDMA .......................................................................................................................... 120
6.1.6 Evaluation of Iterative Techniques for Single-Carrier Modulation................................. 125
6.2 Wide area cellular, down-link .......................................................................................... 129
   6.2.1 IOTA-OFDM ............................................................................................................. 129
   6.2.2 CP-OFDM .................................................................................................................. 131
   6.2.3 Pilot aided channel estimation for OFDM ................................................................. 132
   6.2.4 Approximation of channel estimation errors ............................................................. 136
   6.2.5 OFDM synchronization strategies for the downlink .................................................... 139
6.3 Wide area feeder link ....................................................................................................... 142
6.4 Short range ...................................................................................................................... 142
   6.4.1 CP-OFDM .................................................................................................................. 142
   6.4.2 Pilot aided channel estimation (PACE) for OFDM .................................................... 143
6.5 Evaluation of required power backoff for OFDMA and serial modulation ......................... 145
   6.5.1 Implication for performance comparison of OFDM and serial modulation ............... 148
6.6 Comparison of CP-OFDM and serial modulation in Wide-area ........................................ 149

7. Summary and Conclusions .............................................................................................. 150

A. Appendix ......................................................................................................................... 154
   A.1 Assessment of Coding Techniques .............................................................................. 154
       A.1.1 Decoding Complexity .............................................................................................. 154

Calibration Results ............................................................................................................. 163
   A.2 Simulation Setup .......................................................................................................... 163
   A.3 Wide Area Case (Urban Macro) ................................................................................. 164
   A.4 Short Range Case (802.11n C NLOS) ........................................................................ 167
   A.5 Feeder Link Case (AWGN) ........................................................................................ 168

B. Channel Definitions ........................................................................................................ 170

C. References ..................................................................................................................... 171
List of Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3GPP</td>
<td>3rd Generation Partnership Project</td>
</tr>
<tr>
<td>AMC</td>
<td>Adaptive Modulation and Coding</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point</td>
</tr>
<tr>
<td>APP</td>
<td>A Posteriori Probability</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>BER</td>
<td>Bit error rate</td>
</tr>
<tr>
<td>BICM</td>
<td>Bit-Interleaved Coded Modulation</td>
</tr>
<tr>
<td>BLER</td>
<td>Block Error Rate</td>
</tr>
<tr>
<td>BPSK</td>
<td>Binary phase shift keying</td>
</tr>
<tr>
<td>BS</td>
<td>Base Station</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Distribution Function</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CIR</td>
<td>Channel Impulse Response</td>
</tr>
<tr>
<td>CP</td>
<td>Cyclic prefix</td>
</tr>
<tr>
<td>CSI</td>
<td>Channel State Information</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier transform</td>
</tr>
<tr>
<td>DL</td>
<td>Downlink</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital Signal Processor</td>
</tr>
<tr>
<td>EXIT</td>
<td>Extrinsic Information Transfer</td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplex</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Coding</td>
</tr>
<tr>
<td>FER</td>
<td>Frame Error Rate</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transformation</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array</td>
</tr>
<tr>
<td>GMC</td>
<td>Generalized Multi Carrier</td>
</tr>
<tr>
<td>HARQ</td>
<td>Hybrid Automatic Repeat Request</td>
</tr>
<tr>
<td>HPA</td>
<td>High power amplifier</td>
</tr>
<tr>
<td>HPBW</td>
<td>Half-Power Beamwidth</td>
</tr>
<tr>
<td>ICE</td>
<td>Iterative Channel Estimation</td>
</tr>
<tr>
<td>ICI</td>
<td>Intercarrier Interference</td>
</tr>
<tr>
<td>IFFT</td>
<td>Inverse fast Fourier transform</td>
</tr>
<tr>
<td>IOTA</td>
<td>Isotropic Orthogonal Transform Algorithm</td>
</tr>
<tr>
<td>ISI</td>
<td>Intersymbol interference</td>
</tr>
<tr>
<td>LOS</td>
<td>Line-Of-Sight</td>
</tr>
<tr>
<td>LLR</td>
<td>Log-Likelihood Ratio</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posteriori</td>
</tr>
<tr>
<td>MC-CDMA</td>
<td>Multicarrier code division multiple access</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>MCS</td>
<td>Modulation and Coding Scheme</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Multiple Output</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood</td>
</tr>
<tr>
<td>MLD</td>
<td>Maximum Likelihood Detection</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean-Square-Error</td>
</tr>
<tr>
<td>MS</td>
<td>Mobile Station</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Square Error</td>
</tr>
<tr>
<td>MUD</td>
<td>Multi-user detection</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>OFDMA</td>
<td>Orthogonal Frequency Division Multiple Access</td>
</tr>
<tr>
<td>PACE</td>
<td>Pilot Aided Channel Estimation</td>
</tr>
<tr>
<td>PAPR</td>
<td>Peak to average power ratio</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>PDP</td>
<td>Power Delay Profile</td>
</tr>
<tr>
<td>PER</td>
<td>Packet Error Rate</td>
</tr>
<tr>
<td>PRP-OFDM</td>
<td>Pseudo-Random Postfix OFDM</td>
</tr>
<tr>
<td>PSD</td>
<td>Power Spectral Density</td>
</tr>
<tr>
<td>PSK</td>
<td>Phase Shift Keying</td>
</tr>
<tr>
<td>QAM</td>
<td>Quadrature Amplitude Modulation</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RF</td>
<td>Radio Frequency</td>
</tr>
<tr>
<td>Rx</td>
<td>Receiver</td>
</tr>
<tr>
<td>SC</td>
<td>Single carrier</td>
</tr>
<tr>
<td>SC-FDE</td>
<td>Single carrier with frequency domain equalization</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal-to-Interference-plus-Noise Ratio</td>
</tr>
<tr>
<td>SISO</td>
<td>Single Input Single Output</td>
</tr>
<tr>
<td>SiSo</td>
<td>Soft-Input Soft-Output</td>
</tr>
<tr>
<td>SM</td>
<td>Serial Modulation</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to noise ratio</td>
</tr>
<tr>
<td>TDD</td>
<td>Time Division Duplex</td>
</tr>
<tr>
<td>TDMA</td>
<td>Time Division Multiple Access</td>
</tr>
<tr>
<td>Tx</td>
<td>Transmitter</td>
</tr>
<tr>
<td>UL</td>
<td>Uplink</td>
</tr>
<tr>
<td>WINNER</td>
<td>Wireless World Initiative New Radio</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
</tbody>
</table>
1. Introduction

In order to achieve the goals set by the WINNER vision, the WINNER air interface has to be designed such that it adapts efficiently to the constraints set by environment conditions as well as economical considerations. The approach taken by the WINNER project to achieve these requirements is to base the radio access network on different deployment concept modes that in turn are supported by different physical layer modes [WIND7.2]. The content of this document aims at providing an in-depth analysis of what the authors consider to be the best candidate technologies for a few such modes that has been identified as potential key WINNER physical layer modes. The performance of different link layer techniques is compared and their suitability to specific channel conditions is discussed. The conclusions drawn at the end of this report enable to get a first idea of how the actual WINNER air interface will look like.

A review of the state-of-the-art in link level techniques has already been performed in [WIND2.1], so that this document will present mostly results that are either novel or specific to the WINNER scenarios. This relates to the assessment of the performance-complexity trade-off for FEC techniques, the sensitivity of different modulation formats to RF and other imperfections, the requirements and performance of different channel estimation techniques, and efficient methods for intra- and intercell synchronization. In order to ensure a fair comparison between the schemes investigated by the involved partners, link level simulation chains have been calibrated against each other before starting actual evaluations. Results for this calibration phase can be found in the appendix. Since all main candidate modulation techniques can all be classified into two general areas: serial and parallel modulation (or single and multi carrier techniques), the structure of this document has been adapted to reflect this fact: one chapter is exclusively dedicated to each of the two techniques before simulation results for the different WINNER scenarios are presented in a separate chapter.

The outline of this document is as follows: Chapter 2 provides details on the assumptions made for deriving the results in this report. Chapter 3 introduces the WINNER physical layer modes and channel models used for evaluation. General design issues of OFDM and serial modulation systems are discussed and parameters set for these two basic classes of transmission techniques. Chapter 4 is dedicated to serial modulation. First, different approaches for generating parallelly and serially modulated signals are introduced and compared. In the following, the specifics of this transmission technique are discussed, where the focus is on iterative equalization techniques and pre-distortion. Chapter 5 discussed multi carrier transmission. A general discussion of the coding performance-complexity trade-off is presented and the merits of several advanced multi carrier modulation schemes are discussed. Details on requirements and performance of pilot aided channel estimation are given and mechanisms for intra- and intercell synchronization are studied. This is followed up by a study on iterative channel estimation. The main simulation results are presented in Chapters 4 and 5, while the detailed performance results for the different proposed techniques are presented in Chapter 6, with focus on the different WINNER scenarios. Conclusions are drawn in Chapter 7. The appendix contains further details on the complexity assessment regarding the decoding of different codes proposed for use within WINNER, as well as calibration results for the link level simulation chains of participating partners.
2. Initial assumptions

The results presented in this report were generated keeping in mind the system requirements specified in [WIND7.1], namely a very high spectral efficiencies (up to 10 b/s/Hz), high sustainable peak data rates, and user speeds ranging up to several hundred km/h, in some specific application scenarios. To achieve these challenging targets in a high number of cases, the efficiency of the WINNER link layer must be significantly higher than that of currently deployed wireless communications systems, approaching the physical limits of what amount of information can be transmitted over the wireless channel under the conditions prevailing in the considered scenario. This motivates for the use of very powerful, yet complex algorithms and techniques that might be impractical for current systems. It also necessitates the selection of link layer parameters such that the overhead in terms of guard periods, guard bands, etc. is kept to a minimum – which in many cases sets challenging requirements to the implementation architecture (e.g., oscillator quality). It is, however, assumed that the continuous enhancement of RF circuitry and baseband processing power will make the WINNER system possible at reasonable cost. More economical versions of the system could of course be built by sacrificing some portion of spectral efficiency, yet the focus is on getting as close to the physical limits as possible.

In order to enable a selection of link level parameters, the following general assumptions were made regarding a potential bandwidth allocation for the WINNER system:

- Carrier frequencies of up to 6 GHz, with the focus in this report on a carrier frequency of 5 GHz,
- 100 MHz system bandwidth in the short range mode, using TDD as a natural choice for systems with small cell ranges and low mobility
- 20 MHz paired spectrum for the wide area cellular mode, using frequency division duplex.
3. Physical Layer Parameter Selection

3.1 Overview of WINNER Modes

In [WIND7.2] a set of deployment scenarios for the WINNER system has been defined. These have been categorized into four main classes, according to their transmission range and network layout: wide area cellular, wide area feeder link, short range cellular and short range peer-to-peer. An additional parameterization with respect to the prevailing channel conditions yields a set of modes in which the proposed link layer techniques are to be used. An overview of these scenarios is given in the table below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Wide Area, Cellular</th>
<th>Wide Area, Feeder Link</th>
<th>Short Range, Cellular</th>
<th>Short Range, Peer-to-Peer</th>
</tr>
</thead>
<tbody>
<tr>
<td>[WIND7.2] Rural D.1 B.5 B.3</td>
<td>WINNER WP5 Rural</td>
<td>AWGN</td>
<td>Urban/hotspot NLOS</td>
<td>IEEE 802.11n E NLOS</td>
</tr>
<tr>
<td>Suburban C.1</td>
<td>WINNER WP5 Suburban Macro'</td>
<td>IEEE 802.11n C NLOS</td>
<td>Indoor (office) B.3</td>
<td>IEEE 802.11n C NLOS</td>
</tr>
<tr>
<td>Typical Urban B.1/C.2</td>
<td>WINNER WP5 Urban Macro'</td>
<td></td>
<td>Indoor A.2</td>
<td></td>
</tr>
<tr>
<td>Bad Urban C.3</td>
<td>Unavailable</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Outdoor to indoor B.4</td>
<td>WINNER WP5 Micro'</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Modes and associated channel models used for evaluations in this report

Note that from the single link physical layer perspective, which is the main focus of this document, the short range cellular and peer-to-peer modes are equivalent, since they are based on the same channel conditions. There was no channel model available for the “Bad Urban” mode at the time of writing this document. The channel models enumerated above, together with the scenario-specific information such as, e.g., user mobility given in [WIND7.2] essentially define the environment constraints to which the physical layer has to adapt as good as possible. The following section gives an overview of constraints that can be derived from these figures and how they translate into an appropriate parameter selection for serial modulation and multicarrier techniques.

1 In order to facilitate link level simulation, channel models for the wide area case have been downsampled to 200 MHz (equivalent to 5 ns spacing) where necessary, to allow simulation with an oversampling factor of only 10 at 20 MHz system bandwidth.
3.2 Constraints

3.2.1 Choice of link-layer parameters for OFDM and serial modulation with block transmission and cyclic prefix

The foundations of conventional OFDM and of serial modulation (SM) with block transmission and cyclic prefix have been extensively described in [WIND2.1], Section 4.4 and [WIND2.1], Sections 4.13-4.17, respectively. We consider the subcarrier spacing (and hence, the useful FFT block duration) to be the main design parameters in the design of OFDM, SM, or in general, generalized multicarrier (GMC) systems. In what follows, our analysis centers on OFDM. Most of it also applies to SM, but at the end we indicate the differences.

3.2.1.1 Subcarrier spacing

3.2.1.1.1 Upper bounds

One main motivation for using OFDM as a transmission technique is to avoid the complexity of receiver structures that are able to cope with ISI. It is hence straightforward to see that one fundamental design constraint is that each subcarrier shall be flat fading, i.e., the subcarrier spacing must be smaller than the expected channel coherence bandwidth: \( \Delta f < B_c \). Using the popular rule-of-thumb where the coherence bandwidth is the inverse of the maximum channel excess delay, we see that:

\[
\Delta f < \frac{1}{\tau_{me}} \tag{3.1}
\]

Another constraint is the target system overhead ratio \( \eta \). Since the guard interval of the OFDM system must be at least as long as the worst case expected maximum channel excess delay \( \tau_{me} \), it is easily seen that:

\[
\Delta f < \frac{\eta}{\tau_{me}} \tag{3.2}
\]

where \( \eta \) is defined as

\[
\eta = \frac{T_{Gi}}{T_{OFDM}} = T_{Gi} \Delta f \tag{3.3}
\]

and \( T_{OFDM} \) is the FFT block time duration. Note that the guard interval length should also be appropriately chosen in order to be able to accommodate residual errors in timing synchronization as well as the impulse responses in the transceiver RF chain. The spectral efficiency constraint is obviously the tighter one, that is, whenever the guard interval overhead is at an acceptable level, we can expect subcarriers to be flat fading. More precisely, the inverse of the guard interval overhead roughly denotes the number of adjacent subcarriers that experience the same fading conditions on a single link.

Another important point to be considered in this context is granularity: if we aim for continuous transmission for all devices in the network (which might be desirable due to PAPR considerations [FK05]), the number of subcarriers defines the ratio between the highest and the lowest bit rate user, if OFDMA is used as a multiple access scheme. As soon as the system bandwidth is known, the FFT size can be derived, and such constraints can be mapped to the subcarrier spacing.

3.2.1.1.2 Lower bounds

The subcarrier spacing in an OFDM system is lower bounded by a number of effects that all introduce inter-carrier-interference (ICI):
- Doppler spread/time selectivity
- Phase noise
- Imperfect synchronization.
There exist two rules of thumb considering time selectivity and Doppler spread in OFDM systems:

- The Doppler spread should at maximum be 5% of the subcarrier spacing:
  \[ \Delta f > 20 f_{D,\text{max}} = 20 \frac{v_{\text{max}}}{c} f_c \]  
  \[ \text{(3.4)} \]

- The OFDM symbol length \( T_{\text{OFDM}} \) should be shorter than the channel coherence time. Assuming a Jakes spectrum and targeting the 99% coherence time, this results in:
  \[ \Delta f > 30(1 + \eta) \frac{v_{\text{max}}}{c} f_c \]  
  \[ \text{(3.5)} \]

The two rules-of-thumb essentially relate to the same constraint and obviously produce the roughly the same figures. For a target velocity of 30 m/s (~100 km/h) and a carrier frequency of 5GHz, the resulting minimum subcarrier spacing is 16.5 kHz, assuming a guard interval overhead of 10%.

One rule of thumb for phase noise considerations is that modulation formats up to 64QAM can be supported as long as the 3dB frequency of the phase noise spectrum is lower than 0.05% of the subcarrier spacing:

\[ f_{3\text{dB,PN}} > 5 \cdot 10^{-4} \Delta f \]  
\[ \text{(3.6)} \]

The oscillator figure of merit is \( \mathcal{F}(f_c) \) from which the 3dB frequency is derived as follows:

\[ f_{3\text{dB,PN}} = \pi f_c^2 \mathcal{F}(f_c) \left(\frac{f_c}{f_c}\right)^2 = \pi \mathcal{F}(f_c) f_c^2 \]  
\[ \text{(3.7)} \]

from which the following lower bound on the subcarrier spacing can be established:

\[ \Delta f > 2\pi \cdot 10^3 \cdot \mathcal{F}(f_c) \cdot f_c^2 \]  
\[ \text{(3.8)} \]

Assuming a carrier frequency of 5 GHz and a currently available quality of oscillators of -123 dBC/Hz at 3 MHz, the lower bound on subcarrier spacing, would be around 28 kHz (considering for 64-QAM transmission). With the quality figure of -115dBC/Hz at 1 MHz given in [WIND2.2], the subcarrier spacing could be reduced to even 20 kHz. This figure is obviously mainly influenced by the cost one wants to allow for a handset and the target modulation formats, i.e., the maximum spectral efficiency that one wants to achieve.

Details on requirements and performance of different synchronization techniques are discussed later in this report.

### 3.2.1.1.3 Minimum Cyclic Prefix Overhead when using Orthogonal Signaling

Taking the minimum subcarrier spacing as constrained by Doppler spread allows for deriving the minimum amount of cyclic prefix overhead if we are targeting orthogonal signalling. Taking the two lower bounds for the subcarrier spacing, one can state that:

\[ \eta_D = (1 + \eta_D) \beta_0 \frac{v_{\text{max}}}{c} f_c \tau_{mc} = (1 + \eta_D) \eta_U \]  
\[ \text{(3.9)} \]

and

\[ \eta_{PN} = 20kHz \cdot \tau_{mc} \]  
\[ \text{(3.10)} \]

such that
\[ \eta = \max \left\{ \frac{\eta_u}{1-\eta_u}, \tau_{py} \right\} \]  

(3.11)

The resulting minimum overhead is plotted in Figure 3.2.1.

![Figure 3.2.1: Minimum CP overhead](image)

It is clearly visible that phase noise is the dominant influence for velocities under 120 km/h, under the current assumptions on oscillator quality. It can also be seen that in environments with larger delay spread, the minimum overhead lies in the order of 15-20%.

### 3.2.1.2 FFT Size

Based on the allowable range for the subcarrier spacing, and the known system bandwidth it is then possible to choose an appropriate number of subcarriers for the system, i.e., the FFT size. Note that the FFT bandwidth need not necessarily be the system bandwidth, as the FFT can be used to oversample the channel. This degree of freedom should be exploited whenever compatibility issues arise: fixing the subcarrier spacing to be an integer multiple of the subcarrier spacing of legacy systems may facilitate coexistence and/or upgrading. For the initial WINNER T2.2 parameter set, this degree of freedom has not been exploited.

This rationale obviously calls for different FFT sizes whenever the system bandwidth is changed, in order to optimally adapt to channel conditions.

### 3.2.2 Parameter Sets for WINNER scenarios

#### 3.2.2.1 OFDM Transmission Case

Within the context of WINNER, the following specific assumptions were made:

- The guard interval overhead shall not exceed 10%.
- The maximum channel excess delay supported in the **short range** mode is 800ns.
- The system (and FFT) bandwidth in the short range mode is 100MHz.
- The maximum channel excess delay supported in the **wide area** mode is 5 µs.
- The system (and FFT) bandwidth in the wide area mode is 20MHz.
It can clearly be seen that in the wide area scenario, a 10% overhead for the cyclic prefix and support for velocities beyond 125 km/h cannot be simultaneously supported. Phase noise will also prevent higher order modulation (64-QAM and larger) from being used in the wide area scenario. The above graph motivates the selection of a subcarrier spacing of 48.8 kHz in the short range mode, corresponding to an FFT size of 2048 in 100MHz, and a subcarrier spacing of 19.5 kHz in the wide area mode, corresponding to an FFT size of 1024 in 20MHz. The OFDM system parameters are summarized in the table given below:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Wide Area FDD</th>
<th>Short Range TDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carrier frequency [GHz]</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>FFT size</td>
<td>1024</td>
<td>2048</td>
</tr>
<tr>
<td>System bandwidth [MHz]</td>
<td>20.0</td>
<td>100</td>
</tr>
<tr>
<td>Subcarrier spacing [kHz]</td>
<td>19.531</td>
<td>48.828</td>
</tr>
<tr>
<td>Useful OFDM symbol duration [µs]</td>
<td>51.20</td>
<td>20.48</td>
</tr>
<tr>
<td>Cyclic prefix length [µs]</td>
<td>5.00</td>
<td>0.80</td>
</tr>
<tr>
<td>Total OFDM symbol duration [µs]</td>
<td>56.20</td>
<td>21.28</td>
</tr>
<tr>
<td>Number of used subcarriers</td>
<td>832</td>
<td>1664</td>
</tr>
<tr>
<td>(symmetric, DC not used)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Signal bandwidth [MHz]</td>
<td>16.25</td>
<td>81.25</td>
</tr>
</tbody>
</table>

For IOTA-OFDM, the choice of the parameters is very similar than for the conventional OFDM case expect that no cyclic prefix is added. Thanks to the optimal time-frequency localization of the IOTA filter, the impact of the channel time dispersion doesn't degrades the BER as long as it doesn't exceed approximately 15% of the IOTA-OFDM symbol duration (see the evaluation section for simulation results). As the assumption of a guard interval duration of maximum 10% the OFDM symbol duration is made in the WINNER system, using the same FFT size in IOTA-OFDM (ie. using also the same carrier spacing if we assume a fixed system bandwidth) as for cyclic-prefix OFDM will not impact on the performance of the IOTA-OFDM modulation.
3.2.2.2 Serial Modulation Case

For serial modulation, with cyclic prefixes separating successive blocks, similar considerations apply as for OFDM. However there is a difference in the sensitivity of OFDM and SM to frequency offset and phase noise. Serial modulated systems are known to be much less sensitive to frequency offset and phase noise than OFDM and other multicarrier systems [PVM95]. The phase noise tolerance will be more than for OFDM, i.e. a subcarrier spacing significantly lower than 28 kHz should be possible. As a result, the overhead will be less influenced by phase noise. The parameters for serial modulation transmission, compatible to the corresponding OFDM parameters, are listed in Table 3.2.

Table 3.2 Serial modulation parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Wide Area FDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equalizer FFT size</td>
<td>1024</td>
</tr>
<tr>
<td>System bandwidth [MHz]</td>
<td>20.0</td>
</tr>
<tr>
<td>Modulated symbols per (FFT) block</td>
<td>832</td>
</tr>
<tr>
<td>Symbol rate [Msps]</td>
<td>16.25</td>
</tr>
<tr>
<td>Pulse filtering roll-off (rrc)</td>
<td>0.23</td>
</tr>
</tbody>
</table>

3.2.2.3 Performance loss due to Doppler spread

In order to validate the choice of the WINNER parameters for OFDM and serial modulation, simulations with worst case channel conditions were carried out. The most critical scenario with respect to inter carrier interference (ICI) is the rural channel model with a mobile velocity of 250km/h. In this case the normalized Doppler frequency, \( f_{D,\text{max}} \cdot (T + T_{\text{g}}) \), becomes about 6.5% of the overall OFDM symbol duration (including the guard interval) \( T + T_{\text{g}} \).

In Figure 3.2.3 the BER is plotted against the \( E_b/N_0 \) for simulations assuming a static channel within one FFT block in comparison with time domain simulations. It can be observed that the degradation due to Doppler is rather low, even for very high velocities of 250km/h. Hence, it appears reasonable to simulate the system performance of the WINNER modes in the frequency domain using the block static fading assumption.
3.2.2.4 Guard bands

Figure 3.2.4 illustrates the nearly identical power spectra of transmitted OFDM and serial modulated signals, generated (see Section 4.1) (a) with the time domain approach, with 23% square root raised cosine filtering, and (b) with the frequency domain approach, using 5.6% raised cosine time windowing. If it is required that adjacent signal power spectra intersect at, say their -60 dB points, then with these choices of parameters, adjacent channel signals could be spaced at about 1.3 times the effective data symbol rate (where the effective data symbol rate is (number of used subcarriers)/(useful OFDM symbol duration)) which results in similar guard band overhead as for current IEEE 802.11a systems. Based on these assumptions, the effective data symbol rate for the WINNER physical layer is then 16.25 Msymbols/s and 81.25 Msymbols/s for wide area FDD and short range FDD, respectively. Obviously, if the spectrum mask sets lower requirements on the adjacent channel suppression, the size of the guard bands could be reduced.

The figure also shows the ETSI 3GPP spectral mask [ETSI125104] scaled to fit the spectra shown. In this particular example, the spectrum generated using the frequency domain approach fits into the ETSI mask with room to spare. The time domain-generated spectrum would have been narrower if a root raised cosine rolloff lower than 0.23 were chosen.

![Figure 3.2.4: Examples of OFDM and serial modulation transmitted spectra: (a) generated with time domain method, with 23% square root raised cosine filtering; (b) generated with frequency domain method with 5.6% raised cosine time windowing. Also shown is the scaled ETSI 3GPP spectral mask.](image-url)
4. Serial Transmission

4.1 Approaches for generating OFDM and serial modulated signals: frequency domain approach

There are basically 2 different approaches to generating OFDM and serial modulated (SM) signals: (1) frequency domain (traditionally used for OFDM, and can also be used for SM); and (2) time domain (traditionally used for SM, and can also be used for OFDM). Signal processing, at least at the receiver, is likely to be in the frequency domain, using FFT operations, and therefore we will assume that signals will be transmitted in block format, with successive blocks separated by cyclic prefixes.

The classical way of generating OFDM signals is to take the coded data symbols, say of length $M$, zero-pad them to make a block whose length, say $N>M$, is the desired number of time domain samples per block, and then subject this to an inverse FFT (fast Fourier transform) to generate the $N$-sample time domain waveform. To this is pre-pended the last part of the block waveform as a cyclic prefix. The result is a sampled time domain waveform which has been time-domain-windowed by a rectangular window whose length equals $N$ plus the cyclic prefix length. Because of this rectangular time domain windowing (which corresponds to convolution with sinc($fT$) in the frequency domain), the corresponding frequency spectrum has slowly-decaying tails (sidelobe magnitudes proportional to $1/fT$, where $T$ is the time duration of the window). To achieve faster frequency domain rolloff, it is common to use a non-rectangular time domain window; e.g. a raised cosine window, which will give a spectrum rolloff proportional to $(1/fT)^3 \times (1/\alpha)^2$, where $\alpha$ is the raised cosine rolloff factor.

To generate SM signals in the frequency domain, exactly the same method can be used, except that the original block of $M$ data symbols is now replaced by the FFT of the $M$ data symbols. The combination of FFT, zero-padding and inverse FFT produces a serial modulated signal waveform, in which the original data symbols appear sequentially, at $T/M$-spaced time instants. Each data symbol modulates a pulse waveform proportional to sinc($\pi Mn/N$). The envelope of this waveform decays approximately inversely proportional to time. As with OFDM, the duration of the transmitted block waveform is truncated to $T$ by the rectangular or raised cosine time window. For both OFDM and SM, the data is undisturbed if the cyclic prefix length exceeds the channel delay spread plus the rolloff portion of the time window.

This common-modulator [WIND2.1] frequency domain approach can be designated as a generalized multicarrier [WG00] [FH97], [BR02], [WWN03], [HPL+04] approach. Other variants of multicarrier and serial modulation also fit into this framework; e.g. DS-CDMA, MC-CDMA, SS-MC-MA, FDOSS and IFDMA [FK05]. More details on this approach can be found in [WIND2.1], [WIND2.2] and [FK05].

Multi-band versions of both OFDM and SM signals generated in this manner would result from re-arranging the padded zeroes and data components before doing the inverse FFT [WIND2.2] [FM05].

4.2 Approaches for generating OFDM and serial modulated signals: time domain approach

The classical way of generating a block of SM signals with a cyclic prefix is to form a block of $M$, pre- pend the last part of the block to form a cyclic prefix, and then pass the resulting symbols sequentially through say, a square root raised cosine time domain filter. OFDM can be generated in this way too: do a $M$-point inverse FFT on the data block, add the cyclic prefix, and pass the resulting samples sequentially through the time domain raised cosine or square root raised cosine filter. The resulting SM or OFDM power spectra will have the raised cosine frequency domain shape, with the same rolloff as that of the time domain filter. For raised cosine filtering typically implemented with FIR filters, the effective overall channel delay spread would be increased by half the length of the FIR filter; e.g. typically about 10 symbol periods.

In both approaches, there would be additional analog filtering after the D to A converter, and before the power amplifier. This filtering would add very slightly to the overall channel delay spread.
4.3 Similarities, differences and compatibilities between serial modulation- and multicarrier transmission

Serial (single carrier) modulation and multicarrier (or parallel) modulation can be considered members of the same GMC family, and can be generated and received by the same DSP engines, by appropriate insertion of FFT and inverse FFT operations in the frequency domain approach. Both offer block frequency domain processing with much lower complexity and spectrum flexibility than would be possible with traditional single carrier modulation with pure time domain signal transmitter and receiver processing. Their signal processing similarities also facilitate adaptive reconfigurability among different modulation schemes in response to user requirements, radio environment, etc. For example, a user terminal may switch from transmitting OFDM or OFDMA in a microcell indoor environment, where terminal power amplifier efficiency and cost are paramount importance, to serial modulation in an outdoor environment, where where terminal power amplifier efficiency and cost have paramount importance.

Advantages of serial modulation (adapted from [WIND2.1], section 3.2.17):
- For the uplink, where low peak to average power ratio (PAPR) is desirable, to minimize back-off, serial modulation allows cheaper power amplifiers and/or greater range ([FK05], [FM05], [CND+00].
- Serial modulation has better bit error rate versus peak SNR than does parallel modulation ([MG01]). Its BER performance is also equal to or slightly better than that of OFDM on frequency selective channels for code rates greater than about ¾, and much better performance for uncoded systems [SKJ94].
- Serial modulation is less sensitive to oscillator phase noise and frequency offset than is OFDM [PVM95].

Advantages of parallel (multicarrier) modulation (adapted from [WIND2.1], section 4.1.7.5):
- Coded OFDM(A) offers slightly better BER performance versus average SNR than coded, linearly-equalized serial modulation at code rates below about 3/4, for frequency-selective channels [SKJ94]. This results from the very powerful enhancement of frequency diversity afforded by coding across the subcarriers of OFDM.
- OFDM and OFDMA can employ adaptive loading (optimal allocation of data bits and power to subcarriers), to approach optimal (in an information-theoretic sense) use of bandwidth.
- In the generalized multicarrier signal generation approach, serial modulation requires an extra FFT operation at the transmitter and an extra inverse FFT operation at the receiver.

Both serial and parallel modulation systems can be enhanced by diversity and MIMO techniques, spectrum spreading (variants of CDMA), turbo coding and other iterative detection, channel estimation and decoding techniques. They can employ similar methods for channel and equalizer parameter estimation, including use of training blocks, and (as outlined in Section 4.7) in-block pilot subcarriers.

4.4 Coding

Forward error correcting (FEC) codes are used to increase the transmission reliability by using redundant information in the transmitted signal. In general, design principles of codes for many different channels are well known. The channel scenarios and system parameters considered in the WINNER project are such that the channel can be considered to be nearly block-static over and FFT block. Furthermore, the frequency-domain MMSE equalizer for the single-carrier transmission provides a constant SINR for the symbol estimates it produces in such a case (within a block). As such, the channel seen by the encoder and decoder is essentially a flat fading channel with and SINR varying from block to block. The statistics of the variation are defined by the channel statistics. For all practical purposes, one may employ the same channel codes as for the multicarrier transmission, studied more in Section 5.1, even though the performance of such codes may not be optimal for a serially modulated system.

A practical FEC scheme suitable for a flexible and adaptive wireless system should be able to integrate well with modulation formats having varying bandwidth efficiency, as well as offer good protection to information blocks of varying sizes. To reach the first objective, we evaluate here only bit-interleaved-coded-modulation (BICM) methods. Both convolutional (CC) and parallel concatenated convolutional codes (PCCC), or turbo codes, are considered as outer codes. Code rates above ½ are generated by puncturing from the convolutional mother code of rate ½ having polynomials (133,171). The puncturing patterns are due to [Pro01]. Code rates of 1/2, 3/4 5/6 are considered. The PCCC constituent code has polynomials (1,5/7) and the resulting rate ½ is acquired by puncturing even and odd bits from the outputs of the first and second constituent encoder, respectively.
4.5 Modulation

A traditional single-carrier serial modulation is assumed for the wide-area uplink scenario with BPSK, QPSK and 16-QAM symbol mappings to implement different coding and modulation modes. The modulated transmission is filtered with a root-raised cosine pulse-shaping filter with roll-off 0.23.

The evaluated ACM modes with corresponding spectral efficiencies in terms of bits per channel use are listed in Table 4.1.

<table>
<thead>
<tr>
<th>Coderate/Modulation</th>
<th>BPSK</th>
<th>QPSK</th>
<th>16-QAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>½</td>
<td>½</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2/3</td>
<td>2/3</td>
<td>1 1/3</td>
<td>2 2/3</td>
</tr>
<tr>
<td>¾</td>
<td>¾</td>
<td>1 1/2</td>
<td>3</td>
</tr>
<tr>
<td>5/6</td>
<td>5/6</td>
<td>1 2/3</td>
<td>3 1/3</td>
</tr>
</tbody>
</table>

4.6 Detection

4.6.1 Frequency domain equalization

Receiver working conditions in broadband communications over frequency selective channel are inevitably complicated by the presence of inter-symbol interference. With OFDM this problem is mitigated by transforming a frequency selective channel into a set of frequency flat channels and attaching a cyclic prefix to a payload. This in turn allows for relatively simple signal detection, which can be achieved with per sub-carrier one-tap equalization. On the contrary, in case of serial modulation (single carrier transmission) one has to deal with the excessive amount of intersymbol interference. Correspondingly, efficient and low complexity signal equalization is one of the most critical issues. The elegant way of tackling this problem is that of frequency domain equalization [WIND2.1]. Block static channel assumption that is made in this project allows for cyclic data transmission. This in turn makes it possible to perform equalization on the FFT of the received signal instead on the time domain signal itself, as illustrated in the figure below. By performing such an operation, the equalization turns into a scalar multiplication per frequency bin, similarly as with OFDM. The originally transmitted signal is then recovered simply by an inverse FFT operation [WIND2.1].

![Figure 4.6.1 Frequency-domain linear equalizer](image)

The complexity of such an approach is comparable to that of OFDM. However, the major drawback of the scheme is that the inter-symbol interference in general can not be completely eliminated by the simple procedure described above. This can be elegantly solved by means of iterative (turbo) techniques, that are subject of Section 4.9.

4.7 Estimation

Estimation of frequency domain equalizer parameters for serial modulated systems, using training sequences, time-multiplexed with data blocks, was described in [WIND2.1], Section 3.4.2.2. Evaluation of this approach for WINNER channel models is described in Section 6.1.3. OFDM and other parallel modulated systems can use time-multiplexed or frequency multiplexed training symbols. Frequency multiplexed training symbols are pilot tones transmitted within each FFT block, which can be used to estimate the channel frequency response at the frequencies at which they are placed, and these estimates can be interpolated to estimate the entire block of frequencies (see Section 5.3).

We point out here that IFDMA (Section 5.2.1.3.2 - also called FDOSS), in which the data symbols are a Chu sequence [Chu72], can be used as an in-block pilot signal in serial modulated signals, occupying a subset of subcarriers, just as pilot tones are used in OFDM. In this case, the serial modulated signal is
generated in the frequency domain, leaving single frequency gaps in the spectrum, which are then occupied by the equally-spaced tones of an IFDMA signal, in which the data symbols are the elements of the Chu sequence, as illustrated in Figure 4.7.1. A Chu sequence has the property that both its elements and the FFT of its elements have constant magnitude. The constant magnitude property of its spectrum ensures that each pilot tone frequency will be estimated with a fixed-amplitude tone; the constant magnitude property of its elements, plus the serial modulation property of the IFDMA signal ensures that the IFDMA signal will have minimal peak to average power variation. The sum of the serial modulated data signal and the interspersed IFDMA pilot signal can be shown to have essentially the same peak to average properties as the sum of two serial modulated signals; which will be significantly lower than that of a comparable OFDM(A) signal with in-block pilot tones.

Figure 4.7.1  Spectrum showing placement of pilot tones in a serial modulated signal spectrum; red tones are IFDMA pilots; black tones are FFT components of data symbols,

The channel estimation methods associated with such an in-block pilot tone approach will be essentially identical in implementation and performance to those used with OFDM(A) systems.

Another important variation in channel estimation methods for serial modulation is iterative channel estimation, using soft information feedback. This is treated in Section 4.9.1.2.

4.8 Synchronization

It was pointed out in [WIND2.2], Section 2.1.1.7, that frequency offset and phase noise cause affect the receiver output in serial modulated systems in a different (and more easily-correctable) way than in parallel modulated systems like OFDM. Figure 4.8.1 illustrates the phase error at the output of a linear frequency domain equalizer in a serial modulation system, with a frequency offset equal to 10% of the inter-subcarrier spacing. It is an almost linear variation in time, with a slope proportional to the frequency offset. There are outliers on the ends, whose effects could be eliminated by not transmitting data on those few symbols. A simple decision-directed technique can easily predict this slow variation of the phase error, enabling it to be corrected [MM97].
4.9 Iterative Techniques

This section is intended to provide a brief overview and performance assessment of important iterative techniques employed in the detection of single-carrier modulated signals. These iterative techniques are chosen due to their common merit of being low-complexity and highly effective in minimizing the detection error to a very low level, an advantage that cannot be easily matched by conventional non-iterative technique. The main objective of this chapter is to stimulate further interest in these techniques and to facilitate their use in the WINNER project.

The iterative techniques considered here generally involve three main processes: (1) equalization, (2) decoding, and (3) channel estimation. During each iteration, these three processes are conducted separately while updated/refined information regarding the target signals are continuously generated and shared among these processes. The main challenge for the design of good iterative technique is to achieve low-complexity while optimising the performance by proper handling and generating the outputs required for the exchange of information between the processes.

The outline of this section is as follows. First, perfect channel state information (CSI) scenarios are studied. In section 4.9.1, two novel iterative techniques, namely the frequency-domain equalization with time-domain decision feedback (FDE-TDDF) and the frequency-domain equalization with MMSE interference-cancellation (FDE-MMSE-IC), are presented. Short descriptions of these techniques are included. Next, perfect CSI assumption is removed and channel estimation is considered. In section 4.9.2, FDE-TDDF with iterative channel estimation is presented. Simulations results for all the techniques are presented in chapter 6 (evaluation for wide-area uplink transmissions).

4.9.1 Turbo frequency domain equalization with perfect channel state information (CSI)

4.9.1.1 Iterative Frequency Domain Equalization with Time Domain Decision Feedback (FDE-TDDF)

In this section, perfect channel knowledge is assumed to be available and no channel estimation is considered. Figure 4.9.1 depicts the general structure of the iterative FDE-TDDF equalizer with channel decoding. The iterative FDE-TDDF consists of a forward filter operating in the frequency domain and a backward filter processing the feedback signals in the time domain. The outputs generated by the FDE-TDDF are in the form of extrinsic information (log likelihood ratio, LLR), while will be fed to the binary MAP channel decoder as the intrinsic information (a priori LLR). The outputs from the binary MAP decoder will be in the form of extrinsic information (LLR) and be used to calculate the soft decisions as the feedback signals for the FDE-TDDF. After a certain number of iterations (assuming convergence is
attained), the outputs from the MAP decoders, in the form of a-posterior probability ratio (APP), are utilized to generate hard decisions for the desired signals.

Mathematically, the FDE-TDDF can be described as follows. First, define the following notation,

- \( \mathbf{y} = [y(0), y(1), ..., y(N-1)]^T \), block of received signal samples, where \( N \) = block size
- \( \mathbf{x} = [x(0), x(1), ..., x(N-1)]^T \), block of input signals
- \( \mathbf{W}^{(i)} = \text{diag}(w_0^{(i)}, w_1^{(i)}, ..., w_{N-1}^{(i)}) \), forward filter coefficients matrix for the \( i \)th iteration
- \( \mathbf{G}^{(i)} = \text{circ}(g_0^{(i)}, g_1^{(i)}, ..., g_{N-1}^{(i)}) \), backward filter coefficients matrix for the \( i \)th iteration
- \( \mathbf{h} = [h_0, h_1, ..., h_{N-1}] \), frequency domain channel coefficients
- \( \mathbf{F} \) is the \( N \)-FFT matrix

The outputs of the FDE-TDDF for the \( i \)th iteration can be expressed as follows,

\[
\mathbf{x}^{(i)} = \mathbf{F}^H \mathbf{W}^{(i)} \mathbf{y} + \mathbf{G}^{(i)} \mathbf{x}^{(i-1)}
\]

(4.1)

where \( \mathbf{x}^{(i-1)} \) is the vector containing the soft decisions provided by the MAP decoder during the \( (i-1) \)th iteration. The detailed derivation of the forward and backward filter coefficients can be found in [NLF05]. The results are listed in below,

\[
w_k^{(i)} = \frac{h_k (1 + \rho^{(i-1)} \sum_{l=0}^{N-1} g_l^{(i)} \exp(-j2\pi lk / N))}{|h_k|^2 + \sigma^2}
\]

(4.2)

\[
\mathbf{g}^{(i)} = \left[ g_0^{(i)}, g_1^{(i)}, ..., g_{N-1}^{(i)} \right]^T = -\frac{\mathbf{V}^{-1} \mathbf{v}}{\rho^{(i-1)}}
\]

(4.3)

where \( \mathbf{v} = [v_1, v_2, ..., v_{N-1}]^T \), \( \mathbf{V} = \begin{bmatrix} v_0 & v_{-1} & \cdots & v_{-(N-2)} \\ v_1 & v_0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & v_0 \\ v_{N-2} & \cdots & \cdots & v_0 \end{bmatrix} \)

\[
v_k = \frac{1}{N} \sum_{l=0}^{N-1} \frac{|h_l|^2 (1/\rho^{(i-1)} - 1) + \sigma^2 / \rho^{(i-1)}}{|h_l|^2 + \sigma^2} \exp(-j2\pi lk / N)
\]

In deriving the backward filter coefficients, there is a simply way to avoid calculating the inverse \( \mathbf{V}^{-1} \), see [NLF05]. Next, the outputs fed to the binary MAP decoder, in the form of LLR, can be expressed as follows,

\[
L_e(j) = \frac{2x_j^{(i)} \gamma_j^{(i)}}{E(\epsilon_j^{(i)} \epsilon_j^{(i)} \bar{\gamma}^{(i)})}
\]

where \( \gamma_j^{(i)} \) and \( \epsilon_j^{(i)} \) are the useful signal’s amplitude and residual noise variance contained in \( \mathbf{x}_j^{(i)} \), respectively, see [NLF05] for the computation of these two quantities. The MAP decoder generates the LLR (denoted as \( \lambda_e(j) \)) as its outputs and they are converted to the soft decisions required by the FDE-TDDF.
\[ \hat{x}_j = \tanh(0.5\lambda_\epsilon(j)) \] (4.4)

4.9.1.2 Iterative Frequency Domain Equalization using MMSE-based Interference Cancellation Technique (FDE-MMSE-IC)

Figure 4.9.2 depicts the structure of the FDE-MMSE-IC scheme with channel decoding. The main difference between the FDE-MMSE-IC and the FDE-TDDF described in 4.9.1.1 is that the former contains only one forward filter while the backward filter is replaced with the channel coefficients modulator (which is used to re-create the ISI, using the symbol estimation from the decoder, so that they can be subtracted off from the received signals later). Unlike the FDE-TDDF, subtraction of ISI in FDE-MMSE-IC takes place in the beginning stage prior to the forward filtering in the frequency domain. The outputs (in the form of extrinsic information) from the FDE-MMSE-IC equalizer will be fed to the binary MAP decoder, which will in turn generate the soft decisions as the feedback signals for the FDE-MMSE-IC. For more details regarding the derivation of filter coefficients for the FDE-MMSE-IC, please see [YSS04] and the next section 4.9.1.3. And for other previous work using the same MMSE-IC approach in the iterative equalizer, either implemented in the time domain or frequency domain, readers are referred to [TKS02] [TH00].

![Figure 4.9.2: Iterative FDE-MMSE-IC equalizer](image)

4.9.1.3 Iterative Frequency Domain MIMO Equalization

The equalizer described in Section 4.9.1.2 can easily be extended to the detection of MIMO signals. The most straightforward extension assumes a spatial multiplexing transmission and detects each transmit antenna symbols separately. Using the common modulator framework notation from [WIND2.1], a serially transmitted signal \( x \) due to the modulated symbol vector \( a \) is given by

\[
\begin{align*}
\mathbf{x} &= \mathbf{GZ}_p \mathbf{V Fa} + \mathbf{GZ}_p, \\
& \quad \text{where} \\
\mathbf{Z}_p &= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \\
\mathbf{Z}_p &= \begin{bmatrix} \mathbf{z}_p \\ 0 \\ \mathbf{z}_p \end{bmatrix}
\end{align*}
\]

are the combined cyclic prefix/postfix and pilot symbol placement matrices, and \( \mathbf{G}, \mathbf{V} \) and \( \mathbf{F} \) denote the pulse shaping, frequency allocation and Fourier matrices, respectively. \( N_s \) and \( N_p \) denote the FFT dimension and the prefix length, respectively. The signal is transmitted through a frequency selective MIMO channel \( \mathbf{H} \), and the received signal after receiver filters, embedded in Gaussian noise, is given as
\[ y' = G^H H \bar{G} \bar{Z}_p a + G^H Z_p n. \] (4.8)

For the MIMO system using \( M_R \) receive and \( M_T \) transmit antennas channel matrix is a block matrix with \( M_R \times M_T \) blocks. The channels seen by a single receive antenna are stacked so that

\[
H_{(N_R+N_p) \times N_R} = \begin{bmatrix}
H_1 & \cdots & H_{m_R} & \cdots & H_{M_T}
\end{bmatrix},
\]

(4.9)

where the matrix for each transmit antenna is constructed of matrices for receive antennas as

\[
H_{m_T} = \begin{bmatrix}
H_{m_T}^T & \cdots & H_{m_T}^{m_R, m_T} & \cdots & H_{m_T}^{m_R, M_T}
\end{bmatrix}.
\]

(4.10)

To clarify the above structure, the channel matrix can be shown as

\[
H_{(N_R+N_p) \times N_R} = \begin{bmatrix}
H_{1,1} & \cdots & H_{1,m_T} & \cdots & H_{1,M_T} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
H_{m_T,1} & \cdots & H_{m_T,m_T} & \cdots & H_{m_T,M_T}
\end{bmatrix}.
\]

(4.11)

Each of the blocks describes a single-input-single-output multipath channel, whose matrix representation is given as

\[
H_{m_T} = \begin{bmatrix}
\bar{b}_{m_T,1}(1) & \cdots & \bar{b}_{m_R, m_T}(n_B) & \cdots & \bar{b}_{m_T, M_T}(N_B)
\end{bmatrix},
\]

(4.12)

where the channel response for each time instant between the each transmitter and receiver antenna is given as

\[
\bar{b}_{m_T, m_T}(n_B) = \begin{bmatrix}
0_{(N_R+N_p-N_R)}^T \\
h_{m_T, m_T}(n_B) \\
0_{(N_R+N_p-N_p)}^T
\end{bmatrix},
\]

(4.13)

where the channel coefficient vector is given as

\[
h_{m_T, m_T}(n_B) = \begin{bmatrix}
h_{m_T, m_T}(1) & \cdots & h_{m_T, m_T}(n_B) & \cdots & h_{m_T, m_T}(N_B)
\end{bmatrix}.
\]

(4.14)

When the cyclic postfix is removed with a removal matrix

\[
Z_p' = \begin{bmatrix}
0_{\dim(z_p)}^T \\
1
\end{bmatrix},
\]

(4.15)

the received signal can be given as

\[
y = Z_p' G^H H \bar{G} \bar{Z}_p a + Z_p' G^H Z_p n \\
= H \bar{a} + \tilde{n},
\]

(4.16)
where $H_c$ is a block circulant matrix containing the transmitter filter, propagation channel and the receiver filter. The noise vector $\vec{n}$ is the complex Gaussian with zero mean and covariance matrix $\sigma_0 G^H G$.

If the channel is static over the transmission block, a Fourier transformation block-diagonalizes the block-circulant channel matrix $H_c$ by

$$H_c = F^H H_c F,$$  \hspace{1cm} (4.17)

so that it becomes a $M_R \times M_T$ block matrix with diagonal blocks [Kay93].

With soft MMSE symbol estimates $\hat{a}_d$ from the decoder, interference cancellation and further MMSE filtering for the cancelled signal can be performed, similar to [TKS02] and [YSS04]. If the cancelled signal is given as

$$\tilde{y} = y - H\hat{a}_d,$$  \hspace{1cm} (4.18)

the estimate for the encoded symbols from antenna $m_a$ at the equalizer output is given as

$$\hat{a}_{m_a} = \beta_{m_a} \left( \gamma_{m_a} \hat{a}_{m_a} + H^H H + \sigma_0^2 G^H G \right)^{-1} \bar{Z}_{CP},$$  \hspace{1cm} (4.19)

where

$$\gamma_{m_a} = \frac{1}{N_c} \text{tr} \left( H^H \left( \lambda G^H H + \sigma_0^2 G^H G \right)^{-1} H_m \right),$$  \hspace{1cm} (4.20)

$$\beta_{m_a} = \left( 1 + \gamma_{m_a} (1 - \lambda) \right)^{-1}.$$

In the above, $\lambda$ denotes the average variance of the feedback, i.e.

$$\lambda = \frac{1}{N_c} \sum_{m_a=1}^{N_a} \left[ \text{tr} \left( H^H \left( \lambda G^H H + \sigma_0^2 G^H G \right)^{-1} H_m \right) \right] - \text{tr} \left( \left( \lambda \sigma_0^2 G^H G \right)^{-1} H_m \right).$$  \hspace{1cm} (4.21)

Based on the symbol estimate $\hat{a}_{m_a}$, a symbol de-mapper can extract the encoded bits using a standard MAP or max-MAP approaches.

**4.9.2 Turbo frequency domain equalization with channel estimation**

**Iterative Frequency Domain Equalization with Time Domain Decision Feedback and Channel Estimation (FDE-TDDF-CE)**

In this section, perfect channel knowledge is not available and channel estimation is required. We consider a pilot-aided channel estimation (PACE) method in conjunction with the FDE-TDDF described in section 4.9.1.1. In Figure 4.9.3, the proposed method, FDE-TDDF-CE, is shown. Basically, the FDE-TDDF-CE is identical to the original FDE-TDDF, only with the channel estimation as an additional process in each iteration. In the first iteration, the pilot signals are used to generate the estimated channel statistics (the mean and the variance of each random channel coefficient). The FDE-TDDF filters coefficients are updated accordingly taking into consideration the estimated channel statistics. For the subsequent iterations, the soft outputs from the decoders are used as the intermediate code symbols estimations, which are processed together with the pilot signals by the channel estimator to produce a new set of refined channel statistics estimations. The new channel information is then used by the FDE-TDDF in the next iteration and so on.
The channel estimation is carried out in the time domain based on the least-square criterion. The maximum channel length is assumed to be known or else it is assumed to be equal to the length of the cyclic prefix. In the WINNER scenarios, the latter assumption can be used, i.e., the receiver considers the channel length to be equal to the cyclic prefix.

In the following, it is assumed that one training block (contains only pilot signals) is used for each data block. Figure 4.9.4 shows the packet structure including the data symbols and pilot symbols,

![Packet Structure](image)

In this packet structure, the last $N_{cp}$ pilot symbols from the training block (which has length $M$) are used as the cyclic prefix. Before the first turbo iteration, only the training block is utilized for channel estimation. In the subsequent turbo iterations, the entire block (training + data + CP, with a total block size of $N + N_{cp} + M$) is utilized for channel estimation. It should be noted that the proposed channel estimation can be applied to any other packet structures.

Now, let $\mathbf{h}_r = [h_r(0), h_r(1), \ldots, h_r(N_{cp} - 1)]^T$ be the time domain channel response vector. Then, the channel estimates (which can be alternatively interpreted as the mean values of the random channel coefficients) are given by,

$$\hat{\mathbf{h}}_r = (\Omega^\mathsf{H} \Omega)^{-1} \Omega^\mathsf{H} \mathbf{y}$$

(4.22)

where $\Omega$ is a $(N + N_{cp} + M) \times N_{cp}$ toeplitz matrix with the first row given by the pilot symbols $[p(0), p(N_{cp} - 1), \ldots, p(1)]$ and the first column given by $[p(0), p(1), \ldots, p(M - 1), \hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{N-1}, p(M - N_{cp}), p(M - N_{cp} + 1), \ldots, p(M - 1)]^T$, where $\hat{x}_j, j = 0, \ldots, N - 1$ are the soft decisions of the data symbols obtained from the outputs of the decoder during the last iteration. And the estimation error covariance matrix can be approximated as
\[(h_i - \hat{h}_i)(h_i - \hat{h}_i)^H = \sigma^2(\Omega^H\Omega)^{-1}\]  

(4.23)

In this FDE-TDDF-CE scheme, the filter coefficients of the FDE-TDDF are updated in accordance with the feedback signals and their reliability \((\rho)\), as well as the newly estimated channel statistics. See [NLF05] for the derivation of the filter coefficients when channel estimation becomes part of the iterative process. The results are listed in below (the time domain channel estimates are converted to frequency domain),

\[w_k^{(i)} = \frac{\hat{h}_k (1 + \rho^{(i-1)}) \sum_{n=1}^{N-1} g_k^{(i)} \exp(-j2\pi lk/N))}{|\hat{h}_k|^2 + \sigma^2 + \Delta} \]  

(4.24)

\[\left[ g_1^{(i)}, g_2^{(i)}, ..., g_{N-1}^{(i)} \right] = \frac{V^TV}{\rho^{(i)}} \]  

(4.25)

where

\[v_k = \frac{1}{N} \sum_{n=0}^{N-1} \left[ |\hat{h}_k|^2 \sigma^2 + (\sigma^2 + \Delta)^2 + \left( \frac{1}{\rho^{(i-1)}} - l \right) \right] \exp(-j2\pi lk/N) \]  

(4.26)

\[\Delta = tr\left(\sigma^2(\Omega^H\Omega)^{-1}\right)\]  

(4.27)

### 4.10 Pre-distortion

#### 4.10.1 Power backoff required for typical solid state HPA’s

High power amplifiers (HPA) used in radio transmitters have nonlinear characteristics which can cause significant distortion to signals whose instantaneous power fluctuations come too close to the HPA’s output saturation power. Even small amounts of nonlinear distortion can cause the transmitted power spectrum to have undesirably high sidelobes, which can interfere with signals in adjacent frequency channels. Larger amounts of nonlinear distortion also cause significant nonlinear in-band self-interference, which results in increased received bit error rate. Normally, HPA’s are operated with a certain “power backoff” – which can be defined as the ratio of maximum saturation output power to a lower average output power. The larger the backoff, the less is the nonlinear distortion. However a larger power backoff means lower HPA efficiency. It also means that a more expensive HPA, with a higher maximum output power rating, is necessary to produce a given average output power.

Parallel modulated signals like OFDM, OFDMA and MC-CDMA consist of sums of modulated sinusoids in parallel, and hence have inherently larger ranges of instantaneous power than do serial modulated signals. Therefore, OFDM-like signals require larger power backoffs than do serial modulated signals [CND+00]. One of the following two possible consequences results, affecting the comparison of serial and parallel modulated signals:

1. If both types of systems are required to use the same type of HPA, and to have the same maximum spectrum sidelobe levels, the parallel modulated system will be forced to transmit with an average power which is lower than that of the serial modulated system by the difference in their respective required power backoffs. This means that the parallel system’s received power and \(E_b/N_0\) will be lower, and its bit error rate and coverage will be correspondingly less.

2. If both systems are required to transmit the same average power (e.g. at the maximum power allowed by regulation), the parallel system will have to have a higher-rated, and therefore more expensive power amplifier. The HPA is generally one of the most significant cost components of user terminals, and the relationship of HPA cost to maximum power rating is an important technology issue. However the cost can rise sharply with the output power rating, and is affected not only by the HPA device itself, but can also be affected by thermodynamics: provision of heat sinks, fans, etc [SG01].
4.10.2 Predistortion: methods to reduce instantaneous power variations

There has been much work on peak to average power ratio (PAPR) reduction methods for transmitted OFDM and other parallel modulated signal waveforms, including selective mapping, reference signal subtraction, and clipping and filtering. (See [WIND2.1], section 3.3 and references therein). As reported later in this document, many of these same techniques can also be used to advantage for PAPR reduction of serial modulated signals. Furthermore, generation of serial modulation by the generalized multicarrier approach is equivalent to generation of OFDM or OFDMA that has been precoded by a FFT operation. Thus serial (single carrier) modulation itself can be considered as just a form of OFDM to which a very effective form of PAPR has been applied.

4.10.3 HPA predistorter

It has been reported in many publications that HPA predistortion is a powerful technique of HPA nonlinearity compensation both for single- and multi-carrier signals. Short overview of predistortion methods has already been presented in [WIND2.1]. Among them the LUT predistorter with the variable gain is one of the most popular ones due to relatively simple structure. Figure 4.10.1 S shows the general scheme of the look-up table (LUT) predistorter based on the variable gain principle which is particularly well suited for digital implementation.

The baseband signal in form of the in-phase and quadrature components is converted into polar form. The signal instantaneous magnitude quantized with the necessary accuracy constitutes the address to two LUTs containing the magnitude and phase of the variable gain of the predistorter. Thus, each input signal sample is multiplied by the complex gain which in the predistorter adaptation process is selected to linearize the joint predistorter and HPA AM/AM and AM/PM characteristics. In the LUT-based predistorter each memory cell has to be modified in order to achieve HPA linearization. For signals having large dynamic range the size of LUT is relatively large which slows down the convergence process due to the fact that in order to adjust the predistorter characteristics, each memory cell has to be addressed at least a few times and adaptively modified. It is worth mentioning that in case of solid state amplifiers the AM/PM conversion is negligible, therefore the AM/AM LUT is sufficient.

![Figure 4.10.1 Scheme of the look-up table memoryless predistorter with variable gain](image)

It is shown in Section 6.5 of this document that even when a serial modulation is applied in the uplink, the predistorter is a circuit which brings a valuable increase in the system performance. For multi-carrier systems its meaning is even more important. Therefore the structure and adaptation algorithm of the predistorter has to be carefully considered. This is particularly important for the up-link in which a predistorter is a part of the mobile terminal. Thus, it should be possibly simple, consume small amount of power and should require small surface of the IC chip or a small number of instructions, if implemented in software.

Below we propose such a structure of predistorter and its adaptation algorithm which fulfills these requirements. Instead of a large LUT and its slow adaptation algorithm we propose to approximate the predistorter characteristic by a piece-wise linear curve.

Our proposal is a result of the observation that AM/AM characteristic of the HPA is quite smooth, so it can be well approximated by piece-wise linear function. The same is valid for the inverse characteristic. We conclude that a small number of of piece-wise linear components should be sufficient to well characterize both the HPA and predistorter characteristics. This would result in a small number of adjustable parameters and in consequence a relatively fast convergence could be expected.
First we select the piece-wise characteristic which compensates for the inverse HPA characteristic. In order to do this we divide the range of the input signal magnitudes into smaller \( M \) subranges. This way we choose the \( x \)-coordinates of the knees of the piece-wise linear function. The adaptation algorithm finds the best \( y \)-coordinates of these points such that the the mean square error is minimized:

\[
C = \sum_{k=1}^{M} \sum_{i=1}^{m_k} \left( A(y_k^{(i)}) - x_k^{(i)} \right)^2 = \sum_{k=1}^{M} \sum_{i=1}^{m_k} e_k^{(i)}^2
\]

where
\( x_k^{(i)} \) – the \( i \)-th sample of predistorter input signal belonging to the \( k \)-th subrange
\( y_k^{(i)} \) – \( i \)-th sample of the predistorter output signal belonging to the \( k \)-th subrange
\( A(.) \) – the HPA AM/AM characteristic.

Figure 4.10.2 presents the approximation of the AM/AM characteristic of the predistorter. At the given \( x \)-coordinates of the knee-points we adjust their \( y \)-coordinates to minimize the mean-square error on the output of the HPA. The error is given by the expression

\[
e_k^{(i)} = A(y_k^{(i)}) - x_k^{(i)}
\]

![Figure 4.10.2 Piece-wise linear AM/AM characteristics of the predistorter](image)

Let us recall that the two subsequent \( y \)-coordinates of the characteristics are described by the formulas

\[
y_k^{(i)} = \frac{y_k - y_{k-1}}{x_k - x_{k-1}}(x_k^{(i)} - x_{k-1}) + y_{k-1} \quad \quad \quad y_{k+1}^{(i)} = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}(x_{k+1}^{(i)} - x_k) + y_k
\]

In order to adjust the \( y \)-coordinates adaptively we calculate the gradient of the cost function \( C \). It is

\[
\frac{\partial C}{\partial y_k} = \sum_{i=1}^{m_k} 2e_k^{(i)} \frac{\partial e_k^{(i)}}{\partial y_k} + \sum_{i=1}^{m_k} 2e_{k+1}^{(i)} \frac{\partial e_{k+1}^{(i)}}{\partial y_k}
\]

Calculation of the partial derivatives in the above formula lead to the results

\[
\frac{\partial e_k^{(i)}}{\partial y_k} = \frac{\partial A(y)}{\partial y} \bigg|_{y=y_k^{(i)}} \frac{x_k^{(i)} - x_{k-1}}{x_k - x_{k-1}}
\]

and

\[
\frac{\partial e_{k+1}^{(i)}}{\partial y_k} = \frac{\partial A(y)}{\partial y} \bigg|_{y=y_{k+1}^{(i)}} \frac{x_{k+1} - x_k}{x_{k+1} - x_{k-1}}
\]

Using the following approximation of derivatives

\[
\frac{\partial A(y)}{\partial y} \bigg|_{y=y_k^{(i)}} \approx \frac{x_k - x_{k-1}}{y_k - y_{k-1}}
\]
we obtain the formula describing the predistorter adaptation algorithm for the \( j \)-th OFDM symbol:

\[
\frac{\partial A(y)}{\partial y} \bigg|_{y=y_{k+1,j}^{(i)}} \approx \frac{x_{k+1,j} - x_{j}}{y_{k+1,j} - y_{j}}
\]

where \( \alpha_k \) is the adaptation step of the algorithm and \( n_k \) and \( n_{k+1} \) denote the number of samples contained in the \( k \)-th range of the predistorter signal.

Let us note that the basic part of the iterative adaptation algorithm is the set of \( M \) correlators which correlate the HPA output error with the difference of the input signal and the \( x \)-coordinate of the left edge of the range into which the input signal sample falls.

In order to evaluate the quality of the proposed predistorter algorithm simulations were performed. Their results are shown in Figure 4.10.3. Although the example illustrates the OFDM case the serial transmission would result in similar plots. The OFDM signal was transmitted with 1664 16-QAM modulated subcarriers. The predistorter characteristics consisted of \( M=9 \) linear pieces. The \( y \)-coordinates of the knee-points were updated once per each OFDM symbol. As we see, 300 OFDM symbols are sufficient to achieve a good convergence of the predistorter.

![Figure 4.10.3 Power Spectral Density (PSD) on the output of the linearized HPA after operation of the proposed adaptation algorithm](image)

Concluding, we have proposed the implementationally efficient predistorter algorithm which does not require large LUT memory. Instead it employs a small set of the correlators and for each OFDM symbol practically all the knees of the piece-wise linear characteristics are adjusted.
5. Multi Carrier Transmission

5.1 Coding

The content of this section is intended to provide an assessment of the performance–complexity tradeoff related to different coding techniques envisioned for the use within WINNER. In [WIND2.1], Chapter 3 several well-known methods have been presented and discussed in detail. As a result of the assessment taken there, the following techniques will be the focus of our investigation:

- Convolutional Codes (CC): We use “standard” rate 1/2, memory 6 and 8 codes with generator polynomials \((133,171)^2\) and \((561,753)\), respectively, which have been in widespread use in different wireless communications systems, e.g., 3GPP. Higher code rates are obtained by appropriate puncturing for the specific code, based on the puncturing patterns from [Pro01]. All codes are terminated.

- Parallel Concatenated Convolutional Codes (PCCC, Turbo Codes): We employ codes with equal rate 1/2 constituent recursive systematic convolutional codes of memory 2, 3; using generators \((7,5)\) and \((13,15)\), respectively, with the first polynomial being the recursive one. Random or \((s,t)\) interleaving is used before feeding the information sequence to the second encoder and parity bits are alternately punctured at the output of the two encoders to obtain a rate 1/2 Turbo Code. Higher rate codes are created by using appropriate generator polynomials and puncturing more parity bits, following the specifications in [AcR99]. At least the first encoder is terminated.

- Low-Density Parity-Check Codes (LDPCC): The LDPC codes investigated for this project are irregular codes constructed with the PEG algorithm [WIND2.1] that ensures a high girth and, consequently, a low error floor. The node degree distributions for rate \(1/2\) have a maximum variable node degree \(d_{v,max} = 9\):

\[
\lambda(x) = 0.27684 x + 0.28342 x^2 + 0.43974 x^8
\]

\[
\rho(x) = 0.01568 x^5 + 0.85244 x^6 + 0.13188 x^7
\]

The node degree distributions for rate \(3/4\) have a maximum variable node degree \(d_{v,max} = 11\):

\[
\lambda(x) = 0.12000 x + 0.450977 x^2 + 0.407764 x^6 + 0.312588 x^{11}
\]

\[
\rho(x) = x^{15}
\]

Both are obtained from [Urb05] and were optimized with density evolution for the AWGN channel. However, they showed a good performance for all tested channel types.

The focus of the following investigations is on packet-oriented transmission, i.e., where one aims for error free transmission on top of the physical layer, after some acceptable delay. The existence of automatic retransmission techniques (Hybrid and/or conventional ARQ) is assumed, such that the target block error rate should be somewhere between 1 and 10%. The amount of text and figures in this section has been kept as low as possible, to give a short and concise overview of the findings. Further detail can be found corresponding appendix.

\(^2\) All generators are stated in octal notation.
5.1.1 Methodology

We are interested in assessing the performance and complexity of the proposed coding schemes. However, the results obviously depend on a large number of parameters (channel conditions, transmission technique, etc.). In order to limit the amount of evaluations and reach a certain independence of actual implementation considerations, only single-input single-output, narrowband transmission over following four frequency flat channels is considered (only temporal diversity is available):

- **AWGN Channel**: such channel conditions will for example prevail in the WINNER feeder link scenario, or other line-of-sight conditions with very poor scattering and mobility. It is also the baseline case for link adaptation scenarios, where the aim is to ensure that each addressed time-frequency bin (chunk) exhibits no time or frequency selectivity and the link rate is appropriately scaled to be close to the rate supported by the channel (bit and power loading).

- **Ergodic Rayleigh fading**: corresponds to the case of i.i.d. fading coefficients for each transmitted symbol, i.e., a fading channel with the maximum amount of diversity. However, it can serve as a good baseline comparison case for any system trying to mitigate fading by coding over a large number of fades, i.e., a system that does not have or does not exploit CSI at the transmitter to do fine-scale time-frequency link adaptation (slow link adaptation is of course sensible and should be employed). It can be expected that in environments with high frequency selectivity (i.e., long channel impulse responses), performance close to the figures for ergodic fading can be achieved.

- **Rayleigh fading channel, 5/20 fades/codeword**: a block fading channel with 5/20 independent fades per codeword, corresponding to a medium/low diversity environment. Such conditions will for example occur when coding over the whole OFDM symbol in a low diversity environment (near LOS) or when employing OFDMA as a multiple access scheme, where the diversity order per user might be substantially reduced, compared to the whole diversity present in the system.

Note that while the investigations made here only model temporal diversity, the results are equally applicable to systems where frequency is the dominant source of diversity, as can be expected in low mobility scenarios since from the coding perspective, only the amount of diversity present in the codeword is important, not its source. Time constraints did not allow for investigation of the influence of the chosen modulation alphabet on the performance of the different codes. All presented results in this document have been obtained with BPSK transmission. It is, however, expected that the relative merits of the different coding techniques are largely independent of the employed modulation alphabet. An investigation into this topic should be included in future work.

5.1.1.1 Parameters

Apart from the channel conditions, one factor that significantly influences code performance is the block size. A lower bound on this figure is the minimum scheduling unit, which in the WINNER context is denoted as a “chunk”. The chunk size in the wide area system has been fixed [WIND7.2] to 48 symbols for the wide area and 240 symbols for the short range case. The minimum information block size can hence be expected to be in the order of several tens of bits. An upper bound is hard to establish, but knowing that bit rates in the order of 100Mbps or even 1Gbps should be achieved, it is easily seen that this could be in the range of several thousand bits. We chose 5000 information bits as an upper bound, since code performance can be expected to not substantially improve beyond this point. The block lengths under consideration in performance evaluation were fixed to 50, 100, 250, 500, 1000, 5000 information bits.

Adaptive transmission will obviously require a number of different code rates to be available for link adaptation. In our investigations, we restricted ourselves mainly to the code rates 1/2 and 3/4, which have also been in the focus of the link level evaluations in the remainder of this document. Other code rates, such as 1/3, 2/3, 5/6, and 7/8 should be the focus of future investigations.

In order to enhance numerical stability and reduce complexity, decoding is often performed in the logarithmic instead of the probabilistic domain, leading to well known decoding algorithms: the Viterbi
algorithm for convolutional codes [Vit67], the logMAP algorithm for PCCC [RHV97] and the belief propagation algorithm for LDPCC. For PCCC and LDPCC, low complexity versions of these algorithms have been introduced, that avoid computationally complex non-linear functions: for instance the maxLogMAP and the MinSum algorithms, respectively. The non-linear functions may also be approximated by look-up-tables or linear interpolation. Details to all these algorithms are given in the appendix. We will also investigate the effect of some of these approximations on the decoding performance and complexity.

Last but not least, the performance gains in terms of lower required SNR depend on the target frame error rate. We investigate the cases 10% and 1%, with focus on 1% frame error rate.

5.1.2 Performance Evaluation

Figure 5.1.1 and Figure 5.1.2 show the exemplary performance of different coding techniques in terms of block error rate versus bit SNR on the AWGN and a Rayleigh channel with 20 fades, respectively. Curves for PCCC are given using different number of iterations and logMAP (logMAP: 1) or maxLogMAP (logMAP: 0) decoding. Details on the different decoding algorithms are given in the appendix. LDPC and PCCC clearly outperform convolutional codes in terms of performance. Performance is similar for PCCC and LDPC in the regime of interest with the highest complexity PCCC achieving performance equal to LDPCC under belief propagation decoding while the lower complexity variants show a performance close to that of the LDPCC under low complexity MinSum decoding.

Figure 5.1.1: Performance of different coding schemes on the AWGN channel for information block length 1000 bits. Red curves are for CC, green for LDPC, blue for PCCC. Dashed curves for PCCC are for the memory 3 code, solid for the memory 2 code.
As can be seen in Figure 5.1.2, relative performance of the different coding techniques is comparable on a fading channel with limited diversity. Gains from capacity-approaching codes are lower than on the AWGN. Note that the SNR gain from using capacity approaching codes is higher at lower frame error rates. We also expect LDPC to show better performance at low FER due to their very good distance properties compared to PCCC.

5.1.2.1 Influence of Channel Conditions

Figure 5.1.3 shows the performance of the investigated coding techniques, for code rate 1/2 on the AWGN channel. The Shannon bound for this setup is at 0.19 dB (BPSK signaling). For capacity approaching codes (PCCC and LDPC), the required SNR is steadily decreasing as we increase the block length. This is an expected result, as the theoretical limits predict a larger SNR offset to the Shannon bound for shorter information block lengths [DDP98]. The offset approaches zero only for block lengths in the order of hundreds of thousands of bits, which might be impractical for the application scenarios and target architectures considered in WINNER. Convolutional codes show exactly the opposite behavior: their performance decreases with block size. The reason is evidently that for a larger block length, the probability that a burst error occurs within one block is steadily increasing, and since iterative decoding is usually not applied to CC, block error probability hence scales with the block length. Using an additional outer interleaver and a Reed-Solomon outer code mitigates this problem [WIND2.1] at the expense of additional complexity. One may also use the channel and the detector as inner encoders and thus construct a scheme similar to SCC. However, the resulting decoding complexity will very likely be substantially higher than for the considered convolutional codes, as it scales with the number of detector (demapper) decoder iterations. If a convolutional code of shorter constraint length is used for such a scheme, performance and complexity can be expected to be similar to that of an equivalent PCCC under maxLogMAP decoding. Note that for the memory 2 code, the required SNR is actually higher at block length 5000 than at block length 1000, which is due to the fact that we used an (s,t) interleaver for the shorter block lengths and a random interleaver for information block length 5000. The memory 3 PCCC is obviously less influenced by this distinction.
The performance of LDPCC and PCCC is comparable for higher block lengths, however, PCCC appear to have better performance at shorter block lengths. The difference in performance between the high and low complexity variants of the decoding algorithms lies in the order of 0.5 dB, over a wide range of block lengths. It appears reasonable to use a PCCC with memory 2 or 3 and 8 iterations of maxLogMAP decoding to achieve good performance at acceptable complexity. It is very interesting to observe that for short enough block lengths, CC actually outperform PCCC and LDPC. One explanation for this behavior might be that one main assumption made in iterative decoding is violated for short block lengths: the independence between a priori information. This corresponds to a high number of rather short cycles in the message passing graph and will hence result in relatively low decoding performance.

The crossover point where capacity-approaching codes start to outperform CC is at an information block length of between 50 and 100 bits, for a memory 6 CC, and between 150 and 300 bits for a memory 8 CC. Note that at information block length 5000, the required bit SNR given a performance optimal code selection is made is roughly 1 dB lower than at block length 50 (LDPCC under belief propagation vs. CC memory 8), a loss has to be taken into account when assessing the merits of link adaptation schemes that usually code over very short blocks. The offset is, however, close to zero when only reduced complexity decoders can be applied for reasons of power consumption and/or processing power. The gain of capacity-approaching codes over a memory 6 convolutional code is 2.5-3.5 dB, at an information block size of 5000 bits. The respective loss is below 1 dB at an information block size of 50 bits. The gain of the memory 8 CC over the memory 6 CC is around 0.7 dB. Results are similar for ergodic Rayleigh fading (not shown, the required SNR is of course higher).

Figure 5.1.4 shows results for the same setup for a Rayleigh channel of diversity order 20. The relative merits of LDPCC over PCCC appear to be diminished in such environments, while the relative performance between CC and capacity approaching codes (and the crossover points) are similar to the AWGN case. The performance of LDPCC and PCCC is almost independent of block length. Results for a Rayleigh fading channel of diversity order 5 (not shown) are similar.
Figure 5.1.5 shows the required bit SNR for 1% block error rate at a code rate of 3/4, for the Rayleigh channel of diversity order 20. The performance differences between the investigated codes appear to be largely independent of the code rate. Differences between LDPCC and PCCC range between 0.5 and 1 dB. Relative merits of using capacity-approaching codes appear to decrease, while the difference between memory 6 and 8 CC increases, especially in fading channels with lower diversity order (for 5 fades per block, results not shown). This confirms the observations made in other parts of the deliverable: from a coding perspective, single carrier modulation is advantageous when using CC of higher code rates, since it provides the decoder with a constant SINR within one block, i.e., an AWGN-like channel.

Since good LDPCC can be constructed even for very high code rates, we expect that for even higher code rates, LDPCC will outperform the considered PCCC as the puncturing of parity bits in the PCCC reaches its limits. Duo-binary Turbo Codes [WIND2.1][BJD+01] are one way to increase PCCC performance for higher code rates, by using quaternary instead of binary constituent convolutional codes. Since the input code rate is higher, less redundancy bits have to be punctured to yield a given code rate. The performance gain for an information block length of 1000 bits is expected to be around 0.1 dB with respect to binary PCCC (please refer to the appendix for performance comparison), at frame error rates 1 and 10%. The decoding complexity is approximately 30% lower than that of a “standard” binary PCCC, since the decoder outputs two bits per trellis transition. The results presented in [Brat04] confirm that this is actually a better solution towards solving the puncturing problem for PCCC and that for very high rates, systematic bits should be punctured instead of parity bits. This question should be addressed in future investigations, since punctured codes are highly desirable as rate-compatible codes for hybrid ARQ schemes. Methods for puncturing LDPCC were described in [WIND2.1], but still have to be further evaluated.
Figure 5.1.5: Required SNR to achieve BLER 1% for different coding techniques at rate 3/4, as a function of block size on a Rayleigh channel with 20 independent fades

5.1.2.3 Influence of Target Frame Error Rate

Figure 5.1.6 shows the bit SNR required to achieve a frame error rate of 10%, which can be considered to be quite high, but still acceptable for a link featuring (hybrid) ARQ mechanisms. As could already be seen in the first performance plots, gains from PCCC and LDPC are somewhat lower in this regime of interest. However, the crossover points where these techniques become superior to CC remain the same and the SNR gain of capacity-approaching codes is still substantial (around 1dB at block length 1000, 1.5-2.5 dB at block length 5000).
5.1.3 Complexity Assessment

It is difficult to establish a meaningful absolute measure for the complexity of the considered algorithms, since different cost functions need to be applied when aiming for a silicon implementation:

- the chip area, related to the cost of the (end user) device;
- the energy required for execution of a single operation, related to the power consumption; and
- the time needed for execution of a single operation, i.e., the maximum achievable throughput.

The above costs are often summarized as the “ATE product” which is the target of the optimization in the implementation process. After such a general optimization, the chip can then be designed to meet the specific requirements, e.g., the achievable throughput, at the expense of higher chip area and/or power consumption. Since we cannot make reasonable predictions with respect to the development of the cost of silicon implementations, we will focus our attention on the two factors energy and time.

We can make the following statements, which should hold for a large class of implementation architectures (DSP/ASIC/FPGA). Classifying all operations required for the execution of a typical decoding algorithm according to their implementation simplicity, we find the following:

- **Simple arithmetic operations**: examples are addition, absolute value, sign, shift, multiplication with ±1, maximum, minimum, bit-wise operations. These can usually be implemented to induce only a single cycle delay and consume the least power.
- **Multiplication**: since typical algorithms for signal processing require a large number of multiplications, implementations are usually designed to have comparable or even the same delay for multiplication and addition³, which is achieved through highly parallel execution of the multiplication. However, the cost in terms of energy consumption (and area), a multiplication is roughly 10 times as complex as an addition [JDL+04].

³ Using UMC 0.13µm technology and the Synopsis Design Compiler (compile ultra / Synopsis Design Ware; no pipelining) one can for example achieve 2.5 ns delay for a 32 bit multiplier, for fixed as well as floating point operation; and 1ns and 3.8ns delay for a 32 bit adder in fixed and floating point, respectively. Note that in floating point, an addition is more complex than a multiplication, since it requires alignment of the two exponents.
- **Division, square root, and non-linear functions in general**: such functions are often implemented through iterative approximation algorithms, for example in application of the Newton theorem. The number of required iterations lies usually well below 10, depending on the required precision. Such operations are hence 4-6 times as complex as a multiplication [JDL+04], in terms of cycle count and also energy consumption.

<table>
<thead>
<tr>
<th>Simple Operation</th>
<th>Multiplication</th>
<th>Division</th>
<th>Square root</th>
<th>Non-linear function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative energy cost factor</td>
<td>1</td>
<td>10</td>
<td>40</td>
<td>50</td>
</tr>
<tr>
<td>Relative cycle count</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

**Table 5.1: Relative cost of different operations required for the execution of decoding algorithms**

Table 5.1 summarizes the above statements and is the basis of the complexity assessments given in the subsequent parts of this section and the details given in the appendix. It is easily seen that the cost in terms of energy and cycle count will be equal whenever we require only simple arithmetic operations for the execution of the algorithm. In the following, we will assess the complexity of different coding techniques, both in terms of cycle count and energy consumption, where the latter is mostly of interest for the user terminal, since power consumption is usually not a limiting factor at the access point.

### 5.1.3.1 Parallelization Issues

The maximum achievable clock rate for any implementation is essentially limited by the longest critical path in the chip. The number of cycles required for one run of the algorithm is on the other hand defined by the required arithmetic operations and the above figures. The quotient of these two figures defines the maximum number of times the algorithm can be executed per second. If this figure is insufficient to support the desired data rates (as may well be for the target data rates considered within WINNER), it is necessary to subdivide the available data into several blocks and let the signal processing algorithm run in parallel on different parts of the received data.

One major problem when following this approach for LDPCC and PCCC decoders is that during the exchange of message between the two constituent encoders (two BCJR for the PCCC, the variable node decoder and the check node decoder for the LDPCC), information must be interleaved in order to achieve good code performance. Since the width of the memory interface is generally limited and each “slice” of the parallel architecture is usually assigned a dedicated memory bank, the fact that the two constituent encoders work on different permutations of the information sequence (i.e., the exchanged messages) poses a serious problem to parallel architectures, since read-write operations must be scheduled such that no two “slices” access the same memory bank at the same time – otherwise read-write operations would have to be serialized, which would eat up large portions of the gain from parallelization. Recent results [TBM04] indicate, however, that this problem can be solved and that full parallelism can be achieved without the need of any ad-hoc interleaver design.

The calculations in check and variable nodes in the LDPC decoder are all independent of each other, making this algorithm highly attractive for parallel implementation. Exploiting the convergence behavior of Viterbi and MAP decoders, parallelization of these algorithms is also easily feasible at acceptable overhead (usually equaling the traceback length of the decoder).

### 5.1.3.2 Encoding

The encoding complexity of convolutional codes and Turbo codes is very low compared since they require only linear shift registers (one for CC, two for PCCC) and a few XOR operations to generate their output. Turbo Codes additionally require an interleaver before the second convolutional encoder. The number of XOR operations scales with the memory of the constituent code(s). For our investigated codes, 7 XOR operations are required per information bit for the memory 6 (133,171) convolutional code, and 10 for the memory 8 (561,753) convolutional code. The PCCC with memory 2 constituent CC requires 3 XOR operations per encoder, totaling 6 per information bit for the whole Turbo Code.
The encoding for LDPC has long been considered to increase quadratically with the codeword length $n$, since codes are usually created in non-systematic form, i.e., the right party of the parity check matrix is not lower triangular and parity bits cannot be calculated directly from the information bits. However, in [RU01] an efficient encoding method was presented, that has been described in detail in [WIND2.1]. The general strategy is to bring the parity check matrix of the LDPC into near-systematic form. The factor before the $n^2$ is thus dramatically reduced so that encoding complexity becomes almost linear and encoding even of long LDPC codes becomes practically feasible. The necessary modifications to the parity check matrix can be calculated offline.

The encoding process is summarized as follows algorithm [WIND2.1]: split the codeword vector $c$ (length $n$) into a systematic part $s$ (length $k=n-m$), a first parity part $g$ (calculated via dense rows in the parity check matrix), and a second parity part $p$ (calculated via sparse rows in the parity check matrix), so that $c = (s, g, p)$. Fill $s$ with the $(n-m)$ desired information symbols and determine $g$ and $p$ via back-substitution [WIND2.1]. For all codes used in these investigations, the number of bits $g$, that has to be encoded with $g$ dense rows, could be reduced to one. So the required number of XOR operations $o$ for encoding the whole codeword is determined as:

$$o = k + (m-1)(d_c - 1) = k + (\frac{k}{R-k-1})(d_c - 1) \approx k + (\frac{k}{R-k})(d_c - 1)$$

where $R$ is the code rate and $d_c$ is the average check node degree. The number of XOR operations per information bit $k$ is hence $1 + (1/R-1)(d_c - 1)$ which for the rate 1/2 code simply equals $d_c - s$ for the LDPC considered in the evaluations, we hence require 7 XOR operations per information bit for encoding. We conclude that the encoding complexity of CC, PCCC, and LDPC is comparable, and negligible in comparison to the decoding complexity.

5.1.3.3 Decoding

Since it only required simple operations (additions, max, etc), the cycle count and energy cost for the Viterbi are the same and have been estimated to $10D + 2$ cycles/energy cost per information bit where $D$ is the number of states in the trellis of the considered convolutional code (for details on the decoding algorithms and the derived complexity measures, the reader is referred to the appendix). Not relying on any iterative information exchange, this figure is obviously fixed for all investigated block lengths, channel scenarios, and code rates (remember that higher rates are obtained by puncturing the rate 1/2 mother code). The complexity scales linearly with the number of states in the trellis – a well known fact.

Table A-1 in the appendix summarizes the complexity for decoding one information bit in logMAP and maxLogMAP decoding. With $19D + 1$ the complexity of maxLogMAP decoding is about double that of Viterbi decoding, which is a result of the additional backward recursion and the calculation of soft instead of only hard outputs. This figure obviously has to be multiplied with 2 (for the two encoders) and the number of iterations. It also scales linearly with the number of states in the trellis. The number of iterations executes is usually fixed, however, there exist several approaches to reduce decoding complexity by examining the convergence behavior during decoding to stop the iterations (please refer to the appendix for details).

Table 5.2 summarizes the cost of decoding a single information bit, for different CC and PCCC codes and decoder setups for PCCC. The number of iterations for PCCC is assumed to be fixed; and the energy cost is based on the assumption that the max* operation is implemented via iterative approximation. Cycle count and energy cost are expected some be similar, when look-up-tables with a low number of entries are used instead. It is clearly visible that in terms of complexity, using Turbo Codes with very short memory and maxLogMAP decoding is highly attractive, compared to Viterbi decoding of CC with reasonable memory. Remembering the performance results from the previous section, we conclude that there is no reason to use convolutional codes for medium to large block lengths, even when taking into account decoding complexity.
Table 5.2: Cost for decoding an info bit, for different CC and PCCC Codes and decoder setups

<table>
<thead>
<tr>
<th>Code</th>
<th>Energy cost per info bit</th>
<th>Cycles per info bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC, Rate 1/2, Memory 6, Viterbi decoding</td>
<td>642</td>
<td></td>
</tr>
<tr>
<td>CC, Rate 1/2, Memory 8, Viterbi decoding</td>
<td>2'562</td>
<td></td>
</tr>
<tr>
<td>PCCC, Rate ½, Memory 2 constituent CC, logMAP decoding, 8 iterations</td>
<td>11'616</td>
<td>2'112</td>
</tr>
<tr>
<td>PCCC, Rate ½, Memory 3 constituent CC, logMAP decoding, 8 iterations</td>
<td>24'160</td>
<td>4'288</td>
</tr>
<tr>
<td>PCCC, Rate ½, Memory 2 constituent CC, maxLogMAP decoding, 4 iterations</td>
<td>616</td>
<td></td>
</tr>
<tr>
<td>PCCC, Rate ½, Memory 3 constituent CC, maxLogMAP decoding, 4 iterations</td>
<td>1'224</td>
<td></td>
</tr>
</tbody>
</table>

The complexity for the decoding obviously increases fourfold every time the memory of the code is increased by 2 bit – making the memory 8 CC quite unattractive in terms of complexity.

As a fundamental difference to other decoding algorithms, LDPC decoding checks its convergence behavior during decoding by testing the validity of the codeword in each iteration. While having the nice property that only as much complexity is invested as is required for successful decoding, this approach renders establishing absolute measures for the LDPC decoding complexity extremely difficult. Based on the evaluations in the appendix, where the complexity per information bit per iteration has been derived, we distinguish between two different cases: the average and the worst case complexity, where the former is determined by the average number of iterations at the target block error rate of 1% and the latter is determined by the maximum number of iterations allowed for decoding.

Table 5.3: Cost for decoding an information bit, for the considered LDpCC at information block size 1000, on AWGN and ergodic Rayleigh fading channels

<table>
<thead>
<tr>
<th>Code</th>
<th>Energy cost per info bit</th>
<th>Cycles per info bit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Worst case ⁴</td>
<td>Typical ⁵</td>
</tr>
<tr>
<td>Belief Propagation based on $\Phi$</td>
<td>45'800</td>
<td>10'992</td>
</tr>
<tr>
<td>Belief Propagation based on tanh/artanh</td>
<td>54'500</td>
<td>13'080</td>
</tr>
<tr>
<td>Belief Propagation, Linear Approximation</td>
<td>10'900</td>
<td>2'616</td>
</tr>
<tr>
<td>MinSum</td>
<td>2'900</td>
<td>696</td>
</tr>
</tbody>
</table>

Table 5.3 summarizes the average and worst case complexity for LDPC decoding, for a block length of 1000 information bits and assuming that a maximum of 50 iterations is enough to achieve reasonably good performance. This is confirmed by the results presented in the appendix. The figures for the average complexity are valid for AWGN and Ergodic Rayleigh fading channels and tend to be significantly lower (factor 2-4) in fading channels with lower diversity. It is obvious that if we consider the worst case complexity, LDpCC can hardly compete with PCCC in terms of decoding complexity. Only MinSum decoding will reach comparable complexity – but performance gains over PCCC will be minimal in this case. If we consider, however, the average complexity, LDpCC become highly attractive also in terms of decoding complexity, especially when MinSum decoding or Belief Propagation using linear approximations [RSB+05] is used (implementation of the $\Phi$ function via look-up tables is expected to yield similar results).

⁴ Assuming a maximum of 50 iterations for decoding.

⁵ For block length 1000, the average number of iterations is roughly 12 on AWGN and ergodic Rayleigh channels, for both message passing and MinSum decoding. On Rayleigh fading channels with lower diversity, the figure is significantly lower.
Note that complexity results for LDPC depend to a large extent on the ratio between average and maximum number of iterations, which for the coded presented here has been very large, due to them being optimized solely subject to decoding performance, not complexity. In the standardization process for 802.11n, different LDPC have been proposed which show reasonable performance already at a maximum of 20 iterations [TGn04], with the average number of iterations ranging from 10 to 15 at a block error rate of 1% on the AWGN channel. The proposed code is also systematic, i.e., linear time encodeable.

5.1.4 Decoding Complexity-Performance Trade-Off

We summarize the content of the preceding sections by depicting the performance and computational complexity of decoding for different coding techniques in a single plot. Figure 5.1.7, Figure 5.1.8 and Figure 5.1.9 plot the cycles (and the energy) required for decoding a single information bit over the bit SNR required to achieve out target block error rate of 1%. For LDPC, the average and maximum expected complexity is shown. Blue and red and markers indicate cycle count and energy consumption figures for the considered technique, respectively. Green markers illustrate that the two figures coincide, for all algorithms that require only simple operations (add, min/max, shift, etc.) for execution. The notation for Turbo Codes is “(mx, iy, lgMPz)” where x is the memory of the constituent convolutional codes, y is the number of iterations and z=1 indicates logMAP decoding whereas z=0 indicates maxLogMAP decoding.

![Figure 5.1.7: Complexity-Performance trade-off at block size 1000 bits, on the AWGN channel. Red markers indicate energy requirements, blue indicate cycle count. Green markers are for cases where both figures are equal (for algorithms requiring only simple arithmetic operations) ](image)

The region in the lower left of the diagram obviously corresponds to the best performance-complexity trade-off. Figure 5.1.7 shows results for the AWGN channel, at an information block size of 1000 bits. If we consider only the average complexity, the best trade-off is achieved by LDPC under MinSum or Belief Propagation decoding (depending on the performance requirements). If we consider the maximum decoding complexity, PCCC are the better choice, especially when using maxLogMAP decoding. It has to be stressed that one the one hand results for PCCC were obtained without using any stopping criteria for the iterations; and that LDPC were strictly optimized for performance, not complexity on the other hand. Decoding complexity can be expected to be somewhat lower if we allow for these optimizations. Convolutional codes are clearly unattractive in this regime, since the show rather low performance at almost the same complexity as PCCC under maxLogMAP decoding. The difference in cycle count and
energy consumption for capacity-approaching codes is roughly factor 5, for non-simplified decoding algorithms. In view of this fact, user terminals should probably employ reduced complexity variants of the decoding algorithms (using small look-up-tables) while the access point can run more sophisticated algorithms.

Figure 5.1.8 shows the results for information block length 50 on an AWGN channel. As already outlined, convolutional codes are to be favoured for such short block lengths, both in terms of complexity and performance.

![Figure 5.1.8: Complexity-Performance trade-off at block size 50 bits, on the AWGN channel](image_url)

Figure 5.1.9 illustrates the comparable performance of PCCC and LDPCC on Rayleigh fading channels with limited diversity. LDPCC outperform PCCC in terms of average decoding complexity and could thus be favored whenever a variable decoding delay is acceptable. When considering the maximum decoding delay, again PCCC should be favored.
Figure 5.1.9: Complexity-Performance trade-off at block size 1000 bits, on a Rayleigh fading channel with 20 independent fades per block

Also, the variance in decoding delay is relatively high for LDPCC (since the average number of iterations is reduced while the upper limit is retained). Designing LDPCC with very low variance in decoding delay at acceptable performance should be the focus of further investigations.

5.1.5 Conclusions and Further Work

The results presented in this section indicate the following:

- Convolutional codes with should be favored over PCCC and LDPCC for reasons of performance for short information block lengths, more specifically for block sizes up to 50 bits (for the memory 6 code) or 300 bits (for the memory 8 code).

- The gains of more advanced coding techniques (namely, the investigated LDPCC) over a memory 8 CC are in the order of 2.5 dB for information block lengths of 5000 bits and increase for larger block lengths (gains over the memory 6 CC are ~1 dB higher), for a wide range of channel conditions.

- For systems that can accept variable delay, decoding algorithms that use a variable number of iterations should be used in order to decrease power consumption and free resources for other tasks. LDPCC already include this feature and therefore appear favorable in terms of the performance-complexity trade-off under the current assumptions (namely, a constant number of iterations for PCCC). PCCC and LDPCC show similar performance and maximum decoding complexity, with PCCC outperforming LDPCC for short block lengths.

- Whenever some loss in performance is acceptable or power consumption is a critical factor, low complexity-variants of the standard decoding algorithms become highly attractive. Using maxLogMAP instead of logMAP decoding enables a factor factor 2 reduction in cycle count. Using MinSum decoding instead of belief propagation for PCCC results in a factor 3 reduction in cycle count. The loss in performance is usually below 1 dB for a wide range of channels. A substantial reduction in energy consumption is also achieved (factor 10-15). However, when look-up-tables are used for implementing the max* operation, energy consumption can be already reduced by a factor of around 5 (the figures for energy consumptions and cycle count are then approximately the same).
There are still some open issues that should be addressed in further investigations:

- The number of iterations for PCCC decoding has been fixed for the evaluations made in this deliverable. It should be investigated, at what additional complexity the convergence behavior of PCCC can be detected to limit the number of iterations and lower the total decoding complexity.

- Higher rate PCCC were obtained by puncturing parity bits. Future investigations should concentrate on concepts like higher rate constituent codes (duo-binary Turbo Codes) and puncturing of systematic bits.

- The LDPC investigated in this report were optimized strictly for performance and had a rather high ratio between the average and maximum number of iterations, which makes them less suited for practical implementations. Future code design should take this fact into consideration. As a first step, the codes proposed in [T Gn04] should be assessed in terms of their complexity-performance trade-off.

- The presented results enabled to make some general statements on the relative merits of different coding techniques, as a function of block size, code rate, and channel conditions. However, results could only be presented for a limited number of parameter combinations. In order to provide link-to-system interfaces with meaningful calibration data, appropriate performance measures should be established and code performance evaluated subject to these measures. The results presented in [WIN D2.7] are clearly a path that should be pursued.

### 5.2 Modulation

#### 5.2.1 Specificities of advanced multi carrier schemes

##### 5.2.1.1 Pseudo-Random Postfix OFDM (PRP-OFDM)

The PRP-OFDM modulation scheme has been studied in [MCD+03] and [WIN D2.1] (sections 4.6 to 4.10) for the single-antenna context and in [MDC+04] for the multiple-antennas context. The idea is replace the zero-sequence of ZP-OFDM [MWG+02] by a pseudo-randomly weighted sequence. The corresponding continuous modulator is derived as follows: the transmitted time domain signal \( u(t) \) of the PRP-OFDM modulator is the sum of the filtered data symbols plus the postfix sequence

\[
 u(t) = \sum_{k \in \mathbb{Z}} \alpha(k)p_c(t - kT_g) + \sum_{n=0}^{N_c-1} \tilde{\chi}(k)g_n(t - kT_g) \quad (5.30)
\]

The sub-channels shaping filters \( g_n(t) \) are typically chosen as rectangular functions weighted by a linear phase as defined in [MJ05], \( \tilde{\chi}(k) \) is the \( n \)th data carrier amplitude of the \( k \)th \( N_c \)-carriers OFDM symbol; the choice of the postfix sequence \( p_c(t) \) and its weighting factors \( \alpha(k) \) (a pseudo-random scalar known to both the transmitter and the receiver assuring spectral flatness) is discussed in [MCD04]. In the framework of this paper, all \( \alpha(k) \) are assumed to be a pure phase, i.e. \( \alpha(k) = e^{j\phi_c(k)} \). Concerning the postfix sequence, typically \( p_c(\tau) = 0 \) for all \( \tau \) with \( g_n(\tau) \neq 0 \) and vice versa. The block duration of a PRP-OFDM symbol including its postfix is defined to be \( T_g \).

As illustrated in Figure 5.2.1, the corresponding discrete modulator is obtained by adding the pseudo randomly weighted postfix sequence to the ZP-OFDM modulator outputs:

\[
 x_p(k) = F_Z^H \tilde{\chi}_{N_c}(k) + \alpha(k)e_c \quad (5.31)
\]

with \( P = N_c + D \), \( F_Z^H := \frac{1}{\sqrt{N_c}} W_{N_c}^* \), \( 0 \leq i < N_c, 0 \leq j < N_c \), \( W_{N_c} := e^{-j\pi N_c} \), \( F_Z^H := \left[ I_{N_c} \quad 0_{D,N_c} \right]_{p,N_c} \), \( e_c = \begin{bmatrix} 0_{N_c} \bar{c}_D \end{bmatrix}^T \) and \( e_D \) contains the postfix sequence samples. The expression of the received block is thus:
\[
\begin{align*}
y_p(k) = & \ H_{Pb} \left( F_N^H x_p(k) + \alpha(k) c_p \right) + n_p(k) = \ H_{Pb} \left( F_N^H x_p(k) + \alpha(k) c_p \right) + n_p(k) \\
& \quad (5.32)
\end{align*}
\]

Hereby, the channel matrix is defined as \( H_{Pb} = H_{ub}(P) + \beta_k H_{mb}(P) \) and \( \beta_k = \frac{\alpha(k)}{\alpha(k)} \). Note that \( H_{Pb} \) is diagonalized on a new basis which is different, but still similar to the Fourier basis [MCD+03]. The actual choice of the postfix sequence is discussed in [MCD04].

**Figure 5.2.1:** The PRP-OFDM transceiver chain.

[MCD+03] demonstrates that PRP-OFDM keeps all advantages of ZP-OFDM: different equalization approaches are possible in the receiver ranging from low complexity/medium performance (Overlap-Add based) to high complexity/high performance (MMSE based equalization). Also, symbol equalization is possible even if frequency domain channel nulls fall onto data carriers. Additionally to these features, PRP-OFDM allows simple channel estimation and tracking based on the deterministic postfix sequences: A first idea consists in exploiting that the OFDM data symbols are zero mean; after suitable weighting of the input samples with the inverse weighting factors \( \alpha(k)^{-1} \), a simple mean-value calculation is sufficient to extract the postfix sequence convolved by the channel [MCD+03]. The channel itself is extracted by deconvolution. [MCM+05] demonstrates that such an approach must be refined in practice if higher order constellations are used (QAM64 and higher). Since the postfix sequences are of same power and duration as the guard interval of CP-OFDM, a higher spectral efficiency can be obtained; in particular, the typical overhead in terms of learning symbols and pilot tones for CP-OFDM is avoided.

The disadvantage compared to CP-OFDM is a slight increase in calculation complexity in the receiver (if the low complexity Overlap-Add decoding approach is considered) [MCD05].

### 5.2.1.2 IOTA-OFDM

IOTA-OFDM belongs to the family of OFDM/OffsetQAM modulations [Hir81], a detailed description can be found in [WIND2.1] and [TR25.892]. Contrary to conventional OFDM, OFDM/OQAM modulations do not require the use of a guard interval, which leads to a gain in spectral efficiency. Although a guard interval is a simple and efficient way to combat multi-path effects, better performance can be reached by modulating each subcarrier by a prototype function [LGA01][JDL+04][MJ05]. To obtain the same robustness to the multi-path effects as OFDM with a guard interval, this prototype function must be very well localized in both the time and frequency domains [SR00]. The localization in time aims at limiting the inter-symbol interference and the localization in frequency aims at limiting the inter-carrier interference (e.g. due to Doppler effects).

The orthogonality between the sub-carriers must also be maintained after the modulation. Optimally localized functions having these properties exist but they guarantee orthogonality for real valued symbols.
only. An OFDM modulator using these functions is denoted OFDM/OffsetQAM. We can note that in OFDM/OQAM, each sub-carrier carries a real valued symbol but the density of the sub-carriers in the time-frequency plane is two times greater than in conventional OFDM, also called OFDM/QAM, with no guard interval. This means $\tau_0 V_0 = 1/2$, where $\tau_0$ denotes the OFDM/OQAM symbol duration and $V_0$ denotes the inter-carrier spacing. Thus, OFDM/OQAM has the same spectral efficiency as conventional OFDM with no guard interval. The transmitted signal can be expressed as follows

$$s(t) = \sum_{k \in \mathbb{Z}, n=0}^{N-1} a_{n,k} g_{n,k}(t)$$  \hspace{1cm} (5.33)

where $N$ is the number of sub-carriers, $a_{n,k}$ is the real valued symbol transmitted on the $n^{th}$ sub-carrier at the $k^{th}$ symbol; $g_{n,k}(t)$ denotes the shifted versions in time and frequency of $g(t)$, the real valued prototype function. Therefore the orthogonality condition among the sub-carriers is

$$\Re\left(\int_{-\infty}^{+\infty} g_{n,k}(t) g^{*}_{n',k'}(t) dt\right) = \delta_{n,n'} \delta_{k,k'}$$ \hspace{1cm} (5.34)

In discrete time, the rewriting of equation (3) leads to

$$s[i] = \sum_{k \in \mathbb{Z}, n=0}^{N-1} a_{n,k} g_{n,k}[i]$$ \hspace{1cm} (5.35)

with $g_{n,k}[i] = g((i-N/2)\pi/2) e^{j(n+k)\pi/2} e^{j2\pi i N}$. The demodulation is then performed by applying the real scalar product on the orthogonal basis of functions.

$$\hat{a}_{n,k} = \Re\left(\sum_{i} g^{*}_{n,k}[i]s[i]\right)$$ \hspace{1cm} (5.36)

A particular prototype function called IOTA (Isotropic Orthogonal Transform Algorithm) [LBA95][SR00] satisfies the orthogonality condition. To simplify the notations, we call IOTA-OFDM an OFDM/OQAM system using the IOTA function.

Thanks to the removal of the cyclic prefix, IOTA-OFDM leads to a gain up to 25% in spectral efficiency compared to CP-OFDM. IOTA-OFDM also benefits from a steeper spectrum than CP-OFDM thanks to the good localization in frequency of the IOTA function [WIND2.1]. Therefore, in IOTA-OFDM a larger number of sub-carriers can be modulated within a same spectrum emission mask. The disadvantage of IOTA-OFDM compared to CP-OFDM is an increase of the complexity, but this additional cost can be strongly reduced if a filter-bank approached is used, as shown by Figure 5.2.2.

**Figure 5.2.2: The IOTA-OFDM transceiver chain.**

---

6 In a practical implementation of an OFDM/OQAM transceiver, $\tau_0 = T_p / 2$ where $T_p$ represents the duration of the useful part of a conventional OFDM symbol duration (i.e. without the cyclic prefix)
5.2.1.3 MC-CDMA

This technique has already been largely described within two previous deliverables [WIND2.1], and [WIND2.6]. Therefore in this section we will focus on basic system scheme employed in our simulations. The Figure 5.2.3 hereafter describes the transmission scheme studied.

![Figure 5.2.3: OVSF MC-CDMA Transmitter Scheme](image)

The bit stream (after bit interleaving) is mapped (1) onto multi-level constellation M-QAM (with Gray labelling here), then these symbols \( X' \) are spread with variable length \( L \) Walsh-Hadamard (WH) sequence (2+3). The spreading is processed only in the frequency domain, and each chip code \( C_k \) is mapped onto one subcarrier. The IFFT is then applied (4) as in other Multi-Carrier transmission schemes.

The number of constellation symbols carried by each OFDM symbol, follow then the following constraint relation, involving the available data subcarrier \( N_{subc} \):

\[
N_{subc} = N \cdot L
\]

It can be noted here that MC-CDMA concerns essentially the multiuser access and can a priori be mapped on all multi-carrier techniques (CP-OFDM, PRP-OFDM, IOTA-OFDM) evaluated in this document.

Although WH codes are used based on their null cross-correlation, Multiple-Access Interference (MAI) will arise from the channel fading selectivity. Thus a trade-off has to be found between diversity and MAI robustness. Indeed, on the one hand diversity takes advantage from spreading and the channel selectivity (inversely proportional to the Delay Spread), but in other hand, channel frequency response distorts the chip codes, and thus the cross-correlation among users sequences.

We will first focus on achieving frequency diversity. For this purpose, simulations with single user case (optimal bound) have been carried out with and without frequency interleaving in order to emphasize the impact of the coherence bandwidth on performance results over different wireless channels. The reference algorithms from Single User Detection (SUD) have been implemented: Equal Gain Combining (EGC), Maximal Ratio Combining (MRC), Orthogonal Restore Combining (ORC), and Minimum Mean Square Error (MMSE). All details about these algorithms can be found in [WIND2.1]-[WIND2.6]-[Kai98].

The principal constituent processing blocks are described within the following modular transceiver chain, on Figure 5.2.4.
Preliminary results, SUD comparison with single user case, Downlink:

The single user case, evaluated in Figure 5.2.5, will serve as an optimal bound for performance that the transmission link can reach. Indeed, we will see later on that the behaviour of such algorithms is modified by the MAI.

It is worthy to notice from these results, obtained on Channel C, that EGC leads to slightly same results than MRC, while the ORC (ZF Equalization) has predictable poor performance resulting from the noise enhancement.

Interleaving Impact:

While using spreading, the diversity gain can only be fully reached if and only if each chip experiences uncorrelated fading. In our case, that means each subcarrier carrying one chip code should be
uncorrelated from the others. Due to the coherence bandwidth (inversely proportional to Delay Spread), this might not be the case.

The choice as well as the design of the frequency interleaver is thus of high interest in such scheme. For our scenario, we have chosen uniform random interleaver. As we can notice on the results below (Figure 5.2.6), the performance is slightly the same over the 3 channels. That means the interleaving process managed to reach almost the same diversity amount. Nevertheless, for higher Spreading Factor (SF), the delay spread of each channel still impact the performance. That means that some improvement might be found within the interleaving scheme.

![Figure 5.2.6: Impact of interleaving on performance](image)

**5.2.1.3.1 Preliminary results on coding-Spreading Trade-off in MC-CDMA**

**Impact of the Coding Scheme:**

Then spreading can be seen as a repetition code, therefore the trade-off between spreading and coding can be studied. For this purpose, the system will use the standard de-facto (NASA) Convolutional Coding (CC), R=1/2, G=[133, 171], with 64 states (memory 6) together with Rate Compatible Puncturing Code (RCPC), by applying puncturing on the previous mother code (cf. Figure 5.2.6). This channel coding will act as outer code, serially concatenated with the spreading (inner code).

![Figure 5.2.7: OVSF MC-CDMA SUD with Channel Coding Transceiver chain](image)
For this study we will use the following standardized puncturing matrices, leading to respectively rates R=1/2, R=2/3, and R=3/4:

\[
Puncturing_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad Puncturing_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad Puncturing_3 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.
\]

N.B: We have now to keep in mind that all assessment have been obtained for only 1 OFDM symbol based frame. That means that the size of the packet information is quite small, and thus the bit interleaving depth as well. This is sure to impact the performance while using Convolutive coding with small packet size. Such trend can be related with the channel coding chapter where impact of the packet size has been evaluated with respect to several candidate coding schemes.

While studying the channel coding concatenated with the spreading ([Kai98]-[BB04]), we need also to study the decoding process. Indeed, while tackling enhanced iterative cancelling (IC) techniques, we might need to pass soft information through the feedback loop. We thus start as reference, to study the decoding with hard decision decoding.

**QPSK Hard decision decoding**

Once again, as optimal reference performance bound, we give hereafter the single user case.

The coding gain depends on the Spreading Factor (SF) used, and by increasing the SF, the channel coding gain remains constant. This was expected as with higher SF, main frequency diversity is already recovered, and as the standard deviation of the resulting SNR (in output of detector) is decreased, the channel coding rate will have less and less impact.

Finally in the case studied above (Figure 5.2.8), increasing SF higher than 8 doesn’t lead to any noticeable enhancement when used in conjunction with the CC Channel coding [Kai98].

The system under consideration would use higher order modulations. Therefore in order to give some other performance references, the 16-QAM case is studied in Figure 5.2.9.

**16-QAM Hard decision decoding**
Figure 5.2.9: 16-QAM, Hard decision decoding performance over channel C

Robustness of the coded MC-CDMA with respect to the load (MAI)
After having given these single user case for reference ultimate bound, we study the impact of multiple users on the coded transmission.

Eventhough the MRC detection algorithm is optimal for the single user case, it is well known that its sensitivity to MAI is degrading the performance of the link. Hereafter the results (QPSK) are given with respect to the load (expressed in terms of total number of users), the SF, and the coding rate as well. (The decoding is still hard decision driven).

MRC:

Figure 5.2.10: Sensitivity of MRC w.r.t MAI, with hard decision coding
Once again, the EGC algorithm outperforms the MRC detector as its single user case performance is slightly the same, and its robustness to MAI is better, together with a reduced complexity.

**Impact of the Soft Information with Coding**

In order to prepare the framework for advanced Interference Cancelling (IC) algorithms, we need to investigate the enhancement resulting from soft decision Viterbi decoding.

We thus implement the usual soft symbol demapper [TB01], for M-QAM constellation with Gray labelling, relying on MAX-Log-MAP decoding. These simplified bit LLRs are given for example for the 4-QAM, and 16-QAM case hereafter:

With \( X=\text{real part}, \) and \( Y=(-1)\times\text{Imaginary part}, \)

\[
\Lambda = \begin{cases} 
\Lambda = X \\
\Lambda = -X \\
\Lambda = Y \\
\Lambda = -\left[2 \cdot a - \{X\}\right] \\
\Lambda = -\left[2 \cdot a - \{Y\}\right]
\end{cases}
\]

This resulting real valued bits need then to be weighted by a dedicated function, taking into account the detection process, the number of interferers, as well as the channel fading.

Indeed, for short spreading factor length we will then apply the following weighting function (Eq.4-26) from [Kai98]:

\[
2 \left| \sum_{i=0}^{L} G_i \cdot H \right| (K-1) \left[ \frac{1}{L} \sum_{i=0}^{L} |G_i \cdot H_i| - \frac{1}{L} \sum_{i=0}^{L} |G_i| \sum_{i=0}^{L} |H_i| \right]
\]

\[
+ \frac{\sigma^2}{2} \sum_{i=0}^{L} |G_i|^2
\]

We will first give assessment in the single user case, thus \( K=1, \) and the relation shrinkens to the following:

\[
\frac{4}{\sigma^2} \sum_{i=0}^{L} |G_i \cdot H_i| (5.40)
\]

For larger spreading factor it has be noted that this term is quasi constant, and this weighting has almost no impact, so we can afford to keep the soft demapped bits.
**QPSK with Soft decision decoding.**

The enhancement due to the soft decoding is quite noticeable on the following results given for QPSK and single user case, over channel C:

The gain obtained with such soft information demapping and reliability estimate, is between 3 dB and 2 dB with respectively SF=2, and SF=16.

As it has been underlined in Eq.4-27 from [Kai98], for SF much greater than 2, the reliability estimate can actually be replaced directly with the soft demapped symbols. We will emphasize this phenomenon with the following results, obtained with 16-QAM Gray labelled constellation.
Impact of the soft information estimate

![Impact of the soft information estimate](image)

Figure 5.2.13: Impact of the Soft Information estimate on 16-QAM performance, over channel C

As it can be noticed on these curves, the difference between weighting with Eq.5.39 the soft demapped bits or not is essentially noticeable for SF=2, and SF=4. The difference is around 2 and 1dB respectively. Then for higher SF, both curves are slightly matching, and thus weighting becomes useless.

Interference Cancelling

In order to cope with the MAI, some specific Multi-User Detection (MUD) techniques exist. These can be classified in two groups: linear or non-linear. We will focus on the non-linear cases, in order to provide some inputs to other Tasks, taking into account coding schemes.

Within this non-linear family, we can find Serial-Interference Cancellation (SIC) techniques, or Parallel Interference Cancellation (PIC) algorithms. Roughly, the former technique starts by detecting the data from the strongest interferer, then cancel the newly estimated interfering signal from the received signal. Iterations with respect to decreasing received power interferers would then converge towards a cleaned signal. The main drawback is the delay.

Due to the nature of MC transmission schemes, PIC schemes are preferred. In this scheme, estimation, MAI regeneration and cancelling are done in parallel for each user. Figure 5.2.14 illustrates how these different stages are performed, together with emphasizing the role of the decision process, in this case hard decision.
The received signal (after OFDM demodulation) is fed into the PIC Unit, and the newly estimated MAI for each user is available at the same time (parallel processing of partial summation), and these MAI estimates can be weighted ([YYS00]-[SR98]) before being subtracted from the reference signal. The introduction of such Cancellation Factor filters the successive MAI estimation through the multiple stages, by applying a kind of reliability weighting.

The two algorithms can be implemented with multistage schemes (Figure 5.2.15). In this case, iterating the interference cancelling can sometimes lead to some divergence behaviour in the case of the PIC. Indeed, due to inherent iterative nature of these algorithms, the reliability of the MAI estimate is quite crucial. And error propagation phenomenon known as ‘ping-pong’ effect can be observed ([Bro05]-[RO03]).

The MAI reliability will then depend directly on the combination decision stage and detection. For instance, MRC detection is usually employed after a first IC iterate, as its sensitivity to MAI is too important, and one less sensitive scheme such as EGC, or ORC (decorrelator) detection is preferred for the first stage, before the IC.

On the following Figure 5.2.16, we give results for such kind of combinations. In the current case, a full load with SF=16 is considered in the Downlink. We have implemented a 3-staged PIC, where each IC stage can have a detection scheme varied among MRC, EGC and MMSE. Moreover the 4th final detection stage is fixed to be MRC.
In the present results, the 1st stage is chosen as IC with MMSE detection, as this scheme is the more robust in presence of high MAI. Then we can notice the sensitivity of the BER depending on the further stage detection algorithm. Especially, the choice of EGC for the 2nd stage seems to be optimal, non only in term of complexity reduction, compared with MMSE, but also in term of performance, and error propagation limitation.

Moreover, it has to be noticed that a 2dB performance enhancement can be reached with a 3rd IC staged PIC, even with hard decision.

The same kind of combination study can be performed with a lower SF=8, still full load, and presented in Figure 5.2.17. It is quite clear that as long MMSE detection is performed within the cascaded IC stages, final performance is improved, whatever the stage can be. So once again a 1st stage MMSE PIC detector should be quite sufficient, even if 2 stages MMSE can give a 2dB improvement.

The sensitivity of the staged PIC is due to the reliability of the MAI. This can be estimated by means of Soft Information algorithms. For this case, we will employ the CC previously proposed, whose Soft Outputs are given by APP module with Max-Log-MAP. In order to re-use this Soft Information for accuracy estimate, and weighting of the MAI (Soft-MAI), we have to use the decoding scheme in combination with the IC.
Further studied are thus planned to investigate deeply the impact of the coding scheme (Convolutional Code, Turbo-PCCC, LDPC) within the IC loop, onto the multistage iterative MUD, together with the sensitivity of such advanced coding schemes to the MAI with full load.

5.2.1.3.2 Downlink Layered Multicell MC-CDMA

In the section, the performance of the downlink single and multi-cell multicarrier systems with different channel coding options (single antenna turbo coding and multiple antenna space-frequency turbo coded modulation (SFTuCM) [Tuj03]) and receiver structures is evaluated.

System Model

MC-CDMA is taken to show the system model since with spreading factor=1, MC-CDMA reduces to OFDM system. In Figure 5.2.18 a downlink MC-CDMA cellular system is shown where \( K \) users in each cell are considered. The desired user is located to the central cell. The same transmission parameters are used in all base stations and for all users. It is assumed that the MIMO MC-CDMA system has \( N_c \) subcarriers, \( N \) transmit antennas and \( M \) receive antennas. The input data bits are encoded by the conventional turbo coding (TC) or by SFTuCM which is designed for two transmit antennas [Tuj03]. We divide the \( N \) transmit antennas available in the system into independent layers denoted as \( J = N / J_0 \), where \( J_0 \) is the number of the antennas associated for the encoder. \( J_0 = 1 \) in single antenna turbo code and \( J_0 = 2 \) in SFTuCM. Vertical layering is applied, where the encoded and modulation mapped symbols are multiplexed for \( J \) transmit antenna groups.

\[
\sum_{i=\infty}^{\infty} s_n^k(t) = \sum_{i=-\infty}^{\infty} \sum_{p=1}^{P} \sum_{g=1}^{G} x_{s,o}^k(i) \delta^k_p(t-iT_s) \exp(j2\pi(g+pP)\Delta f(t-iT_s)), \tag{5.41}
\]

where \( \Delta \) is the guard interval \( T_s \) is the symbol duration, \( \Delta f = 1/(T_s - \Delta) \) is the minimum subcarrier separation, and \( p_s(t) \) is the rectangular pulse waveform. A combination of the cell specific scrambling code and spreading code sequence is denoted as \( \delta^k_g \). No subcarrier interleaving is used since powerful channel coding is always employed to obtain the frequency diversity. Moreover, by placing the spreading chips at adjacent subcarriers, the multiple access interference will be reduced due to the fading correlation.
The received signal consists of the desired signal, multiple access interference (MAI), inter-cell interference and Gaussian distributed thermal noise. After OFDM demodulation, the equivalent frequency domain received signal of the MIMO system at \( p \)th code symbol interval can be rearranged to form

\[
r_p = C_{p,q} x_{p,q} + \sum_{q=1}^{6} C_{p,q} x_{p,q} + \eta_{p},
\]

where \( C_{p,q} \) is the equivalent channel-spreading matrix from \( q \)th base station to central cell. Let us consider the received signal from the base station of the desired cell \( q = 0 \). To simplify notation, after omitting \( p \) and \( q \), the received signal vector, transmit symbol vector, and noise are written as

\[
\mathbf{r} = \left[ r_1^1, \ldots, r_k^G, \ldots, r_N^G \right]^T \in \mathbb{C}^{G \times M}, \quad \mathbf{x} = \left[ x_1^1, \ldots, x_k^K, \ldots, x_N^K \right]^T \in \mathbb{M}^{K \times N}, \quad \eta = \left[ \eta_1^1, \ldots, \eta_k^G, \ldots, \eta_N^G \right]^T \in \mathbb{C}^{MG},
\]

The channel--spreading matrix can be presented as

\[
\mathbf{C} = \left[ \mathbf{c}_{1,1}, \ldots, \mathbf{c}_{K,N} \right] \in \mathbb{C}^{MK \times NK},
\]

\[
\mathbf{c}_{k,n} = \left[ \mathbf{c}_{k,n}^1, \ldots, \mathbf{c}_{k,n}^G \right]^T \in \mathbb{C}^{MG},
\]

\[
\mathbf{c}_{k,n}^G = \left[ \mathbf{H}_{n,n}^G d_k^1, \ldots, \mathbf{H}_{n,n}^G d_k^G \right]^T \in \mathbb{C}^G,
\]

where \( H_n^G \) contains Rayleigh fading channel coefficient, path-loss and shadowing between transmit antenna \( n \) and receive antenna \( m \) at subcarrier \( g \). Note that for a certain received symbol \( r_p \), the subcarrier index is \( (p-1)G + g \) due to the assumption of quasi-static fading. Since the received signal is multiplied by the desired cell scrambling code, \( \mathbf{C} \) includes spreading codes only.

**MMSE Detection in MIMO MC-CDMA**

The MMSE based detectors for multiple antenna MC-CDMA receivers used herein are based on the techniques presented in [VTL04] [VTL04a], where the detailed description of the receivers can be found. We describe the receiver structures briefly. Furthermore, a slight modification to take inter-cell interference into account is presented.

**Linear Symbol-Level MMSE Detector**

In the joint space-frequency MMSE detector, the coded symbols \( x_0 \) is estimated over subcarriers and all transmit antennas. We assume that the channel of the desired cell is perfectly known at the receiver. The MMSE criterion based detector for the MIMO MC-CDMA system is

\[
\mathbf{W} = \left( \mathbf{C}_y \mathbf{R}_s \mathbf{C}_0^H + \mathbf{R}_{int} + \mathbf{R}_{\eta \eta} \right)^{-1} \mathbf{C}_0 \mathbf{R}_s
\]

where the matrix filter \( \mathbf{W} \in \mathbb{C}^{MG \times NK} \) is able to estimate the coded symbols for all users. The symbols are assumed to be uncorrelated between different antennas and subcarriers so that \( \mathbf{R}_s = E_s \mathbf{I}_N \), where \( E_s \) is the average power of the transmitted symbols. Fast fading of the inter-cell interference channels is not known and interference is assumed to be spectrally and spatially white when \( \mathbf{R}_{int} = N_{int} \mathbf{I}_{MG} \), where \( N_{int} \) is the average power of the interference defined by the path loss and shadow fading. Thermal noise term is \( \mathbf{R}_{\eta \eta} = N_\eta \mathbf{I}_G \).

The input for the symbol level space-time maximum a posteriori (MAP) based turbo decoder is the soft output of the MMSE filter written as \( \hat{x} = \mathbf{W}_x^H \mathbf{r} \) with an equivalent channel and noise model of estimate \( \hat{x} \in \mathbb{C}^N \). The residual interference in the output of the multiuser MMSE detector can be approximated to be Gaussian distributed. The equivalent system model for the decoder is

\[
\hat{x} = \mathbf{W}_x^H \mathbf{r} = \mathbf{\Omega}_s x_k + \varphi_k
\]
where the residual inter-cell interference-MAI-plus-noise term $\varphi_z$ is considered to be Gaussian distributed.

**Linear Chip-Level MMSE Detector**

The spreading codes of all user's must be available in the joint symbol-level MMSE multiuser detector. Spatial filtering and chip combining can be separated whereupon only the desired user signature is needed. Also the inversion of the $MG \times MG$ matrix is reduced to an inversion of a $M \times M$ matrix when a significant save of the complexity is achieved in the chip-level receiver over the symbol-level one. The system model for the receiving signal at the $g$th subcarrier of the $p$th symbol can be written as

$$r^g = H^g_0 z^g_0 + \sum_{q=1}^N H^g_q z^g_q + \eta^g,$$

(5.51)

where $r^g \in \mathbb{C}^{M_t}$, $H^g_q \in \mathbb{C}^{M \times N}$ is the channel matrix and $\eta^g \sim N(0, N_0 I_M)$ is Gaussian noise. The symbols $z^g = [z^g_1, ..., z^g_L]$ to be detected are $z^g_n = \sum_k d^g_k x_{n,k}$, $n = 1, ..., N$. The filter matrix for the $g$th subcarrier of the $p$th

$$W^g = (H^g_0 R_{zz} H^g_0 + R_{inter} + R_{yy})^{-1} H^g_0,$$

(5.52)

where $R_{zz}$ is the covariance of $z^g$ and $R_{inter} = N_{inter} I_M$. We assume that the spreading codes are random and the coded symbols are independent between antennas and users. Consequently, $R_{zz} = E_x (K / G) I_N$. Symbol estimate for user $k$ in antenna $n$ is got by simple combining of $G$ detected chips so that

$$\hat{x}_{k,n} = \sum_{g=1}^G d^g_k \hat{z}^g_n,$$

(5.53)

where $\hat{z}^g_n$ is $n$th element from output of the spatial filter $\hat{x} = (W^g)H^g r^g$.

**Iterative Detection and Decoding**

In order to achieve better spatial receiver diversity compared to the linear receiver, the principle of iterative detection and decoding (IDD) can be employed with the considered symbol level SF-MMSE receiver. The decoded decisions can be used for parallel interference cancellation (PIC) to mitigate the co-antenna interference (CAI) in IDD based receiver. If hard decisions are used in interference cancellation, the error propagation may destroy the benefit of the IDD receiver. Thus, soft interference cancellation with IDD is seen to be advantageous and is considered here in.

The initial soft decisions for the next iteration detection are obtained by using the linear symbol level MMSE based detector and iterative decoder. The IDD receiver in which detection is based on the MMSE filtering has been derived for MIMO MC-CDMA. When the receiver enters the IDD phase, MMSE detector has to be updated so that the estimated signal from the desired user can be obtained. Based on the estimation of the signal transmitted from different antennas, the co-antenna interference between different antenna signals can be obtained by using the estimated fading channel and the estimated data. Then the interference can be subtracted from the received signal. So more accuracy signal can be achieved and the performance can be improved a lot by cancelling the co-antenna interference.

Now the received signal model in (5.41) is decomposed to the form

$$r_p = C_{0,k,j} x_{0,k,j} + \bar{C}_{0,k,j} \bar{x}_{0,k,j} + \tilde{C}_0 \tilde{x}_0 + \sum_{q=1}^N C_q x_q + \eta,$$

(5.54)

where the first right hand side term represent the desired received signal of the user $k$ from layer $j$, the second one is the self-CAI term from layers $j' = j$, third term is MAI and the fourth term includes inter-cell interference. The MMSE minimisation criteria in IDD phase is written as

$$\langle W_{k,j}, \Psi_{k,j} \rangle = \arg \min_{\langle W, \Psi \rangle} \mathbb{E} \left\| x_{k,j} - W^H r - \Psi \right\|^2,$$

(5.55)
where matrix filter is $W_{k,j} \in \mathbb{C}^{M \times J_b}$ and $\Psi_{k,j} \in \mathbb{C}^{M \times J_b}$ represents self-CAI cancellation term for the user of the interest. The MMSE based filter with soft self-CAI cancellation in MIMO MC-CDMA systems has been derived in [VTL04] [VTL04a]. The MMSE criterion is differentiated with respect to the $W$ and $\Psi$. The solution for the CAI cancellation term is

$$\Psi_{k,j} = W_{k,j}^{H} \tilde{C}_{k,j} E[\tilde{x}_{k,j}]. \quad (5.56)$$

Thus, the symbol estimates of the layer $j$ at the output of the MMSE detector are given by

$$\hat{x}_{k,j} = W_{k,j}^{H} \left[ -\tilde{C}_{k,j} E[\tilde{x}_{k,j}] \right]. \quad (5.57)$$

### Simulation Results

The simulation results to illustrate the performance of the considered systems in single cell and cellular environment are presented in this section. The number of the subcarriers is $N_c = 1024$, of which 832 subcarriers are used. The frame length is $P = 832$ coded symbols per transmit antenna and the antenna configurations of $N = M = 1$, $N = M = 2$, $N = M = 4$ are considered. Parallel concatenated convolutional code with $[7,5]_{\text{oct}}$ generation polynomial in octal form or SFTuCM is applied in encoding. The coding rate is half which yields $N$ b/s/Hz spectral efficiency with QPSK. The spatial multiuser detection is performed using the detectors described previously. Decoding is based on the conventional log-MAP decoding in turbo coded (TC) case and on iterative space-time decoder with SFTuCM. Eight decoder iterations are used. Uncorrelated (UC) fading is applied for the different transmit-receive antenna pairs. The quasi-static fading is assumed due to the assumption of low mobility when the duration of the frame is much shorter than the coherence time of the channel. We consider short range cellular system herein and the length of the cyclic prefix is 800 ns. The number of the subcarriers is chosen to be 1024 instead of the 2048 to save the simulation time. Orthogonal Walsh-Hadamard spreading codes with length of 8 are used in MC-CDMA system.

In multi-cell environment, the average power of the received signal and the interfering signals depend on the shadow fading and on the path losses between the base stations and the desired mobile station in the central cell. The path loss obeys COST 259 non-line-of-sight microcell model and it is

$$L(d) = 26 \log(d) + 10 \log(4\pi/\lambda)), \quad \text{where} \quad d \quad \text{is distance from the base station in meters and} \quad \lambda \quad \text{is wavelength.}$$

The shadow fading is assumed to be log-normally distributed with standard deviation of 8 dB. The equal transmission power is assumed for every base stations and co-channel interferers are transmitting all the time. We assume that signal-to-noise-plus-interference ratio (SINR) can be kept positive with soft handover. Thus, the channel realizations with SINR below zero are not considered due to the lack of soft handover in the simulation model.

Figure 5.2.19 illustrates the performance of the OFDM and MC-CDMA in single-cell environment. The IEEE802.11n model C and model E defined power delay profiles (PDP) are used. The SNR is given per data symbol so that the loss due to the cyclic prefix is not taken into account. Note also that the same results are attained with 2048 subcarriers when the loss caused by the guard interval decreases. It can be seen that the performance of the SISO turbo coded MC-CDMA system is the same as OFDM system in IEEE802.11n model C channel. This is also valid in 2x2 for TC and SFTuCM. The reason for this is that the orthogonality of the codes removes the MAI efficiently in chip combining. The remaining of the orthogonality among the users’ sequences means that the channel is nearly constant between $G = 8$ adjacent subcarriers. OFDM outperforms MC-CDMA with negligible improvement in IEEE802.11n model E channel due to the increased frequency selectivity compared to the model C channel. If channel varying exists during $G$ adjacent subcarriers the OFDM system obtains better performance than MC-CDMA. The use of multiple antennas with TC yields improvement for performance compared to the SISO TC system when SNR is higher than 6 dB in model C channel. SFTuCM provides about 2 dB improvement for FER performance compared to the TC in two-by-two MIMO system.
Figure 5.2.19: FER vs. SNR performance of OFDM and MC-CDMA system in single cell environment. IEEE802.11n model C and model E channels used.

Figure 5.2.20 considers the performance of the OFDM and MC-CDMA systems in the seven-cell environment. The PDP of IEEE802.11n model E is used. The distance of the user from the base station is normalized by the cell radius. The OFDM system provides a slightly better performance than that of the MC-CDMA as seen from SISO TC case. Figure 5.2.19 illustrates that good performance is not achieved in fully loaded case because of the high interference due to the low soft handover threshold. In MC-CDMA the FER performance can be easily controlled by adjusting the number of the users at the cost of the reduced spectral efficiency of the system. The performance improves rapidly when the number of the users is decreased from 8 to 5 and SFTuCM is used, but with TC not so high enhancement of the performance is not achieved.

In order to further show the powerful ability of the proposed IDD receivers, it is considered in Figure 5.2.21 and Figure 5.2.22. Performance of the single cell OFDM systems with turbo and SFTuCM is illustrated in Figure 5.2.21. Different from other figures, here only 4 iterations are adopted too reduce the simulation time. Performance advantages of SFTuCM over TC can be seen clearly, around 1.3 dB performance gain with two iterations IDD receivers. In addition to this, the performance gain provided by IDD receiver with two iterations is about 2 dB for SFTuCM coded system. Figure 5.2.22 shows the performance gain of the IDD receiver with the soft CAI cancellation compared to the linear receiver. The antenna configuration is $N = M = 4$, $K = 6$, $G = 8$ and layered transmission is used. The results indicate that MIMO MC-CDMA with SFTuCM and IDD receiver provides superior performance in the cellular environment. The number of the users can be 75% of the spreading factor that acceptable performance for the high rate downlink data packet transmission is achieved with suitable automatic repeat request (ARQ) protocol.
Figure 5.2.20: FER vs. normalised distance in a multi-cell environment. OFDM is considered in the case of $K=1$. IEEE802.11n model E channel used.

Figure 5.2.21: FER vs. SNR performance of OFDM system with turbo and SFTuCM and IDD receiver in single cell environment. IEEE802.11n model C channel used ($N=M=4$, 4 decoding iteration).
Summary

Downlink MIMO MC-CDMA and OFDM systems with turbo code and space-frequency turbo coded modulation is considered in both single cell and multi-cell environments. The linear MMSE based receivers and iterative receiver structure with soft CAI cancellation are considered. The performance of the discussed systems in the presence of inter-cell interference is studied via computer simulations. MC-CDMA is able to provide the same performance as that of the OFDM when the correlation between the adjacent subcarriers is high. The soft CAI cancellation iterations are shown to possess the ability to further improve the performance of the layered systems. Our results indicate that the considered OFDM and MC-CDMA systems are feasible candidates for future cellular high date rate downlink packet transmission when equipped with a suitable ARQ protocol.

5.2.1.4 IFDMA

5.2.1.4.1 System Overview

Interleaved Frequency Division Multiple Access (IFDMA) can be considered both as a serial modulation and as a multi carrier transmission scheme. More specifically, it can be described as

- variant of a single carrier scheme, i.e., DS-CDMA using Frequency Domain Orthogonal Signature Sequences (FDOSS) as introduced in [WIN2.1], and also in, e.g.,[ChC00], [Fal04], and
- multi carrier scheme, i.e., unitary pre-coded OFDMA with interleaved sub carrier distribution, cf. [GRC02].

IFDMA signal generation is based on compression in time by factor \( L \) and subsequent \( L \)-fold repetition of a modulated data signal, where \( L \) designates a positive integer. Afterwards, a user dependent frequency shift is applied to the compressed and repeated signal of each user. As in OFDMA, IFDMA signal generation results in a set of overlapping but mutually orthogonal sub carriers applied to each user. The sub carriers of each user are equidistantly distributed over the whole bandwidth.

IFDMA combines many advantages of single and multi carrier transmission schemes:

- Due to the sub carrier arrangement, IFDMA provides high frequency diversity.
- Even for transmission over time variant multipath channels, the multiple access interference is very low.
- It has been shown that compared to MC-CDMA, IFDMA has better performance [SDS98].
- IFDMA has a low peak to average power ratio.
- IFDMA has low complexity for signal generation as well as for user separation.
- The principle of IFDMA signal generation has been shown to be a low complexity method for OFDMA signal generation [FCS04].

However, IFDMA has some disadvantages:
- As for OFDM, in order to provide robustness to time offsets and to avoid inter block interference of successive IFDMA modulated data blocks, a guard interval is necessary.
- For IFDMA, equalization is necessary in order to compensate inter symbol interference (ISI). Computationally efficient one-tap-equalization is possible for IFDMA using a cyclic extension as guard interval.
- Due to the interleaved sub carriers of different users, similar to OFDMA with interleaved sub carriers, IFDMA is expected to be sensitive to frequency offsets.

In the following, IFDMA is introduced as an OFDMA variant. We consider one user with index $k$ transmitting the data vector $d^{(k)} = [d_0, ..., d_{Q-1}]^T$. According to the signal model in Figure 5.2.23, after serial-to-parallel conversion, firstly, a linear operation described by the $(Q \times Q)$-matrix $M$ is applied to the input vector $d^{(k)}$ where matrix $M$ can be interpreted as a linear precoding matrix.

Let $N$ designate the total number of subcarriers available in an OFDMA system and let $Q$ designate the number of subcarriers assigned to each user. The total number of subcarriers shall be assumed as

$$N = K \cdot Q,$$  \hspace{1cm} (5.58)

where $K$ designates the number of users. In order to avoid inter user interference, the $Q$ subcarriers of each user have to be mapped to the $N$ subcarriers available in the system in such a way that subcarriers of different users are orthogonal to each other. The assignment of the precoded data symbols after mapping to the subcarriers can be described by an $N \times N$ Inverse Discrete Fourier Transform (IDFT) matrix $\text{IDFT}_N$. The output vector of the OFDMA system after parallel-to-serial conversion is represented by $s^{(k)} = [s^{(k)}_0, ..., s^{(k)}_N]^T$. If the matrix $M$ is chosen according to

$$M = I,$$ \hspace{1cm} (5.59)

where $I$ designates the $(Q \times Q)$-identity matrix, the output vector represents an uncoded OFDMA signal. For a unitary matrix $M$, the output vector represents a unitary precoded OFDM (UP-OFDM) signal as it is described, e.g., in [XZG03]. One possible choice for a unitary matrix $M$ is given by

$$M = \text{DFT}_Q,$$ \hspace{1cm} (5.60)

where $\text{DFT}_Q$ designates $Q$-point Discrete Fourier Transform (DFT) matrix. In this case, the generalised signal model depicted in figure 1 is equal to the structure described by [GRC02]. Let us assume the $Q$ subcarriers of each user $k$, $k = 0, ..., K - 1$, to be equidistantly allocated and equally distributed over the whole bandwidth. Allocating the $Q$ subcarriers of one user equidistantly over the total bandwidth is equal to the insertion of $K-1$ subcarriers modulated by zero inbetween two adjacent subcarriers modulated by elements of the precoded input vector $M \cdot d^{(k)}$. The insertion of zeros combined with the precoding matrix $M$ can be shown to be equal to

$$\tilde{M} = \text{DFT}_N \cdot R,$$ \hspace{1cm} (5.61)

where $\text{DFT}_N$ designates the $N$-point DFT matrix and $R$ represents a vector of $K$ $Q \times Q$ identity matrices $I_Q$ according to $R = [I_Q, ..., I_Q]^T$. In order to avoid multiple access interference, let us further assume that the subcarrier set of user $k$ has a frequency offset of

$$\Phi^{(k)} = \frac{2\pi}{N} k.$$ \hspace{1cm} (5.62)

In this case, the generalised OFDMA scheme in figure 1 can be described by
\[ \mathbf{s}^{(k)} = \Theta^{(k)} \cdot \text{IDFT}_N \cdot \mathbf{M} \cdot \mathbf{d}^{(k)} = \Theta^{(k)} \cdot \text{IDFT}_N \cdot \mathbf{DFT}_N \cdot \mathbf{R} \cdot \mathbf{d}^{(k)}, \]  

\[ (5.63) \]

where \( \Theta^{(k)} \) designates the frequency shift operation by rotation with \( e^{j \phi^{(k)} n} \), \( n = 1, \ldots, N-1 \) in time domain and is given by the \((N \times N)\)-matrix

\[ \Theta^{(k)} = \text{diag}(e^{\frac{2\pi j k n}{N}}). \]  

\[ (5.64) \]

Hence, using eq. (5.63) an OFDMA signal with DFT pre-coding and equidistantly distributed subcarriers and frequency offset according to eq. (5.64) for each user can be written as

\[ \mathbf{s}^{(k)} = \Theta^{(k)} \cdot \mathbf{R} \cdot \mathbf{d}^{(k)}. \]  

\[ (5.65) \]

Thus, equation (5.65) represents a data vector \( \mathbf{d}^{(k)} \) which is repeated \( K \) times and user dependently shifted in frequency domain by \( \Phi^{(k)} \) which is equal to the signal generation of an IFDMA signal. It should be noted that compared to OFDMA, for IFDMA instead of the \( Q \) elements of the data vector the \( Q \) elements of the DFT of the data vector are transmitted by the subcarriers.

### 5.2.1.4.2 Computational Complexity

In the following, the numbers of complex multiplications for OFDM and IFDMA signal generation at the transmitter side are compared to each other. In order to fairly compare IFDMA with OFDM, also for OFDM multiple users have to be taken into account, i.e. OFDMA is considered, where, similar to IFDMA, it is assumed that a number \( Q \) of subcarriers out of \( N \) is assigned to each of the \( K \) users. According to [NeP00] an OFDM signal using \( N \) subcarriers can be generated by an IFFT operation. For that reason, for an OFDMA system where the \( N \) subcarriers are assigned to different users, the number of complex multiplications \( M_{\text{OFDMA}} \) for signal generation is given by

\[ M_{\text{OFDMA}} = N \log_2 N. \]  

\[ (5.66) \]

As described, e.g., in [SDS98], for IFDMA the \( Q \) subcarriers of each user are equidistantly spread over the whole bandwidth and the subcarriers of different users are interleaved. According to [FCS04], for an OFDMA system with the same subcarrier distribution as for IFDMA, the computational complexity, i.e., the number of complex multiplications for OFDMA can be further reduced to

\[ M_{\text{OFDMA,red}} = N + Q \log_2 Q. \]  

\[ (5.67) \]

IFDMA signal generation consists of compression in time and subsequent repetition by factor \( K \) of a block of \( Q \) input data symbols. Afterwards, the resulting \( N \) IFDMA chips have to be user dependently rotated by a complex exponential. Hence, the number of complex multiplications \( M_{\text{IFDMA}} \) for IFDMA is given by the number \( N \) of the IFDMA chips to be rotated, that is

\[ M_{\text{IFDMA}} = N. \]  

\[ (5.68) \]

### 5.3 Estimation

#### 5.3.1 Pilot aided channel estimation (PACE)

In this section pilot aided channel estimation (PACE) for OFDM system is described. The principle of PACE is to periodically insert pilot symbols in time and frequency. Provided that the spacing of the pilots is sufficiently close to satisfy the sampling theorem, the channel response may be reconstructed by appropriate estimation and interpolation techniques. The principle of PACE was thoroughly described in deliverable D2.1 [WIND2.1] section 3.4.2.1, as well as in the references therein.

The channel response of an OFDM modulated signal is typically correlated in two dimensions, time and frequency. The optimum solution for PACE is given by a two dimensional (2D) Wiener interpolation filter (WIF), which minimizes the MSE between the desired response and the received pilot sequence. However, for practical implementation a cascaded filter consisting of two one dimensional (1D) filters, one operating in time and the other in frequency direction appears more appropriate, named 2x1D-PACE. First, channel estimation is performed in frequency direction, yielding tentative estimates for all subcarriers of that OFDM symbol. The second step is to use these tentative estimates as new pilots, in
order to estimate the channel for the entire frame. If estimators with the same complexity are compared, 2x1D-PACE typically outperforms 2D-PACE.

According to the requirements of the wide area mode as well as the short range mode, a mobile terminal should not transmit and receive at the same time. This means that for channel estimation purposes, one OFDM frame must be considered as an isolated stream of $N_{frame}$ OFDM symbols. In other words, pilot symbols transmitted in preceding and subsequent frames are not taken into account for channel estimation. Note, unlike for the results presented in this deliverable, in [WIND2.1] a continuous stream of FFT blocks was assumed, in which case the channel estimation accuracy would be somewhat superior.

The evaluation part of PACE can be found in chapter 6, and is structured as follows: selection criteria for an appropriate pilot grid are given in sections 6.2.3.1 and 6.4.2.1 for the wide area FDD mode and the short range TDD mode respectively. Results in the form of mean squared error (MSE) plots are presented in sections 6.2.3.2 and 6.4.2.2 for the wide area and the short range modes.

The optimum minimum mean squared error (MMSE) estimator performs smoothing and interpolation jointly. To reduce the complexity of the optimum MMSE estimator, we propose to separate the smoothing and interpolation tasks in section 6.2.3.3. The separated smoothing and interpolation estimator (SINE) consists of a MMSE based smoother which only operates at the received pilot symbols, and an interpolator which is independent of the channel statistics. We show in section 6.2.3.3 that the separated approach gets close to the optimum MMSE, while the complexity is grossly reduced.

In section 6.2.4 a simple way is found to assess the effects of channel estimation errors, given the MSE as a function of the SNR. With the typical urban (TU) channel model for the wide area mode, the degradation in $E_b/N_0$ for some selected pilot grids and estimators are compared. The benefit of using a pilot boost is also addressed in section 6.2.4. Conclusions on the choice of an appropriate pilot grid and channel estimator are given in section 5.3.1.1.

### 5.3.1.1 Conclusions

The main advantage of PACE with respect to schemes relying on the transmission of one full OFDM training symbol is twofold: first, even rapidly changing channels can be accurately estimated with modest pilot overhead. In the considered modes the pilot overhead was between 3% and 9% for the wide area mode and 1% to 2% for the short range mode. Second, channel estimation can be carried out independent of the chosen modulation cardinality, code rate, etc. This supports the deployment of a flexible air interface.

In order to apply PACE to the WINNER air interface, the following conclusions can be drawn:

- The preferred implementation of PACE is a cascaded filter consisting of 2 one dimensional Wiener interpolation filters. If possible channel estimation should be performed, such that estimation in frequency direction is performed first, since the computational cost is in all cases lower.
- For channel estimation in frequency direction about 20% oversampling factor appears to be sufficient. This results in a pilot spacing of $D_f=8$ and $D_f=20$ in the wide area and short range mode, respectively.
- For interpolation in time direction the oversampling should be around 100% or larger. Unless the velocity is very high (as in the rural channel model), placing a pilot at the beginning and end of the frame is sufficient. This results in a pilot spacing determined by the frame length according to $D_t=N_{frame}=1$.
- A simple model to approximate the effects of channel estimation errors was presented. This allows to assess the degradation in $E_b/N_0$ without implementing the channel estimation unit. Only the MSE curves for a given estimation scheme as well as the BER curves assuming perfect CSI are required.
- The degradation in $E_b/N_0$ can be reduced by inserting a pilot boost. In many cases a pilot boost appears favorable compared to a denser pilot grid, since a pilot boost only compromises the power efficiency, not the spectral efficiency of the system.
- Alternatively, the degradation in $E_b/N_0$ can also be reduced by more sophisticated signal processing. One possibility which was presented in this deliverable is the use of a pre-smoothing filter. A pre-smoothing filter achieves a performance close to the MMSE, having significantly less computational cost with respect to the optimum Wiener interpolation filter.
5.3.2 Cyclic channel estimation for high Doppler

High Doppler causes time variation of the channel impulse response, which results in significant performance degradation. One semi-blind channel estimation method for MIMO-OFDM systems in high Doppler scenarios is briefly presented in section 3.4.2.5.2 of [WIND2.1]. This channel estimation method is described and simulation results for wide area scenarios are presented in this section. Simulation results indicate that the method of channel estimation is suitable for high Doppler wide area scenario, while it does not reduce the bandwidth of system.

It is easy to prove that the inverse DFT of a periodic circular array of complex numbers given by

\[ p = [a \, 0 \cdots 0 \, a \, 0 \cdots 0 \, a \cdots] \]

with \( Q \) zeros between every two \( a \)'s is also an array of the same form but with a different number of zeros. Hence, if a known impulse value \( a \) is periodically added to the data samples \( d \) in the frequency domain, i.e., the transmitted signal is denoted as \( z = F^{-1}[d + p] \), the received signal is

\[ y = h \ast F^{-1}[d] + h \ast w + n \]  \hspace{1cm} (5.69)

where \( w = F^{-1}[p] \), and the form of \( w \) is also given by \([c \, 0 \cdots 0 \, c \cdots 0 \cdots] \) with a different number of zeros, say \( R-1 \).

Now compute the expectation value of \( y(m) \) denoted by \( E[y(m)] \), but defined as a time average of every \( R \) samples

\[ E[y(m)] = \frac{1}{Q} \sum_{q=0}^{Q-1} y(m + qR) \]  \hspace{1cm} (5.70)

where the product of \( Q \) and \( R \) is equal to the number of subcarriers. Assuming that the data and noise have zero mean, since \( h \ast w \) is periodic with period \( R \), it is found that

\[ E[y(m)] \approx c \cdot h(m) \]  \hspace{1cm} (5.71)

In other words, this channel estimator computes an average value of close-to periodic samples of received signals containing zero-mean orthogonal impulse trains (such as Walsh codes) to obtain a channel estimate under the zero-mean data and zero-mean noise assumption. The estimator requires simple computation and modest channel tracking capability under high Doppler channel environment without any bandwidth penalty since the pulse sequence is added to the data, not to replace data.

The only limitation is that the period of the impulse train in the time domain must be greater than the delay spread of the channel. Also, the larger make the amplitude of the impulse train, the better the channel estimate becomes but the worse the peak to average ratio of the time domain signal to be transmitted becomes and more transmitted signal power is allocated to impulse signals over the data symbols.

To eliminate this DC bias term, a bipolar Walsh sequence impulse train is used instead of a constant amplitude periodic impulse train. In addition, Walsh sequence will bring more benefit for multiple antennas systems due to its zero cross correlation property among different Walsh sequences. This implies that the Walsh impulse train of each transmitter antenna can be uniquely recovered by correlating its own Walsh impulse train thus enabling MIMO channel CIR estimation.

The process of channel estimation and detection at the receiver can be in time domain or in frequency domain. Figure 5.3.1 shows the time domain approach. The received signal will be processed for channel estimation as described previously. The next step is to extract the impulse train out of the received signal.

For this purpose, the channel estimate \( \hat{h}(n) \) is used to convolve with the scaled impulse train \( c \cdot w(n) \) to obtain the impulse train estimate embedded in the received signal. After subtracting out this impulse train influence, the received signal \( \tilde{y}(n) \) is converted into the frequency domain through FFT. Finally a usual per-tone equalization is performed in the frequency domain to obtain the symbol estimate \( \hat{D}(k) \).
For MIMO systems cyclic channel estimation can also be employed. A set of $M$ Walsh sequences can be used for $M$ transmit antennas. Walsh sequences are chosen since they render zero cross-correlation to each other. If assuming the transmit antennas are sending each impulse train synchronously, one Walsh sequence used by one transmit antenna won’t be interfering other Walsh sequence used by other transmit antenna for the channel estimation process. At the receiver, each receiver antenna will cross-correlate the received signal with the Walsh sequence of a desired transmit antenna to obtain the channel estimation of the desired transmit antenna. This estimation process occurs simultaneously across receiver antennas. After MIMO channel CIR estimates are available, each receiver will subtract out undesirable Walsh impulse train influence, and then the received data will be passed to a MIMO symbol detector along with MIMO channel estimates.

It can be observed that the error concentrates at subcarrier locations of the impulse train in the frequency domain. It is found out that noise enhancement and the data symbol cancellation are occurring at those subcarrier locations. To alleviate this problem, an additional random timing jittering can be added to the impulse train in the time domain. In this way, the impulse train itself is not periodic such that the data error associated by channel estimate won’t cancel the data symbols at those frequency bins and at the same time the noise enhancement will be spread to adjacent subcarriers. There is no change in the overall estimator error, however, through this jittering the estimate error characteristics has changed to avoid the actual data cancellation and noise enhancement.

Simulations are conducted to show the performance of the algorithm. The simulation environment is according to wide area parameter and scenarios in Winner. Figure 5.3.2, Figure 5.3.3, Figure 5.3.4 and Figure 5.3.5 show BER and PER curves of the cyclic channel estimation against existing LS channel estimation and perfect channel estimation for SISO case. The number of subcarriers is 1024 within 20 MHz system bandwidth and 832 out of them are used. A 100 samples cyclic prefix is added to each OFDM symbol to alleviate ISI. QPSK modulation is used and the channel code used is $(133,171)$ $K=7$ convolutional code. The simulation is under urban macro and rural model with vehicular velocities of 70 km/h and 250 km/h, respectively. One coded packet contains 10 OFDM symbols. The existing Least Square (LS) channel estimation uses 2 OFDM preambles to estimate channel while the cyclic channel estimation utilizes 64 or 128 impulse samples spanning over one OFDM symbols. The simulation result shows the obvious advantage of the cyclic channel estimation algorithm over the conventional preamble-format channel estimation. It shows that the preamble-format channel estimation won’t be a reliable method for high Doppler channel environment unless additional training sequences are inserted throughout the packet to update channel estimates meaning further loss of bandwidth efficiency. Whether for urban macro scenario or for rural macro scenario, performance of this cyclic channel estimation is better than the LS method. When vehicular velocity is high e.g. 250 km/h, performance of the method is also good. In addition, it is noted that the cyclic channel estimation doesn’t have any bandwidth loss since the impulse train is added to the data symbol in the transmitter and removed in the receiver as shown in the receiver structure.

The above cyclic channel estimation method is semi-blind. It does not replace the subcarrier tones in OFDM symbols. Instead a pulse train is added to the data. So there is no loss of bandwidth efficiency. The estimator requires simple computation and modest channel tracking capability under high Doppler channel environment. This channel estimator can be well-suited for the long physical data packet system such as future high data rate wireless system.
Figure 5.3.2: BER performance for urban macro at 70 km/h

Figure 5.3.3: PER performance for urban macro at 70 km/h
Figure 5.3.4: BER performance for rural at 250 km/h

Figure 5.3.5: PER performance for rural at 250 km/h
5.4 Synchronization

5.4.1 OFDM based synchronization on the downlink

5.4.1.1 OFDM synchronization strategies

For synchronization we have to distinguish between coarse synchronization (acquisition), where all other relevant system parameters are still unknown and fine synchronization, termed tracking, where a previous coarse estimate is further improved and adapted to the channel variations. Furthermore, there are three quantities which are to be synchronized in an OFDM system: first, OFDM symbol and frame timing synchronization; second, carrier frequency synchronization; and finally, sampling rate synchronization.

One can distinguish between pre-DFT and post-DFT synchronization algorithms, i.e. synchronization is performed prior and after the demodulation of the subcarriers, respectively. Note that in general, post-DFT algorithms essentially require at least a coarse symbol timing information, which must be obtained in the time domain. According to the discussion in [WIND2.1], [SFF01], pre-DFT algorithms are commonly used for acquisition while post-DFT algorithms are more appropriate for fine tuning of the synchronization offset.

A synchronization strategy for an OFDM based downlink may consist of several algorithms. In current OFDM systems, mainly two different strategies are adopted. In the following their properties as well as the major advantages and drawbacks are summarized.

In the following the synchronization strategy for the terrestrial digital video broadcasting (DVB-T) system is outlined:

- **Acquisition** in time and frequency can be achieved by synchronization utilizing the guard interval. Coarse symbol timing synchronization, as well as the fractional part of the frequency offset, are compensated prior to the DFT by exploiting the structure of the cyclic prefix. As a result ICI is eliminated or can be at least kept to an acceptable level. The performance of this algorithm is evaluated in section 0.
- Averaging over several OFDM symbols may be necessary to achieve sufficient accuracy required for subsequent stages.
- The remaining task of finding the start of the OFDM frame, as well as the integer part of the frequency offset (i.e. the part of the frequency offset which is multiple of the subcarrier spacing) can be determined in the frequency domain, after OFDM demodulation.
- **In the frequency domain** the integer part of the frequency offset, $\Delta f_I$, is estimated using continuously transmitted pilot tones, located at certain subcarrier positions. The set of transmitted pilots shows up at the FFT output bins shifted by $\Delta f_I$, which can be easily detected by correlating two consecutive OFDM symbols.
- By assigning these pilot tones an appropriate signature, the start of an OFDM frame can also be determined.
- **Tracking**: the fine tuning of the carrier frequency offset, $\Delta f$, can be obtained by employing continuous pilot tones. A carrier frequency offset translates to a certain phase shift of adjacent subcarriers, and can therefore be detected in the frequency domain. The estimate of $\Delta f$ is then fed to a voltage controlled oscillator which tunes the local oscillator. The performance of frequency domain tracking is evaluated in section 6.2.5.2. Instead of continuous pilot tones, decision directed tracking, where decisions from previous symbols are used as pilots, could be used. In an iterative receiver utilizing the turbo principle, also soft information could be used to improve the reliability of these pilots.
- Finally, the synchronization performance can be improved by iteratively updating the different synchronizer stages.

Synchronization for wireless LAN systems, such as Hiperlan/2 or 802.11a is achieved as follows:

- **Acquisition** is based on an OFDM training symbol. The unique structure of the training symbol allows to instantly estimate the timing and frequency offset in the time domain. Hence, acquisition is a one-step procedure.
- **Tracking** is equivalent to the case described above.

The main advantages and drawbacks of the two described strategies are:
• The main advantage of acquisition based on correlating the cyclic prefix is that no additional pilot overhead is required.
• The main drawback of the cyclic prefix based approach is that its performance degrades in frequency selective channels [BSB97].
  o The main advantage of using a dedicated OFDM training symbol is that very fast and reliable acquisition is feasible.
  o Its main drawback is that additional resources for synchronization are consumed. For the WINNER air interface with its relatively few numbers of OFDM symbols per frame, this gives rise to a significant training overhead.

5.4.2 Synchronization in OFDM based single cell and cellular networks

5.4.2.1 Introduction

In a classical Single Frequency Network (SFN) all radio stations operate in the same frequency band and transmit identical signals. The pure concept of SFN has been applied in some broadcasting network like Digital Audio Broadcasting (DAB) or Digital Video Broadcasting Terrestrial (DVB-T) [ETSI2001]. In order to increase the flexibility and efficiency of the cellular communication system, an adapted SFN can be applied based on the OFDM transmission technique.

There are some major differences between an SFN network and a conventional cellular network. Firstly, in a cellular SFN network, all base stations (BSs) and mobile terminals (MTs) share the whole resources and can access them at anytime. It means that resources used by BSs and MTs in different cells are not separated by any guard bands. Secondly, in such a network, there is no BS controller. Instead of that, a radio resource management (RRM) using a self-organized dynamic channel allocation (SO-DCA) is operated in each BS. MTs in different cells communicate with their BSs by the means of a multiple access scheme [Galda03].

Implementing such a technique described above requires an accurate time and frequency synchronization of all BSs within the network. Therefore a concept for synchronization has been proposed, using simple sync signals in the downlink (DL) and uplink (UL) periods. In this report a general structure of such self-organized network is described. The feasibility of this proposed synchronization technique in a full-coverage cellular network in short-range and wide area scenarios is then investigated. Finally, results in different deployment scenarios are shown by means of computer simulations.

5.4.2.2 System Concept

A cellular OFDM network is considered, in which every cell is allowed to transmit in the same frequency band at the same time. Such network is well known from the application of the OFDM transmission technique for broadcasting applications and is referred to as single frequency network (SFN).

In contrast to a broadcasting system, where the same data is transmitted by different base stations to achieve a macro-diversity gain, in this proposal a cellular communication network is considered. In the considered system, radio resources are managed and selected by the RRM operating at each BS. Resource assignments to MTs are carried out independently among BSs within a network. All BSs and MTs share the whole common resources by means of a multiple access scheme. A general structure of such a network is shown in Figure 5.4.1. An MT can listen to different BSs, but it will always be associated with the BS, from which it receives signals with the highest power. The other way around, a BS can listen to MTs in its own cell and MTs in the vicinity as well.

A TDD frame structure will be assumed for separating the UL and DL transmission within the network. Therefore, a BS cannot, in general, directly receive the signal transmitted by another BS. It allows supporting a traffic asymmetry between the UL and the DL. To combine this frame structure with a flexible resource management, the frame timing of different cells has to be aligned.
To avoid both ICI and ISI between MTs located within the same cell as well as in adjacent cell during the uplink and between adjacent BS all terminals within the network shall be synchronized in time and frequency.

Figure 5.4.1: General structure of an OFDM based cellular single frequency network

5.4.2.3 Description of the Sync Signal

Due to the system concept described in the previous section all MTs located in the same as well as in adjacent cells have to be tightly synchronized in time and frequency to avoid severe multiple access interference. Moreover, in an SFN each BS within the network should synchronize itself with its neighbours. The MTs can be synchronized according to received information from BSs in the DL. However, it is not possible for a BS to receive signals, which are transmitted directly from other BSs. Hence a concept has been proposed, namely using MTs as the means to synchronize BSs. In the UL each BS receives signals from MTs in adjacent cells, which contain timing and frequency information about other BSs. Each BS then estimates time and frequency offsets and adjusts its own time and carrier frequency according to the estimations.

To overcome this challenge, two dedicated sync signals are introduced, one for the synchronization of MTs during the DL and another one for the synchronization of BSs during the UL. The sync signals are located at the beginning and at the end of each MAC frame as a preamble and postamble, respectively. In the DL, the preamble sync signal is used to synchronize all MTs and in the UL, the postamble sync signal is used to synchronize all BSs within the network. The concept is illustrated in Figure 5.4.2.

Figure 5.4.2: TDD frame structure including a DL and UL sync signals
The sync signal structure is defined in the frequency domain, which is illustrated in Figure 5.4.3. Each sync signal consists of only two adjacent subcarrier signals inside the system bandwidth, which are transmitted with the maximum transmission power.

![Figure 5.4.3: Each sync signal allocates a pair of two adjacent subcarriers](image)

At the receiver side, each sync signal compared to the useful data will have a gain of SNR defined by

\[
P_g = 10 \log\left(\frac{N_C}{2}\right)[dB]
\]

where \(N_C\) is the total number of used subcarriers in the whole bandwidth. For example, with 2048 subcarriers, the gain should be 30 dB. It assures that the sync signal can be detected even in case the SNR of transmitted data is low. It is shown in Figure 5.4.4.

![Figure 5.4.4: The gain of sync signal compared to data signal](image)

In the time domain each subcarrier pair is transmitted in three adjacent OFDM symbols. The phase difference between the two adjacent subcarrier signals is zero and the phase difference between two adjacent OFDM symbols on a single subcarrier is also zero at the transmitter side. An overview of the sync signal is depicted in Figure 5.4.5.
### 5.4.2.4 Estimation Procedure

The reason to construct a sync signal in the way described in the previous section is that time and frequency synchronization can be carried out simultaneously at the receiver.

- The phase difference between the two adjacent subcarriers of the sync signal is directly related to the time difference between transmitter and receiver.
- The phase difference of the same subcarrier but of two adjacent OFDM symbols of the sync signal is directly related to the carrier frequency offset between transmitter and receiver.

Based on this property, time and frequency offset estimation procedure is carried out in the DL for MTs and in the UL for BSs as follows. For each frame, in the DL every BS transmits a sync signal, which is different from others. Therefore at an MT side only a sync signal is evaluated, which is either the one with the highest signal amplitude or the one that has been signalled by the BS. The case is more complicated for the UL, because as many as possible of the sync signals are included in the estimation procedure. The MTs, which are associated with the same BS, also transmit an identical sync signal. The signals from these MTs superimpose themselves at the receiving BS on the same pair of subcarriers. Finally different signals from MTs in different cells are located on different subcarriers. This procedure is illustrated in Figure 5.4.6.
Figure 5.4.6: Superposition and locations of sync signals at a BS

The structure of the synchronization system at a receiver is depicted in Figure 5.4.7.

Figure 5.4.7: Synchronization system at a receiver

There are 3 main functions, namely *Estimate time and frequency*, *Detect the reliable estimation* and *Adaptation in a control loop*.

*Estimate time and frequency offset*: a sync signal consisting of 3 OFDM symbols is received in the time domain. There is an ISI free interval from the latest beginning symbol until the earliest ending symbol due to the phase continuity property. Within this interval 2 consecutive FFT windows can be placed in order to transform the signal into the frequency domain. Both the time offset and the frequency offset are estimated in the frequency domain. The estimation process is based on the phase rotation between adjacent symbols or subcarriers. Firstly the frequency offset is estimated by the phase difference between the same subcarriers but of two adjacent OFDM symbols as given in the following formula

\[
    f_I = \frac{\Delta F}{2\pi} \tan^{-1}\{R_k(I)^*R_2(l)\} \tag{5.73}
\]

where \( R_k(I) \) is the output of the \( k \)-th symbol FFT on subcarrier \( I \), \( f_I \) is the frequency offset estimated on that subcarrier and \( \Delta F \) is the subcarrier spacing.

Secondly, depending on the phase difference between the two adjacent subcarriers of the sync signal, the time offset is estimated as follows
\[ t_i = \frac{T_s}{2\pi} \tan^{-1}\{R_i(l)R_i(l+1)^*\} \] (5.74)

where \( t_i \) is the time offset estimated on that subcarrier and \( T_s \) is the symbol duration.

The estimation procedure is depicted in Figure 5.4.8.

**Figure 5.4.8: Two FFT windows are placed within the ISI free interval**

Detect the reliable estimation: as the accuracy of a phase estimation solely depends on the SNR of the observed signal, only the sync signals received with a sufficient signal amplitude (greater than a detection threshold) are evaluated during the estimation process. Therefore an amplitude threshold is defined to guarantee a reliable estimation.

Adaptation in a control loop: after the estimation phase, MTs and BSs will adapt their own time and carrier frequency based on the estimations. However, with one-step adaptation they cannot reach the common point. The new time and frequency offsets should be continuously updated in following frames according to the evaluated results of the previous frame. It is obtained by the following formula

\[ \delta^*_k[n+1] = \delta^*_k[n] + \eta \sum_{m=1}^{K} \delta^*_m[n] - \delta^*_k[n] \] (5.75)

where \( \delta^*_k[n] \) is the time/frequency offset of BS \( k \) during frame \( n \) and \( \eta \) is the feedback factor. In the next section an influence of this feedback factor on the synchronization process will be discussed. The adaptation process should be iterated in such a loop in order to have a convergence of time and frequency parameters of all MTs and BSs.

### 5.4.2.5 Network Simulation

A cellular single frequency network based on the OFDM transmission technique is considered. In the analysis a regular grid of hexagonal cells is considered. All cells are assumed to have an identical radius \( R \), which is a simulation parameter. In the considered simulation scenario, the cell radius has a value of 100m. For the network analysis a cellular network having two additional rings around a central cell is used. All MTs are generated within the network following a uniform distribution and the number of MTs is also a simulation parameter. The whole cellular network is depicted in Figure 5.4.9.
The transmit power $P_{TX}$ is kept constant for all terminals during the whole simulation time. No power control is used to adapt the transmit power of a mobile terminal to the channel conditions. The transmit power of the terminals is adjusted such that, a predefined average SNR $\gamma$ at the cell border is obtained.

An open space situation is analysed with the WSSUS channel model 802.11n. Influences of radio channel characteristics like path loss, fast fading, shadowing, multi-path delay are considered.

In the control loop, the update process is repeated for 100 consecutive frames to achieve convergence. Such a frame sequence is considered as a simulation run.

**Figure 5.4.9: Cellular network for simulation**

Some typical parameters that are used and varied during the simulation are shown in Table 5.1. The average number of MTs per cell is 7. The investigation is carried out with the value of the SNR at the cell border of 20 dB. The mean of the estimated offsets is weighted by a feedback factor $\eta$, which is identical for time and frequency offset updates. This factor decides how fast timing and frequency synchronization converge. But there is a trade-off: the faster the convergence is, the higher is the offset variance.
Table 5.1: Simulation parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System bandwidth</td>
<td>$B = 100$ MHz</td>
</tr>
<tr>
<td>Number of subcarriers</td>
<td>$N = 2048$</td>
</tr>
<tr>
<td>Subcarrier spacing</td>
<td>$\Delta F = B/N = 48.8$ KHz</td>
</tr>
<tr>
<td>Symbol duration</td>
<td>$T_s = 1/\Delta F = N/B = 20.48 \mu$s</td>
</tr>
<tr>
<td>Guard interval length</td>
<td>$N_G = 80$</td>
</tr>
<tr>
<td>Guard interval duration</td>
<td>$0.8 \mu$s</td>
</tr>
<tr>
<td>Number of cells</td>
<td>$N_{bs} = 19$</td>
</tr>
<tr>
<td>Cell radius</td>
<td>$R = 100$ m</td>
</tr>
<tr>
<td>Path-loss coefficient</td>
<td>$2.5$</td>
</tr>
<tr>
<td>Feedback factor</td>
<td>$\eta = 0.3$</td>
</tr>
<tr>
<td>SNR at propagation distance $R$</td>
<td>$20$ dB</td>
</tr>
<tr>
<td>Average number of MTs per cell</td>
<td>$7$</td>
</tr>
<tr>
<td>Channel model</td>
<td>$802.11n$</td>
</tr>
</tbody>
</table>

5.4.2.6 Synchronization in single cell

As mentioned, an MT will always be associated with the BS from which it receives the signal with the highest power. The MT will evaluate the DL sync signal to synchronize itself to the BS it is associated with. A suitable sync signal has to allow an MT to establish a tight synchronization with the strongest BS it is receiving. The task of MT synchronization is therefore closely related to synchronizing to the strongest propagation path in a multipath transmission environment.

The accuracy of the employed synchronization procedure determines the resulting amount of interference, which will be observed by a BS during the UL transmission. Thus, the performance that can be achieved in the data transmission during the UL only depends on the synchronization accuracy that can be achieved by the MTs during the DL synchronization phase. No additional synchronization is carried out at the receiver during the UL.

In an OFDM system small timing offsets between the transmit signals of different MTs within the same cell can be mitigated by the guard interval. If this time offset exceeds the range which can be compensated, severe inter symbol and inter carrier interference will result. Thus, all transmitters in the UL have to be synchronized to the common timing reference. Typically, this reference is defined by the DL transmission of the BS. In this case the remaining deviation in the arrival times of the MTs signals result from the propagation delays. These delays are much smaller than the OFDM symbol duration and can therefore be compensated by the guard interval.

The frequency synchronization of an OFDM-FDMA uplink is much more susceptible to a deviation of the state than the time synchronization. An offset of the transmit carrier frequencies between different MTs can cause severe interference between subcarriers assigned to different users. The more the transmit frequencies deviate from each other the higher will the resulting ICI be. To avoid severe performance degradation, an accuracy of the frequency synchronization in the order of a few percent of the subcarrier spacing is required [Mohamed01].

Due to the superposition of the different MT signals at the air interface their frequency offsets can only be compensated at the receiver by the means of multi-user detection like MMSE linear or iterative interference cancellation techniques [Cao02][Ibars03]. Even though possible in principle those techniques will not be part of this research because of their high computational complexity. The technique proposed here rather aims at removing the frequency offset at the transmitter than compensating it at the receiver.
In the considered system any data transmission will be suspended until a sufficient synchronization of BSs and MTs has been achieved. Thus, the residual time offset will be smaller than the OFDM guard interval and the residual frequency deviation between different terminals will be small enough not to cause severe interference.

Figure 5.4.10 shows the convergence of all MTs’ frequency to their BS’s frequency. Within a cell, frequency synchronization between all MTs and their BS is correctly achieved after 20 frames. Frequency offset is about 0.5% of the subcarrier spacing. Figure 5.4.11 shows the convergence in time of all MTs regard to their BS. Inside a cell, after 10 frames, time synchronization between all MTs and BS is achieved. Time offset is about 8% of the guard interval.

![Convergence of MTs to their BS frequency](image)

**Figure 5.4.10: Frequency synchronization in a cell**
5.4.2.7 Synchronization in a Self-organized Cellular Network

An MT will receive superimposed signals from different BS. As it has been explained before these signals must be synchronized in time and frequency to avoid interferences. Therefore, in an SFN not only have the MTs to be synchronized to their BS but also the BS have to be synchronized with their neighbours.

As all MTs located within the same cell are synchronized to their own BS, the sync signals transmitted by these MTs do not convey any information to their BS. Thus, for the synchronization carried out during the UL, only sync signals, which have been received from adjacent cells, are of interest to a BS. Every BS will, therefore, evaluate the sync signals it has received from MTs located in other cells in order to obtain an “indirect” estimate of the time and frequency offset of its adjacent BS.

Unfortunately, all sync signals from adjacent cells are received with a much lower signal power than the sync signal transmitted by the BS’s own MTs. Moreover, these sync signals will be completely unsynchronized until the tracking phase has been reached and will interfere with each other. It is this fact which makes the estimation of the time and frequency offsets of adjacent BS much more difficult than in a single cell situation. It is important to design a suitable sync signal that allows a reliable estimation of time and frequency offsets despite of this situation.

The synchronization accuracy which is achieved by evaluating the sync signals of adjacent cells during the UL determines the overall system performance. On the one hand, it influences the DL, as signals of different BS will superimpose at an MT during the DL data transmission phase. On the other hand, the time and frequency offset of the MTs within one cell depends on the state of their BS. In case when adjacent BS are not sufficiently synchronized, the MTs located in their cells will not be adjusted in time and frequency either.

On the one hand, in the short range scenario, the FDMA multiple access is proposed. Therefore both time and frequency synchronizations are highly required. On the other hand, the TDMA multiple access is proposed for the wide area scenario. In the OFDM-TDMA system, in each time slot, which consists of some symbols, the full bandwidth can be used by a single user. A certain number of time slots inside a frame is assigned for each user. That means in each cell different resources in time domain are allocated.
to each MT. In this described system, frequency synchronization between cells does not play important role anymore. Instead of that, time and frame synchronization are more significant.

Figure 5.4.12 shows the convergence of all BSs’ frequency to the reference center BS’s frequency. After 20 frames, frequency synchronization between all BSs within the cellular network is correctly achieved. Frequency offset is about 1% of the subcarrier spacing. Figure 5.4.13 shows the convergence in time of all BSs regard to the reference center BS. After 20 frames, time synchronization between all BSs within the network is achieved. Time offset is about 10% of the guard interval.

![Figure 5.4.12: Frequency synchronization in a cellular network](image-url)
5.4.2.8 Conclusion

In this section, time and frequency synchronization of MTs and BSs in an OFDM-based cellular single frequency network has been analysed. The proposed technique is based on the sync signal as preamble or postamble of a MAC frame in the DL or UL, respectively. A sync signal consists of 3 phase-continuous OFDM symbols. In the frequency domain, it is represented by a pair of subcarriers, chosen from a subset of subcarriers in the whole bandwidth. At the receiver, 2 FFT windows are placed within the ISI free interval and according to the phase difference between 2 modulated OFDM symbols; time and frequency offset can be estimated. A control loop is used in order to achieve the convergence of time and frequency synchronization. After each frame timing and frequency are adjusted to the newly estimated values.

A network simulation has been carried out in a full-coverage network. As observed from the simulation results, time and frequency offsets can be estimated simultaneously. It can be concluded that this synchronization algorithm is feasible for time and frequency synchronization of MTs and BSs in an OFDM-based cellular single frequency network.

5.4.3 Synchronization for MIMO-OFDM

One synchronization scheme for MIMO-OFDM systems with improved immunity to co-channel interference is briefly presented in [WIND2.1]. For the wideband (WB) case a frequency reuse of one is foreseen, so the synchronization scheme robust to co-channel interference is needed. This scheme has been simulated for the DL case, but its modified versions can be adopted also for the UL case. The concept of the proposed scheme is described and simulation results are presented in this section.

5.4.3.1 Description

Synchronization process includes at least packet detection, carrier frequency offset correction and symbol timing recovery. Phase tracking and sample timing recovery are also likely to be necessary. Synchronization is achieved via special training symbols placed at the head of the packet. The time domain representation of the training sequence is given in Figure 4.1.17 of [WIND2.1]. For every TX antenna it is transmitted first a signal that is the combination of signals with short and long periodicity.
(used for packet detection, carrier frequency offset estimation and symbol timing recovery). Then, the part dedicated for channel estimation is transmitted.

The spectral representation of the first part of the training sequence is shown in Figure 5.4.14. The letters a, b, c indicates the frequency packets transmitted by the up to three interfering BTS [Pri04]. With proper network planning, direct co-channel interference among signals with the same time period is avoided because of frequency multiplexing.

![Figure 5.4.14: Spectral representation of the first part of the training sequence. Small guard bands are provided to separate every packet of subcarriers.](image)

Let us assume a system with \( N \) subcarriers, \( M \) TX antennas and \( P \) RX antennas. Let the OFDM signal at the \( m \)-th TX antenna be \( x_m(t) \), then let the received signal at the \( p \)-th RX antenna be:

\[
r_p(t) = \sum_{m=1}^{M} \sum_{l=1}^{\Delta} c_l^{mp}(t) x_m(t-l) + n_p(t), \quad p = 1\ldots P
\]

where \( \Delta \) represents the maximum delay spread of the channel (the time unit is the sampling time) and the first tap is placed in the time origin, \( n_p \) is an additive noise contribution.

The \( M \) TX antennas are divided in \( \beta \) different groups \( \Gamma_l, \, l = 1\ldots \beta \). If \( \Gamma_m \) is the group containing the \( m \)-th TX antenna, let \( R_m \) be the number of its elements. Moreover, a set of subcarriers \( \Omega_m \) is assigned to each group. Antenna assignment has to be carried out such that the reciprocal correlation between antennas in the same group is minimized. If e.g. \( M = 4 \) and \( \beta = 2 \), antennas \{1, 3\} should be assigned to the first group and \{2, 4\} to the second group. Choosing different groups like \{1, 2\} brings a drop of performance especially for highly-correlated channels and/or small number of RX antennas.

During the transmission of the training sequence, assuming that the total length of the sequence is given by an integer number \( S \) of OFDM symbols without CP, the following time-domain signal is transmitted from the \( m \)-th TX antenna:

\[
\tilde{x}_m(t) = \sum_{k=0}^{S\xi_m-1} C_m^{\Omega_m}(t-kD_m), \quad t = 0\ldots S \cdot N - 1
\]

where \( D_m = N / \xi_m \) represents the time period of the training sequence relative to group \( \Gamma_m \). This corresponds to use \( \xi_m \) subcarriers in the set \( \Omega_m \), while the remaining subcarriers are set to zero. \( C_m^{\Omega_m} \) is a pseudo-random sequence of length \( D_m \).

Looking at Figure 5.4.14, it can be noticed that for the implementation used in the simulations a slightly different training sequence has been adopted, which having a single group (\( \beta = 1 \)). This group \( \Gamma_1 \) will
not use a single set of subcarriers, but two sets named $\Omega_1$ and $\Omega_2$. The definition (5.77) of the training sequence is modified as follows:

$$\tilde{x}_n(t) = \sum_{k=0}^{N_m-1} C_m^{D_{m,1}}(t - kD_{m,1}) + \sum_{k=0}^{N_m-1} C_m^{D_{m,2}}(t - kD_{m,2}) \quad t = 0 \ldots S \cdot N - 1$$  (5.78)

where $C_m^{D_{m,1}}$ is a sequence with time period $D_{m,1} = \frac{N}{T_{m,1}}$, using the subcarriers in $\Omega_1$, and $C_m^{D_{m,2}}$ is a sequence with time period $D_{m,2} = \frac{N}{T_{m,2}}$, using the subcarriers in $\Omega_2$. This means that the signal transmitted from every TX antenna is the sum of two signals with different period and different spectral band.

The various $C_m^{D_{m}}$ are chosen in such way that a pseudo-orthogonality condition between different TX antennas is achieved:

$$\forall m', m'' \in \{1 \ldots M\}, \sum_{i=0}^{N_m-1} \tilde{x}_n(t) \tilde{x}_n^*(t + k) = \begin{cases} \alpha_n & \text{if } m' = m'' = m \text{ AND } k = 0 \\ 0 & \text{otherwise} \end{cases}$$  (5.79)

The pseudo-orthogonality condition is broader than the orthogonality condition. It is inclusive of the cases in which the random sequences for the training sequence are orthogonal codes like Walsh-Hadamard codes.

In practical design of the training sequence, $S$ will typically be in the order of 1 or 2. Simulations with particularly high interference scenarios have also been performed, for which $S = 4$ and over performing better. $C_m^{D_{m}}$ can be produced in the time domain and then filtered to eliminate the subcarriers not belonging to $\Omega_m$. In the simulations the signal is produced in the frequency domain and then transformed to the time domain via IFFT.

The training sequence is used for packet detection first. This task is performed by looking at the threshold-crossing of an auto-correlation function normalized to signal energy. This is a traditional way of achieving packet detection. In this way the optimal threshold value is not too closely tied to the amplitude of the incoming signal [HT02]. In the implementation, for every RX antenna there are three different auto-correlation windows. The value of auto-correlation is summed to smooth out the effect of random noise. This implementation is a direct extension of the algorithm developed in [HT02] for the SISO case to MIMO case. It is well known that this algorithm needs to be preceded by a DC-blocking filter in order to avoid that any DC component is interpreted as periodic signal, which leads to triggering a false alarm.

In the proposed design the subcarriers for every TX antenna have been divided into two sets $\Omega_1$ and $\Omega_2$. Let $D_1 < D_2$, such that $\Omega_1$ contains the signals with the shortest time period. Only the signal relative to $\Omega_1$ is used for the packet detection. The separation of the various groups of subcarriers is performed via FFT or preferably via a set of digital filters which is more selective than the FFT.

After packet detection, the training symbols defined in (5.77) can conveniently be exploited for a multi-step frequency offset estimation process as follows. There are two sets of subcarriers $\Omega_1$ and $\Omega_2$, and offset estimation is performed in two steps. The received signal related to $\Omega_1$ will be processed first and then $\Omega_2$. Let us filter the signal expressed in (5.76) so that $r_p^{\Omega_1}(t)$ contains only subcarriers belonging to the set $\Omega_1$. The first offset estimation step (coarse estimation) is based on the following auto-correlation:

$$\Psi_p(k) = \left\{ \sum_{l=0}^{L} r_p^{\Omega_1}(t) r_p^{\Omega_1}(t + kD_1) \right\} \quad L \leq S$$  (5.80)

In the case considered here of $\beta = 2$, it can be assumed in (5.80) that $l = 1$. If $f_s$ is the sampling frequency, the coarse estimated frequency offset on the $p$-th RX antenna is given by:
The coarse offset estimate has a detection range of \( \frac{\xi}{(2k_{\text{max}})} \) inter-carrier spacing. Assuming that all RX antennas are subject to the same frequency offset (that applies in case a single local oscillator is used), the frequency offset estimation can be averaged:

\[
f_{\text{coarse}} = \frac{1}{P} \sum_{p=1}^{P} f_{\text{coarse},p}
\]  

(5.82)

A coarse frequency adjustment is then applied to the received signal for every RX antenna:

\[
\hat{r}_p(t) = r_p(t) \cdot e^{-2\Psi_{\text{coarse}}}, \quad p = 1 \ldots P
\]

(5.83)

Further offset estimation steps are based again on (5.80), performed on a different group \( \Omega_l \), where the time period is longer than that in the previous step. In the specific case the second estimation step is also the final one (fine estimation). This step is performed with \( l = 2 \):

\[
\Psi_p(k) = \sum_{\tau=0}^{\infty} \hat{r}_p(\tau + kD_j), \quad L \leq S
\]

(5.84)

Based on (5.81) and (5.82) it is possible to compute a fine offset estimate:

\[
f_{\text{fine}} = \frac{1}{P} \sum_{p=1}^{P} f_{\text{fine},p}
\]

(5.85)

Finally, the fine correction is applied to the received signal for every RX antenna:

\[
\tilde{r}_p(t) = \hat{r}_p(t) \cdot e^{-2\Psi_{\text{fine}}}, \quad p = 1 \ldots P
\]

(5.86)

It is noted that the overall estimation process corresponds to an frequency offset of:

\[
\Delta f_{\text{est}} = f_{\text{coarse}} + f_{\text{fine}}
\]

(5.87)

It has been shown that this design of the training sequence plus the above way to process it increases the performance of the offset estimator of a factor close to 6 dB compared to the traditional approach.

The last synchronization step is the fine symbols timing, which is performed on the set of subcarriers \( \Omega_2 \). The algorithm is a standard cross-correlation plus averaging that guarantees excellent performance in MIMO thanks to space diversity. One can observe that when the received signal has a higher amplitude and lower distortion, the cross-correlation with the TX training symbols will have a higher absolute value. Thus, summing the appropriate correlations on all TX antennas and taking one decision only, one can expect higher performance. These decisions can be averaged on all RX antennas.

5.4.3.2 Simulation

Simulations have been done for 4 x 4 MIMO case in downlink using short range scenario parameters, i.e. 2048 subcarriers within about 100 MHz bandwidth at 3 km/h. The simulation environment is such that the operating point is above or equal to SNIR = -5 dB, with 2 x co-channel interferers and AWGN. The signal and the interferers have different, independent MIMO channel realizations (still, the interferers are picked up from a limited set of different cases).

The total interference has been arbitrarily chosen so that it satisfies to the following condition:

SNR = SNIR + 10 dB, meaning that interference always has a stronger power than AWGN. E.g. if the simulation is performed with SNIR = -5 dB, then SNR = 5 dB and SIR = -4.542 dB.

Concerning network timing, simulations may be performed in three different scenarios:
a. Completely asynchronous network. In this case the signal and the interferers are randomly delayed in time with respect to each other.
b. Quasi-synchronous network. In this case the signal and the interferers are delayed of a random amount from 0 to 10 samples with respect to each other.
c. Like (b) but where the amount of delay is 0 to 100 samples. This corresponds to a difference in propagation path of up to 300 m.

In the case (a) the training sequence transmitted from a given BTS has a high probability to interfere directly with an in-band interferer (payload from a different BTS). In the case (b) the co-channel interference is extremely limited (apart from a short time interval). The case (c) has only slightly stronger interference power than (b), but might be more realistic from a network point of view.

Simulations results are presented below. They are related only to the cases (b) and (c), which should be more realistic from a network perspective.

Starting from the scenario (b), the performance of the synchronization algorithms is summarized in the following figures.

Figure 5.4.15 shows the performance of packet detection algorithm for scenario (b) with two different thresholds, i.e. 0.4 and 0.5. We can see that for these two thresholds missing detection rates are both zero. With 0.4 of threshold, the false alarm rate is a little high when SNIR is below –2 dB. However, with 0.5 of threshold, the false alarm rate is zero within –5~15 dB region. Therefore, selecting an optimal threshold can improve the packet detection algorithm a lot.

Figure 5.4.16 shows the performance of frequency offset estimation for scenario (b) with 4096-samples training sequence. From the figure, we can see that the average module of normalized estimation error is less than 0.01 when SNIR is above –4 dB. When SNIR is –5 dB, the error is just a little higher than 0.01. Therefore, the performance of the frequency offset estimation algorithm is essentially good in the SNIR region.

Figure 5.4.17 shows the probability density functions (PDF) of module frequency offset estimation error in SNIR region of –5~15 dB for scenario (b) with 4096-samples training sequence. Each curve represents the PDF for one specific integer SNIR. The curve for –5 dB is the rightmost one while that for 15 dB is the leftmost. From the figure, we can see that the probable region of estimation error is also left shift when SNIR is increased.

Figure 5.4.18 shows simulation results of symbol timing algorithm for scenario (b) with 4096-samples training sequence. We can see that in the full SNIR region the average module estimation error is almost zero. Thus the performance for symbol timing is good.
Figure 5.4.15: Performance of packet detection with threshold values 0.4 (thr a) and 0.5 (thr b) for scenario (b).

Figure 5.4.16: Performance of frequency offset estimation for scenario (b). The requirement on the maximum estimation error depends on the detection range of the phase tracking algorithm.
Figure 5.4.17: PDF of frequency offset estimation error for SINR = -5 ~ 15 dB for scenario (b)

Figure 5.4.18: Performance of symbol timing recovery with $4^2 \times 2048$ samples X-correlation windows for scenario (b)
For the scenario (c) performance is as follows:

Figure 5.4.19 shows the performance of packet detection algorithm for scenario (c). We can see that packet detection performance for scenario (c) is similar to that for scenario (b). For two thresholds missing detection rates are both zero. With 0.4 of threshold, the false alarm rate is a little high when SNIR is below −2 dB. However, with 0.5 of threshold, the false alarm rate is zero within −5~−15 dB region.

Figure 5.4.20 shows the performance of frequency offset estimation for scenario (c) with 4096-samples training sequence. Similar to the results for scenario (b) the average module of normalized estimation error is less than 0.01 when SNIR is above −4 dB. When SNIR is above −4 dB, the error is just a little higher than that for scenarios (b) because of larger amount of delay. But the performance is also good for this scenario.

Figure 5.4.21 shows the probability density functions (PDF) of module frequency offset estimation error in SNIR region of −5~−15 dB for scenario (c) with 4096-samples training sequence. Each curve represents the PDF for one specific integer SNIR. The curve for −5 dB is the rightmost one while that for 15 dB is the leftmost. Compared to Figure 5.4.17, probable region of estimation error at high SNIR like 15 dB for scenario (c) is on the left of that for scenario (b).

Figure 5.4.22 shows simulation results of symbol timing algorithm for scenario (c) with 4096-samples training sequence. The average module estimation error is essentially less than 0.002. The performance for scenario (c) is worse than that for scenario (b), but it is acceptable.
Figure 5.4.20: Performance of frequency offset estimation for scenario (c). The requirement on the maximum estimation error depends on the detection range of the phase tracking algorithm.

Figure 5.4.21: PDF of frequency offset estimation error for SINR = -5 ~ 15 dB for scenario (c).
From above simulation results, we can see that the performance of packet detection in the two considered scenarios starts to degrade below –2 dB for a non-optimal threshold, but an accurate choice of the threshold can ensure sufficient performance in the whole operating range down to –5 dB.

Frequency offset estimation does not show particular loss of accuracy above –4 dB for the scenarios (b) and (c). Performance in scenario (c) is only marginally worse than that in (b).

Symbol timing recovery does not show any problem for scenarios (b) and (c).

Simulation results obtained for the 100 MHz channel show that the training sequence with frequency interleaving (multiplexing) of the training symbols from different BTSs is suitable to operate down to SNIR = -5 dB.

5.5 Iterative Techniques

5.5.1 Iterative Channel Estimation

5.5.1.1 Iterative channel estimation technique for OFDM systems - Description

In this section we evaluate the performance of iterative channel estimation (ICE) procedures applied specifically to OFDM systems, used as modulation scheme in most of the WINNER air interface modes described in the section 3.1. More specifically, we focus on the so-called separated approach in which symbol detection and channel estimation are performed separately within the receiver chain. This is motivated by the need of a low-complex receiver structure for the physical layer design of the WINNER system. If one considers the joint approach which performs jointly symbol detection and channel estimation with the help of an extended MAP algorithm, a significant processing resource must be allocated to that stage of the decoding chain as the complexity of the trellis-based MAP detection is greatly enhanced (cf. [WIND2.1], section 3.4.3.2).

In the following, one provides various channel estimation quality measurements for each transmission scenario, including the wide area case (urban and urban macro environment) and the short range scenario.
5.5.1.1 Detector design

A chart depicting the general structure of the detection device under discussion is given below in the Figure 5.5.1. Assuming the separated approach, symbol-wise detection and channel estimation are carried out separately in the receiver chain. First the incoming signal from the channel is passed onto the soft-in/soft-out demapper which outputs soft information on the coded bits under the form of Log Likelihood Ratios (LLRs) to a deinterleaver. These LLRs proceed then to the soft-in/soft-out channel decoder. After reinterleaving of the soft information on the coded bits generated by the outer decoder, the a priori information on the coded bits is processed by a soft mapping device which maps soft bit information onto so-called soft symbols, generally different from elements of the modulation alphabet employed for transmission. The obtained soft symbols are used as pilots for the channel estimation stage which consists of two one-dimensional Wiener filters arranged in cascade.

![Diagram of detector design](image)

Figure 5.5.1: Iterative Channel Estimation for OFDM systems

Note that only the extrinsic information $L^e$ is forwarded along the receiver chain for later processing, implying that the a priori LLRs of each decoder must be subtracted from their corresponding a posteriori LLRs in order to ensure the convergence of the iterative process. The gray filled rectangle from Figure 5.5.1 refers to the specific part of the receiver dedicated to the channel estimation step. In the case of iterative channel estimation, the estimates are generated according to the two following steps:

Before the first turbo iteration, no a priori soft information on the coded bits is available to the receiver. Consequently, one has to perform at first a conventional channel estimation via interpolation based solely on the subcarriers at pilot locations. Here, we consider for that purpose two cascading Wiener interpolation filters (cf. [WIND2.1], section 3.4.2.1) in frequency and time directions which provide estimates over the complete OFDM grid. This is represented graphically in the Figure 5.5.2 through the red marked upper path.
On a second stage, soft symbols obtained from the soft mapper are multiplexed with the pilot symbols and processed by the two successive one-dimensional Wiener filters in order to get new channel estimates. Obviously, no interpolation is required anymore in that case because LS estimates not only at pilot positions but also at data symbol locations are utilized for Wiener filtering in both frequency and time directions. This process is pictured on Figure 5.5.3 through the violet marked paths.
The whole decoding procedure is repeated until a stable state is achieved or when one reaches a certain number of turbo iterations to keep the decoding processing time reasonable.

5.5.1.1.2 General simulation parameters

In the following, we take the transmission parameters considered for the WINNER wide area and short range scenarios given in the section 3.2.2.1. For channel coding, the memory 6 convolutional code with generator polynomials (133,171) with max-log-MAP decoding is utilized. For the wide area scenario, we consider a maximal user velocity of 70 km/h (urban macro model) or 250 km/h (rural propagation model) which corresponds respectively to the Doppler frequencies $f_d = 324$ Hz and $1156$ Hz.

5.5.1.1.3 Parameters for the iterative channel estimation (ICE)

In contrast to a conventional interpolation technique, the subcarrier spacing is not a relevant issue for the second part of an ICE process anymore. In fact, a genuine Wiener filter is equivalent to a Wiener interpolation filter (WIF) with subcarrier spacing equal to 1. However, the first channel estimate obtained with the help of the interpolation filter determines greatly the performance of the future channel estimation steps. Indeed, a too loose pilot grid induces bad channel estimation when the sampling theorem is violated, resulting in a poor reliability of the soft information provided by the channel decoder. Obviously, this affects negatively the performance of the subsequent iterative channel estimation steps. This error propagation could even result in a performance degradation of the channel estimation which is of course to be avoided because an iterative scheme is only introduced when its use enhances the overall receiver performance. Moreover, we employ a mismatched Wiener filter as in section 5.3.1 for the channel estimation through interpolation and the subsequent iterative channel estimations. This is mainly motivated by the fact that the channel correlation properties in time and frequency domains are not necessarily known at the receiver side, and even if they are known, a frequent update of the filter coefficients would be computationally prohibitive.

Therefore, we will refer in this part to parameters similar to those mentioned for a conventional channel estimator based on a WIF, i.e. the subcarrier spacings in frequency and time, $D_f$ and $D_t$, the filter sizes respectively in time and frequency directions $M_t$, $M_f$ at the initial estimation step, the filter sizes $M_t^*$, $M_f^*$ for the crude iterative channel estimation steps. An additional parameter specific to iterative techniques is the number of iterations $N_{\text{it}}$. For instance $N_{\text{it}} = 0$ means that no turbo loop is employed at the receiver. From $N_{\text{it}} = 1$, an all-symbols channel estimation is performed, according to Figure 5.5.3.

Following [WIND2.1], section 3.4.2.1, the sampling theorem in time and frequency domains gives upper bounds on the acceptable subcarrier spacing in time and frequency directions. For the wide area case, we have $D_{f, \text{max}} = 10$, $D_{t, \text{max}} = 27$ in the urban macro cell environment, $D_{f, \text{max}} = 17$, $D_{t, \text{max}} = 7$ in the rural scenario, assuming a mobile terminal velocity of up to 250 km/h. Concerning the short range scenario, we get $D_{f, \text{max}} = 25$, $D_{t, \text{max}} = 100$ which means that the channel coefficients can be reconstructed with a much lower pilot overhead in the short range case compared to the wide area propagation scenario. In Table 6.6, candidate pilot subcarrier spacings for the wide area, urban macro case together are listed with the induced pilot overhead. Not that for the simulations, we do not employ the subcarrier spacing $D_t = 4$ for the sake of simplicity. The pilot subcarrier spacings utilized for the rural environment are listed below in the Table 5.2. In order to prevent channel estimation error propagations, we keep the values of the spacings below those suggested by the maximum thresholds mentioned above.

| Table 5.2: Pilot subcarrier spacings and related pilot overhead, wide area rural case |
|---------------------------------|--------|--------|
| $D_t$ | $D_f$ | $3$    |
| $8$   | $5.0\%$ | $3.6\%$ |

For the short range scenario, we are free to choose larger spacings in the frequency direction while for the time direction, pilot symbols are placed solely at the edges of the OFDM frame as the sampling theorem allows overwhelming subcarrier spacings in that direction.
Table 5.3: Pilot subcarrier spacings and related pilot overhead, short range scenario

<table>
<thead>
<tr>
<th></th>
<th>(D_f)</th>
<th>(12)</th>
<th>(20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D_t)</td>
<td>10</td>
<td>1.7%</td>
<td>1.0%</td>
</tr>
</tbody>
</table>

The filter sizes of the interpolation filter are similar to those provided in section 5.3.1 for non-iterative channel estimation. We select \(M_f = 8\) and \(M_t = \) number of OFDM tones containing pilot symbols per frame. When performing all-symbols channel estimation, the filter sizes are set to \(M'_f = 8\) and \(M'_t = \) number of OFDM symbols per frame, usually 10. That way, the filter sizes are approximately equal in frequency and time direction while the complexity is kept rather low.

Concerning the number of iterations, we simulate up to \(N_{it} = 3\) turbo iterations and try to determine the required number of iterations before reaching a stable state with the help of so-called EXtrinsic Information Transfer (EXIT) chart analyses. A trade-off between receiver performance and decoding complexity might also be found in order to find a suitable number of turbo iterations. It should be stressed that a higher number of iterations is solely advantageous if a non-gray mapping is employed at the modulation/demodulation stages [Brink99].

5.5.1.1.4 Reference modes

The BER curves and the EXIT charts obtained with the aforementioned simulation parameters are compared with those produced when perfect channel knowledge, or CSI, is assumed at the receiver. For the EXIT chart analysis, mutual information transfer characteristics referring to channel estimation via interpolation are given as mean of comparison with iterative schemes. The latter curves are otherwise required to draw the EXIT charts for ICE because the first channel estimation is performed with the help of a WIF, giving the bit-wise mutual information value fed to the channel decoder before the first turbo iteration.

5.5.1.2 Performance assessment – BER and MSE results

**Short range:** We first address the performance analysis of iterative channel estimation for OFDM systems by means of BER curves and MSE measurements. In the short range scenario, the proposed channel estimation technique allows to achieve the bound given by the case where perfect channel knowledge is assumed at the receiver. It should be stressed that this results is achieved with a rather loose pilot grid as the pilot symbol overhead remains almost negligible, here below 2%. One can notice that additional turbo iterations are not necessary to improve the detection performance in terms of Bit Error Rate. On the other hand, the MSE, after a drop at around 5 dB, tends to stabilize in the high SNR regime. This is nevertheless a general effect resulting from the utilization of a mismatched Wiener filter for channel estimation (section 5.3.1).

![Figure 5.5.4: BER curves, short range case, \(D_t = 10, \) \(D_f = 12\) (left plot) \(D_t = 10, \) \(D_f = 20\) (right plot)](image-url)
Cellular case, urban macro environment: The four following figures depict the receiver performance in the urban macro propagation environment. We observe again that the gain of iterative techniques for channel estimation only become significant when the pilot subcarrier spacings approach the theoretical limits given by the sampling theorem ([WIN2.1], section 3.4.2.1.1.2). The Figure 5.5.6 and Figure 5.5.7 refer to cases where the subcarrier spacings still are still far from the limits imposed by the sampling theorem. In Figure 5.5.8 and Figure 5.5.9 however, as the pilot symbol density decreases, a performance gain can be observed. At BER = 10^{-2}, we get a positive shift of 0.5 dB after only one turbo iteration in Figure 5.5.9, even though the performance bound given by the perfect CSI scenario is still far from being achieved. Regarding the CE MSE averaged over the whole OFDM frame plotted on page 101, the same trend can be observed. With spacings $D_t = 8$, $D_f = 8$, the ICE technique outperforms the non iterative CE stage from $E_b/N_0 = 6$ dB. In other pilot grid configurations, the gain in terms of MSE induced by the use of an iterative CE method vanishes. It should again motivate the use of ICE techniques if the pilot subcarrier spacings approach the theoretical bounds in time and/or frequency directions. Otherwise, a pilot-based channel estimation stage is sufficient.
Figure 5.5.7: Receiver performance, urban macro channel model with perfect CSI (dashed violet curve), channel estimation via interpolation (dashed blue curve) or using ICE (solid curves), $D_t = 3$, $D_f = 5$

Figure 5.5.8: Receiver performance with perfect CSI (dashed violet curve), channel estimation via interpolation (dashed blue curve) or using ICE (solid curves), $D_t = 5$, $D_f = 8$
Figure 5.5.9: Receiver performance with perfect CSI (dashed violet curve), channel estimation via interpolation (dashed blue curve) or using ICE (solid curves), $D_t = 8$, $D_f = 8$

Figure 5.5.10: MSE against $E_b/N_0$ for various pilot symbol spacing pairs $D_t \times D_f$ obtained with a Wiener interpolation filter (WF) or with an iterative channel estimation technique after three turbo iterations (ICE 3 iters), urban macro channel model
**Cellular case, rural scenario:** Concerning the rural propagation environment, the iterative channel estimation technique allows also to improve the receiver performance both in terms of MSE and BER. However, the perfect CSI bound cannot be approached as the accuracy of the channel estimation step is greatly degraded at a Doppler frequency of $f_D = 1120$ Hz, which corresponds to a maximal user velocity of 250 km/h in the rural propagation scenario.

A more robust channel estimation algorithm, i.e. a pilot-based method with a denser pilot grid, may be employed in this case to fill the performance gap observed in Figure 5.5.11 and Figure 5.5.12. With respect to the urban macro environment, we note similarities concerning characteristics of the ICE behaviour. When the pilot grid is sufficiently dense, the ICE method does only improve slightly the system performance. This fact is illustrated by a performance gain of 1 dB at BER = $10^{-3}$ in Figure 5.5.12.

The maximal interval between two adjacent pilots in the time direction is $D_t_{max} = 7$ according to the sampling theorem. For $D_t = 6$, the pilot symbol spacing in time direction is close to the maximum level which explains the poor receiver performance visible in Figure 5.5.11 before the first turbo iteration. In that case, the ICE method helps to improve the transceiver performance by 2.5 dB at BER = $10^{-3}$ after two iterations, while the MSE is significantly reduced as well. In the low SNR regime however, the output signal of both CE stages is so distorted that the channel decoder cannot provide reliable information on the information bits, which is translated in a BER of almost 0.5 for $E_b/N_0 \leq 4$ in Figure 5.5.11.

![Figure 5.5.11: BER results, the rural propagation model, $D_o = 6$, $D_f = 8$, mobile velocity $v = 250$ km/h, CE performed with a Wiener interpolation filter ("No iteration") or using an iterative channel estimation technique (solid curves)](image)
Figure 5.5.12: BER results, rural propagation model, $D_t = 3$, $D_f = 8$, mobile velocity $v = 250$ km/h, CE performed with a Wiener interpolation filter ("No iteration") or using an iterative channel estimation technique (solid curve).

Figure 5.5.13: MSE against $E_b/N_0$ with pilot symbol spacing $Dt \times Df$, and, rural propagation model.
Remarks: A common observation from the BER and MSE curves from all propagation scenarios leads to the conclusion that the utilization of ICE becomes uniquely relevant above a certain $E_b/N_0$ threshold. In most cases, this level is set to about 5dB and corresponds to the value above which the soft information on the coded bits produced by the channel decoder becomes reliable enough in order to perform the all-symbols channel estimation shown in Figure 5.5.3. Under that threshold, only the channel estimates obtained through interpolation should be employed for the demapping stage. This fact is illustrated in the EXIT charts given below as well.

5.5.1.3 Performance assessment – EXIT charts

The mutual information transfer characteristics of the demapper and the channel decoder describe the relationship between the incoming and the outgoing flow of mutual information for the decoding element [Brink99],[WIND2.1], section 3.4.3.5). The exchange of extrinsic information between the demapper and the channel decoder is plotted in an EXIT chart. This graphical representation gives a good insight into the behaviour of the iterative process. The trajectory of the iterative decoding loop starts from the y-axis at a level which corresponds to the output bit-wise mutual information of a demapper fed with the first channel estimates, obtained via interpolation (green line on the following charts). Afterwards, the trajectory is built by successive vertical and horizontal projections on the curves that refer to the channel decoder (light blue curve) and the detection block with ICE (red curve). As a consequence, the number of turbo iterations required before achieving a stable state can be derived directly from the EXIT chart.

Short range scenario:  

Figure 5.5.14: EXIT chart, detection and channel decoding at $E_b/N_0 = 6$ dB with the simulated trajectory (dashed lines), short range propagation environment

On the figure above is depicted the EXIT chart for the pilot spacings $D_t = 10$, $D_f = 20$ at $E_b/N_0 = 6$ dB. The trajectory gets stuck after only one turbo iteration so that no performance improvement shall be observed afterwards, owing to the good quality of the first channel estimation via interpolation. The degradation due to the channel estimation errors is therefore reduced and explains the good performance of the ICE technique in terms of BER or MSE already observed in section 5.5.1.2.
Figure 5.5.15: EXIT chart, detection and channel decoding at $E_b/N_0 = 6$ dB with the simulated trajectory (dashed lines), urban macro propagation environment, $v_{\text{max}} = 70$ km/h.

Figure 5.5.16: EXIT chart, detection and channel decoding at $E_b/N_0 = 4$ dB with the simulated trajectory (dashed line), urban macro propagation environment, $v_{\text{max}} = 70$ km/h.
The Figure 5.5.15 and Figure 5.5.16 represent the evolution of the iterative decoding loop in the cellular case (urban macro) with pilot spacings $D_t = 8$, $D_f = 8$, respectively at $E_b/N_0 = 6$ dB and $E_b/N_0 = 4$ dB. While the trajectory slope obtained at 6 dB is similar to that from Figure 5.5.14, the second case shows a pathological configuration. The first pilot-aided CE stage is not as good as at 6 dB and consequently, the trajectory gets stuck immediately in the iterative process, at the low position $I_{e1} = 0.15$, $I_{e2} = 0.41$. The performance of the receiver is thus very poor at $E_b/N_0 = 4$ dB, even with iterative channel estimation.

Figure 5.5.17: EXIT chart at $E_b/N_0 = 8$ dB, rural propagation environment, $v_{max} = 250$ km/h

Figure 5.5.18: EXIT chart at $E_b/N_0 = 4$ dB, rural propagation environment, $v_{max} = 250$ km/h
At a maximal speed of 250 km/h in the rural propagation scenario, the impact of the Doppler effect becomes significant in comparison to the urban macro case. Indeed, at a Doppler frequency of $f_d \approx 1100$ Hz, the pilot CE stage is particularly vulnerable to estimation errors, especially if a mismatched filter is employed for interpolation. As a result, the gap between the output bit-wise mutual information of the demapper with pilot-aided CE and that obtained with perfect CE broadens. On the EXIT charts, this fact is also visible through the slope of the transfer characteristic of the demapper with ICE, marked in red. While at $E_b/N_0 = 8$ dB (Figure 5.5.17), the curve is still steep on the left-hand side of the graph, the slope evolves significantly in that part at $E_b/N_0 = 4$ dB. Instead of a rapid increase in terms of output bit-wise mutual information at high SNR, the progression tends to be much slower in the low SNR regime on Figure 5.5.18. Thus, below a certain SNR level, neither the pilot CE stage nor the ICE technique can provide sufficiently reliable bit-wise information to the channel decoder. It explains in particular the poor receiver performance at $E_b/N_0 < 4$ dB appearing in Figure 5.5.12.

5.5.2 Iterative Interference Suppression for Pseudo-Random-Postfix OFDM based Channel Estimation

5.5.2.1 Introduction

The PRP-OFDM modulation scheme has been studied in [MCD+03], [MDC+04] and [WIND2.1] (sections 4.6 to 4.10) for the single-antenna context and in [MDC+04] for the multiple-antennas context. With PRP-OFDM, an estimate of the CIR can be derived by a simple averaging of the summation of the postfix with the head of the time domain OFDM received block [MCD+03]. The averaging is required in order to cancel the interference of the samples carrying useful information on the pseudo-random postfix. A practical trade-off needs to be established between the length of the averaging window and the resulting amount of residual interference impacting the channel estimation accuracy. For a large averaging window [SMS+04] shows that PRP-OFDM CIR estimation outperforms schemes relying on rotating pilot patterns interpolation in terms of mean-square-error (MSE) up to an SNR of approx. 15dB. When targeting higher SNRs (e.g. using high order constellations, 64QAM) the requirement in CIR MSE is such that the constraints put on the averaging window length lead to solutions which cannot be considered for implementation.

In order to solve this issue, this chapter proposes an iterative CIR estimation scheme refining the first rough CIR estimates based on exploiting the outputs of a soft output decoder which makes PRP-OFDM suitable for high throughput systems.

5.5.2.2 Notations and PRP-OFDM modulator

This section settles the baseband discrete-time block equivalent model of a $N_c$ carrier PRP-OFDM system. The $i$ th $N_c \times 1$ input digital vector $\mathbf{x}_{N_c}(i)$ is first modulated by the IFFT matrix

$$F_{N_c}^H = \frac{1}{\sqrt{N_c}} \left[ W_{N_c}^H \right]^H,$$

$0 \leq k < N_c, 0 \leq l < N_c$ and $W_{N_c} = e^{-j\frac{2\pi k l}{N_c}}$. Then, a deterministic postfix vector $\mathbf{c}_D = [c_{D,1}, \ldots, c_{D,N_c}]^T$ weighted by a pseudo random value $\alpha(i) \in C, |\alpha(i)| = 1$ is appended to the IFFT outputs $\mathbf{s}_{N_c}(i)$. A pseudo random $\alpha(i)$ prevents the postfix time domain signal from being deterministic and avoids thus spectral peaks [MCD04]. With $P = N_c + D$, the corresponding $P \times 1$ transmitted vector is $\mathbf{x}_P(i) = F_{NP}^H \mathbf{x}_{N_c}(i) + \alpha(i) \mathbf{c}_P$, where

$$F_{NP}^H = \begin{bmatrix} \mathbf{I}_{N_c} \\ 0_{D,N_c} \end{bmatrix} \mathbf{F}_{N_c}^H \quad \text{and} \quad \mathbf{c}_P = \{0_{N_c}, \mathbf{c}_D\}^T$$

(5.88)

The samples of $\mathbf{x}_P(i)$ are then sent sequentially through the channel modeled here as a $L$ th-order FIR $H(z) = \sum_{n=0}^{L-1} h_n z^{-n}$ of impulse response $\mathbf{h} = (h_0, \ldots, h_{L-1})$. The OFDM system is designed such that the postfix duration exceeds the channel memory $L \leq D$.

Let $\mathbf{H}_{\text{in}}(P)$ and $\mathbf{H}_{\text{out}}(P)$ be respectively the Toeplitz inferior and superior triangular matrices of first column: $[h_0, h_1, \ldots, h_{L-1}, 0, \ldots, 0]^T$ and first row $[0, \ldots, 0, h_{L-1}, \ldots, h_0]$. As already explained in [MCD+03], the channel convolution can be modeled by $\mathbf{r}_P(i) = \mathbf{H}_{\text{in}} \mathbf{x}_P(i) + \mathbf{n}_P(i)$. $\mathbf{H}_{\text{in}}(P)$ and $\mathbf{H}_{\text{out}}(P)$ represent respectively the intra and inter block interference. Since $\mathbf{x}_P(i) = F_{NP}^H \mathbf{x}_{N_c}(i) + \alpha(i) \mathbf{c}_P$, we have:

$$\mathbf{r}_P(i) = (\mathbf{H}_{\text{in}} + \beta \mathbf{H}_{\text{out}}) \mathbf{x}_P(i) + \mathbf{n}_P(i)$$

(5.89)
where $\beta = \frac{a(i-1)}{\sigma^2}$ and $n(i)$ is the $i$th AWGN vector of element variance $\sigma^2_n$. Note that $H_\beta = (H_{in} + \beta H_{in})$ is pseudo circulant: i.e. a circulant matrix whose $(D-1) \times (D-1)$ upper triangular part is weighted by $\beta$.

The expression of the received block is thus:

$$r_p(i) = H_\beta \left( F_{in}^H \tilde{x}_n(i) + \alpha(i) c_p \right) + n_p(i)$$

Please note that equation basic.circul is quite generic and captures also the CP and ZP (Zero Padding) modulation schemes. Indeed ZP-OFDM corresponds to $\alpha(i) = 0$ and CP-OFDM is achieved for $\alpha(i) = 0$, $\beta(i) = 1 \forall i$ and $F_{in}^H$ is replaced by $F_{cp}^H$, where

$$F_{cp}^H = \begin{bmatrix} 0_{D,N_c \times D} & I_D \\ I_{N_c} & 0_{N_c, N_c} \end{bmatrix} F_{cp}^H.$$  

With these notations, CIR estimation is discussed in the following.

### 5.5.2.3 Channel Estimation

The standard low-complexity PRP-OFDM CIR estimation technique based on interference suppression by mean value calculation is is discussed in [MCD+03] and [WIND2.1] (sections 4.6 to 4.10). This section presents a novel, improved approach based on iterative interference suppression.

The iterative estimation scheme presented here requires an initial CIR estimate which is for example obtained by the technique presented in [MCD+03] and [WIND2.1].

The iterative CIR estimation is performed in several steps:

1. Initial CIR estimation: at iteration $k = 0$, perform an initial CIR estimation $\hat{h}_i^0$, for example as proposed in [MCD+03].
2. Increment iteration index: $k \leftarrow k + 1$
3. Perform FEC decoding based on latest CIR estimates $\hat{h}_i^{k-1}$: buffer the outputs of the soft-output decoder which indicate the bit-probabilities of the $l$th encoded bit of the constellation on carrier $n$ of OFDM symbol $i$: $p_l^i(\tilde{x}_n(i))$ with $n \in [0, \cdots, N_c - 1]$ and $l \in [0, \cdots, \log_2(Q) - 1]$; $Q$ is the constellation order.
4. Interference estimation: as detailed in 5.5.2.5, the interference estimation $u_p^i(i)$ from OFDM data symbol $i$ is generated based on the bit-probabilities $p_l^i(\tilde{x}_n(i))$ and the latest CIR estimates $\hat{h}_i^{k-1}$ as given by theorem 1 (cf annex 1).
5. Interference suppression: subtract estimated interference from received vector $r_p(i)$ and form a new observation vector: $\hat{r}_p^i(i) = r_p(i) - u_p^i(i)$.
6. CIR estimation: derive a new CIR estimate $\hat{h}_i^k(i)$ from $\hat{r}_p^i(i)$ e.g. as proposed in [MCD+03]. The resulting $\hat{h}_i^k(i)$ yields to a more accurate estimate since interference of the OFDM data symbols on the postfix convolved by the channel has been reduced.
7. Iterate: until a given performance criterion is met go to step 2.

The iterative CIR estimation is compatible to any FEC decoder which delivers at its output bit-probabilities of encoded information bits among which are the SOVA (Soft-Output-Viterbi-Algorithm) decoders and forward backward algorithm. If such a decoder is applied for the sake of CIR estimation only, the complexity increase is considerable. However, if the proposed technique is used in a system where iterative decoding is used anyhow (e.g. in the context of Turbo Codes, etc.), the additional complexity can be considered for implementation.
5.5.2.4 Simulation Results (note: preliminary results are given here, which are not based on the WINNER channel models)

In order to illustrate the performances of our approach, simulations have been performed in the context of a typical WLAN system: a $N_c = 64$ carrier 20MHz bandwidth broadband wireless system using a 16 sample postfix. The CP-OFDM modulator is replaced by a PRP-OFDM modulator. A rate $R = 1/2$, constraint length $K = 7$ Convolutional Code (CC) (o171/o133) is used before bit interleaving followed by 64QAM constellation mapping.

Monte Carlo simulations are run and averaged over 2500 realizations of a normalized BRAN-A [ETSI98] frequency selective channel without Doppler in order to obtain BER curves.

![Figure 5.5.19: Simulation results for iterative interference suppression.](image)

Based on a SOVA decoder, Figure 5.5.19 illustrates for a fixed carrier-over-interference (C/I) ratio of $C/I = 24dB$ that the MSE of the CIR is decreased by approx. 12dB after three iterations using the new algorithm proposed in Section 5.5.2.3 compared to the initial estimates obtained by the algorithm proposed in Section 5.5.2.3. This gain varies only slightly with the size of the observation window for the mean-value calculation of the CIR convolved by the CIR plus noise.

Note that the short size of the mean-value calculation window also makes the proposed scheme applicable in high Doppler scenarios and to packet based transmission schemes where a packet contains a small number of OFDM symbols.

As a result, it can be stated that a novel iterative interference cancellation scheme for PRP-OFDM based systems has been proposed. In a typical example the MSE of the resulting CIR estimated is improved by approx. 12dB over three iterations. This makes PRP-OFDM modulators applicable to higher order constellations, e.g. 64QAM, etc. The proposed scheme can be applied in high mobility scenarios without losing throughput nor spectral efficiency compared to CP-OFDM systems designed for a static environment, since no additional redundancy in terms of pilot tones, learning symbols, etc. is necessary.

5.5.2.5 Annex 1: Interference Estimation

Let describe in this section the details of the proposed interference estimation at iteration $k$.

**Theorem 1.** Define the last CIR estimated $\hat{h}^{k-1}(i)$ represented by matrix $H^{k-1}(i)$ multiplication. Denote by $y^{k-1}(i) = s^{k-1}(i) + \hat{w}^{k-1}$ the frequency domain equalized vector $r_p(i)$ ($\hat{w}^{k-1}$ representing the residual error) preformed with the CIR estimate $\hat{h}^{k-1}(i)$ of the previous step. The optimum time domain
interference estimate in the minimum MSE sense is thus given by
\[ u_{c}^{(i)}(i) = \sum_{a_{n} \in \{0, \cdots, Q-1\}^{n}} p(\tilde{x}_{N_{c}}(i) = a_{n}, \tilde{y}_{N_{c}}^{(i)}(i))H_{c}^{H}F_{c}^{H}(a_{n}) \] (5.92)
\[ \{a_{n} \in C, n \in [0, \cdots, Q-1]\} \] is the set of constellation symbols (alphabet) and \( Q \) the constellation order.

Proof: See [MCM+05]

Practical aspects:
The expression \( p(\tilde{x}_{N_{c}}(i) = a_{n}, \tilde{y}_{N_{c}}^{(i)}(i)) \) is calculated using Bayes’ rule:
\[ p(\tilde{x}_{N_{c}}(i) = a_{n}, \tilde{y}_{N_{c}}^{(i)}(i)) = \frac{p(\tilde{y}_{N_{c}}^{(i)}(i)|\tilde{x}_{N_{c}}(i) = a_{n}) p(\tilde{x}_{N_{c}}(i) = a_{n})}{p(\tilde{y}_{N_{c}}^{(i)}(i))} \] (5.93)

\[ p(\tilde{x}_{N_{c}}(i) = a_{n}) = \prod_{n=0}^{N-1} p(\tilde{x}_{n}(i) = a_{n}) \] is obtained by exploiting the bit-probabilities \( b_{i}(a_{n}) \) of the soft-decoder outputs: \( p(\tilde{x}_{n}(i) = a_{n}) = \prod_{i=0}^{\text{log}((Q)-1)} p(b_{i}(a_{n})) \) assuming that the bits are independent. This property is usually assured by a large interleaver. \( p(\tilde{y}_{N_{c}}^{(i)}(i)|\tilde{x}_{N_{c}}(i) = a_{n}) \) is given by a multivariate Gaussian probability density function (PDF) with \( R_{\tilde{x}_{N_{c}}^{(i)}, \tilde{y}_{N_{c}}^{(i)}} = E[\tilde{w}_{N_{c}}^{(i)}(\tilde{w}_{N_{c}}^{(i)})^{H}] \):
\[ p(\tilde{y}_{N_{c}}^{(i)}(i)|\tilde{x}_{N_{c}}(i) = a_{n}) = \pi^{-ny} \det[R_{\tilde{x}_{N_{c}}^{(i)}, \tilde{y}_{N_{c}}^{(i)}}]^{\frac{1}{2}} \exp\left\{-\frac{(\tilde{y}_{N_{c}}^{(i)}(i) - a_{n})^{H}R_{\tilde{x}_{N_{c}}^{(i)}, \tilde{y}_{N_{c}}^{(i)}}^{-1}(\tilde{y}_{N_{c}}^{(i)}(i) - a_{n})}{2}\right\} \] (5.94)
The expression \( p(\tilde{x}_{N_{c}}(i)) \) is calculated according to (9) by exploiting
\[ \sum_{a_{n} \in \{0, \cdots, Q-1\}^{n}} p(\tilde{x}_{N_{c}}(i) = a_{n}, \tilde{y}_{N_{c}}^{(i)}(i)) = 1. \]

If \( R_{\tilde{x}_{N_{c}}^{(i)}, \tilde{y}_{N_{c}}^{(i)}} \) is diagonal (or approximated by a matrix containing its diagonal elements only), (8) can be considerably simplified, since \( p(\tilde{x}_{N_{c}}(i) = a_{n}, \tilde{y}_{N_{c}}^{(i)}(i)) = \prod_{n=0}^{N-1} p(\tilde{x}_{n} = a_{n}, \tilde{y}_{n}^{(i)}(i)) : \)
\[ u_{c}^{(i)}(i) = \sum_{n=0}^{N-1} \sum_{a_{n} \in \{0, \cdots, Q-1\}} p(\tilde{x}_{n} = a_{n}, \tilde{y}_{n}^{(i)}(i))H_{c}^{H}F_{c}^{H}(a_{n}) \] (5.95)
with \( a_{n}^{(i)} = (0, \cdots, 0, a_{n}, 0, \cdots, 0)^{T} \) is derived from vector \( a_{N_{c}} \) in which only the \( n \) th element is non-zero.

5.6 Pre-distortion

As already considered in deliverables [WIND2.1] and [WIND7.2] and the chapter on a single-carrier modulation, high peak-to-average power ratio (PAPR) is one of the main drawbacks of OFDM and OFDMA signals. Its consequence is the necessity of the high back-off of the high power amplifier (HPA) in order to avoid signal clipping, at least at a tolerable probability. Another effect of the high PAPR is the influence of HPA nonlinearity. The latter creates nonlinear distortions leading to the appearance of new spectral components both inside and outside the band of the amplified signal.

Two approaches seem to be appropriate to solve the problem of nonlinear amplification and high PAPR of OFDM signals. The first one is the application of some kind of a PAPR-reduction technique. Among such techniques clipping and filtering and selective mapping (SLM) are good examples. The second approach relays on the application of a predistorter preceding the nonlinear HPA. Both techniques can be combined to improve performance of the transmitter. One has to admit that all above techniques very often create additional computational load which should be carefully evaluated when comparing the considered solution with less complicated ones resulting however in a lower performance.

Typically, in the HPA predistorter the instantaneous amplitude and phase of the input signal to be amplified is first modified in such a way that the HPA output signal is proportional to the input signal for any of its amplitude values. Figure 5.6.1 illustrates a typical location of the predistorter in the OFDM TDD transceiver. The predistorter operates on the block of OFDM samples supplemented with the cyclic prefix. After modifying their in-phase and quadrature components the samples are D/A- and upconverted. Subsequently, the continuous signal is processed by the RF stage usually consisting of the modulator shifting the signal to the target frequency range and the filter limiting the signal bandwidth.
Finally, the signal is amplified in the HPA. The HPA output is sampled and attenuated in order to obtain the signal needed as a reference for the predistorter adaptation.

In cases different from TDD or TDMA transmission a separate feedback path has to be implemented.

Among predistortion methods and structures the variable gain predistorter is the most popular one. Another novel possibility of driving the variable gain by the input signal magnitude has been described in Section 4.10.3. Let us shortly recall that the predistorter AM/AM characteristic is approximated by a piece-wise linear adaptively adjusted curve. In the experiments reported below exactly such a predistorter was applied.

![Figure 5.6.1 Location of the predistorter in the TDD OFDM transceiver](image)

A simulation experiment has been performed in order to validate the sense of application of a PAPR-reduction technique and an adaptive predistorter (or both of them simultaneously). First we concentrated on the SLM method [MH97]. Recall that this method relies on generation of a certain number $M$ of pseudorandom blocks of the length equal to the OFDM data symbol block. Thus, the input data block is multiplied symbol-by-symbol by $M$ pseudorandom blocks and on each resulting frequency domain vector IFFT is performed. Out of $M$ resulting time domain sequences that one which features the lowest PAPR is transmitted.

For the purpose of simulation, the following parameters have been set: the number $N$ of subcarriers was equal to 1664, $M=4, 8$ and 16 pseudorandom complex sequences were selected and stored. The complex sequences consisted of elements taken from the set \{±1, ±j\}, so they could be considered as 4-PSK symbols. The subcarrier modulating data symbols were drawn from the 16-QAM symbol set.

Figure 5.6.2 presents the influence of the applied SLM procedure on the cumulative complementary distribution function of the IAPR. From this figure we conclude that application of 8 different pseudorandom sequences in the selection process of the best OFDM symbol is worth further considerations. Let us note that only marginal additional information equivalent of 3 bits has to be transmitted which determines the number of the selected pseudorandom sequence. However, the computational complexity of the transmitter increases considerably because the IFFT has to be performed $M=8$ times in order to select the best OFDM symbol. In the next considerations we will apply SLM method with $M=8$ possible choices of the multiplying sequences.
Let us now report our results on the influence of the SLM method on the decrease of the sidelobe of the OFDM signal spectrum. It is well known that the performance limit set for a potentially applied predistorter is an HPA characteristic which is ideally linear up to a certain saturation level. This observation allows us to investigate the influence of the SLM PAPR-reduction technique on the sidelobe level, when the transmitted signal is distorted by an ideal clipper, which is linear up to the clip level. If we tolerate the level of sidelobes of about 40 dB below the main lobe, it turns out that the input back-off level of the OFDM signal should be equal to about 7 dB.
As we see in Figure 5.6.3, the performance improvement due to the applied SLM technique is very moderate. It is important to stress that the curves shown in this figure constitute a limit for the performance of the predistorter if a nonlinear HPA is applied.

Now let us consider the application of the HPA predistorter and a combination thereof and the SLM technique with $M=8$ randomizing sequences. The Rapp HPA model with $p=2$ (see Figure 6.5.4) was applied in our simulations. The PSD on the HPA output was measured after transmitting of 300 OFDM symbols which was found to be sufficient the predistorter to converge to the characteristics which linearizes the HPA. The signal output spectrum is shown in Figure 5.6.4.

![Figure 5.6.4](image)

**Figure 5.6.4** Power spectrum density on the HPA output: without predistortion and PAPR reduction (curve 1), with the application of the predistorter (curve 2), with the application of the predistorter and SLM technique ($M=8$)

We conclude from Figure 5.6.4 that the improvement in the performance is mostly achieved due to the predistorter, however, at the expense of a substantial increase in computation complexity further moderate improvement is still possible by applying the SLM PAPR reduction technique. Let us stress that the applied SLM technique required a negligible increase in the information transmitted to the receiver. We suppose that higher improvement in the sidelobe suppression could be achieved at the substantial increase of the number of randomizing sequences and performed IFFTs.

Generally, application of the HPA predistorter in front of the solid state high power amplifier results in attenuation of the spectrum sidelobes by about 8 dB at the frequencies close to the signal passband edge, whereas additional application of the SLM technique results jointly in 11 dB sidelobe suppression.

One has to admit that there are also other performance measures of the application of a predistorter and PAPR reduction techniques. One such an approach is to plot the total degradation TD [dB] versus the HPA output back off OBO [dB] for a target bit error rate, say $10^{-4}$ for different configurations of the HPA, predistorter and PAPR reduction techniques [ChP03]. The total degradation is described by formula

$$TD = \left[ \frac{E_b}{N_{0,NL}} - \frac{E_b}{N_{0,L}} \right] - OBO \quad [dB]$$

(5.96)

where $E_b / N_{0,NL}$ represents the required $E_b/N_0$ to obtain the target BER when the nonlinear HPA is used, and $E_b / N_{0,L}$ denotes the required $E_b/N_0$ to maintain the same BER for a linear HPA. The minimum of TD shown in such plot indicates the optimum level of OBO. Such a minimum measured for transmission using the nonlinear HPA only occurs for a higher value of OBO. The difference between these two values is the achieved gain due to application of PAPR reduction and/or predistortion techniques.
6. Evaluation of proposed techniques

The results presented in this chapter illustrate the performance of the investigated link layer techniques in the defined WINNER scenarios. Results for cyclic prefix OFDM are given as baseline reference cases.

6.1 Wide area cellular, uplink (SM),

6.1.1 Linear equalization with perfect channel state information

Figure 6.1.1 shows bit error rate simulation results for rate $\frac{1}{2}$-coded QPSK serial modulation, in which 832-symbol blocks, with 80-symbol cyclic prefixes, are transmitted over four WINNER channel scenarios: urban macro, rural, suburban macro and urban micro. The symbol rate in each case is 16.26 MSymbols/s. Square root raised cosine spectrum shaping is used, with 23% rolloff, so that the total occupied bandwidth is 20 MHz. The code used is the (133,171) $K=7$ convolutional code, with soft-decision Viterbi decoding. Coding and interleaving are over 10 blocks. The receiver uses perfect channel state information. Doppler is present, corresponding to 70 km/hr. with a 5 GHz carrier frequency. In these curves, the $E_b/N_0$ ratio includes a 0.4 dB addition which accounts for the 80-symbol cyclic prefix overhead in each block.

![Graph](image)

**Figure 6.1.1: Bit error rate for ideal channel state information for serial modulation: QPSK, soft Viterbi decoding**

The rural and urban macro curves are similar, while the urban micro and suburban macro curves are up to 2 dB worse, as a result of reduced frequency diversity for those channels.

In the results shown in this section, the receiver uses minimum mean squared error linear frequency domain equalization. The use of additional decision feedback equalization, with a small number of time domain feedback taps would have provided a small improvement in performance, by overcoming the noise enhancement characteristic of linear equalization [FAB+02], [FA02]. Furthermore, substantial BER performance improvement is obtained by turbo equalization, as described in Section 4.9 and evaluated in Section 6.1.6.

The effect of higher code rates for the urban macro and rural channels, with ideal state information is shown in Figure 6.1.2.
6.1.2 Linear equalization: coding and modulation evaluation

Different coding and modulation combinations were tested as described in Section 0. The linear MMSE frequency-domain equalizer was assumed. The de-mapping uses the sub-optimal max-approach and the channel decoder is of the max-log-map –type. Perfect channel information and synchronization is assumed at the receiver. The results are reported in Figure 6.1.3. In general the slope of the curves is constant over all coding rates, except for the extreme cases of high-rate codes and 16-QAM modulation. It can be seen that BPSK rate 5/6 performs worse than QPSK rate ½ and is useless. Also BPSK rate ¾ performs only a little better than QPSK rate ½ and may be discarded as well. A saturation of performance with high-rate coding and 16-QAM can be seen. Later we will show the effect is suppressed when turbo equalization is performed.

If the desired maximum spectral efficiency through modulation and coding is 2 bits/s/cu, 8PSK with 2/3 coding can replace ½ rate 16-QAM and offer a constant envelope modulation. The performance comparison is presented in Figure 6.1.4 where both linear and turbo equalization are considered. These two AMC modes have essentially identical performance when linear equalization is used, while there is a performance gain <0.5dB for the 16-QAM when turbo equalization is used.
Figure 6.1.3: AMC mode evaluation for single-carrier linear equalization

![Graph showing AMC mode evaluation for linear equalization](image)

Figure 6.1.4: 8PSK 2/3-rate vs. 16-QAM ½-rate

![Graph showing 8PSK 2/3 vs. 16-QAM ½](image)

Different channel scenarios are considered in Figure 6.1.5 and Figure 6.1.6 for the convolutionally coded QPSK and turbo coded 16-QAM, respectively. Except for the suburban macro channel, the performance is very similar across the channel scenarios. Examining Figure 6.1.6 and Figure 6.1.3 more carefully, we notice a 1dB gain for turbo codes in the ½ rate 16-QAM mode at FER 10%, and 2dB at FER 1%. It should be noted, however that the symbol demapping with turbo codes is the optimal one-shot log-map de-mapping, which increases the difference slightly.
Figure 6.1.5: QPSK transmission in all channel scenarios, linear and turbo equalization

Figure 6.1.6: 16-QAM with PCCC in all channel scenarios, linear and turbo equalization
A final evaluation for linear equalization is made regarding the Doppler modeling in simulations. The performance of single-carrier signaling in correctly modeled Doppler (with classical spectrum) is compared to a block-static channel. The results in Figure 6.1.7 show the block-static assumption is valid with the selected system parameters.

### 6.1.3 Linear equalization with training sequences

Channel estimation training, using varying numbers of 128-symbol training blocks, was added to the simulations. Each training block was a 128-symbol Chu sequence [Chu72]. The curve with 4 training blocks per frame represents the frame format shown in the example of Figure 6.1.8. The 80-symbol training CP, shown in green, is the last 80 symbols of the Chu sequence. Channel frequency response estimates are derived from the 128-symbol training blocks, as described in [WIND2.1], Section 3.4.2.2. Simple frequency domain interpolation, with zero-padding to the 832-symbol block size is carried out, as described there (similar to a corresponding channel estimation technique described in [AK05], when there is no a priori knowledge of channel delay profile statistics). In this example, the overhead ratio would be $(4 \times 128 + 2 \times 80) / (4 \times 128 + 2 \times 80 + 832) = 47\%$. This ratio, which includes cyclic prefixes overhead, would be halved for continuous transmission of blocks, where four training sequences are shared by two adjacent blocks.

![Frame format with 4 training blocks per frame](image)

To effect a format with less overhead, the frame format shown in Figure 6.1.9 was tested. In this case, only two sets of 4 training blocks are used, one set at each end of a train of 10 data blocks. Bearing in mind that the channel is changing slowly due to Doppler, linear interpolation, combined with the channel
estimation procedure of [WIND2.1], Section 3.4.2.2., is used to estimate the channel during each data block. The overhead ratio in this case would be $(8 \times 128 + 12 \times 80) / (10 \times 832 + 8 \times 128 + 12 \times 80) = 19\%$. Again, this ratio would be roughly halved in a continuous stream of blocks, in which adjacent groups of 10 blocks share training blocks.

Figure 6.1.9: Frame format with linear interpolation over 10 blocks

Figure 6.1.10 shows the bit error rates for the rate $\frac{1}{2}$ QPSK serial modulation system on the urban macro channel, for perfect state information and for various numbers of training blocks. The results for 4 training blocks per frame (~47% overhead) and for the scheme with interpolation over 10 blocks (~21% overhead) are almost identical, and are slightly more than 2 dB worse than for the case of ideal channel state information at a BER of $10^{-4}$. Increasing to 8 training blocks gives roughly 0.5 dB improvement, while reducing to 2 training blocks per frame causes 2 dB additional degradation.

Figure 6.1.10: Bit error rate for ideal channel state information and for different channel estimation configurations
6.1.4 Effect of frequency offset

Frequency offsets equal to 5% and 10% of the inter-subcarrier spacing (977 Hz and 1.953 KHz respectively) were added to the previously mentioned serial modulation simulations. Frequency offset compensation, as described in Section 4.8, according to the technique proposed in [MM97] was used. The results are shown in Figure 6.1.11. Curves labelled 'DA' use ideal decisions. Those labelled 'NDA' use use a frequency quadrupler [Vit83]. Curves labelled 'no FC' represent the case where there is no attempt to compensate for frequency offset. A frequency offset equal to about 10% of the inter-subcarrier spacing causes little degradation when the frequency offset compensation is applied.

![Figure 6.1.11 Performance for 5% and 10% frequency offset, with ideal channel state information, with and without adaptive frequency offset compensation [MM97].](image)

6.1.5 IFDMA

In the following, the performance of an IFDMA system is investigated. The sensitivity to frequency and time diversity of the radio channel and the sensitivity to frequency synchronisation errors are considered. Moreover, results of the analysis of the computation complexity in comparison to OFDMA are reported under the parameters assumptions of wide area cellular scenario.

For IFDMA, different data rates can be achieved by allocation of different numbers of subcarriers per user. In Figure 1, BER curves are reported for coded QPSK transmission with rate ½ convolutional coding (133, 171) over an urban macro channel and for different numbers \( Q \) of subcarriers per user. The corresponding net bit rates \( R_b \) per user for the different values of \( Q \) are given in Table 6.1.

<table>
<thead>
<tr>
<th>( Q )</th>
<th>4</th>
<th>8</th>
<th>16</th>
<th>64</th>
<th>1024</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_b )</td>
<td>78 kBit/s</td>
<td>156 kBit/s</td>
<td>313 kBit/s</td>
<td>1.25 MBit/s</td>
<td>20 MBit/s</td>
</tr>
</tbody>
</table>

An interleaving depth of 0.5 ms is assumed which corresponds to an interleaving over 10 IFDMA symbols. Since perfect frequency synchronization of the signals of different users is assumed, the results are almost independent of the number of active users. For that reason, only results for one active user are considered.
In Figure 6.1.12 the performance of an IFDMA system with different data rates per user is presented. The curve for $Q=1024$ represents an IFDMA signal in a system where all subcarriers belong to one user. Hence, this system is comparable to OFDM for single user applications. The more sub-carriers are assigned to one user, the higher is the frequency diversity for the user’s signal and hence, the performance improves. It can be observed that the higher the value of $Q$, the lower the sensitivity to its decrease.

In Figure 6.1.13 the simulation results for different channel models are compared to each other under the assumption that the maximal mobile speed envisaged for each model is considered. On the one hand, due to the higher velocity, for the suburban macro channel more time diversity can be expected compared to the urban macro channel. On the other hand, the suburban macro channel provides less frequency diversity. In this case, the effect of lower frequency diversity over-compensates the effect of higher time diversity and hence, performance is better for the urban macro channel.
For the urban micro channel performance is better than for suburban macro channel since the frequency diversity effect over-compensates the effect of low time diversity of the urban micro channel. The system can get advantage of the high velocity and hence, high time diversity of 250 km/h in the rural channel, for which it exhibits the best performance.

In Figure 6.1.14, the simulation results for the different channel models are shown for a velocity 70 km/h which is assumed for all channels. The beneficial impact of the higher frequency diversity of the rural channel can be inferred.
As shown in Section 5.2.1.4.1, IFDMA can be described by a precoded OFDMA scheme with subcarriers equidistantly allocated over the whole bandwidth. Hence, it can be expected that similar to OFDM, IFDMA is sensitive to frequency offsets. Therefore, in the following the sensitivity of a fully loaded IFDMA system is investigated by simulation of transmission over an additional white Gaussian noise (AWGN) channel in an up-link scenario. We consider frequency errors due to oscillator inaccuracies and Doppler shifts. Let $\Delta \Theta_{\text{max}}$ designate the maximum frequency error. For each user, the frequency error is described by a random variable $\Delta \Theta$ which is assumed to be uniformly distributed in $[-\Delta \Theta_{\text{max}}, \Delta \Theta_{\text{max}}]$. The simulation parameters are given in Table 6.2. Due to the frequency offsets, orthogonality between the signals of different users is lost, because the signal spectrum of each user is not positioned in the zero points of the others. Hence, we have multiple access interference (MAI). In the simulations it is assumed that the frequency shift of the useful signal can be estimated and the receiver is adjusted accordingly, but the different received users signal are asynchronous. The impact of MAI on BER performance is reported in Figure 6.1.15, where it can be observed that, e.g., at BER of $10^{-3}$ and $\Delta \Theta_{\text{max}}T = 6\%$ the asynchronous reception causes a degradation of the $E_s / N_0$ of $\approx 0.4$ dB.

Table 6.2: Simulation parameters (AWGN channel)

<table>
<thead>
<tr>
<th>Compression/repetition factor:</th>
<th>$K=L=16$</th>
<th>Subcarrier spacing:</th>
<th>$1/T=20$ kHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of symbols:</td>
<td>$Q=64$</td>
<td>Relative frequency offset:</td>
<td>$\Delta \Theta \cdot T = 0,0.02,\ldots,0.1$</td>
</tr>
<tr>
<td>Total number of subcarriers:</td>
<td>$N=1024$</td>
<td>Bandwidth:</td>
<td>$B=20$ MHz</td>
</tr>
</tbody>
</table>
Figure 6.1.15: Simulation results with knowledge of the frequency error for user of interest at the receiver

For the system parameters assumed for wide area cellular scenario, the computational complexity per user of the IFDMA transmitter calculated as in Section 5.2.1.4.2 is given in Table 6.3, where $R_b$ designates the net bit rate per user.

Table 6.3: Computational complexity for N=1024

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$R_b$</th>
<th>$M_{OFDMA}$</th>
<th>$M_{OFDMA,red}$</th>
<th>$M_{IFDMA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19.5 kBit/s</td>
<td>10240</td>
<td>1024</td>
<td>1024</td>
</tr>
<tr>
<td>16</td>
<td>315 kBit/s</td>
<td>10240</td>
<td>1088</td>
<td>1024</td>
</tr>
<tr>
<td>64</td>
<td>1.25 MBit/s</td>
<td>10240</td>
<td>1408</td>
<td>1024</td>
</tr>
<tr>
<td>128</td>
<td>2.5 MBit/s</td>
<td>10240</td>
<td>1920</td>
<td>1024</td>
</tr>
<tr>
<td>1024</td>
<td>20 MBit/s</td>
<td>10240</td>
<td>10240</td>
<td>1024</td>
</tr>
</tbody>
</table>
### 6.1.6 Evaluation of Iterative Techniques for Single-Carrier Modulation

The simulation parameters for assessing the performance of the iterative techniques are listed in Table 6.4.

<table>
<thead>
<tr>
<th>Table 6.4: Simulation parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Channel models</strong></td>
</tr>
<tr>
<td><strong>Training block (M)</strong></td>
</tr>
<tr>
<td><strong>Training sequences</strong></td>
</tr>
<tr>
<td><strong>Modulation</strong></td>
</tr>
<tr>
<td><strong>Channel code</strong></td>
</tr>
<tr>
<td><strong>Block size (N)</strong></td>
</tr>
<tr>
<td><strong>Cyclic prefix (N_{cp})</strong></td>
</tr>
<tr>
<td><strong>Pulse shaping</strong></td>
</tr>
<tr>
<td><strong>Number of iterations</strong> (with channel estimation)</td>
</tr>
<tr>
<td><strong>Number of iterations</strong> (Perfect CSI, no CE)</td>
</tr>
<tr>
<td><strong>Performance measure</strong></td>
</tr>
</tbody>
</table>

First, we consider the scenarios assuming that perfect CSI is available and no channel estimation is needed. For the urban macro channel model, it is clear from Figure 6.1.16 that the FDE-TDDF and FDE-MMSE-IC essentially yield the same BER performance. They both improve the performance by about 1.5dB compared with the single-carrier (SC) calibration case in which non-iterative technique (which is identical to the iterative schemes when the number of iteration is equal to one) is employed. Interestingly, we may compare the iterative techniques performance with that of the multi-carrier (MC) calibration case, as both support the same spectral efficiency and channel coding scheme. Notice that no iterative technique is needed for MC as it has no ISI (inter-symbol-interference) while MC suffers from higher PAPR. On the other hand, the SC requires the iterative techniques to suppress the ISI while enjoying the low PAPR. Thanks to the effectiveness of the iterative techniques, the SC yields a better performance than the MC for these chosen sets of simulation parameters (with perfect CSI), as noted in Figure 6.1.16. We therefore conclude that SC is a suitable technology, as it has lower PAPR while it is possible to employ iterative techniques to yield excellent performance. The price to pay is that the receiver has a little higher complexity, but this is considered to be feasible and economical in the uplink direction. Figure 6.1.17 and Figure 6.1.18 contain similar simulations results for the suburban macro and rural channel models.

In terms of complexity, the FDE-TDDF has a complexity of order (N) (measured by the number of complex multiplications per data symbol) while the FDE-MMSE-IC has a complexity of order (log N). Note that the feedback filter of the FDE-TDDF can also be implemented in the frequency domain to yield a complexity of order log N. While the FDE-TDDF may incur a slightly higher complexity, it is robust to channel estimation error, which is explained next.

Figure 6.1.16, Figure 6.1.17 and Figure 6.1.18 show the simulation results for the proposed FDE-TDDF-CE scheme (no perfect CSI is assumed). It is clear that its performance is excellent and it approaches the performance of a linear FDE with perfect CSI. It is only 1-2 dB off from the FDE-TDDF scheme with perfect CSI. The only apparent drawback is the occurrence of an error-floor-like performance at high SNR (due to the channel mismatch using the least-square estimation). The remarkable performance of the FDE-TDDF-CE can be explained as follows. The filter coefficients in the FDE-TDDF are updated in accordance with the estimated channel statistics provided by the channel estimator. Hence, it will take into consideration the possible channel error level predicted by the estimator. (This method is different from the conventional iterative schemes in which the estimated channel coefficients are treated as the true channel values.) Hence, we conclude that the FDE-TDDF-CE is a suitable scheme in single-carrier signals detection when channel estimation is required.
Finally, the order of complexity of the FDE-TDDF-CE scheme can be shown to be upper bounded by $\max(N^2, N_{cp}^3)/N$. Since $N_{cp}$ is much smaller than $N$ in most cases, this complexity is affordable. Furthermore, there is a way to simplify the calculations required for the channel estimation and the complexity can be reduced to be linear with $N_{cp}$, i.e., total complexity is on the order of $\max(N^2, N_{cp})/N$. The performance loss due to this simplification is small and for further results, please see [NLF05].

![Figure 6.1.16: Urban macro channel](image1)

![Figure 6.1.17: Suburban macro channel](image2)
Different adaptive coding and modulation modes were tested for turbo equalization. The equalizer performs one linear equalization operation and two turbo iterations. Comparing the results with those of the linear equalizer in Figure 6.1.3, we can see that turbo equalization gives either a 1-3dB gain keeping the modulation constant, or additional throughput if the modulation mode is changed in a given channel scenario. Furthermore, the modes with highest spectral efficiency become usable with turbo equalization, since they gain most from turbo iterations. For QPSK, the turbo gain is around 2dB in most channel scenarios, as presented in Figure 6.1.20. A gain of 2-3dB can be found in the case where turbo coding and 16-QAM is used for transmission.

In summary, we can show that the link throughput can be increased by the simple combination of turbo methods and adaptive coding and modulation. By using turbo methods, higher throughput modes can be used than with the simple linear equalizer.
Figure 6.1.19 AMC mode evaluation for turbo equalization

Figure 6.1.20 QPSK transmission in all channel scenarios, linear and turbo equalization
6.2 Wide area cellular, down-link

6.2.1 IOTA-OFDM

In that section we simulated both Cyclic Prefix OFDM and IOTA-OFDM over Sub-urban and Urban channel models with a velocity of 70 km/h. Channel coding is PCCC with rate 1/2 (4 iterations are assumed in the decoder) and the bits are mapped over QPSK symbols. Channel is generated in frequency so the results do not take ISI nor ICI into account.

IOTA-OFDM parameters are very similar to Cyclic Prefix OFDM (CP-OFDM) ones, except for the symbol duration (see Table 6.5).

Table 6.5: CP-OFDM and IOTA-OFDM parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>CP-OFDM</th>
<th>IOTA-OFDM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carrier frequency [GHz]</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>FFT size</td>
<td>1024</td>
<td>1024</td>
</tr>
<tr>
<td>System bandwidth [MHz]</td>
<td>20.0</td>
<td>20.0</td>
</tr>
<tr>
<td>Subcarrier spacing [kHz]</td>
<td>19.531</td>
<td>19.531</td>
</tr>
<tr>
<td>Useful OFDM symbol duration [μs]</td>
<td>51.20</td>
<td>26.60 for real valued symbols</td>
</tr>
<tr>
<td>Cyclic prefix length [μs]</td>
<td>5.00</td>
<td>0.0</td>
</tr>
<tr>
<td>Total OFDM symbol duration [μs]</td>
<td>56.20</td>
<td>26.60 for real valued symbols</td>
</tr>
<tr>
<td>Number of used subcarriers (symmetric, DC not used)</td>
<td>832</td>
<td>832</td>
</tr>
<tr>
<td>Signal bandwidth [MHz]</td>
<td>16.25</td>
<td>16.25</td>
</tr>
</tbody>
</table>
Figure 6.2.1: Bit error rate and Packet error rate for IOTA-OFDM and CP-OFDM over Urban Macro channel model (70 km/h)

Figure 6.2.2: Bit error rate and Packet error rate for IOTA-OFDM and CP-OFDM over Suburban Macro channel model (70 km/h)

From these results we observe that OFDM-IOTA leads to a 0.5 dB improvement on CP-OFDM. This gain comes essentially from the cyclic prefix removal in IOTA-OFDM. It can be deduced that the frequency diversity provided by the channel is also fully exploited with the IOTA modulation. That performance gain in term of Eb/No could also be viewed as an improvement of the throughput of about 10 % (corresponding to the cyclic prefix overhead in CP-OFDM).

In [WIND2.2] it was also shown that the IOTA-OFDM spectrum is steeper than the conventional OFDM one. Thus, for a given power mask, more sub-carriers could be modulated by using IOTA-OFDM. This would lead to a further improvement of the effective throughput.
6.2.2 CP-OFDM

Simulation results for CP-OFDM with QPSK and 16-QAM modulation, rate $\frac{1}{2}$ coding, are reported in Figure 6.2.3. Further results on different coding and modulation modes in the urban macro channel are reported in Figure 6.2.4.

**Figure 6.2.3:** Frame error rate over SNR for the different wide area scenarios (rural, suburban, typical urban, outdoor to indoor), for QPSK and 16-QAM, using standard CP-OFDM

**Figure 6.2.4:** Frame error rate over SNR for the different modulation alphabets and code rates in the WINNER Urban Macro channel, using standard CP-OFDM
6.2.3 Pilot aided channel estimation for OFDM

In this section the performance of two dimensional (2D) pilot aided channel estimation (PACE) applied to OFDM is evaluated for the wide area mode, as described in section 5.3.1.

6.2.3.1 Channel estimation parameters & Pilot grid selection

In the following we study the application of PACE to the WINNER wide area FDD mode, with 20MHz bandwidth from 3.2.2.1. The system parameters are listed in Table 3.1. Two channel models are considered: the typical urban (TU) and the rural channel model will be considered. For the TU and rural channel models the maximum velocities are set to 70km/h and 250 km/h, resulting in a maximum Doppler frequency of 324 Hz and 1157 Hz, respectively. According to Deliverable ([WIND2.1] section 3.4.2.1.1.2) the sampling theorem in frequency and time requires for the pilot spacings:

\[
D_f \leq \frac{1}{\beta_f} \frac{N_{\text{FFT}}}{N_{\text{GI}}} = 10 \quad \text{and} \quad D_t \leq \frac{1}{2\beta_t} \frac{1}{f_{\text{D,max}} T_{\text{sym}}} = \{27,7\}
\]

(6.1)

where \(N_{\text{FFT}}, f_{\text{D,max}}, T_{\text{sym}}\) denote the size of an FFT block, the OFDM symbol duration including the guard interval, and the maximum Doppler frequency. Accordingly, the oversampling factors in frequency and time become \(\beta_f \geq 1\) and \(\beta_t \geq 1\). Since the pilot spacing in frequency is lower bounded by the guard interval length, \(N_{\text{GI}}, D_f\) is independent of the chosen channel model.

The pilot spacing in time, on the other hand, is inversely proportional to the normalized Doppler frequency, \(f_{\text{D,max}}, T_{\text{sym}}\), and becomes 27 and 7 for the TU and rural models, respectively. In order to allow short frame durations of approx. 0.5ms, the number of OFDM symbols per frame should not exceed 10. This implies that the number of OFDM symbols in time direction will be much smaller than the number of pilots in frequency direction. Hence, the oversampling factor in time, \(\beta_t\), should be larger than \(\beta_f\). However, for velocities to be expected in an urban environment, \(\beta_t\) is determined by \(N_{\text{frame}}\) rather than the sampling theorem. For extremely high velocities such as experienced in the rural model, \(D_t\) should not exceed 4. Note, unlike for the results presented in this deliverable, in [WIND2.1] a continuous stream of FFT blocks was assumed. However, due to the requirements of the hybrid FDD duplex scheme, a mobile terminal cannot transmit and receive at the same time. Therefore, we will consider the problem of estimating the channel for one isolated OFDM frame.

The pilot spacing in time is set to 3, 4 and 8 in the following, i.e. \(D_t=\{3,4,8\}\). Note, in order to allow that a frame begins and ends with pilots, for \(D_t=\{4,8\}\) the OFDM frame length needs to be adjusted to \(N_{\text{frame}}=9\). Trials suggested that in frequency direction appropriate values for the oversampling factor are in the range \(\beta_f=\{1.2,2\}\), corresponding to an oversampling factor of 20% and 100%, respectively. Thus, the pilot spacing in frequency direction will be set to \(D_f=\{5,8\}\). Out of the possible combinations four parameter sets are picked; these are denoted by (a)−(d), and are shown in Table 6.6. According to section 3.4.2.1.1.2 in [WIND2.1] and also shown in Table 6.6 are the pilot overheads due to channel estimation, associated to these parameter sets. It is seen that the overhead is roughly between 3% and 9%.

Table 6.6: Pilot overhead \(N_p/N_D\) for pilot spacings of and for an OFDM frame with 822 subcarriers and \{10,9,9\} OFDM symbols per frame.

<table>
<thead>
<tr>
<th>(D_f)</th>
<th>(D_t)</th>
<th>(M_f)</th>
<th>(M_t)</th>
<th>Pilot overhead</th>
<th>Complexity / subcarrier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>freq. first</td>
<td>time first</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(d)</td>
<td>5</td>
<td>3</td>
<td>8</td>
<td>4</td>
<td>8.7%</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(c)</td>
<td>8</td>
<td>3</td>
<td>8</td>
<td>4</td>
<td>5.3%</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b)</td>
<td>8</td>
<td>4</td>
<td>8</td>
<td>3</td>
<td>4.4%</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(a)</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>2</td>
<td>2.9%</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For channel estimation a cascaded filter consisting of two 1D filters, one operating in time and the other in frequency direction can be implemented, named 2x1D-PACE. First, channel estimation is performed in frequency direction, yielding tentative estimates for all subcarriers of that OFDM symbol. The second step is to use these tentative estimates as new pilots, in order to estimate the channel for the entire frame.
The computational complexity of 2x1D-PACE, in terms of required multiplication per subcarrier, is dependent on the order of the filtering [WIN2.1], section 3.4.2.1.1.3, however, the performance is not. The number of required multiplications per subcarrier is

\[
\text{frequency first} = \frac{M_f M_f}{N_{\text{frame}}} + M_f \tag{6.2}
\]

if channel estimation in frequency direction is performed first, and

\[
\text{time first} = \frac{M_p M_p}{N_c} + M_f \approx \frac{M_f}{D_f} + M_f \tag{6.3}
\]

if channel estimation in time direction is performed first. In the above equation the approximation \(N_p \approx N_c / D_f\) has been used. For the considered parameter sets the numerical values are listed in Table 6.6. It can be observed that for the chosen parameters, (6.3) is dominated by \(M_f\), which is typically larger than \(M_t\). On the other hand, (6.2) is far less dependent on \(M_t\). Therefore, performing channel estimation in frequency direction first is always less complex. The difference becomes particularly significant, when the filter order in frequency, \(M_f\), is large and the filter order in time \(M_t\) is low.

For the Wiener interpolation filter (WIF) the auto and cross-correlation functions need to be estimated at the receiver. It may be prohibitive to estimate the filter coefficients during operation in real time. Alternatively, a robust estimator with a model mismatch may be chosen. That is to assume a rectangular shaped power delay profile with maximum delay \(T_w\) and a rectangular shaped Doppler power spectrum with maximum Doppler frequency \(F_w\). By using a mismatched estimator the filter coefficients can be precomputed and stored.

It is important to note that the parameters of the robust estimator should always be equal or larger than the worst case channel conditions, i.e. largest propagation delays and maximum expected velocity of the mobile user. In order to determine the channel estimator only \(T_w\) and \(F_w\) are required. If the maximum delay of the channel \(\tau_{\text{max}}\) is not known it can be upper bounded by the guard interval duration \(T_{\text{G}}\). Since the filter should also satisfy the 2D sampling theorem, the filter passbands can be chosen within the range

\[
\frac{T_{\text{G}}}{2} \leq T_w \leq \frac{T_{\text{G}}}{D_f} \quad \text{and} \quad \frac{F_{\text{sym}}}{2} \leq F_w \leq \frac{1}{2 D_f T_{\text{sym}}}
\]

The SNR at the input of the channel estimation filter is assumed to be known at the receiver.

If the pilot sequence is of infinite duration the mismatched WIF becomes an ideal lowpass interpolation filter, whose coefficients are described by a sinc function [AK05]. However, for pilot sequences of finite length, the filter coefficients will be different, but similar to a lowpass interpolation filter with windowing. The difference becomes most severe near the beginning and end of the sequence. There edge effects give rise to an increased estimation error. According to [LCS98], the mismatched WIF has superior performance in terms of the MSE, with respect to low pass filtering with windowing.

6.2.3.2 Results & Discussion

In this section, results are presented for the typical urban and the rural channel models of the wide area mode. The mobile velocity was set to 70km/h for the typical urban and 250km/h for the rural channel model.
Figure 6.2.5. MSE vs SNR for different pilot grids and filter dimensions in time and frequency.

Figure 6.2.5 shows the MSE averaged over all subcarriers and symbols of one frame against the SNR for the typical urban and rural channel models. For channel estimation 2x1D PACE with model mismatch was used, employing the pilot grid selection from Table 6.6. In time direction, the filter dimension was set equal to the number of pilots per frame which is \( M_t = \{4, 3, 2\} \) for \( D_t = \{3, 4, 8\} \). In frequency direction a filter order of \( M_f = 16 \) was chosen. It is seen that the denser the pilot spacing the better the performance. This is an expected result, since a denser pilot grid corresponds to a higher degree of oversampling. Furthermore, an error floor is observed at high SNR. This MSE floor is typically experienced for the used estimator with model mismatch; and its level depends on the pilot grid, the estimator order, as well as the channel model. For the typical urban channel model only parameter set (a) experiences an error floor in the considered SNR range. For the rural channel model parameter set (a) exhibits poor performance since the sampling theorem in time direction is violated; also parameter set (b) has an error floor above 30dB.

Figure 6.2.6. MSE vs filter dimension \( M_f \) in frequency for different pilot grids.

How the filter order affects the performance is investigated in Figure 6.2.6 for the typical urban and the rural channel model. Since the filter dimension in time direction \( M_t \) is limited by the number of pilots per frame, we only consider the filter order in frequency direction, \( M_f \). Figure 6.2.6 allows us to determine to what extent the MSE can be lowered by increasing \( M_f \). For parameter set (d), with \( D_f = 5 \), an estimator with \( M_f = 4 \) (\( M_f = 8 \)) coefficients appears to be a good compromise between performance and complexity, at an SNR of \( E_s/N_0 = 20\)dB (40dB). For grids (a)-(c) \( M_f = 8 \) (\( M_f = 16 \)) coefficients appears reasonable. Hence, a denser pilot spacing allows for a less complex estimator. Of course, this is traded with reduced bandwidth efficiency. How much a certain pilot grid affects the BER is addressed in section 6.2.4.
6.2.3.3 PACE with pre-smoothing in frequency direction

Instead of a denser pilot spacing the performance may be improved by more sophisticated signal processing. The optimum estimator for PACE is given by the Wiener interpolation filter (WIF). Unfortunately, an optimum WIF requires information about the channel statistics, which means that generating the filter coefficients is in many cases prohibitive. One alternative, a WIF with model mismatch has been studied in the previous section. However, for some applications the accuracy attained by the mismatched WIF may be insufficient.

In this section the objective is to close the performance gap between matched and mismatched WIF. Unlike the WIF which averages over the noise and interpolates jointly, we follow a separated approach for smoothing and interpolation [HW98]. There the received pilot symbols are passed through a smoothing type filter which is matched to the channel statistics. Subsequently, the smoothed set of channel estimates at pilot positions are interpolated to yield the estimates for the entire frame. Unlike the smoother, the interpolation filter does not require any knowledge of the channel statistics. The motivation for the proposed smoothing and interpolation estimator (SINE) is twofold: first, according to the sampling theorem perfect interpolation of a bandlimited noiseless signal is possible without any knowledge of the channel statistics; second, a MMSE-based smoother which only reduces the effects of noise at the pilot positions can be implemented with significant less computational cost. In fact, it is shown in [AK05] that this separated approach for smoothing and interpolation can approach the performance of the optimum MMSE estimator for long sequences; and if edge effects are not dominant.

The fact that long sequences are favorable limits the application of the SINE to estimation in the frequency direction. For estimation in the time direction where only a few pilots are available per frame, the WIF with model mismatch as described in the previous section was used.

Since the smoother assumes knowledge of the channel statistics, an adaptive filter, which updates the filter coefficients online, needs to be implemented. For pre-smoothing an adaptive Kalman filter [SA03], or a low rank estimator (LRE) based on the singular value decomposition (SVD) may be employed [ESB+98]. Since the channel statistics change relatively slowly, the filter coefficients need only to be updated in the order of several ms or so. In any case, for the generation of the smoother coefficients, perfect knowledge of the channel statistics is assumed in the following.

For the results shown in Figure 6.2.7 and Figure 6.2.8 the pilot grid (b) with $D_f=8$, $D_t=4$ and $M_t=3$ is selected since it offers a good trade-off between pilot overhead and achievable performance for all channel models.

![Figure 6.2.7. MSE vs SNR for pilot grid (b) and various interpolation filter dimensions in frequency direction.](image)

In Figure 6.2.7 the MSE is plotted against the average received SNR for various dimensions of the interpolation filter $M_f$ for the typical urban channel model. For pre-smoothing and perfectly matched MMSE estimator was used which utilizes all available pilots. For comparison results for the mismatched and perfectly matched Wiener interpolation filter, marked with mismatched WIF and MMSE, are also included. It is seen that the SINE approaches the MMSE for low SNR. On the other hand, for high SNR
where the interpolation error becomes dominant, the results for channel estimation with and without pre-smoothing merge.

![Graph showing MSE vs filter dimension](image)

**Figure 6.2.8. MSE vs filter dimension $M_f$ in frequency for different pilot grids.**

In Figure 6.2.8 the MSE is plotted against the interpolation filter order $M_f$. Also the dimension of the pre-smoothing filter, $M_{sm}$, is varied. Again the performance for the mismatched and perfectly matched Wiener interpolation filter, marked with mismatched WIF and MMSE, are also included. According to Figure 6.2.8, an interpolation filter order exceeding $M_f=16$ hardly improves the performance. On the other hand, only a smoother which uses all available pilots approaches the MMSE.

### 6.2.4 Approximation of channel estimation errors

In this section a simple model for the approximation of channel estimation errors is evaluated, as described in section 5.3.1. The proposed model was applied to the OFDM wide area mode.

We assume that the channel estimation error $\hat{H}_{\ell,k} = H_{\ell,k} - \hat{H}_{\ell,k}$ is a Gaussian random variable with zero mean and variance equal to the MSE [WIND2.1], section 3.4.4. This yields an effective system model taking into account channel estimation errors given by

$$Y_{\ell,k} = H_{\ell,k}X_{\ell,k} + \eta_{\ell,k}$$

(6.4)

where $\eta$ represents the noise term with variance $\sigma_{\ell,k}^2 = N_0 + \text{MSE}_{\ell,k} \cdot E_s$ and zero mean. The MSE of an arbitrary two dimensional (2D) pilot aided channel estimation scheme which relies on FIR filtering is determined by

$$\text{MSE}_{\ell,k} = \mathbb{E}[|\hat{e}_{\ell,k}|^2] = \mathbb{E}[|H_{\ell,k} - \hat{H}_{\ell,k}|^2]$$

(6.5)

The MSE is dependent on subcarrier and OFDM symbol index $\ell$ and $k$. In order to keep the modeling of channel estimation errors as simple as possible, we use the MSE averaged over all subcarriers and OFDM symbols of one frame

$$\text{MSE} = \frac{1}{N_{frame}N_c} \sum_{\ell=1}^{N_{frame}} \sum_{k=1}^{N_c} \text{MSE}_{\ell,k}$$

(6.6)

In order to assess the degradation due to channel estimation for different pilot grid and estimation filter parameters, the three parameter sets (a), (b) and (d) are selected from Table 6.6.

The effective noise term in (6.4) can be transformed to determine the loss in SNR due to channel estimation errors, to obtain

$$\Delta = \frac{E_s}{N_0} = 1 + \text{MSE} \cdot \frac{E_s}{N_0}$$

(6.7)
Figure 6.2.9 shows the loss in SNR according to (6.7) for the chosen pilot grid and estimation filters. Clearly, the denser the pilot grid the smaller the degradation due to channel estimation. On the other hand, the spectral efficiency of the system decreases. One way to reduce the SNR loss without sacrificing bandwidth is to use a pilot boost. Curves showing the SNR loss of a 3dB pilot boost are drawn with dashed lines in Figure 6.2.9. It is seen that a pilot boost is only effective as long as the MSE floor of the estimator is not reached. So, parameter set (a.) with a pilot overhead of less than 3% (see Table 6.6) achieves acceptable performance if the SNR is below 20dB. On the other hand, parameter set (b.) appears reasonable for SNRs up to 35dB. However, compared to set (c.), parameter set (b.) requires a more complex channel estimation filter with $M_f=16$ coefficients to reduce the MSE floor (which can be extracted from Figure 6.2.6).
checked that the estimator does not experience an MSE floor in the SNR region of interest, in which case a pilot boost becomes ineffective.

<table>
<thead>
<tr>
<th>$D_t$</th>
<th>3</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0 dB</td>
<td>0.9 dB</td>
<td>2.7 dB</td>
</tr>
<tr>
<td>8</td>
<td>2.2 dB</td>
<td>3 dB</td>
<td>4.8 dB</td>
</tr>
</tbody>
</table>

Table 6.7: Relative pilot boost for a certain pilot grid provided that the overall transmit power is the same for all pilot grids

![Figure 6.2.11. Performance of an OFDM system with 2x1D pilot symbol aided channel estimation (PACE). Dashed lines show simulations by performing channel estimation, while solid lines denote curves applying the proposed approximation of channel estimation errors.](image)

The accuracy to the channel estimation error model is validated in Figure 6.2.11. The system parameters of the WINNER wide area mode and the urban macro (typical urban) channel model were assumed. Convolutional channel coding with rate 1/2 (133,171) and memory 6, with soft Max-Log MAP decoding was used. The performance of an OFDM system with 2x1D PACE is plotted and compared to the proposed estimation error approximation. The BER curve for the receiver with perfect CSI is shifted according to (6.7) is plotted together with a receiver having implemented a channel estimation unit. Dependent on the pilot grid there is a mismatch of about 0.5dB. So, the proposed estimation error approximation models the performance reasonably well.
6.2.5 OFDM synchronization strategies for the downlink

6.2.5.1 Synchronization utilizing the guard interval

In [BSB97] a blind algorithm for symbol timing and frequency offset estimation, which utilizes the guard interval was proposed, which has the great advantage that it does not require any overhead in form of training data. The basic idea is that for OFDM systems employing a cyclic prefix within the guard interval the periodic structure is readily available in form of the cyclic prefix. In other words, the cyclic prefix and the part of the data symbols from which is copied are identical. Tracking of the symbol time and frequency offset are achieved by computing the correlation metric

\[
\rho(\Delta n) = \sum_{n=\Delta n}^{\Delta n+\frac{N_{\text{FFT}}}{2}-1} y_n y^*_{n+N_{\text{FFT}}}
\]  

where the received signal \( y_n \) is sampled at the time instants \( t=nT_{\text{spl}} \). The symbol timing offset, normalized to the sampling duration \( T_{\text{spl}} \), is denoted by \( \Delta n = \Delta T / T_{\text{spl}} \), and the frequency offset is denoted by \( \Delta f \).

The magnitude of \( \rho(\Delta n) \) peaks at the beginning of an OFDM symbol, while its phase at this time instant is proportional to \( \Delta f \). The synchronization range of the algorithm is \( \Delta f T < \pm \frac{1}{2} \) of the subcarrier spacing \( 1/T \). The timing estimator can only detect the start of an OFDM symbol, but not the start of the frame. Hence the algorithm has ambiguities in the range of one OFDM symbol duration.

Since the time and frequency offset will be constant during some OFDM symbols, averaging over several estimates considerably improves the performance [BBBL99]. For the performance evaluations carried out in this deliverable, a simple 1st order lowpass filter is used. Hence, the smoothed time offset is given by

\[
\Delta \hat{n}_k = \alpha \cdot \Delta \hat{n}_{k-1} + (1-\alpha) \Delta \hat{n}_k
\]

where \( \alpha < 1 \) is a positive constant, and \( \Delta \hat{n}_k \) is the estimated symbol timing offset of OFDM symbol \( k \), which is essentially the maximum of \( \rho(\Delta n) \) [BSB97]. For the simulation results presented in this deliverable we set \( \alpha = 0.9 \). At the first run \( (k=1) \), \( \Delta \hat{n}_1 \) is initialized by \( \Delta \hat{n}_1 = \Delta \hat{n}_1 \).

Unfortunately, the performance of this approach degrades in a frequency selective channel, since the time dispersive channel impulse response reduces the ISI free part of the guard interval. To assess the accuracy of the cyclic prefix based synchronization scheme, computer simulations have been carried out for the WINNER wide area FDD mode, with system parameters taken from Table 3.1.

![Typical Urban channel (\( \alpha=0.9 \))](image)

Figure 6.2.12. Variance and mean of the symbol timing offset normalized to the sampling duration \( \Delta n = \Delta T / T_{\text{opt}} \).

Figure 6.2.12 shows the variance and mean of the symbol timing offset normalized to the sampling duration \( \Delta n = \Delta T / T_{\text{opt}} \). It is seen that the variance \( \text{var}(\Delta n) \) decreases with the number, \( N_{\text{avg}} \), of averaged timing estimates \( \Delta \hat{n}_k \). For low SNR, the variance is still in the range of one hundred. This however corresponds to a standard deviation of about ten samples, which appears to be reasonably accurate.
Furthermore, the symbol timing estimate becomes biased, i.e. $\Delta \hat{n}$ is no longer zero mean. In fact, this bias is beneficial, since it increases the robustness towards ISI. This is because a negative symbol timing estimate $\Delta \hat{n}$ (the estimated start of the OFDM symbol is earlier than the actual start) will give rise to ISI, while there is some tolerance for positive $\Delta \hat{n}$, in case the excess delay of the channel is shorter than the cyclic prefix.

![Typical Urban channel ($\alpha=0.9$)](image)

**Figure 6.2.13.** Variance of the fractional part of the carrier frequency offset normalized to the subcarrier spacing $\Delta f T$.

The variance of the fractional part of the normalized carrier frequency offset $\Delta f T$, where $1/T$ denotes the subcarrier spacing, is shown in Figure 6.2.13. Also the variance of $\Delta f T$ decreases with the number of averaged estimates $N_{\text{avg}}$. Even for low SNR the variance is around $10^{-3}$, which corresponds to a standard deviation lower than 5%. For acquisition this appears reasonably accurate.

### 6.2.5.2 Tracking of the carrier frequency offset

Fine tuning of the carrier frequency offset is done in the frequency domain after acquisition. Furthermore, assuming that the integer part of the frequency offset has been removed by detecting the position of the continuous pilot tones, a residual frequency offset, which is a fraction of the subcarrier spacing, remains to be detected. The frequency offset is determined by measuring the phase shift between the subcarriers of two adjacent OFDM symbols $\arg(Y_{ik}^* Y_{i,k-1}^*)$, where $Y_{ik}$ denotes the received pilot subcarrier $i$, of OFDM symbol $k$. In a multipath fading channel the variations of $\arg(Y_{ik}^* Y_{i,k-1}^*)$ are due to the Doppler spread, while the frequency offset corresponds to its mean. In order to estimate the mean, $\arg(Y_{ik}^* Y_{i,k-1}^*)$ is averaged over several pilot subcarriers, which should be placed such that the fading on each pilot tone becomes mutually uncorrelated. On the pilot tones, the following metric is computed [SFFM99]

$$ R(\Delta f T) = \sum_{i,f} Y_{ik} Y_{i,k-1}^* $$

(6.10)

Then, the estimate of the carrier frequency offset is given by [SFFM99]

$$ \Delta \hat{f} T \approx \frac{1}{2\pi (1 + N_{\text{avg}} / N_{\text{FFT}})} \arg(R(\Delta f T)) $$

(6.11)
Figure 6.2.14. Tracking of the normalized carrier frequency offset $\Delta f / T$ as a function of OFDM symbols.

For simulations 16 continuous pilot tones are transmitted, spaced 50 subcarrier apart from each other, which corresponds to approximately 2% overhead in the WINNER wide area mode.

Figure 6.2.14 shows the tracking behaviour of the normalized carrier frequency offset $\Delta f / T$ over time, at an SNR of 5dB. At time instant $k=1$, the tracking loop was initiated with a certain frequency offset. The tracking was performed by a 1st order loop filter with filter constant $\alpha = 0.9$. One can notice that, after approximately 25 OFDM symbols, the loop filter locked to the carrier frequency, even if the frequency offset is as large as $\Delta f / T = 0.3$. The residual estimation error is below 1%, which is below the frequency variations due to the Doppler spread. A lower value for $\alpha$ leads to a quicker steady state, however, the residual estimation error increases.

6.2.5.3 Conclusions

In this section the feasibility of two synchronization strategies for the downlink of the WINNER air interface were studied. For acquisition, cyclic prefix based synchronization has proven acceptable performance. Acquisition using an OFDM training symbol is both very fast and robust. However, for the WINNER system parameters, a training symbol causes significant pilot overhead.

The choice upon the most appropriate synchronization scheme depends also on the requirements and features of the WINNER system. In case an OFDM training symbol is required anyway, e.g. for channel estimation, to use this also for synchronization purposes would be a straightforward choice. If this is not the case, the large training overhead one full OFDM training symbol brings about is hardly justified. Then, a synchronization strategy which does not need an OFDM training symbol may be preferable.
6.3 Wide area feeder link

Figure 6.3.1: Frame error rate over SNR for the different modulation alphabets and code rates for the feeder link (short range system parameters) on an AWGN channel, using standard CP-OFDM

6.4 Short range

6.4.1 CP-OFDM

Simulation results for CD-OFDM in the short range scenario with QPSK, 16-QAM and 64-QAM modulation and rate ½ coding are reported in Figure 6.4.1. Further results for different coding and modulation modes are reported in Figure 6.4.2.

Figure 6.4.1: Frame error rate over SNR for the different short range scenarios (outdoor, indoor), for rate ½ coded QPSK, 16-QAM and 64-QAM, using standard CP-OFDM
Figure 6.4.2: Frame error rate over SNR for different modulation alphabets and code rates for the short range system on an IEEE 802.11n C NLOS channel, using standard CP-OFDM

6.4.2 Pilot aided channel estimation (PACE) for OFDM

In this section the performance of two dimensional (2D) pilot aided channel estimation (PACE) applied to OFDM is evaluated for the short range mode, as described in section 5.3.1.

6.4.2.1 Channel estimation parameters & Pilot grid selection

For the short range TDD mode, the performance of PACE is evaluated using the power delay profile proposed for the WLAN standard IEEE 802.11n model C NLoS. Both propagation delay and maximum velocities are significantly lower than for the wide area channel models. Hence, the overhead due to pilot symbols decreases accordingly. Applying (6.1), which is a consequence of the sampling theorem, the pilot spacings in frequency and time are lower bounded by \( D_f \leq 25 \) and \( D_t \leq 100 \) (at a velocity of 50km/h). According to the discussion for the wide area mode in section 6.2.3.1 we select \( D_f \) such that an oversampling factor of 20\% and 100\% can be maintained, so \( D_f = 20 \) and 12, respectively.

For the pilot spacing in time direction the sampling theorem becomes meaningless, bearing in mind that the frame length is significantly below 100. In the following we set \( D_t = 3 \) and 9, which results in \( M = 4 \) and 2 pilot symbols per frame of 10 OFDM symbols. Note, unlike for the results presented in this deliverable, in [WIND2.1] a continuous stream of FFT blocks was assumed. However, due to the requirements of the TDD duplex scheme, a mobile terminal cannot transmit and receive at the same time. Therefore, we will consider the problem of estimating the channel for one isolated OFDM frame.

The resulting pilot overheads are given in Table 6.8.

<table>
<thead>
<tr>
<th>( D_f )</th>
<th>( D_t )</th>
<th>( \frac{N_p}{N_d} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>3</td>
<td>3.4%</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>1.7%</td>
</tr>
</tbody>
</table>

Although, the pilot overhead for all grids is rather low, it should be noted that for MIMO system as well as for uplink reception \( N \) times as many pilots are required, where \( N \) is the number of links to be estimated. Hence, keeping the pilot overhead as low as possible remains important.
6.4.2.2 Results & Discussion

In Figure 6.4.3 the MSE is plotted against the average received SNR for the pilot grids given in Table 6.8 with the urban micro channel model. In the SNR range considered no MSE floor is observed.

![Figure 6.4.3. MSE vs SNR of the urban micro channel model (short range) for different pilot grids.](image1)

In Figure 6.4.4 the MSE is plotted against the interpolation filter order $M_f$. For the short range estimator similar conclusions can be drawn as for the wide area mode estimator. That is, a denser pilot spacing in frequency allows for a less complex estimator. However, provided that an estimator complexity of $M_f=16$ is feasible, the scarcer pilot grid is preferable due to the larger bandwidth efficiency.

![Figure 6.4.4. MSE vs filter dimension $M_f$ in frequency direction of the urban micro channel model (short range) for different pilot grids.](image2)
6.5 Evaluation of required power backoff for OFDMA and serial modulation

One of the main advantages of serial modulation over parallel modulation is its lower amplitude dynamic range and its consequent decreased high power amplifier (HPA) power backoff requirement [FK05], [FM05], [CND+00]. This difference in amplitude, or instantaneous power, dynamic range, is illustrated in Figure 6.5.1, which shows complementary cumulative distributions of QPSK serial modulated and OFDMA signals generated by (a) frequency domain method, with 5.5% windowing of the time domain waveform, and (b) time domain method, with 23% square root raised cosine filtering of the time domain waveform.

![Figure 6.5.1 Distribution of instantaneous power for comparable OFDMA and serial modulated waveforms: (a) frequency domain generation, with 5.5% raised cosine time domain windowing; (b) time domain generation with 23% square root raised cosine frequency domain filtering.](image)

Figure 6.5.2 shows the comparable CDF for 16-QAM signals generated by the frequency domain method.

![Figure 6.5.2 Instantaneous power distribution for 16 QAM frequency domain-generated serial modulated and OFDMA signals](image)

It is well known that the dynamic range of the instantaneous power of OFDM signals can be reduced by a variety of techniques mentioned in Section 4.10.2 and in [WIND2.1]. It is perhaps not so well appreciated
that many of these techniques can also be applied to serial modulation. For example SLM (selective level mapping) can be applied successfully in modified form to serial modulated signals, as well as to OFDM [FM05]. Even clipping and filtering (see [DG04] and references therein) can be applied to serial modulation, as to OFDM, with only moderate effects of nonlinear distortion on the received signal. An example of this is shown in Figure 6.5.3. Clipping and filtering with 1 and 4 iterations as in [DG04], respectively, are applied to QPSK frequency domain-generated serial modulated and OFDMA signals. The clipping level for the OFDMA signal is twice its standard deviation, while that for the serial modulated signal is 1.5 times its standard deviation. It is seen that the dynamic range is markedly reduced, and that the serial modulated signal maintains its lower dynamic range advantage over the OFDMA signal. The clipping operation introduces some nonlinear distortion in the received signal, which increases with the number of iterations of clipping and filtering. However with 4 iterations in this example, each waveform’s mean square distortion at the receiver output is more than 25 dB below the desired signal output level.

![Figure 6.5.3 Instantaneous power distribution for OFDMA and serial modulated waveforms with iterative clipping and filtering applied: (a) 1 iteration; (b) 4 iterations.](image)

The above results show that the instantaneous power dynamic range can be significantly reduced by repeated clipping and filtering (and by other means) for both OFDM and serial modulated signals. This reduction in the dynamic range will certainly reduce the power backoff requirements if the HPA has a nonlinear input/output characteristic which closely approximates that of an ideal clipper (i.e. linear up to an abrupt saturation level). Real HPA’s, especially moderate cost ones, may be expected to have nonlinear characteristics which are less ideal. A Rapp model [Rap91] (see Figure 6.5.4), with a parameter \( p = 2 \), has been identified as a fairly good approximation to the amplitude-to-amplitude conversion characteristic of a typical solid state power amplifier. With \( p = 10 \) or higher, the characteristic approaches that of an ideal clipper. Examples of spectral regrowth due to a \( p = 2 \) nonlinearity for frequency domain-generated OFDMA and serial modulated QPSK signals are shown in Figure 6.5.5(a) and Figure 6.5.5(b). In each case the power backoffs are adjusted so that the maximum sidelobe level is about -48 dB, consistent with the scaled ETSI spectral mask assumed in Section 3.2.2.4. Figure 6.5.5(a) shows the results with no clipping or filtering applied before the nonlinearity. Figure 6.5.5(b) shows the results when 4 iterations of clipping and filtering are applied, exactly as in Figure 6.5.3 (b). Two facts are apparent:

1. The clipping and filtering, which is very successful in reducing the signals’ dynamic range, has almost no effect on the output power spectrum. Presumably this is because most of the sidelobe regrowth is due to the part of the HPA nonlinear characteristic which affects small and moderate signal levels.

2. In this example, the serial modulation signal requires almost 3 dB less power backoff than does the OFDMA signal\(^7\).

\(^7\) Similar results, not shown here, are obtained for the time domain method of generating serial modulated and OFDM signals.
Figure 6.5.4 Rapp model of HPA nonlinearity

Figure 6.5.5 Output power spectra of signals passed through a Rapp model \( p = 2 \) nonlinearity; (a) no clipping or filtering prior to nonlinearity; (b) 4 iterations of clipping and filtering prior to nonlinearity. Backoffs of OFDMA and serial modulated signals are adjusted to produce approximately -48 dB relative spectral sidelobe levels. Scaled ETSI 3GPP mask is also shown.

Corresponding results are shown in Figure 6.5.6 for an HPA with \( p = 10 \) (approximating an ideal clipper). The change from \( p = 2 \) to \( p = 10 \) results in an almost 3 dB reduction in the required backoff. Furthermore, in contrast to the \( p = 2 \) HPA, the application of 4 iterations of clipping and filtering yields a further reduction in required backoff of 1.6 to 2.2 dB. Thus the iterative clipping and filtering procedure is most effective when combined with a HPA predistortion (linearization) procedure, as exemplified in Section 4.10.3 and Section 5.6. In any case, serial modulation still requires roughly 2.5 to 3 dB less backoff than OFDMA.
Figure 6.5.6 Output power spectra of signals passed through a Rapp model $p=10$ nonlinearity; (a) no clipping or filtering prior to nonlinearity; (b) 4 iterations of clipping and filtering prior to nonlinearity. Backoffs of OFDMA and serial modulated signals are adjusted to produce approximately -48 dB relative spectral sidelobe levels.

### 6.5.1 Implication for performance comparison of OFDM and serial modulation

One way to look at these results in terms of OFDM peak to average power reduction methods for spectrum sidelobe suppression, is to observe that a “FFT precoding” operation (which turns the OFDM signal into a serial modulated signal) is more effective than measures like repeated clipping and filtering to reduce the required power backoff.

We have seen in the preceding examples that serial modulation requires 2.5 to 3 dB less power backoff than does a comparable OFDM system under a variety of conditions. In light of the discussion in Section 4.10.1, this is equivalent to imposing a roughly 2.5 dB penalty on the received average $E_b/N_0$ of OFDM systems, if the systems are compared on the basis of using the same transmitting HPA, and constrained to have the same transmitted output power spectrum sidelobe level. The alternative view, comparing on the basis of equal average transmitted power, depends on the relative costs of power amplifiers with 2.5 dB to 3 dB differences in peak power output. In any case, for uplink transmission, the serial modulation approach appears to be a clear winner in terms of easing the user terminal’s HPA requirements.
6.6 Comparison of CP-OFDM and serial modulation in Wide-area

The comparison with QPSK and 16-QAM modulations and code rates $\frac{1}{2}$, $\frac{3}{4}$ and $\frac{5}{6}$ in the urban macro channel is presented in Figure 6.6.1 and Figure 6.6.2. With QPSK serial modulation offers comparable or better performance than OFDM even with simple linear equalization, while turbo equalization provides roughly 2dB more gain. In the 16-QAM case OFDM performs significantly better than the linear equalizer and roughly comparably to the turbo equalizer. Unless very high spectral efficiencies are assumed, serial modulation offers a better performance in the considered case.

![Figure 6.6.1 QPSK comparison between OFDM and serial modulation](image1)

![Figure 6.6.2 16-QAM comparison between OFDM and serial modulation](image2)
7. Summary and Conclusions

This document makes a first assessment of the performance of candidate link-level technologies that have been selected for investment in [WI]N D2.1. Since the relative merits of the considered techniques depend crucially on a large number of parameters, such as carrier frequency, cost of the devices, etc., final decisions on the selection of a technique for a specific purpose (e.g., modulation, channel estimation, forward error correction, etc.) can only be made once such constraints are known. The results in this deliverable are intended to give an overview of what performance can be achieved at link level, under certain assumptions, and what the main issues are, that limit this performance. In Chapter 3, the parameters used for evaluation of serial and parallel modulation systems throughout this report are derived. The findings of these evaluations are summarized in the following.

Coding (Sections 4.4, 5.1): Convolutional Codes should be favored over PCCC/LDPCC for short information block lengths, more specifically for block sizes up to a few hundred bits (when using a large memory CC). The gains of advanced coding techniques over a memory 8 CC are in the order of 2.5 dB for block lengths of several thousand bits (and increase for larger block lengths), for a frame error rate of 1% and a wide range of channel conditions. The losses for using short instead of long codewords (50 vs. 5000 information bits) is around 1dB on the AWGN channel, given a performance optimal choice is made for the code. PCCC and LDPCC show comparable performance and maximum decoding complexity. LDPCC offer the inherent capability to reduce power consumption since they check for successful decoding during iterations and can stop the decoding process as soon as possible. However, comparable concepts exist for PCCC and applying such techniques results in a variable decoding delay which might be undesirable from the implementation perspective. Low complexity-variants of the standard decoding algorithms (maxLogMAP and MinSum decoding) are highly attractive for the user terminal, since power consumption and required cycle count can be reduced by a factor of 2 (assuming non-linear functions are implemented via look-up-tables otherwise). The loss in performance is usually below 1 dB. Open issues include using stopping criteria for PCCC and optimizing LDPCC for a low variance in decoding delay. Serial modulation is favorable from the coding perspective, since it provides the decoder with a constant SINR per block, which yields the better error correction performance, compared to coding over a whole OFDM symbol where the SINR varies within the codeword.

Modulation (Sections 3.2.2, 4.1-4.3, 4.5, 5.2): Parallel modulation (OFDM and its variants) and serial modulation ("single carrier" and its variants), when transmitted in blocks separated by cyclic prefixes, are both members of a family of "generalized multicarrier" (GMC) transmission methods, in which transmission and reception signal processing operations are done in the frequency domain, with benefits in simplicity and performance in frequency selective channels. The choice of link layer parameters would be similar for both cases. Either time domain or frequency domain processing can be used to generate transmitted waveforms. Fig. 3-2 shows examples of power spectra generated by each of these approaches.

Single carrier transmission (Section 6.1): provides a constant diversity in adaptive modulation and coding modes up to 2/3 16-QAM with simple linear equalization. 8-PSK with code rate 2/3 can be used instead of 16-QAM, code rate 1/2, if the maximum required spectral efficiency required is 2 bit/s/channel use. 16-QAM with linear equalizer and high-rate codes does not work very well. There are no significant losses due to Doppler even in high mobility scenarios.

Pseudo-Random-Postfix OFDM (PRP-OFDM; Sections 5.2.1.1, 5.5.2) has been shown to be of advantage compared to standard Cyclic Prefix OFDM (CP-OFDM) and Zero-Padded OFDM (ZP-OFDM) if the target application requires i) a minimum pilot overhead, ii) low-complexity channel tracking (e.g. high mobility context) and iii) adjustable receiver complexity/performance trade-offs without requiring any feedback loop to the transmitter.

IOTA-OFDM (Section 5.2.1.2): The use of IOTA-OFDM presents various advantages: first, thanks to the removal of the cyclic prefix, IOTA-OFDM leads to a gain up to 25% in spectral efficiency compared to CP-OFDM. IOTA-OFDM also benefits from a steeper spectrum than CP-OFDM thanks to the good localization in frequency of the IOTA function. Therefore, in IOTA-OFDM a larger number of sub-carriers can be modulated within a same spectrum emission mask, leading to a further increase of the spectral efficiency.
MC-CDMA (Section 5.2.1.3): right combination of spreading and coding can increase the robustness of MC-CDMA to the MAI, and thus increase the load of the system. Moreover, very basic iterative MUD algorithm, multistagged can lead to spectral efficiency increase by approaching single-user bound. The introduction of soft information relying on new candidate coding schemes is sure to outperform such basic hard detected/regenerated iterative MUD schemes. Finally, it is shown that MC-CDMA is a promising air interface candidate for Layered Multi-cell environment.

Interleaved Frequency Division Multiple Access (IFDMA; Sections 5.2.1.3, 6.1.5): IFDMA combines the advantages of single and multi carrier systems such as high frequency diversity, constant envelope, and at the same time robustness to time offsets by appropriate choice of the guard interval. However, similar to OFDMA, IFDMA is expected to be sensitive to frequency offsets. It is shown that, compared to OFDMA, IFDMA provides lower complexity at the transmitter.

Channel Estimation (Sections 4.7,5.3)
Serial Modulation: It was pointed out that pilot tones can be inserted into frequency domain-generated serial modulation signals, just as they can for OFDM and other parallel modulated signal waveforms. A minimal PAPR is obtained if the pilot tones are generated by taking the FFT of a Chu training sequence; the resulting composite pilot signal is equivalent to an IFDMA (or FDOSS) signal. Adaptive linear equalization of serial modulated signals, using interpolation between training sequences at the beginning and end of a frame of 10 blocks was evaluated for the WINNER urban macro channel model; the degradation was about 2 dB from the case of ideal channel state information; in a continuous stream of frames, the overhead, including cyclic prefixes, would be about 10%.

Pilot aided channel estimation (PACE) for OFDM (Section 5.3.1): Under the assumed parameters for the WINNER system, a pilot overhead of about 3% to 9% is required for the wide area FDD mode, while for the short range TDD mode the overhead is only about 1% to 2%. For channel estimation in frequency direction about 20% oversampling factor appears to be sufficient. For interpolation in time direction placing a pilot at the beginning and end of the frame is sufficient, unless the velocity is very high (as in the rural channel model). The degradation in $E_b/N_0$ with respect to perfectly known CSI can be reduced by inserting a pilot boost, which in many cases appears favorable compared to a denser pilot grid. Another possibility to reduce the degradation is the use of a pre-smoothing filter which achieves a performance close to MMSE at significantly less computational cost.

Cyclic channel estimation for high Doppler (Section 5.3.2): In high Doppler scenario, the traditional preamble LS channel estimation cannot work. The cyclic channel estimator does not replace the subcarrier tones in OFDM symbols while a pulse train is added to the data. The estimator requires simple computation and modest channel tracking capability under high Doppler channel environment. There is no loss of bandwidth efficiency using this estimation approach. This channel estimator can be well suited for wide area scenario, especially in high Doppler case.

Synchronization (Section 4.8, 5.4):
Synchronization for Serial Modulation (Section 4.8): Serial modulation systems are more robust with respect to synchronization errors and phase noise since the receiver output in serial modulated systems is affected in a more easily-correctable way than for parallel modulated systems like OFDM, since the phase error at the output of the linear frequency domain equalizer varies only linear in time. Adaptive decision-directed frequency offset compensation was proposed and found to be effective for frequency offsets of 10% of the inter-subcarrier spacing.

Synchronization strategies for OFDM Downlink (Section 5.4.1): For acquisition, cyclic prefix based synchronization has proven acceptable performance. Acquisition using an OFDM training symbol is both very fast and robust. However, for the WINNER system parameters, a significant training overhead is required. In case an OFDM training symbol is required anyway, e.g. for channel estimation using it for synchronization would as well be a straightforward choice. If this is not the case, a synchronization strategy which does not need an OFDM training symbol may be preferable.
Synchronization in OFDM based single cell and cellular networks (Section 5.4.2): Time and frequency synchronization of MTs and BSs in an OFDM-based cellular single frequency network is proposed, by using sync signals as preamble or postamble of a MAC frame in the DL or UL, respectively. A sync signal consists of 3 phase-continuous OFDM symbols which are represented by a pair of subcarriers the frequency domain. At the receiver, 2 FFT windows are placed within the ISI free interval and according to the phase difference between 2 modulated OFDM symbols, time and frequency offset can be estimated. A control loop is used in order to achieve the convergence of time and frequency synchronization. Simulation of a full-coverage network has shown that using this technique, time and frequency synchronization of MTs and BSs is feasible in an OFDM-based cellular single frequency network.

Synchronization for MIMO-OFDM (Section 5.4.3): For the WB (Wide Band) case a frequency reuse of one is foreseen, so the synchronization scheme robust to co-channel interference is needed. The training sequence at the head of the packet designed with frequency multiplexing is appropriate to be used in this situation. Simulation results on this synchronization scheme for MIMO-OFDM system show that under 100 MHz channel, the training sequence with frequency multiplexing of the training symbols from different BTSs is suitable to operate in co-channel interference scenarios down to e.g. SNIR = -5 dB.

Predistortion (Sections 4.10,5.6,6.5): Realistic HPA nonlinearities, modelled by the Rapp model with parameter $p=2$, will cause transmitted spectrum sidelobe regrowth. The amount depends on the HPA backoff. PAPR-reduction methods, such as iterated clipping and filtering are effective for reducing PAPR for both serial and parallel modulation. However this has little effect on reducing spectral sidelobe splatter for Rapp model with $p=2$. Their effectiveness becomes more apparent for a Rapp model nonlinearity with $p=10$ or higher.

Use of a HPA, whose nonlinear characteristic is close to that of an ideal clipper (linear up to a saturation level, corresponding approximately to a Rapp model with $p=10$ or more), is effective for reducing required backoff, for a fixed allowable spectrum sidelobe level, for both serial and parallel modulation. Its effectiveness is further enhanced when combined with PAPR reduction methods for both serial and parallel modulation. An adaptive HPA predistortion algorithm was proposed and found to be effective for reducing spectral sidelobe regrowth for OFDM. Its effectiveness is further enhanced when combined with selective level mapping (SLM).

Serial modulation requires 2 to 3 dB less power backoff than parallel modulation methods (OFDM and its variants), even when peak to average power ratio (PAPR)-reduction methods are applied. This has two possible consequences:

- If both types of systems are required to use the same type of HPA, and to have the same maximum spectrum sidelobe levels, the parallel modulated system will be forced to transmit with an average power which is lower than that of the serial modulated system by the difference in their respective required power backoffs. This means that the parallel system’s received power and $E_b/N_0$ will be lower by 2 to 3 dB, and its coverage will be correspondingly less.

- If both systems are required to transmit the same average power (e.g. at the maximum power allowed by regulation), the parallel system must have a higher-rated, and therefore more expensive power amplifiers used. The HPA is generally one of the most significant cost components of user terminals, and the relationship of HPA cost to maximum power rating is an important technology issue.

Iterative Techniques (Sections 4.9,5.5)

Turbo-equalization (TEQ) for Single Carrier (Sections 4.9.1, 4.9.2): TEQ gives around 2dB gain over a linear equalizer in single carrier systems. Higher AMC modes than with linear equalization can be used in the same environments, e.g., 16-QAM works with up to code rates 3/4 and 5/6 with TEQ. Single carrier with TEQ outperforms OFDM with conventional equalization. In this deliverable, a class of powerful iterative technique, namely the turbo frequency domain equalization (FDE), is proposed as a promising receiver algorithm for the detection of single-carrier modulated signals. The performance of various types of turbo frequency domain equalizers has been evaluated for the suggested WINNER channel scenarios.
and for different combinations of channel codes and modulation schemes. With only three to four iterations, a performance gain of 1-3 dB over the linear FDE can be generally observed for all the scenarios. For high-order modulation (such as 16-QAM), turbo code can be employed to yield a significant gain of 3dB at high SNR, suggesting that adaptive modulation and coding is a very attractive approach to be used with the iterative technique. Furthermore, in a situation when pilot-assisted channel estimation is employed, it is shown that the turbo FDE can incorporate the iterative channel estimation process to yield a remarkable performance, which is shown to be very close to the performance of a linear FDE with ideal channel state information. Finally, it should be remarked that the turbo FDE has the advantage of smaller complexity compared with its time-domain equalization counterpart. This, coupled with the advantage of small PAPR associated with the single-carrier modulated signals, makes the turbo FDE a suitable receiver technology for the uplink transmission employing single-carrier modulated signals.

Iterative channel estimation in OFDM (Section 5.5.1.1): It was shown that the technique is of particular interest if the pilot subcarrier spacings in time and frequency directions approach the theoretical limits given by the sampling theorem. In this case, a significant gain is observed after solely one or two turbo iterations with respect to the performance obtained with a conventional pilot-aided channel estimation method. However, the gains vanish when the pilot symbol overhead increases or when the transmission operates in the low SNR regime. The detection complexity enhancement induced by the introduction of the iterative scheme remains otherwise reasonable. The ICE algorithm requires indeed only an extra soft mapping device and additional filtering operations for the all-symbols channel estimation step. The complexity of the aforementioned filtering stage is nevertheless limited by the use of constant filter coefficients as far as the propagation conditions do not evolve significantly. These observations should thus motivate the inclusion of ICE techniques in the receiver design of an OFDM system.

Iterative Interference Suppression for PRP-OFDM based Channel Estimation (Section 5.5.2): In section 5.5.1.2, a novel PRP-OFDM related channel estimation scheme has been introduced based on iterative interference suppression which makes PRP-OFDM applicable to higher order constellations such as 64-QAM and above. The inherent need for soft outputs of the maximum-likelihood decoder rather limits the application of the proposed technique to advanced coding schemes, such as Turbo Codes and/or LDPC codes where iterative decoding combined with soft output decoders is common.

The obtained results clearly indicate that a very high efficiency can be achieved at link level. Iterative techniques are a very promising path towards approaching the fundamental performance limits at the physical layer (i.e., minimum channel estimation overhead, error rate versus SNR performance reasonably close to the bounds defined by channel capacity). Imperfections in the analog front-end, such as non-linear HPA, phase noise in oscillators, and I/Q-imbalance will be a significantly issue as wireless systems go to higher carrier frequencies and implementations to lower supply voltages (cf. also results in [WIND2.7]). Overcoming these limitations in the RF frontend requires either very costly RF components, or sophisticated signal processing that either avoids or corrects the resulting deteriorations in performance. Finally, interference becomes a critical issue for systems with a frequency reuse close to, or equal to one. While intercell operation has not been in the focus of this report, the results presented in the synchronization chapter illustrate that intercell timing synchronization can be achieved in OFDM based systems, which is one contribution to avoiding serious UL/DL interference problems for single frequency networks.
A. Appendix

A.1 Assessment of Coding Techniques

A.1.1 Decoding Complexity

A.1.1.1 PCCC

The standard approach for decoding PCCC is letting two BCJR [BCJ+74] decoders exchange extrinsic information for a predetermined number of iterations. Let us consider a rate \( R_c = p / q \) constituent convolutional code of memory \( m \) (and hence \( D = 2^m \) states in the trellis, and \( B = 2^b \) branches entering/leaving each state). The decoder must first use the soft output from the detector together with the a priori information from the other MAP decoder to calculate appropriate branch metrics, here denoted as \( \gamma_k(s',s) \). This is followed by a forward-backward recursion through the trellis, where the forward and backward state probabilities \( \alpha_k(s) \) and \( \beta_k(s) \) are calculated, before soft outputs can be derived from these values. The different messages are calculated as follows (for further details please refer to [WIND2.1] and the references stated therein):

- **Branch metrics**: 
  \[
  \gamma_k(s',s) = \frac{L_v(k)x_v(k)}{2} + L_v^e(k)x_v(k) + L_v^s x_v(k) \quad (A-1)
  \]

- **Forward recursion**: 
  \[
  \alpha_k(s) = \max_{s'} (\alpha_{k-1}(s') + \gamma_k(s',s)) \quad (A-2)
  \]

- **Backward recursion**: 
  \[
  \beta_k(s) = \max_{s'} (\beta_{k-1}(s') + \gamma_k(s',s)) \quad (A-3)
  \]

- **Calculation of LLRs**: 
  \[
  L(k) = \max_{s'} (\alpha_{k-1}(s') + \gamma_k(s',s) + \beta_k(s)) - \max_{s} (\alpha_{k-1}(s) + \gamma_k(s',s) + \beta_k(s)) \quad (A-4)
  \]

where \( \max^* \) is defined as 
\[
\max^* (a,b) = \max(a,b) + \log(1 + \exp[-|a-b|]),
\]

Calculating \( \alpha_k(s) \), \( \beta_k(s) \), \( \gamma_k(s',s) \) and \( L(k) \) essentially permits to go a single step further down the trellis, i.e., performs all necessary operations required to calculate the decoder output for a single BCJR for \( p \) information bits. The number of operations required to do so is summarized in the table below:

<table>
<thead>
<tr>
<th>Simple Operations</th>
<th>( DB(p + 2q) )</th>
<th>( DB )</th>
<th>( DB )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward recursion</td>
<td></td>
<td>( D(B-1) )</td>
<td>( D(B-1) )</td>
</tr>
<tr>
<td>Backward recursion</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Calculation of LLRs and extrinsic values</td>
<td>( DB + 2p )</td>
<td></td>
<td>( (D-1)p )</td>
</tr>
</tbody>
</table>

The only difference between logMAP and maxLogMAP is obviously that the \( \max^* \) operation is replaced by a simple max operator. Summing up the above figures and taking this into account, we can formulate
the complexity for the two variants of the BCJR (similar complexity results have been reported in [RHV97]):

<table>
<thead>
<tr>
<th>Simple Operations</th>
<th>max* operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>logMAP</td>
<td>$DB(p + 2q + 3) + 2p$</td>
</tr>
<tr>
<td>maxLogMAP</td>
<td>$DB(p + 2q + 5) + D(p - 2) + p$</td>
</tr>
</tbody>
</table>

For the rate ½ constituent convolutional codes which are the main interest of our investigations, this translates into the following figures in terms of energy consumption and cycle count:

<table>
<thead>
<tr>
<th></th>
<th>Energy cost per info bit</th>
<th>Cycle count per info bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>logMAP</td>
<td>$196D - 58$</td>
<td>$34D - 4$</td>
</tr>
<tr>
<td>maxLogMAP</td>
<td>$19D + 1$</td>
<td>$19D + 1$</td>
</tr>
</tbody>
</table>

**Table A-1: Cost for decoding a single code bit, for logMAP and maxLogMAP decoding**

The energy required for running the logMAP algorithm will obviously be dramatically reduced, if the max* operation can be implemented via a look-up table. In this case, energy requirements can be expected to be similar to the cycle count (roughly double the complexity of maxLogMAP). Inserting $D$ for the considered codes and multiplying the figures resulting from the above formulae with double the number of iterations (since we have two BCJRMs) then gives the total number of operations required for decoding one transmitted information bit, if we assume the max* operation to be implemented via iterative approximation.

### A.1.1.1.1 Stopping criteria for iterative decoding of PCCC

Throughout our evaluations, we left the number of iterations for PCCC be fixed and did not use any stopping criteria to detect early convergence in the decoding process. It has to be mentioned, however, that several such approaches exist and that employing them will probably lead to very similar results in the complexity-performance tradeoff as for LDPC. In the following we will present and discuss some of these approaches. Some of these techniques work on extrinsic values, while other on soft bits.

A very good overview of stopping criteria is given in [MDP00]. The proposed algorithms can be divided into hard-decision rules, soft decision rules, and CRC rules. Some remarks regarding complexity and buffering requirements are included in the analysis. Hard decision rules allow to stop iterations when tentative decisions in two successive half iterations, or successive full iterations are identical. Soft decision rules compare a metric on bit reliabilities (soft bit decisions) with a threshold. If the calculated metric is smaller then threshold, iterations are stopped. These rules can be divided into minimum bit reliability, average bit reliability and bit-by-bit reliability checks. CRC rules depend on calculating the syndrome of an outer CRC code after each iteration, (effectively reducing the code rate). If no errors were detected, iterations are stopped. Another hard decision rule has been proposed in [GCG04]. This technique is MLSE-based and relies on comparison of MLSE decisions after each half iteration.

A stopping rule derived from observing the variance of the MAP output (second moment of LLRs) has been introduced in [Rob94]; this method requires calculating the variance of the LLRs at every decoding step and comparison of the calculated variance with the SNR of an equivalent AWGN channel, depending on the required BER level. It is stated that blocks having a high variance require more iterations for successful decoding. Hoefer [LaH01] proposes to instead use the mean reliability of LLRs, which requires less computational effort than the method in [Rob94]. In [HOP96], a so-called cross entropy criterium has been proposed. Iterations are stopped when the cross entropy between the LLRs of the component decoders is smaller than some threshold. In [SLF99] the Sign Change Ratio technique, which is related to the method from [HOP96], was proposed. This technique computes number of sign changes
of the extrinsic information between two iterations. A modification of this technique has been presented in [WWE00]. A quite different approach, the joint optimization of number of iterations and BER, is proposed in [SPM+01]. Another quite broad overview of low complexity stopping rules can be found in [GKW03]. Additionally, authors propose new criterion based on calculation sum of reliability instead of average reliability, which does not require division.

A.1.1.2 CC Decoding

The Viterbi algorithm is the standard method of decoding convolutional codes. This algorithm is actually somewhat similar to the BCJR algorithm and in fact the maxLogMAP and the Soft Output Viterbi will yield the same results [FBL+98]. The main differences to the maxLogMAP decoding are:

- No a priori information is included in the calculation of the branch metrics
- The algorithm uses only the forward recursion
- Output bits are calculated from tracing the “survivor path” with the highest metric and the end of the codeword back.

In view of the above description for BCJR decoding and these statements, the number of operations required for Viterbi decoding of a single information bit is easily derived. Note that we added one operation in the forward recursion, to account for saving the origin of the survivor path at each state, which will facilitate backtracking and thus the generation of the decoding result.

<table>
<thead>
<tr>
<th>Simple Operations</th>
<th>Calculation of branch metrics</th>
<th>Forward recursion</th>
<th>Calculation of output bits</th>
<th>Sum</th>
<th>For rate 1/2 convolutional codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculation of branch metrics</td>
<td>$DB(2q - 1)$</td>
<td>$2DB$</td>
<td>$2$</td>
<td>$DB(2q + 1) + 2$</td>
<td>$10D + 2$</td>
</tr>
</tbody>
</table>

A.1.1.3 LDPC Decoding

LDPC codes are typically decoded with the help of the message passing algorithm (MPA), which has been described in detail in [WIN2.1]. This decoding algorithm computes the distribution of reliabilities of the Tanner graph by iteratively exchanging messages between the variable and check nodes. Depending on the context this algorithm as also referred to as sum-product (SPA) or belief propagation (BPA) algorithm. This algorithm checks in each iteration whether the codeword is valid and aborts the decoding process if a maximum number of iterations has been reached. This leads to a variable decoding delay. Although the average number of iterations is typically quite small, real time implementations might have to be designed for the maximum decoding delay. Let us consider a LDPC of codeword length $n$ and information length $k$. The number of variable nodes in the LDPC decoder is hence $n$ and the number of check nodes is $m = n - k$. The complexity of any message passing is obviously not only influenced by the number of nodes, but also the number of messages per node, i.e., the number of edges in the graph. Let us therefore also define $d_v$ and $d_c$ as the average variable and check node degrees, respectively.

The log domain messages of the BPA are given as follows. The message from the $i$-th variable node to the $j$-th check nodes is calculated as

$$Q_{i,j} = F_i + \sum_{i,j} R_{i,j} \quad \text{(A-6)}$$
The message from \( j \)-th check nodes to the \( i \)-th variable node is calculated as

\[
R_{j,i} = \prod_{l \neq j} \text{sgn}(Q_{l,j}) \Phi\left( \sum_{l \neq j} \Phi(Q_{l,j}) \right)
\]

(A-7)

or, equivalently

\[
R_{j,i} = 2 \text{ar tanh}\left( \prod_{l \neq j} \text{tanh}(Q_{l,j}/2) \right)
\]

(A-8)

The non-linear functions \( \Phi, \tanh, \text{ar tanh} \) are the source of the main complexity in LDPC decoding, since for practical decoder architectures, they would have to be implemented either as iterative loops or look-up tables. There exist several approaches to reducing this complexity by linear approximation of these non-linear functions. Appropriate selection of the interpolation parameters enables close-to-original performance at significant reduction in the node complexity [CDE+02][RSB+05]. Another approach exploits two interesting properties of \( \Phi(x) \), namely the fact that \( \Phi(\Phi(x)) = x \), and the fact that \( \Phi(x) \) is decaying very fast. The sum in (A.1.1.2-2) is hence clearly dominated by the maximum value over \( \Phi(Q_{l,j}) \), which corresponds to the minimum value of \( |Q_{l,j}| \). This motivates for the introduction of the so-called MinSum decoding algorithm with considerably reduced complexity where check note messages are defined as follows:

\[
R_{j,i} = \prod_{l \neq j} \text{sgn}(Q_{l,j}) \cdot \min_{l \neq j} |Q_{l,j}|
\]

(A-9)

Efficient implementations of the BPA will first calculate some “aggregate” messages before then calculating the individual messages sent out over the edges of the currently considered node, namely \( \sum R_{j,i} \) at the variable nodes and \( \sum \Phi(Q_{l,j}) \) at the check nodes. When using (A-3) for the check node calculations, a similar approach can be followed, relying on forward-backward calculation of parts of the total product. The complexity of calculating the different messages and tentative decoding for one single decoder iteration is summarized in the table below:

<table>
<thead>
<tr>
<th>Simple Operations</th>
<th>Multiplications</th>
<th>Non-linear functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable node messages</td>
<td>2( d_c n )</td>
<td>-</td>
</tr>
<tr>
<td>Check node messages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BP Variant I</td>
<td>(5( d_c - 2 ))( m )</td>
<td>-</td>
</tr>
<tr>
<td>BP Variant II</td>
<td>2( d_c m )</td>
<td>(2( d_c - 1 ) + ( d_c ))( m )</td>
</tr>
<tr>
<td>BP, Linear Approximation(^8)</td>
<td>(37( d_c - 45 ))( m )</td>
<td>-</td>
</tr>
<tr>
<td>MinSum</td>
<td>(4( d_c - 1 ))( m )</td>
<td>-</td>
</tr>
<tr>
<td>Tentative decoding</td>
<td>(2( d_c + 2 ))( m )</td>
<td>-</td>
</tr>
</tbody>
</table>

Normalizing this figure to the number of transmitted code bits \( n \), we get the following figures (note that \( d_c m = d_c n \)):

\(^8\) Assuming the linear approximation function requires 3 simple operations: 1 COMP, 1 ADD, 1 SHIFT; please refer to [RSB+05] and the references for details.
If we now consider the properties of actual implementation architectures described above and knowing that the average node degrees are $d_v = 3.55, d_c = 7.1$ for the rate 1/2 code developed within the framework of this report, we can now state the complexity per iteration and transmitted information bit as:

<table>
<thead>
<tr>
<th></th>
<th>Simple Operations</th>
<th>Multiplications</th>
<th>Non-linear functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP Variant I</td>
<td>$9d_v$</td>
<td>-</td>
<td>$2d_v$</td>
</tr>
<tr>
<td>BP Variant II</td>
<td>$6d_v + 2d_v / d_c, 3d_v - 2d_v / d_c, 2d_v$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>BP, Linear Approximation</td>
<td>$37d_v - 45d_v / d_c$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>MinSum</td>
<td>$8d_v + d_v / d_c$</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Energy cost per information bit</th>
<th>Cycle count per information bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP Variant I</td>
<td>916</td>
<td>150</td>
</tr>
<tr>
<td>BP Variant II</td>
<td>1090</td>
<td>150</td>
</tr>
<tr>
<td>BP, Linear Approximation</td>
<td>218</td>
<td>218</td>
</tr>
<tr>
<td>MinSum</td>
<td>58</td>
<td>58</td>
</tr>
</tbody>
</table>

To determine the worst case decoding complexity, the maximum number of iterations is needed. The frame error rate performance of a LDPC code under BPA decoding for several maximum iteration numbers is plotted in Figure A.1.1 exemplary for the AWGN channel. Simulations for several other channel types showed very similar results. The number of iteration can be chosen in the range of 50 to 80, without loosing much performance. The maximum number of iteration was set to 100 for the simulations in this document, which is in fact a quite conservative number. The average number of iterations is typically quite low, especially at high SNR. Examples for measured values are presented in Table A-1.

The BPA with linear approximated correction term considered in the table above showed no measurable loss in performance compared to the exact BPA. However, the further complexity reduction of the MinSum approximation has to be paid with a considerable performance loss, which depends on the channel and the used code. The FER performance for several channels and rate $R=0.5$ LDPC codes with different lengths is plotted in Figure A.1.2 to Figure A.1.5. The loss of $E_b/N_0$ is between 0.5 to 1.1 dB at a target frame error rate of $FER = 0.01$. 

If we now consider the properties of actual implementation architectures described above and knowing that the average node degrees are $d_v = 3.55, d_c = 7.1$ for the rate 1/2 code developed within the framework of this report, we can now state the complexity per iteration and transmitted information bit as:
Figure A.1.1: Frame error rate performance of a LDPC Code with information length $k=500$ and rate $R=0.5$ using BPA for several maximum number of iterations.
Table A-1: Performance loss and average number of iterations for BPA and MinSum decoder

<table>
<thead>
<tr>
<th>Channel</th>
<th>Information length</th>
<th>Loss $E_b/N_0$ at $FER = 0.01$ [dB]</th>
<th>average # iterations at $FER = 0.01$ [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Exact</td>
<td>MinSum</td>
</tr>
<tr>
<td>AWGN</td>
<td>500</td>
<td>0.5</td>
<td>9.4</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.6</td>
<td>14.5</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>0.7</td>
<td>27.9</td>
</tr>
<tr>
<td>ergodic Rayleigh</td>
<td>500</td>
<td>0.5</td>
<td>9.7</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.8</td>
<td>13.7</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>0.8</td>
<td>25.4</td>
</tr>
<tr>
<td>Rayleigh with 20 fades per codeword</td>
<td>500</td>
<td>0.7</td>
<td>5.7</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.8</td>
<td>6.1</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>0.9</td>
<td>8.1</td>
</tr>
<tr>
<td>Rayleigh with 5 fades per codeword</td>
<td>500</td>
<td>0.7</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>0.8</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>1.1</td>
<td>5.4</td>
</tr>
</tbody>
</table>

Figure A.1.2: Frame error rate performance of exact BPA and MinSum decoder for rate $R=0.5$ LDPC codes with different information length $k$ and AWGN channel.
Figure A.1.3: Frame error rate performance of exact BPA and MinSum decoder for rate $R=0.5$ LDPC codes with different information length $k$ and ergodic Rayleigh channel.

Figure A.1.4: Frame error rate performance of exact BPA and MinSum decoder for rate $R=0.5$ LDPC codes with different information length $k$ and Rayleigh channel with 20 independent fades per codeword.
Figure A.1.5: Frame error rate performance of exact BPA and MinSum decoder for rate $R=0.5$ LDPC codes with different information length $k$ and Rayleigh channel with 5 independent fades per codeword.

7.1.1.1 Duo-binary PCCC performance

Figure A.1.6: Frame error rate performance of Duo-Binary PCCC and binary PCCC (3GPP) over AWGN channel, coding rates 1/2 and 3/4.
Calibration Results

A.2 Simulation Setup

The following parameters were defined for calibration of link level simulation chains of all partners:

System: parameters as defined in section 3.2.2.
Modulation: QPSK, Gray mapping, MAP detection
Coding: (133,171) memory 6 CC, terminated, Viterbi decoding (equiv. MaxLogMAP)
Frame length: 1 symbol; optional for Wide area case: 10 OFDM symbols, random interleaving, 70 km/h (only time diversity, no ICI modeling)

Channel models:
- Wide area case: WINNER WP5 Urban Macro MIMO
- Short range case: IEEE 802.11n C NLOS
- Feeder link case: AWGN

Advanced calibration cases:
- 16-QAM, with maxMAP detection, rest as specified
- 16-QAM, rate $\frac{1}{2}$ (7,5) PCCC, puncturing \[s,p1,0,s,0,p2\]
- MaxLogMAP decoding, 4 internal decoder iterations

The simulation chains of involved partners have been successfully calibrated before running evaluations for this deliverable, as the results in the following subsections show.
A.3  Wide Area Case (Urban Macro)

Wide Area (WP5 Urban Macro, 1 OFDM symbol), QPSK, Rate 1/2 Memory 6 CC

Wide Area (WP5 Urban Macro, 10 OFDM symbols), QPSK, Rate 1/2 Memory 6 CC
Symbol rate: 16.25 Ms/s  
Sampling rate: 195 Ms/s  
Rolloff factor: 0.23  
Prefix length: 80 symbols  
RC Filter taps: 96  
Modulation: QPSK  
Channel: WP5 urban macro rounded to 5.128 ns, with v=0 Km/h  
Equalization: symbol rate frequency domain linear MMSE
A.4 Short Range Case (802.11n C NLOS)
A.5 Feeder Link Case (AWGN)
B. Channel Definitions

For the wide area feeder link, AWGN channels have been used for evaluation, as proposed in [WIND7.1]. For the short range scenario, IEEE 802.11n models have been used – more specifically, IEEE 802.11n Model C NLOS to model indoor, small office environments and IEEE 802.11n Model E NLOS to model outdoor environments. For evaluations in the wide area mode, slightly modified WINNER WP5 channel models have been used. Channel power delay profiles have been downsampled to 200MHz, in order to enable partners to run simulations with an oversampling factor of only 10 for the wide area case (WINNER WP5 models had been given at a precision of 0.1ns, corresponding to 10GHz channel bandwidth). The resulting PDPs are given in the tables below:

Table B-1: Channel power delay profiles adapted from WINNER WP5 Channels, used for evaluations in this deliverable

<table>
<thead>
<tr>
<th>Suburban Macro</th>
<th>Urban Macro</th>
<th>Urban Micro</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3.0000</td>
<td>0.000</td>
<td>-3.0000</td>
</tr>
<tr>
<td>-5.2200</td>
<td>0.010</td>
<td>-5.2200</td>
</tr>
<tr>
<td>-6.9800</td>
<td>0.025</td>
<td>-6.9800</td>
</tr>
<tr>
<td>-5.6682</td>
<td>0.140</td>
<td>-5.2204</td>
</tr>
<tr>
<td>-7.8882</td>
<td>0.150</td>
<td>-7.4404</td>
</tr>
<tr>
<td>-9.6482</td>
<td>0.165</td>
<td>-9.2004</td>
</tr>
<tr>
<td>-9.2147</td>
<td>0.060</td>
<td>-4.7184</td>
</tr>
<tr>
<td>-11.4347</td>
<td>0.070</td>
<td>-6.9384</td>
</tr>
<tr>
<td>-13.1947</td>
<td>0.090</td>
<td>-8.6984</td>
</tr>
<tr>
<td>-13.4132</td>
<td>0.400</td>
<td>-8.1896</td>
</tr>
<tr>
<td>-15.6332</td>
<td>0.410</td>
<td>-10.4096</td>
</tr>
<tr>
<td>-17.3932</td>
<td>0.430</td>
<td>-12.1696</td>
</tr>
<tr>
<td>-19.4735</td>
<td>1.380</td>
<td>-12.0516</td>
</tr>
<tr>
<td>-21.6935</td>
<td>1.390</td>
<td>-14.2716</td>
</tr>
<tr>
<td>-23.4535</td>
<td>1.410</td>
<td>-16.0316</td>
</tr>
<tr>
<td>-25.1898</td>
<td>2.830</td>
<td>-15.5013</td>
</tr>
<tr>
<td>-27.4098</td>
<td>2.835</td>
<td>-17.7213</td>
</tr>
<tr>
<td>-29.1698</td>
<td>2.855</td>
<td>-19.4813</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
C. References


ETSI TS 125 104 V4.2.0 (2001-09) “Universal Mobile Telecommunications System (UMTS); UTRA (BS) FDD; Radio Transmission and Reception”, (3GPP TS 25.104 version 4.2.0 release 4.


[Urb05] Prof. Rüdiger Urbanke’s website: http://lthcwww.epfl.ch


