CONFIDENTIAL

FP7-ICT Future Networks
SPECIFIC TARGETTED RESEARCH PROJECT
Project Deliverable

PHYDYAS Doc. Number	PHYDYAS_009-a
Project Number	ICT - 211887
Project Acronym+Title	PHYDYAS – PHYsical layer for DYnamic AccesS and cognitive radio
Deliverable Nature	Report
Deliverable Number	D9.1-a
Contractual Delivery Date	January 15th, 2009
Actual Delivery Date	January 26, 2009
Title of Deliverable	WiMAX FBMC-OFDM comparison scenarios
Contributing Workpackage	WP9
Project starting date; Duration	01/01/2008; 30 months
Dissemination Level	CO
Author(s)	Frank Schaich (ALUD)

Abstract:
To assess the performance filter bank based multicarrier (FBMC) systems can provide, a simulator is developed within work package 9. In order to gain acceptance within the community the simulator is based on an existing wireless transmission standard (WiMAX, Worldwide interoperability for Microwave Access) and enhanced with algorithms developed by other work packages to include the filter bank based multicarrier principle. This deliverable describes the principles of the simulator and its usage. Naturally, as the project still continues, the simulator will be further developed. Later deliverables will include those developments. Finally some exemplarily simulation results are provided.

Contents

Notations

A, B input streams of the CTC encoder
Y_1, Y_2
W_1, W_2 output streams of the CTC encoder

M overall number of subcarriers, FFT size
K overlapping factor in prototype filter design

m time index at SFB output/AFB input
N_{ovs} oversampling factor

i user index in multiuser cases
U number of users

$x_{k,n}$ observed ideal (without channel) complex sample of subcarrier k in symbol n
$y_{k,n}$ observed channel-distorted complex sample of subcarrier k in symbol n
$s(m)$ transmitted sequence at SFB output in the downlink
$s_i(m)$ transmitted sequence at SFB output of user i in the uplink

P number of paths in the multipath channel model
$P_{i,p}$ path power of path # p in the multipath channel model of user i
P_i transmit power of user i
S_i path-norm factor

$c_{i,p}$ complex gain of the pth path of the channel of user i
$\tau_{i,p}$ delay of the pth path of the channel of user i
m_i path independent time offset due to a failed synchronization
$m_{i,p}$ path dependent time offset due to different path lengths
$\Delta\rho_i(m)$ user specific phase and frequency offset
σ_n^2 channel noise variance

$r_i(m)$ received complex sequence at AFB input at the mobile of user i (downlink)
$r(m)$ received complex sequence at AFB input at the basestation (uplink)

$H_{k,n}$ channel response for subcarrier k and symbol n (assuming flat-fading and time variant case)
$\tilde{H}_{k,n}$ estimate of the channel response for subcarrier k and symbol n
$n_{k,n}$ noise sample disturbing subcarrier k in symbol n
$\tilde{n}_{k,n}$ noise sample disturbing the estimate of the channel response for subcarrier k in symbol n

1 Introduction

The objective of PHYDYAS is to propose a physical layer for future radio systems that is more efficient than present OFDM based solutions and better suited to the new concepts of DASM (Dynamic Access Spectrum Management) and cognitive radio.

One of the basic ideas is to concentrate the signal power a single subcarrier transmits stronger within its frequency range within the multi carrier spectrum. With OFDM parts of the signal power leaches into neighbored subcarriers (a single subcarrier follows a sinc function), leading to a higher sensitivity to frequency offsets and to the need of capacity reducing band guards. In reverse the new physical layer proposed in PHYDYAS thus ought to be less sensitive to carrier offsets and smaller guard bands should be affordable, leading to a more efficient use of the spectrum. Additionally spectrum sensing in cognitive radio is more reliable due to lesser interference.
To allow for synchronization and to battle the frequency selective behavior of multi path channels, OFDM based systems introduce a so called cyclic prefix to the signal, again reducing the capacity. With the new physical layer investigated in PHYDYAS there is no necessity for the cyclic prefix anymore.

To reach these goals a filter bank is introduced into the transmission chain. This filter bank is the heart of the project. A basic solution was provided at the beginning [1], which was and will be further developed within the project. Many aspects of the transmission chain, such as synchronization and initialization, channel estimation and tracking, equalization and demodulation, MIMO (multiple input multiple output) processing, need to be addressed. This is done within other work packages and thus this deliverable will not go into detail with respect to these algorithms.

To gain acceptance within the community the physical layer proposed in PHYDYAS needs to be compared to the state of the art with respect to complexity and performance. To get performance measures a link level simulator gets developed reproducing a complete wireless transmission chain including the algorithms produced by other work packages needed to realize the new physical layer. IEEE 802.16e [2] is a worldwide accepted standard defining a general air interface for broadband wireless access systems. To accelerate the introduction of such systems and to ensure compatibility and interoperability the so called WiMAX forum (an industry-led, not-for-profit organization) defined a subset out of this [3]. A system fulfilling this guideline gets certified and may use the WiMAX brand (‘Worldwide interoperability for Microwave Access’). The basic version of the simulator developed in PHYDYAS will be WiMAX based. The description of this simulator and its usage is the main scope of this deliverable.

The rest of this deliverable is structured as follows: Next is a description of the simulator and its components. Afterwards the usage of the simulator is described. Finally some exemplarily simulation results are presented. Later deliverables will present further simulation results obtained with the simulator to compare the new physical layer to the performance WiMAX can provide. At this point it shall be emphasized that the principles investigated in PHYDYAS are not limited to WiMAX. With minor modifications they easily can be translated to other OFDM based transmission systems such as LTE (Long Term Evolution).

2 Transmission system

WiMAX is an OFDM based point to multipoint (in the downlink) and multipoint to point (in the uplink) respectively wireless cellular transmission system. Diverse possibilities of abstraction are possible. As PHYDYAS is concerned with the processing of the signals at the physical layer a single cell link level chain without abstraction of the elements of the digital processing stage is the way to go. This means schemes like e.g. forward error correction (FEC), channel estimation and equalization and data protection by hybrid automatic repeat request (HARQ) are directly to be implemented. The performance measures produced are the coded/uncoded bit error rate (BER) and coded packet error rate (PER) depending on the signal-to-noise ratio (SNR). As the FBMC principle is applicable both in uplink and downlink, two single transmission chains are developed in PHYDYAS accordingly.
As already mentioned the core of the simulator is based on WiMAX. The extension to FBMC mode will be twofold, one solution trimmed for maximal compatibility to WiMAX and one solution trimmed for maximal performance. Up to now just the former one is realized, the latter is up to further research.

This chapter shall provide a general view onto the architecture of the simulation chains and the used algorithms. Block diagrams will be used to depict the general architecture. The names within the single blocks are the names of the modules realizing the respective functionality within the simulation chains. Where needed a more precise description of the algorithms will be given and especially the adjustments needed to transform them to FBMC mode will be addressed.

The usage of the simulator is handled within a later chapter.

2.1 General overview

Figure 1 shows the block diagram of the general structure of the downlink chain, Figure 2 that of the uplink chain:

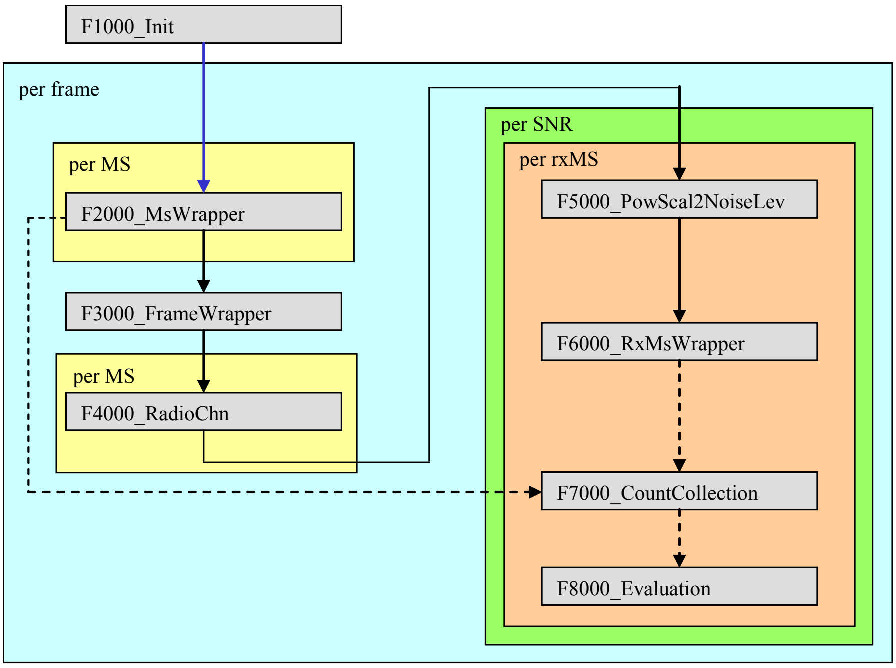

Figure 1: General architecture of the downlink simulator

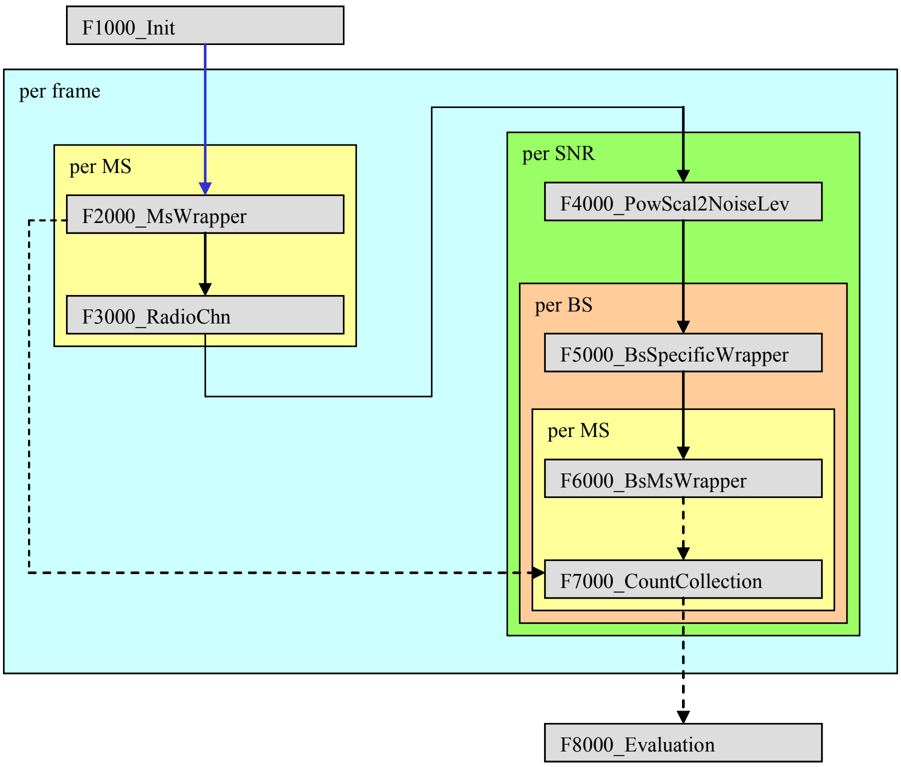

Figure 2: General architecture of the uplink simulator

Before the actual simulation the initialization is done (‘F1000_Init’). Calculations which just have to be executed once are done here and stored within the workspace for later use.
Data transmission in WiMAX is frame based. Thus the actual transmission part of the simulators is called frame wise. This is depicted via the blue colour within the diagrams. A more detailed view on framing in WiMAX is given later.
Next is the so called MS wrapper (‘F2000_MsWrapper’). Here the mobile specific elements of the transmitter such as data generation and forward error correction are located. Several mobiles can be served in downlink or are allowed to transmit data in uplink. Thus the MS wrapper may be called several times, once per defined mobile.
In the uplink every mobile generates its own data frame, transforms it into time domain and transmits it to the BS (multipoint to point). Hence the frame building here is located within the MS wrapper. This is different in the downlink. Here just a single frame is built by the BS containing the data of all mobiles (point to multipoint). Therefore a separate frame wrapper is needed within the downlink chain (‘F3000_FrameWrapper’). Naturally it is just called once per frame independent of the number of mobiles.
Then the time signals are transmitted to the BS from all mobiles (uplink) and from the BS to all mobiles (downlink). So the module representing the mobile channel again gets called per mobile (‘F3000_RadioChn’, ‘F4000_RadioChn’). A closer view onto the used models will be given later.
Link level results are typically obtained as a function of the SNR. Thus the receiver block is called several times, depending on the number of predefined SNR values. The modules ‘F4000_PowScal2NoiseLev’ and ‘F5000_PowScal2NoiseLev’ respectively are adding the noise to the received signals. In uplink we have a single receiver (the BS) and thus this module is called just once per SNR value. However, in downlink several receiver may be present (the mobiles) and thus this module gets called per receiving mobile and per SNR value in this case.
Next is the detection of the data with the help of the received signals. Again slightly differences between uplink and downlink are present. In uplink first a basestation specific wrapper (‘F5000_BsSpecificWrapper’) performs the frame specific operations (e.g. FFT). Then a wrapper called per receiving mobile (‘F6000_BsMsWrapper’) collects the data symbols out of the frame and performs the actual data detection. In downlink, as every mobile gets its own signal, the whole signal processing can be performed collectively within a single wrapper (‘F6000_RxMsWrapper’).
Diverse numbers are collected in ‘F7000_CountCollection’, such as the number of Bits transmitted and the number of erroneous Bits.
Finally, again once per simulation, the evaluation is done (‘F8000_Evaluation’).
Up to this point just a general view of the simulators is provided. The following chapters will go more into the details of the single blocks.

2.2 Initialization ('F1000_Init')

The principal parameters of the simulator can be chosen with the help of several parameter files (more on that later). To keep the complexity of setting up a simulation as low as possible just the most general settings are to be provided (e.g. channel model, logical placement of the data). Any parameter that can be derived with the help of those, is determined within the initialization (e.g. channel characteristics, physical placement of the data).
Figure 3 depicts the general structure of the initialization phase for the downlink, Figure 4 that for the uplink:

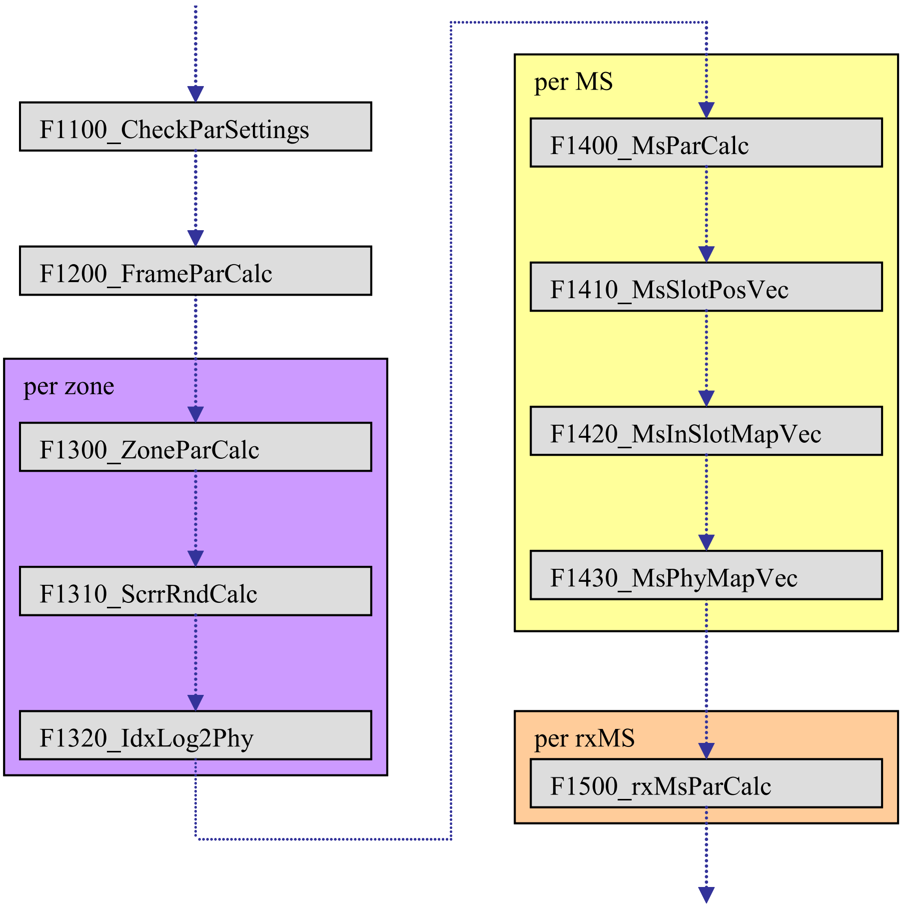

Figure 3: Parameter initialization (downlink)

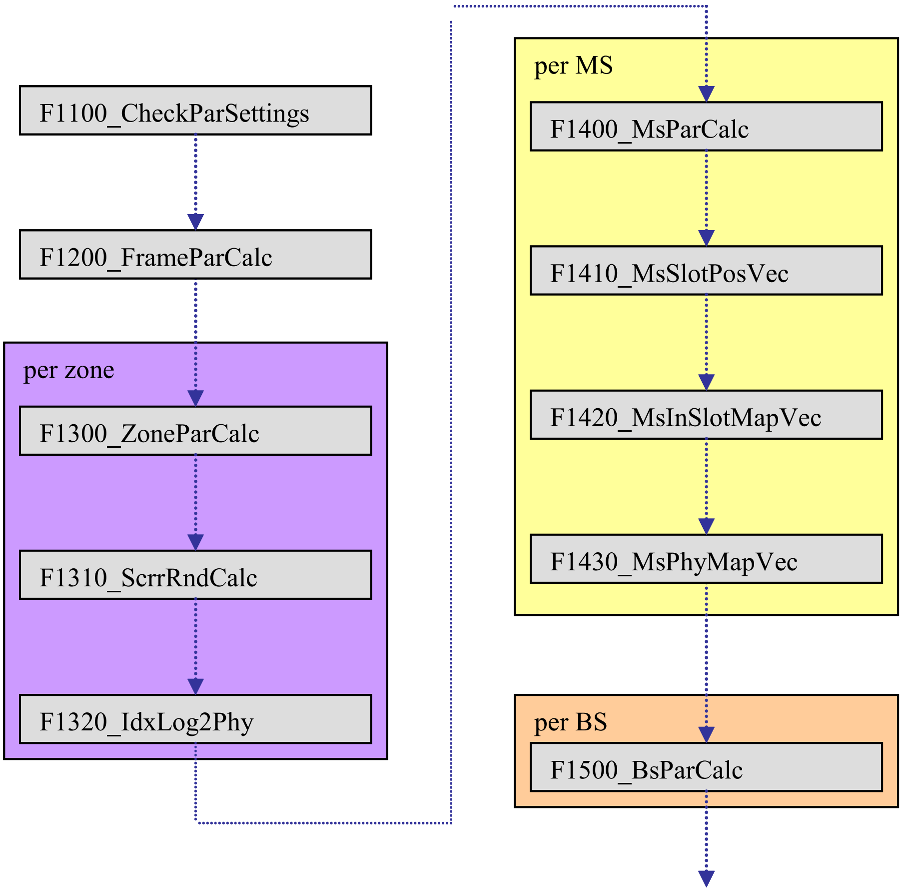

Figure 4: Parameter initialization (uplink)

Obviously the general structure of the initialization is rather similar between uplink and downlink. This is clear, as the general principles used for uplink and downlink are comparable.
Both in uplink and in downlink a number of consistency checks are accomplished before the actual initialization ('F1100_CheckParSettings'). This way an erroneous parameter setting shall be identified. Due to the vast number of parameters these checks never will be exhaustive. Thus, when preparing the parameter files, the user still has to be attentive.
A closer look into the details of the initialization phase is left out here, as later chapters, describing the algorithms using the parameters initialized here, will provide them.

2.3 Signal generation ('F2000_MsWrapper', 'F3000_FrameWrapper')

This chapter will give a closer look into the way the signals to be transmitted are generated and the algorithms used therefore. As mentioned earlier the functions described here and in the following chapters are called once per frame.
The generation of the transmit signals in downlink is twofold. First the MS wrapper is called once per defined mobile and performs the MS specific functions from the generation of the random bit stream up to the symbol stream ready to be transmitted. Then the frame wrapper builds the actual frame and performs the IFFT.

2.3.1 Symbol generation

A general structure of the downlink MS wrapper is shown in Figure 5:

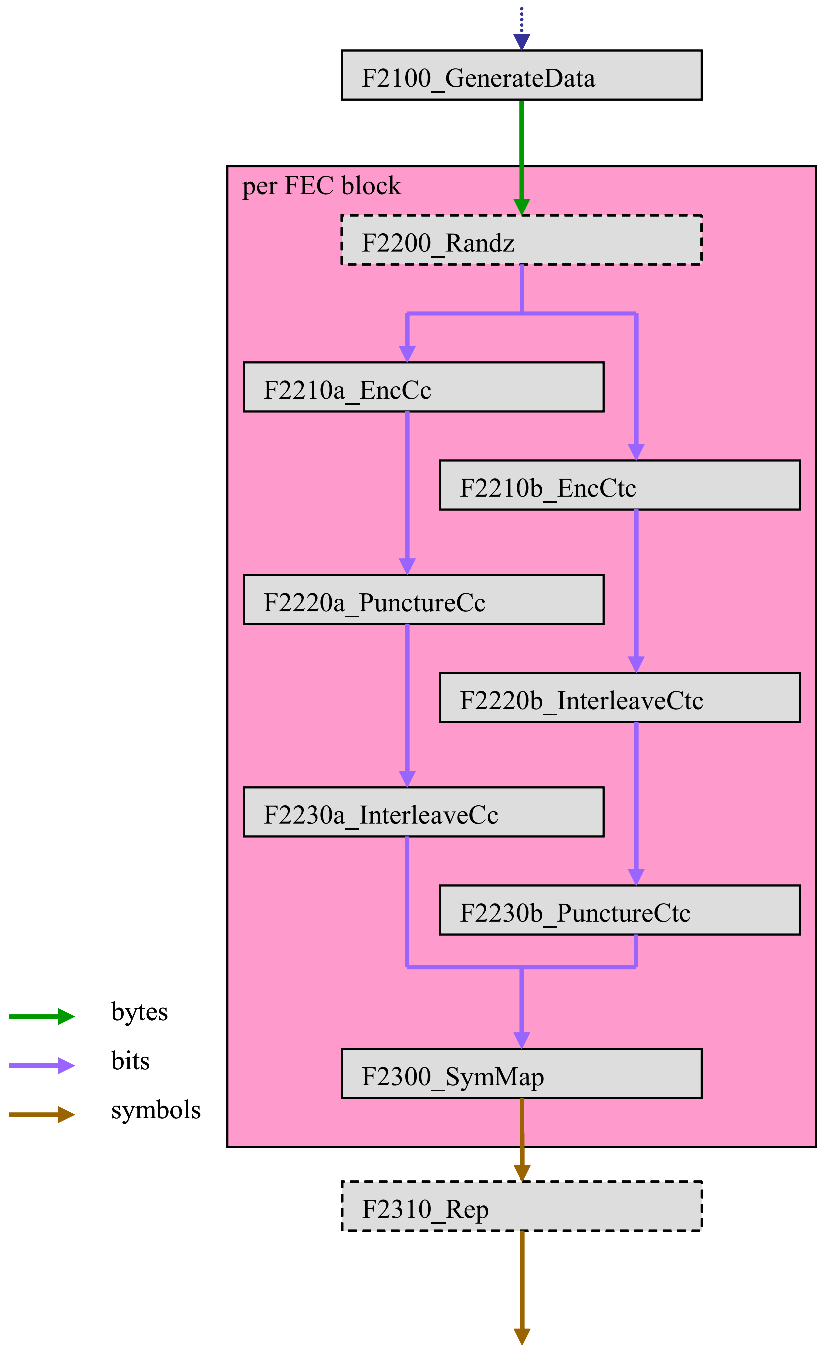

Figure 5: MS wrapper (downlink)

First the random data to be transmitted within the actual frame is produced (‘F2100_GenerateData’). The number of bits to be generated and with it the transfer rate depends on the respective size of the allocation and the used modulation and coding scheme (MCS) and is precalculated within the initialization phase. The complete bit sequence is parted into so called FEC blocks. After an optional randomization (‘F2200_Randz’) (bitwise multiplication with a given PRBS sequence) the FEC (forward error correction) encoding takes place. Two possible schemes are available:

convolutional coding (CC) and convolutional turbo coding (CTC). Even if the former is mandatory and the latter just optional, networks already deployed today and in future are using and will use CTC. Therefore it is highly recommended to use CTC in simulation runs performed within PHYDYAS, either. Figure 6 depicts the used CTC encoder ('F2210b_EncCtc'):

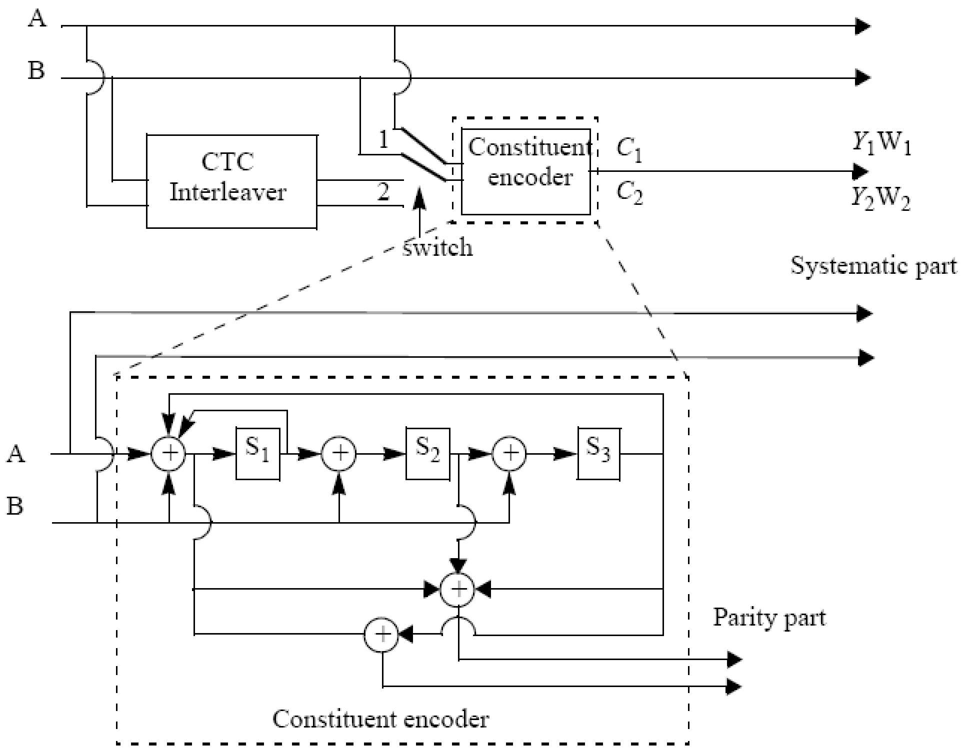

Figure 6: CTC encoder

The basic building blocks of a turbo encoder are an interleaver and in this case a convolutional encoder. The sequence to be encoded is alternately fed into A and B. Two bits entering the encoder cause six bits leaving it (the unchanged streams: A and B; two directly convolutional encoded streams: Y_1 and W_1; and two convolutional encoded streams after interleaving: Y_2 and W_2). The principal functionality of a convolutional encoder and an interleaver are assumed to be known. As the following blocks expect a single stream multiplexing is performed (A, B, Y_1, Y_2, W_1, W_2…).

To avoid error bursts the coded bit stream is interleaved ('F2220b_InterleaveCtc').

Obviously the rate of the CTC encoder is 1/3. However, the supported code rates (1/2, 2/3, 3/4 and 5/6) in WiMAX are higher and thus puncturing ('F2230b_PunctureCtc') is performed.

For the sake of completeness the implemented convolutional encoder (‘F2210a_EncCc’):

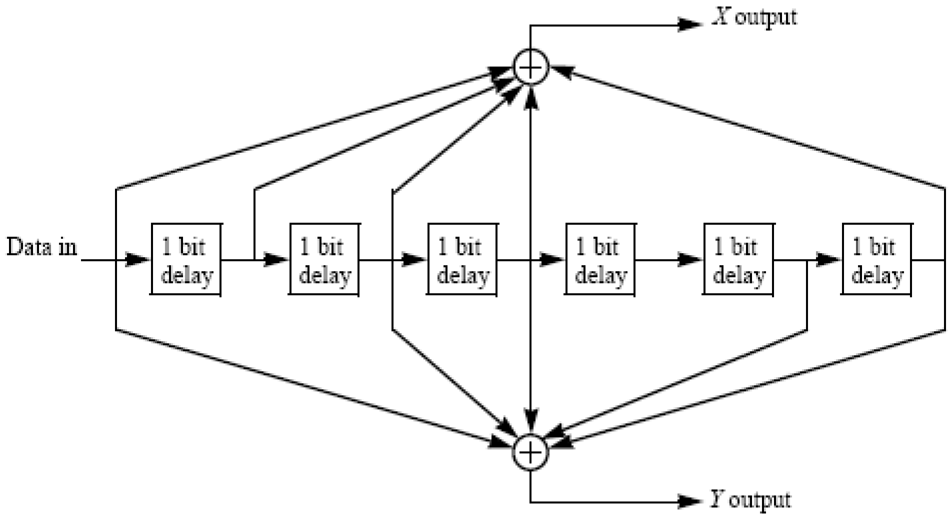

Figure 7: CC encoder

The tail-biting principle is applied. This means that the encoder memory is initialized with the last bits of the sequence to be encoded. The native rate of the CC encoder is 1/2. Puncturing is used to adjust the chosen rate (‘F2220a_PunctureCc’). As within the CTC chain, interleaving is done to spread burst errors (‘F2230a_InterleaveCc’).

Finally the symbol mapping (‘F2300_SymMap’) is done. Supported are QPSK, 16 QAM and 64 QAM.

Indicated by the dashed box an optional repetition coding can be performed (‘F2310_Rep’). Possible choices are 2, 4 and 6 times repetition.

The algorithms used up to here are state of the art of communications for a long time, thus it is desisted from a more detailed description here.

Before heading to the details of the frame wrapper, it is sensible to first start talking about the MS wrapper of the uplink. Figure 8 illustrates its block diagram:

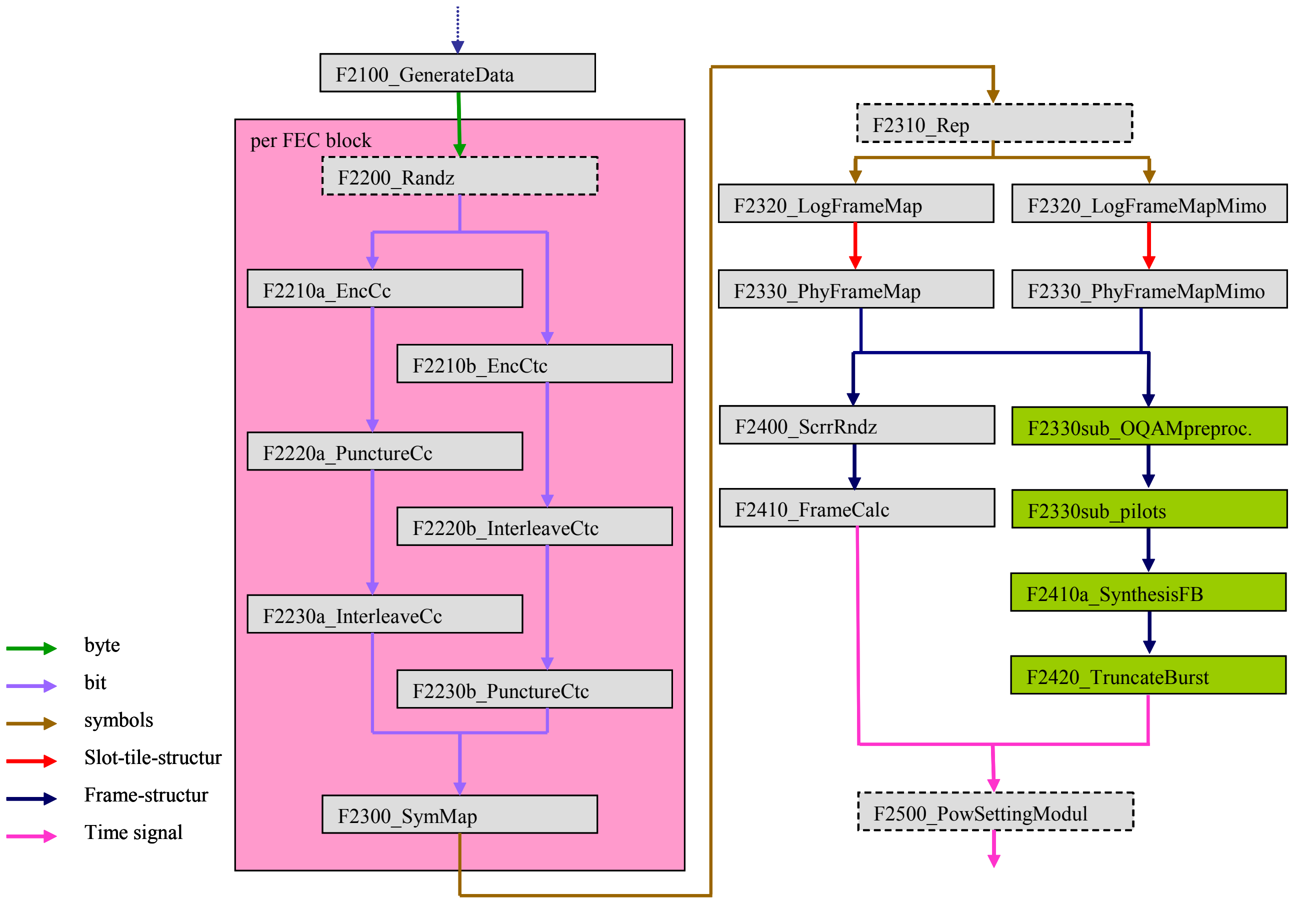

Figure 8: MS wrapper (uplink)

The additions needed to transform the base version of the simulator to FBMC are marked in Figure 8 and Figure 9 by the green colouring. Grey boxes are representing unchanged functions; either as they can be used in FBMC mode the same way as in WiMAX mode or they are not called if FBMC mode is active.

The generation of the symbol stream to be transmitted is similar to the procedures used in the downlink. The only difference is the fact, that in uplink in contrast to the downlink 64QAM is not mandatory, but only optional. However, the simulator used within the project supports 64QAM in the uplink, either. The remaining parts of the MS wrapper within the uplink chain provide the same functionalities as the elements of the frame wrapper in the downlink chain. The rest of this chapter is dedicated to their description.

The cause for these slightly diverse architectures is the fact, that in uplink every mobile generates its own frame and thus the frame building functions are called per MS. In downlink just a single frame is built by the basestation, even if several mobiles are defined. Thus the frame wrapping functionalities are separated from the symbol stream generating elements within the downlink chain.

Up to the point where the symbol streams to be transmitted are available no adjustments are necessary to evolve the simulators to FBMC mode. All the implemented principles are applicable to

the new physical layer the same way as they are to WiMAX. This changes when building the frames.

2.3.2 Framing

The general structure of the downlink frame wrapper is as follows:

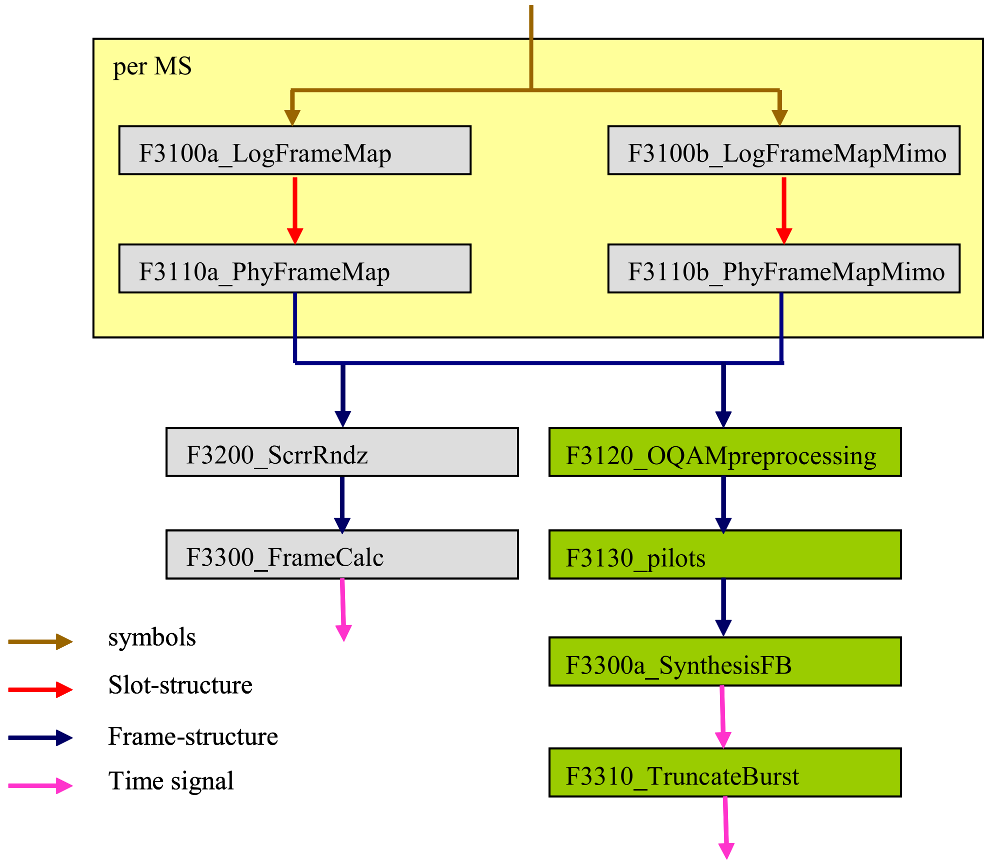

Figure 9: Frame wrapper (downlink)

The task of this part of the simulation chain is to build the frame with all its parts, namely preamble, frame control channels and the data bursts. Before heading to the actual implementation, a general description of frame building in WiMAX follows.

2.3.2.1Framing in WiMAX – a general description

In OFDM based transmission systems data is transmitted via a stream of OFDM symbols. Each symbol consists of a given number of subcarriers, each carrying a complex data symbol, excluding the guards, the dc carrier and the carriers transmitting pilots. Thus the data, that is to be transmitted, can be distributed both in time and in frequency direction. To keep the transmission manageable this stream of OFDM symbols is portioned into frames, each consisting of a given number of OFDM symbols. Within this frame both the control data and the user bursts are located.

Before heading to the details some basic parameter settings are provided:

Table 1: Basic parameter choices

access scheme	OFDMA, TDD
frequency band	2.5 GHz
frame length	5 ms
# of OFDM symbols per frame (UL + DL)	47
duration of one OFDM symbol	91.4 µs
length of cyclic prefix	1/8 of the duration of a single OFDM symbol
FFT size	1024
bandwidth	10 MHz
carrier spacing	10.94 kHz
sampling frequency	11.2 MHz
TTG	105.7 µs
RTG	60 µs

Figure 9 shows the general frame structure used in WiMAX in TDD mode (time division duplexing):

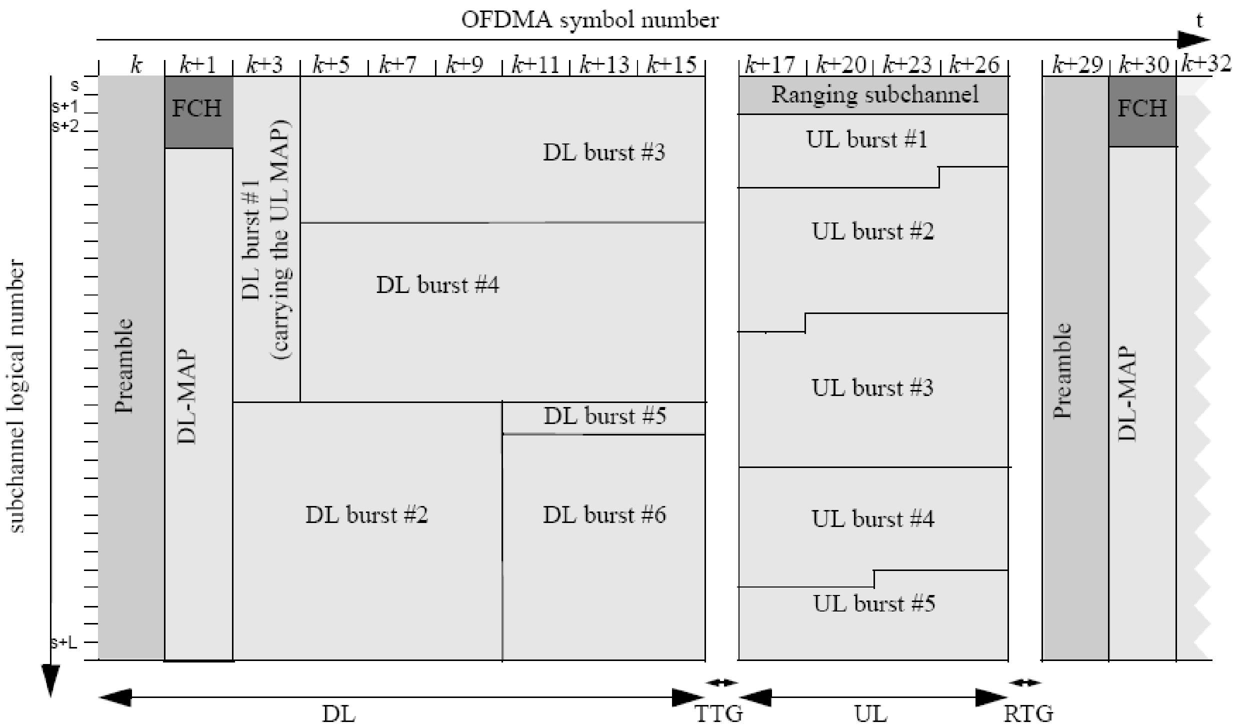

Figure 10: OFDMA frame

A single frame consists of 47 OFDM symbols. The first x (x=26…35) symbols build up the DL subframe, the rest (y=47-x) the UL subframe. Uplink and downlink are sharing the frequency band. Separation is achieved by time division. If uplink transmissions are active, downlink is silent and vice versa. To allow the base station to turn around small time gaps between the subframes are inserted (TTG and RTG). The length of those gaps is a rather conservative choice by the WiMAX Forum. The standard itself claims minimum gaps of 5 µs. In FDD mode (frequency division

multiplexing) both uplink and downlink may transmit data without interception, they are separated in frequency domain.

The first symbol of the frame carries the preamble. Figure 11 depicts its structure:

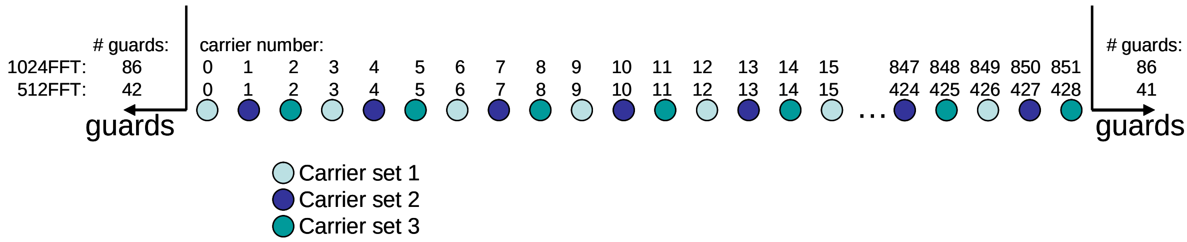

Figure 11: Preamble

The dots are representing the single subcarriers. On both ends of the symbol guard bands are located. The overall preamble consists of three carrier sets. The frame can be parted into 3 segments, e.g. to serve three sectors. The three carrier sets are directly associated to that segments. Within the link level simulator for PHYDYAS always just a single segment is active (using the complete frequency band). Thus just carrier set 1 is needed. The subcarriers belonging to carrier set 2 and 3 are not active, they carry null symbols. The single carriers of the preamble are BPSK modulated (mapping: 0 → 1, 1 → -1; relative amplitude: $2\sqrt{2}$), the DC carrier shall not be modulated. 114 possible binary sequences are provided by [2]. By correlating the received symbol with all possible ones, the mobile can determine which one was transmitted. This way it can detect in which segment it is located and evaluate the parameter IDcell. This parameter affects the data permutation of the first zone (more on this later in this deliverable). Up to now just a single binary sequence per FFT size is incorporated (segment 0, IDcell=2). In hexadecimal format:

1024 FFT: 1C75D30B2D F72CEC9117 A0BD8EAF8E 0502461FC0 7456AC906A DE03E9B5AB 5E1D3F98C6 E
512 FFT: 8EB62664E3 B2C5222DE1 8E9000561F 25AAFC

Other purposes of the preamble is a first coarse time synchronization of mobiles entering the network and channel estimation.

Once the mobile has roughly synchronized to the frame timing, it needs to know the general structure of the frame, such as length of the downlink and uplink subframe in number of OFDM symbols and position of the ranging subchannels (the latter are used for a finer synchronization of the mobile to the frame timing). This information is carried within the frame control channels, namely the FCH (frame control header), the DL MAP and the UL MAP. The parameters used to transmit the FCH are always the same (such as modulation and coding scheme, position within the frame, length). Thus all mobiles are always able to extract its content, the so called DL frame prefix. After a mobile has received the frame prefix, it is able to collect the symbols of the DL-Map and its content, a MAC message (media access control) containing several information elements. These information elements e.g. specify the positions of the data bursts within the downlink subframe, the IDs of the mobiles the single bursts are dedicated for, optionally existing zone switches within the frame (more on zones in WiMAX later) and the position and the length of the UL MAP. The task of the latter is comparable to that of the DL MAP, however, for the uplink subframe. Once the DL MAP and the UL MAP are decoded, the mobile fully is aware of the structure of the frame. As PHYDYAS mainly is concerned with physical layer aspects, it is assumed that the simulated

mobiles always are capable of decoding the frame control messages correctly. This assumption is valid, as the transmission of the control messages typically is much more robust than the transmission of the data (the most robust modulation scheme (QPSK), more redundancy (code rate 1/2, several repetitions)).
The remaining part of the subframe is used for data transmission.

Both the downlink and the uplink subframes can be divided into so called zones. Figure 12 shows this exemplarily:

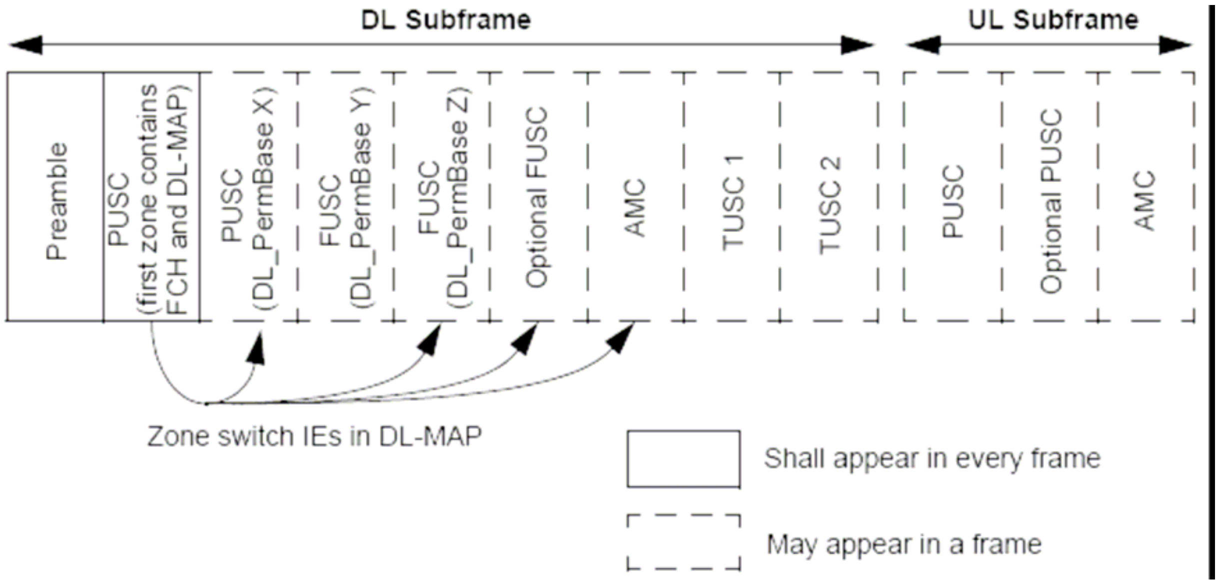

Figure 12: Division of the subframes into zones

The preamble is not part of any zone. Many different zone types are defined in [2], however, in [3] just a few of them are chosen to reduce complexity. They are PUSC (partial usage of subchannels,), FUSC (full usage of subchannels) and AMC (adaptive modulation and coding) in downlink and PUSC and AMC in uplink. Although FUSC is implemented, it is recommended to concentrate the investigations to PUSC and AMC within PHYDYAS, as FUSC is rather similar to PUSC and does not provide any new aspects. Obviously just the first zone must be a PUSC zone. The layout of the rest of the frame is up to the system designer.

The main difference between PUSC and AMC is the distribution of the data onto the available subcarriers. PUSC follows a frequency diverse approach, AMC a frequency selective one. Furthermore the placement of the pilots is different. As these aspects are fundamental for the link level performance and as substantial adjustments have to be done here to introduce FBMC into the system, a detailed description follows.

The smallest possible allocation unit in WiMAX is the so called slot. A slot contains x subcarriers over y OFDM symbols, depending on the zone type:

Table 2: Slot properties depending on the zone types

	PUSC, uplink	PUSC, downlink	AMC
x, y	24, 3	24, 2	18,3
# data carriers	48	48	48
# pilot carriers	24	0	6

Common to all types is the number of data carriers per slot. This way the partitioning of the data stream to be transmitted into packages is independent of the allocation type. Important for data transmission in general is the approximate knowledge of the channel. Pilots are placed regularly into the frame, with which the receiver can gain this knowledge (more on this later). For PUSC in the uplink and AMC those pilots are part of the slots. Figure 13 and Figure 14 depict cut-outs of the zones, to illustrate the architecture of the slots:

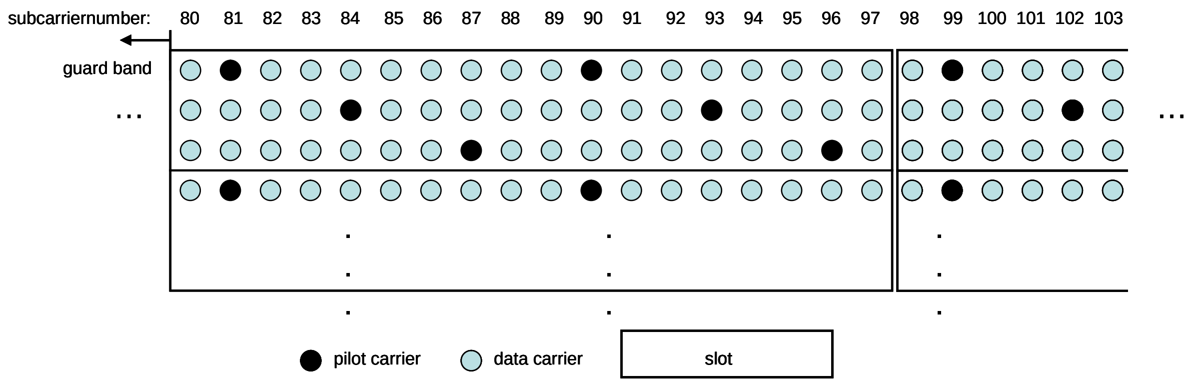

Figure 13: Pilot and data placing in AMC

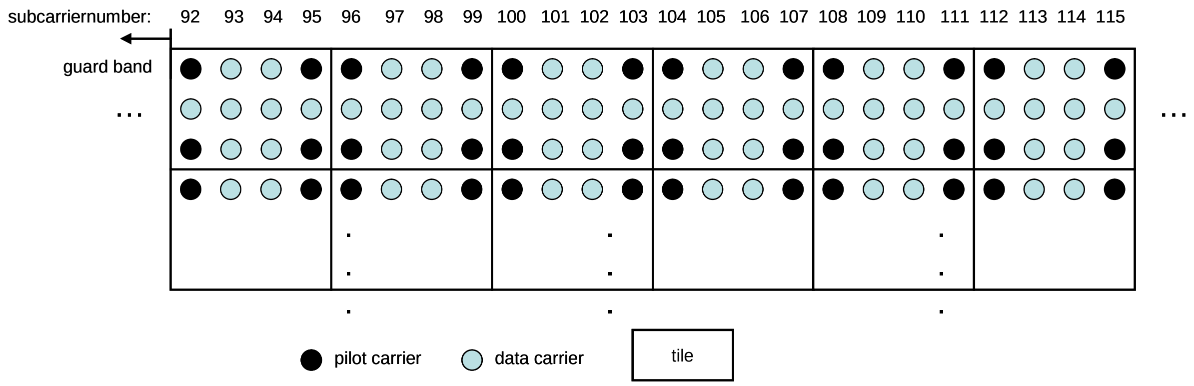

Figure 14: Pilot and data placing in uplink PUSC

Horizontally frequency changes, indicated by the growing subcarriernumber, downwards time flows. The light blue dots represent data carriers, the black ones pilots. Obviously all carriers within a slot in AMC are contiguous to gain frequency selectivity. With PUSC in the uplink, slots are further divided into tiles. A single tile consists of 8 data carriers, surrounded by 4 pilots. 6 not contiguous (in frequency) tiles sharing the OFDM symbols compose a slot. This way frequency diversity is gained without loosing the capability to estimate the channel efficiently.

With PUSC in the downlink the situation is different. Here the pilots are not part of the slots. Furthermore, to maximize frequency diversity, the data carriers belonging to the same slot are spread over the complete frequency range. Detached from the grouping of the data carriers into slots, the pilots are placed regularly into the zone. To do this placement the zone is arranged into clusters:

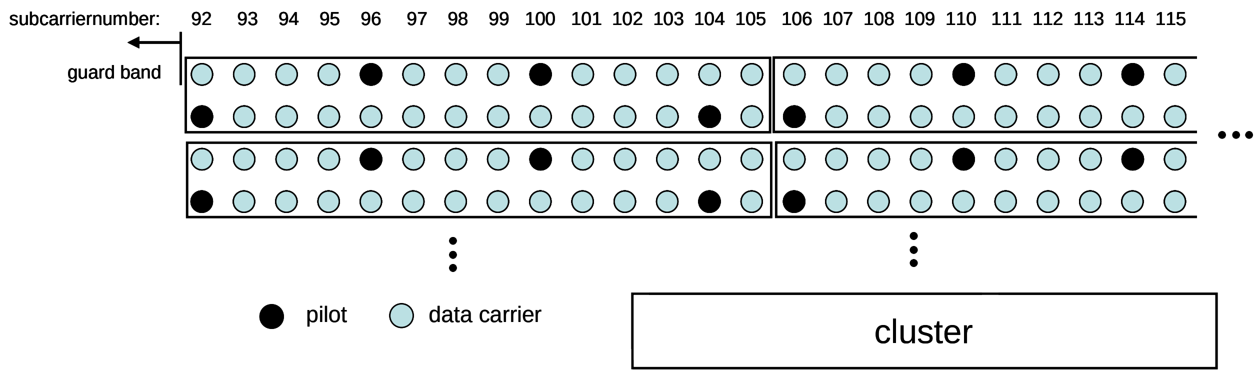

Figure 15: Pilot and data placing in downlink PUSC

The data carriers within the clusters are assigned to different slots. Thus they typically are allocated to different mobiles. Because of this common pilots unlike dedicated pilots are to be used in downlink PUSC. Special cases do exist (catchword beamforming), where it is made sure, that a single mobile gets all data carriers within the clusters allocated, then the pilots are dedicated. To illustrate the frequency diversity of PUSC and the frequency selectivity of AMC Table 3 includes the subcarriernumbers of the first slot both in time and frequency direction within the zone exemplarily (the fat numbers represent pilot positions):

Table 3: Subcarrier mapping to slots

	subcarriernumbers	time index
PUSC (DL)	925 931 767 772 836 846 617 623 893 898 682 692 98 104 458 463 752 762 743 749 290 295 528 538	1
	925 930 768 772 837 845 617 622 894 898 683 691 98 103 459 463 753 761 743 748 291 295 529 537	2
PUSC (UL)	**60** 61 62 **63 144** 145 146 **147 192** 193 194 **195 297** 298 299 **300 337** 338 339 **340 393** 394 395 **396**	1
	60 61 62 63 144 145 146 147 192 193 194 195 297 298 299 300 337 338 339 340 393 394 395 396	2
	60 61 62 **63 144** 145 146 **147 192** 193 194 **195 297** 298 299 **300 337** 338 339 **340 393** 394 395 **396**	3
AMC	80 **81** 82 83 84 85 86 87 88 89 **90** 91 92 93 94 95 96 97	1
	80 81 82 83 **84** 85 86 87 88 89 90 91 92 **93** 94 95 96 97	2
	80 81 82 83 84 85 86 **87** 88 89 90 91 92 93 94 95 **96** 97	3

Important to note is, that it is mandatory to transmit all pilots in downlink, even if not all slots are allocated to mobiles. In uplink naturally this is different, as in this case collisions between transmitted pilots of different mobiles would occur. Finally in downlink PUSC and in AMC (both downlink and uplink) the pilots are boosted by 2.5 dB compared to the data.

Table 4 summarizes the carrier allocations for 1024-FFT:

Table 4: 1024-FFT carrier allocations

	PUSC uplink / downlink	AMC
# of guard subcarriers (left)	92 / 92	80
# of guard subcarriers (right)	91 / 91	79
# of pilot carriers	420, 0 / 120	96
# of data carriers	420, 840 / 720	768
# of slots in freq. direction	35 / 30	48

An important measure in data transmission is the achievable data rate. It depends on the used modulation scheme, the code rate and the number of allocated subcarriers. Being the smallest possible allocation unit, the slot dictates the granularity of the possible data rates. Table 5 includes the number of transmitted Bits per slot and the resulting data rate for all combinations of modulation scheme and code rate are computed:

Table 5: Data rates per slot

	QPSK	16 QAM	64 QAM
code rate 1/2	48 bits/slot 9600 bits/s	96 bits/slot 19200 bits/s	144 bits/slot 28800 bits/s
code rate 2/3	not part of the WiMAX profile	not part of the WiMAX profile	192 bits/slot 38400 bits/s
code rate 3/4	72 bits/slot 14400 bits/s	144 bits/slot 28800 bits/s	216 bits/slot 43200 bits/s
code rate 5/6	not part of the WiMAX profile	not part of the WiMAX profile	240 bits/slot 48000 bits/s

The choice of the modulation scheme and the code rate to use is done on the basis of channel quality measurements. Depending on the targeted data rate and on the used modulation scheme and code rate, the user gets the corresponding number of slots allocated.
Downlink PUSC has 30 slots spanning 2 OFDM symbols available to transmit data, uplink PUSC 35 slots spanning 3 OFDM symbols and AMC 48 slots spanning 3 OFDM symbols, either. Thus, if a zone includes e.g. 18 OFDM symbols, the number of slots usable for data transmission are 270 for downlink PUSC, 210 for uplink PUSC and 288 for AMC (all numbers are valid for 1024 subcarriers per OFDM symbol).

The allocation of transmit opportunities is done on the basis of the slots. Each mobile allowed to transmit/receive data within a frame gets a number of slots assigned. Their number, the frequency the user gets a portion of the frame (user burst) and the MCS used determine the transmission rate (if HARQ is activated, this number is slightly reduced, but the transmission of the data gains robustness). In the downlink the shape of the bursts always are rectangular (see Figure 10). Figure 16 is a zoomed in version of the frame. Two horizontally adjacent rectangles (→ this zone is in PUSC mode, in case of AMC it would be three adjacent rectangles) are representing a single slot (e.g. k and k+1 in line 7). The grey ones are assigned by the scheduler to an exemplary user burst. The allocation of the slots to the burst is done frequency first. The same holds for the placing of the data into the slots.

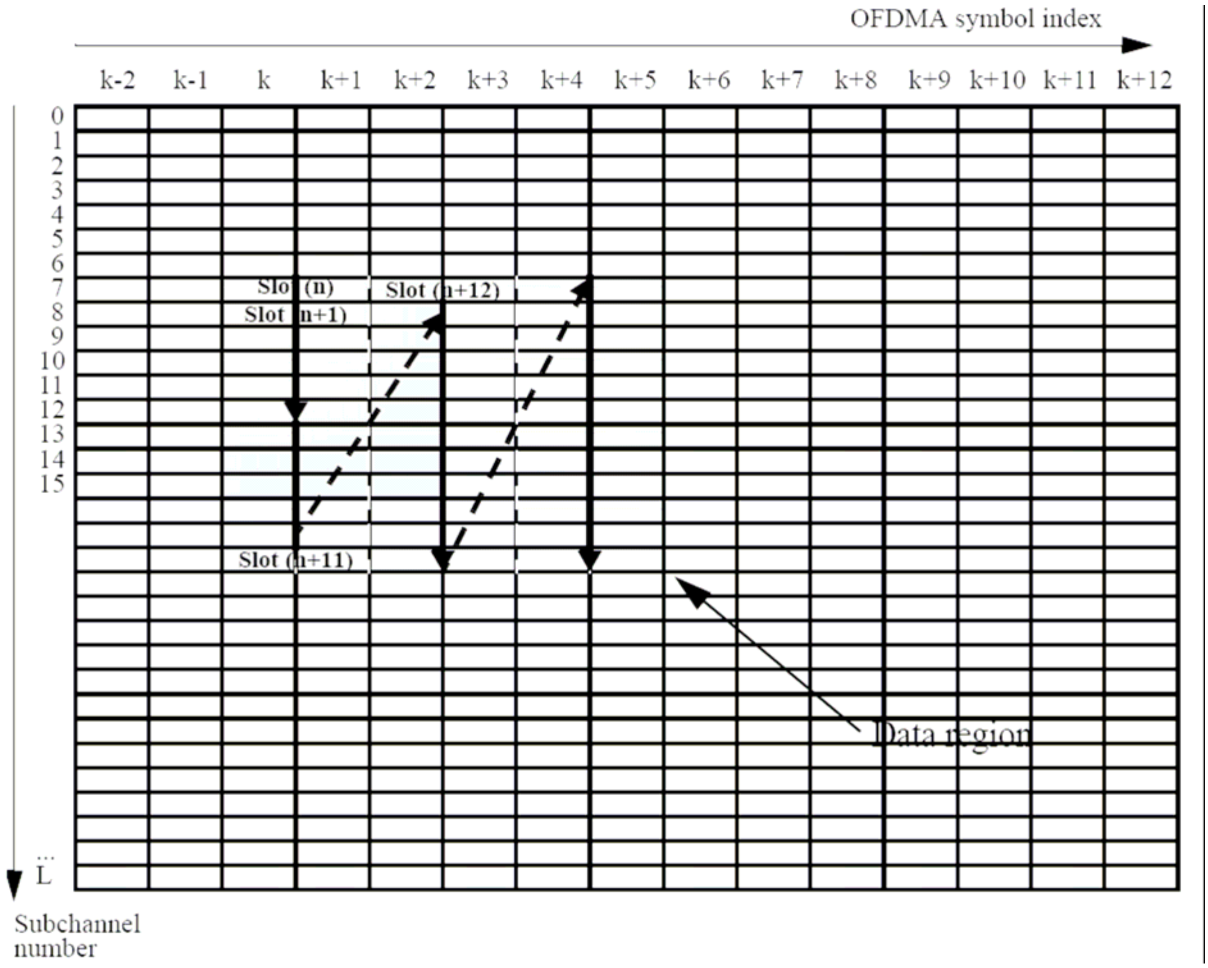

Figure 16: Burst shaping in downlink

In the uplink the situation is somewhat different. As already observed, data allocations in the uplink are not necessarily rectangular. The allocation is as follows: the scheduler arranges the users (indicated in Figure 10 by the fact, that the user numbers within the UL subframe grow top down). Then the scheduler starts placing the allocation of user 1 within subchannel 0 and timeslot 0. The allocation then grows in time direction until the end of the zone is reached (→ time first allocation). After that the allocation of slots starts at the next subchannel. This is done until the allocated number of slots for user 1 is reached. Then user 2 get's its allocation. It starts where the allocation of user 1 ended, again in time direction first. This is done until all users received their allocation. The placement of the data within the given allocation region then again is frequency first. Figure 17 depicts this:

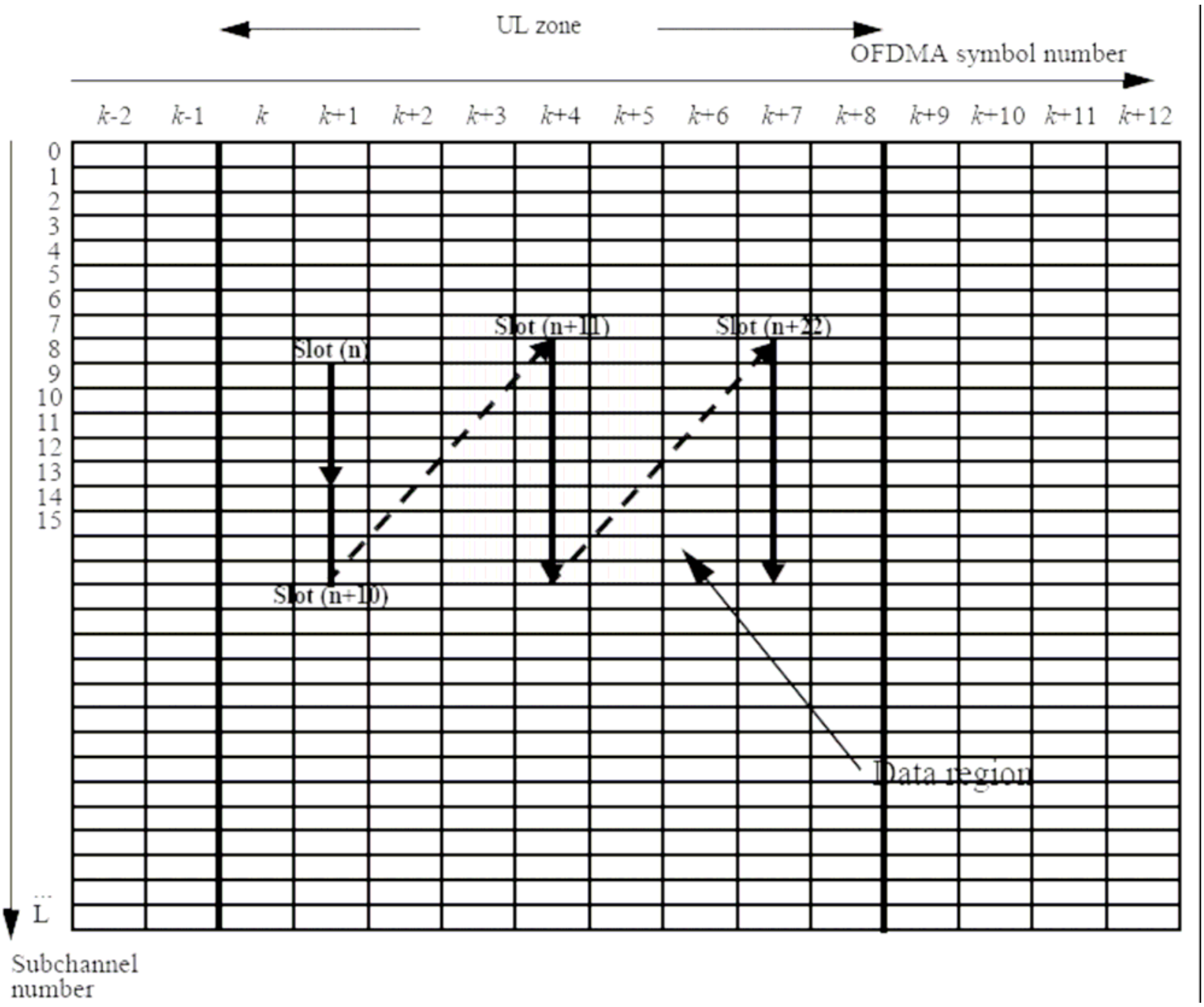

Figure 17: Burst shaping in uplink

As described earlier, in uplink the placing of the users is done time first. Without further intervention, such an approach would harm the target of PUSC to maximize frequency diversity, as the same subcarriers would be used for different time slots. Therefore the so called slot rotation is used. Figure 18 illustrates this:

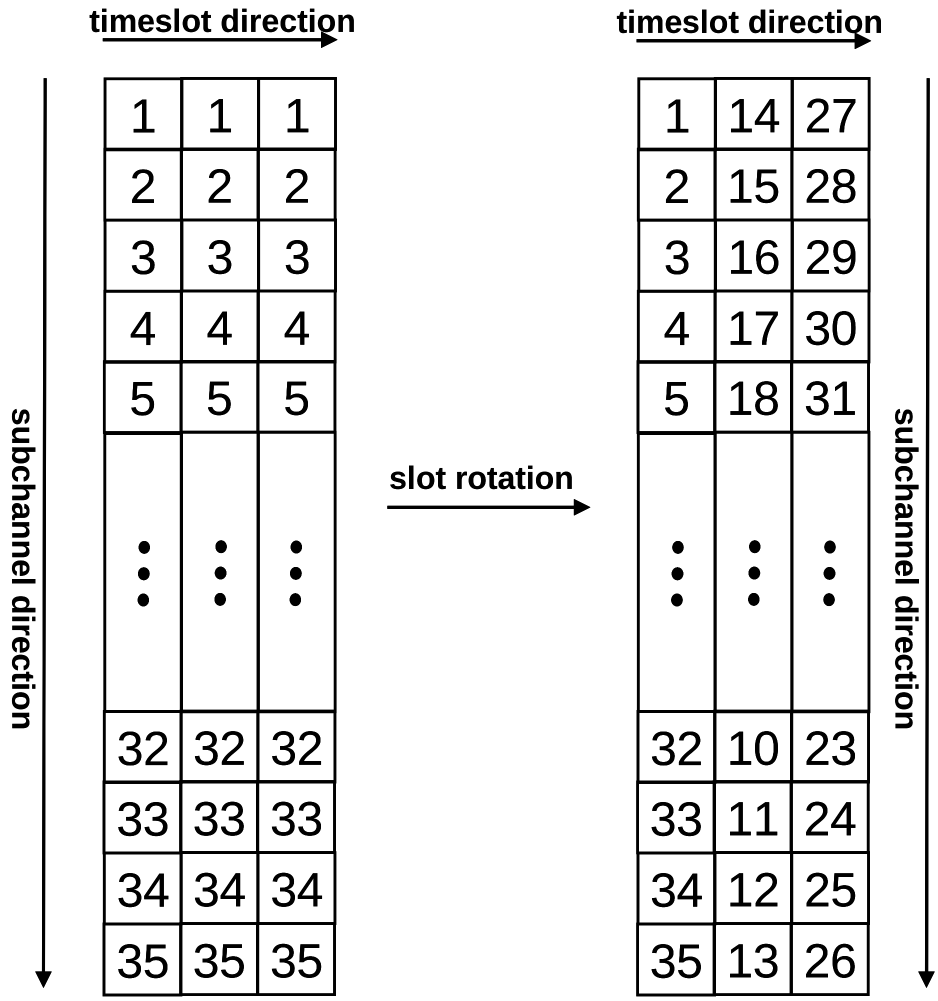

Figure 18: Slot rotation

On the left side a PUSC zone is shown before slot rotation. The slots lying inside the same timeslot are numerated in direction of growing subchannel number (1024 FFT → PUSC zone has 35 subchannels; timeslot length = 3 → 9 OFDM symbols). After applying the slot rotation, slots formerly located inside the same subchannel and thus sharing the physical subcarriers now are separated. The allocation of the data described earlier in this document is based on the slot locations after the slot rotation. Thus, e.g. user one (8 slots burst size) gets slot numbers 1, 2 and 3 within the first timeslot, 14, 15 and 16 in the second and finally 27 and 28 in the third timeslot. The placing of the data is frequency first thus the order is: 1, 2, 3, 14, 15, 16, 27, 28 within the respective time slot. Without the slot rotation a burst would use the same subcarriers within the zone. With the slot rotation this is prohibited, thus frequency diversity is reached.

Once all subcarriers are modulated the so called subcarrier randomization is performed. Here all subcarriers are multiplied with either +1 or -1 depending on a precalculated PRBS sequence. The target is the randomization of the interference in multi-cell deployments.

Finally, before the signal gets transmitted, the cyclic prefix (1/8 of the OFDM symbol time) is added and the signal is transformed into time domain with the help of an IFFT.

2.3.2.2 Framing in WiMAX – implementation issues and transformation to FBMC mode

A stream of symbols is available to the downlink frame wrapper and the respective elements in the MS wrapper of the uplink chain exactly fitting into the allocated bursts for every user. The task is to first rearrange this symbol streams into logical blocks (‘F3100a_LogFrameMap’ and ‘F3100b_LogFrameMapMimo’ in the downlink chain, ‘F2320_LogFrameMap’ and ‘’F2320_LogFrameMapMimo’ in the uplink chain), place them into the physical frame (‘F3110a_PhyFrameMap’ and ‘F3110b_PhyFrameMapMimo’, ‘F2330_PhyFrameMap’ and ‘F2330_PhyFrameMapMimo’), add further frame elements (pilots, preamble and the frame control data, namely the FCH and the MAPs), perform the subcarrier randomization (‘F3200_ScrrRndz’, ‘F2400_ScrrRndz’) and transform the signal into time domain (‘F3300_FrameCalc’, ‘F2410_FrameCalc’).

As mentioned earlier two different versions are planed to introduce the new FBMC based physical layer. One with maximal compatibility to WiMAX and another designed with optimal performance in mind. The former one is fully integrated into the simulator today. The latter is up to future work. However, even if compatibility is to be maximized, adjustments have to be done to include the FBMC functionality. As in FBMC adjacent subcarriers are interfering each other, OQAM (offset QAM) is to be used (‘F3120_OQAMpreprocessing’ and ‘F2330sub_OQAMprocessing’). In OQAM the real and the imaginary part of the complex QAM symbols are transmitted alternately both in frequency and in time direction. This way data symbols on adjacent subcarriers are orthogonal again. Another aspect influenced by the overlap between adjacent subcarriers is the construction of the pilots. In WiMAX pilots are plain real symbols placed regularly into the frame. These probes can be used by the transmitter to directly estimate the channel at these positions. Typically channel coefficients are complex values. Thus, even if the general structure of alternating real and imaginary symbols within the frame is fulfilled making the interference to the data symbols nonhazardous, the interference still distorts the channel estimation, if FBMC mode is active To overcome this problem two alternative schemes are used: pair of pilots (POP) and auxiliary pilots (auxPilots). The processing and the placement of the pilots are done in ‘F3130_pilots’ and ‘F2330sub_pilots’ respectively. A detailed description of the processing can be found in [4] and [5]. The placement is the same as in WiMAX to maximize compatibility. Alternative placements optimized with respect to performance are up to future work.
Next is the transformation into time domain, the processing with the synthesis filter bank (‘F3300a_SynthesisFB’ and ‘F2410a_SynthesisFB’) [1][6][8]. Burst truncation is performed (‘F3310_TruncateBurst’ and ‘F2420_TruncateBurst’) [8] and finally in ‘F2500_PowSettingModul’ the user specific transmit power is set.

At this point transmit processing is finished. Baseband signals now are available ready to be transmitted over the channel. The modelling of the channel is topic of the next chapter.

2.4 Channel modelling

Both line of sight and fading scenarios and combinations can be simulated:

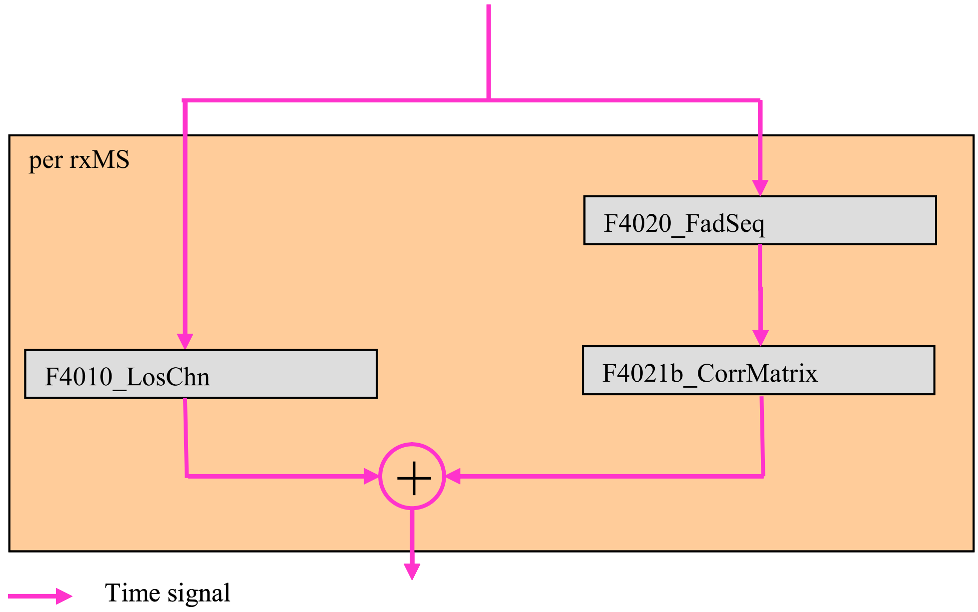

Figure 19: Channel (downlink)

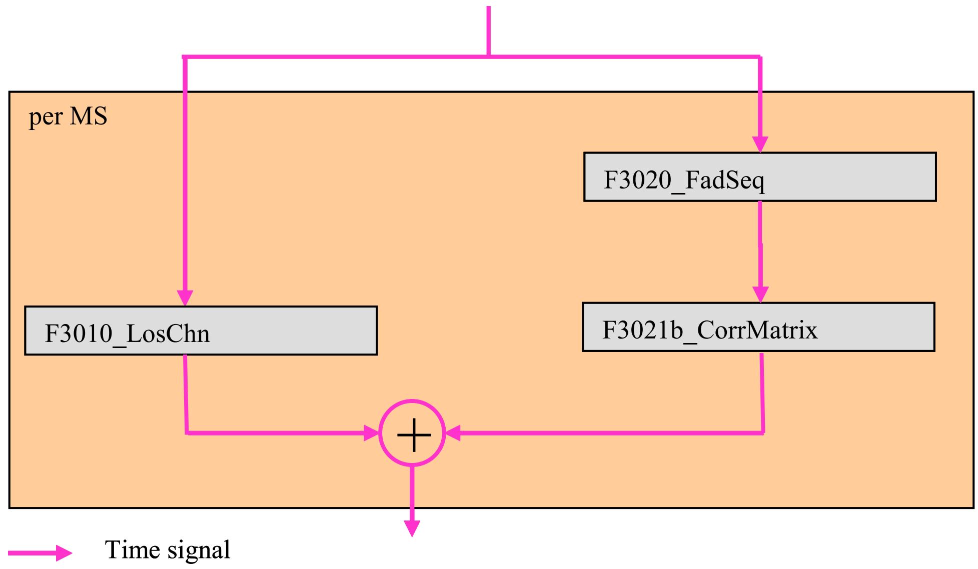

Figure 20: Channel (uplink)

Parameters describing the line of sight channel ('F4010_LosChn' and 'F3010_LosChn') are path power (in dB), Doppler shift (in Hz), direction of arrival (in degree) and the zero phase (in radian). The fading channel ('F4021b_CorrMatrix' and 'F3021b_CorrMatrix') uses the widely accepted ITU path delay model. Here the impulse response includes 6 differently delayed and differently attenuated paths, each encountering independent fast fading ('F4020_FadSeq' and 'F3020_FadSeq'). Several different profiles are defined. Table 6 summarizes the ones implemented into the simulator:

Table 6: Power delay profiles

	Ped B		Veh A	
p	path power $P_{i,p}$ [dB]	path delay $\tau_{i,p}$ [ns]	path power $P_{i,p}$ [dB]	path delay $\tau_{i,p}$ [ns]
0	0	0	0	0
1	-0.9	200	-1	310
2	-4.9	800	-9	710
3	-8	1200	-10	1090
4	-7.8	2300	-15	1730
5	-23.9	3700	-20	2510
	Veh A extended		**Veh B**	
p	path power $P_{i,p}$ [dB]	path delay $\tau_{i,p}$ [ns]	path power $P_{i,p}$ [dB]	path delay $\tau_{i,p}$ [ns]
0	0	0	-2.5	0
1	-1	310	0	300
2	-9	710	-12.8	8900
3	-10	1090	-10	12900
4	-15	1730	-25.2	17100
5	-20	10000	-16	20000

Ped B reflects the situation a pedestrian experiences. Veh A, Veh A extended and Veh B reflect the transmission of data from or to a vehicular. Beside the choice of the model the speed of the user is an important measure for characterising the channel. Frequency selectivity is dominantly influenced by the choice of the model, time variance by the speed. Another important feature of modelling wireless channels is the correlation between the different antenna links. Although the WiMAX part of the simulator already features all multiple antenna schemes recommended in [3], the description of the used models handling correlation is delayed to a later deliverable, as the FBMC versions of those schemes are still under development within other workpackages.

As the mobile typically is not aware of its position within the cell, it has to be synchronized. Initial ranging (IR, needed for network entry) and periodic ranging (PR, needed to realign the moving mobile) are done in WiMAX via special regions within the uplink frame [9] to achieve this. First the mobile synchronizes itself roughly to the frame timing with the help of the preamble. Then, after decoding the FCH, the DL-MAP and the UL-MAP, it knows those ranging regions and can transmit a randomly chosen ranging code within a randomly chosen slot in the IR region. After this message is sent, the MS waits a given amount of time (backoff window) if the BS could detect the request and responds. Once the BS could successfully detect the mobile, periodic ranging possibilities are granted. With their help the mobile gets synchronized with respect to transmit power, timing and frequency. Corrections are communicated via MAC messages. The ranging mechanism itself is rather a system level issue und thus is not incorporated in such detail into the link level simulator.

Instead the simulated MS will be treated as already ranged. Residual offsets can be set. This way the impact of non-ideal synchronization can be evaluated.
The accuracies to be fulfilled before a mobile is allowed to transmit data are defined in the standard [2]. The residual timing offset after ranging must not exceed $\pm 8 \cdot N_{ovs}$ time samples (1024 FFT) and $\pm 4 \cdot N_{ovs}$ respectively (512 FFT) with N_{ovs} being the oversampling factor. At the BS, the reference frequency accuracy shall be better then $\pm\ 2 \cdot 10^{-6}$ (= 5 kHz at 2.5 GHz). The transmitted centre frequency, the receiver centre frequency and the symbol clock frequency shall be derived from the same reference oscillator. At the MS, the transmitted centre frequency and the sampling frequency shall be derived from the same reference oscillator. MS UL transmission shall be locked to the BS and its centre frequency shall not deviate more than 2% of the subcarrier spacing (≈ 0.22 kHz) compared to the BS centre frequency.

Let $s_i(m)$ be the complex baseband signal to be transmitted by user i in uplink. Then the signal received by the basestation is, if U users are active (multiple-access channel):

$$r(m) = \sum_{i=1}^{U} 10^{\frac{P_i}{20}} e^{j\Delta\varphi_i(m)} \frac{1}{S_i} \sum_{p=0}^{P} 10^{\frac{P_{i,p}}{20}} c_{i,p}(m) s_i(m - m_{i,p} - m_i)$$

with the path-norm factor:

$$S_i = \frac{1}{\sqrt{\sum_{p=0}^{P} 10^{\frac{P_{i,p}}{20}}}}$$

Via P_i non-ideal power control can be introduced. $P_i = 0\ \forall\ i$ reflects a perfect calibration of the transmit powers. $\Delta\varphi_i(m)$ includes the user specific phase and frequency offset, m_i the path independent time offset due to a failed synchronization, $m_{i,p}$ the path dependent due to different path lengths. $c_{i,p}(m)$ represents the sequence of complex fading coefficients. P is the number of paths reaching the receiver. A look into Table 6 shows, that $P = 6$ holds in any case.
In downlink user i receives, if $s(m)$ is the signal transmitted by the basestation (broadcast channel):

$$r_i(m) = e^{j\Delta\varphi_i(m)} \frac{1}{S_i} \sum_{p=0}^{P} 10^{\frac{P_{i,p}}{20}} c_{i,p}(m) s(m - m_{i,p} - m_i)$$

Due to the random nature of fading, the received power naturally is a random variable, either. However, the mean power of the QAM constellations and the long term mean power of the fading are adjusted that way, that the received long term mean power per used data subcarrier always is unity.
Another source of distortion is noise. Typically many sources are present within a transmission chain. Within the simulator all sources are gathered and modelled by a single AWGN source located directly after the signal reception. In 'F5000_PowScal2NoiseLev' and 'F4000_PowScal2NoiseLev' the noise is added. Typically simulation runs are done with different signal-to-noise ratios. Within the simulators the generated noise always has the same mean power. Therefore the received signal either is attenuated or amplified before the noise is added, to adjust the actual SNR.

2.5 Signal processing at the receiver

Once the signal is available at the receiver, signal processing algorithms have to be applied to detect the transmitted bit sequences with highest reliability. Again slight architectural differences are necessary between the uplink and the downlink chain. In downlink several receivers may be present while a single transmitter is active (broadcast channel). Thus every receiving mobile has to perform every single reception algorithm including the frame specific ones (Figure 23). In uplink a single receiver is served by several possible transmitters (multiple access channel). Thus frame specific actions just have to be performed once (Figure 22). Only the user specific algorithms have to be performed once per user (Figure 24). The meaning of the line colouring is shown in Figure 21.

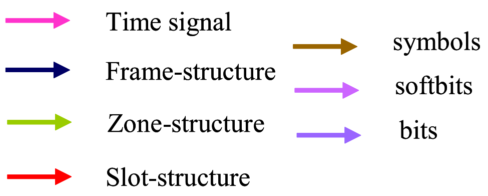

Figure 21: Line coloring

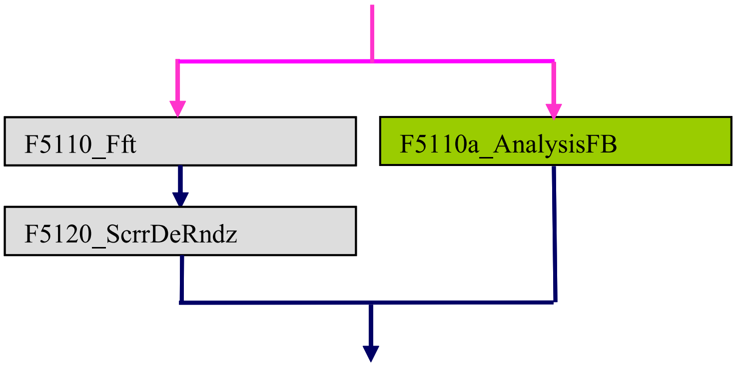

Figure 22: Common frame processing in uplink

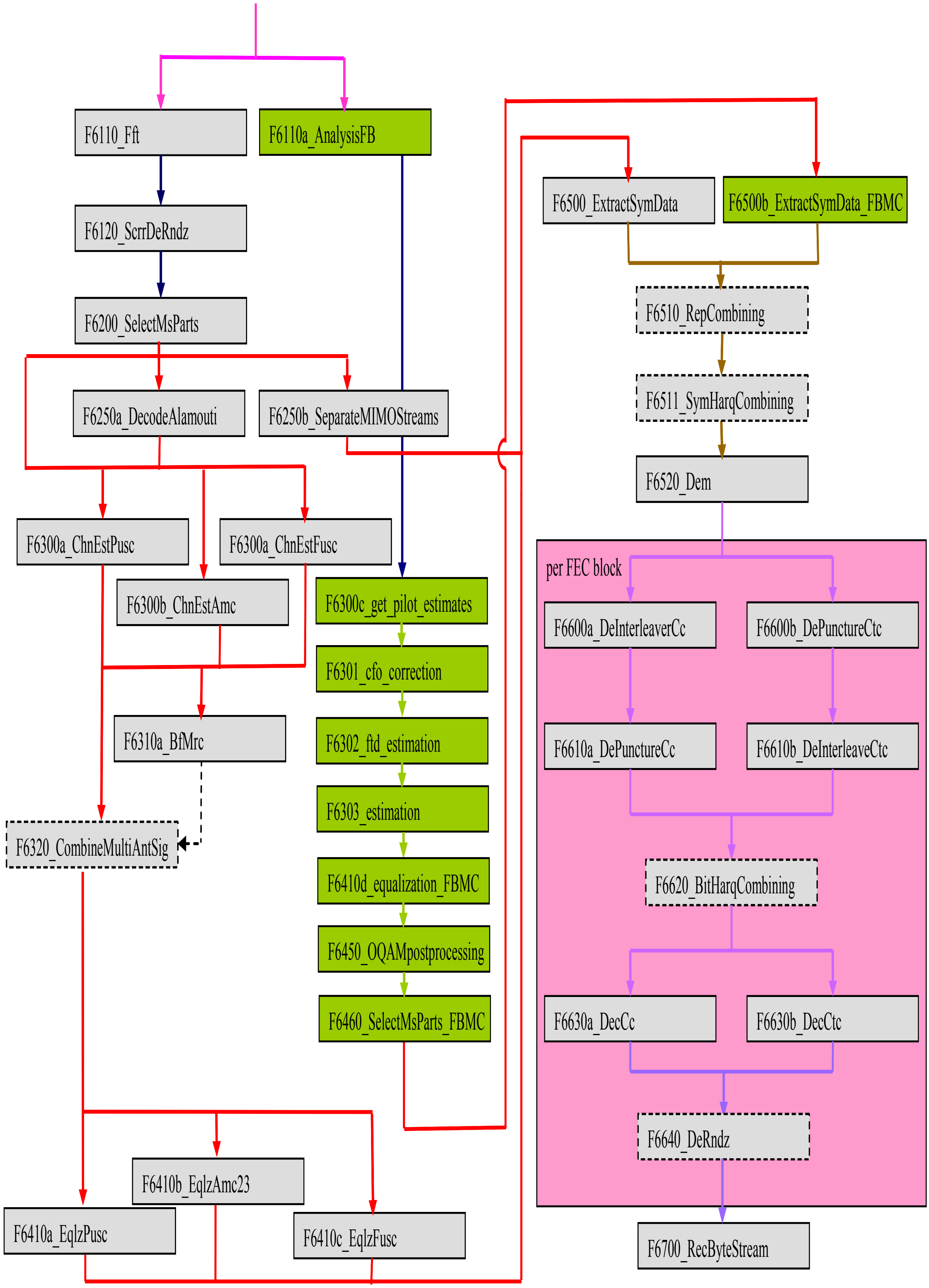

Figure 23: MS specific processing (downlink)

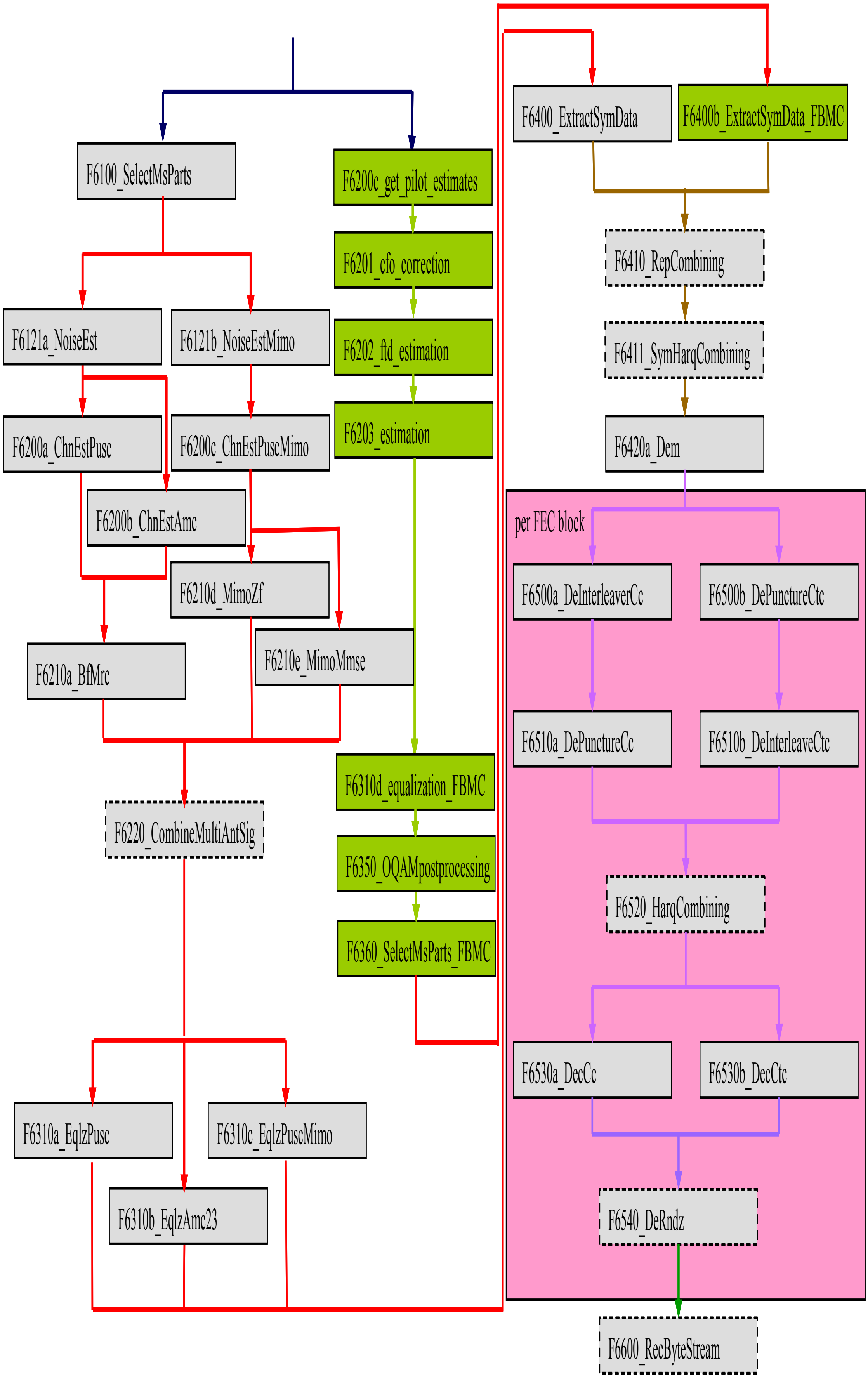

Figure 24: MS specific processing (uplink)

Again the green colour indicates new functions needed for implementing the FBMC mode.

The first step is the transformation of the disturbed and noisy time signal into frequency domain. In WiMAX classical OFDM is performed, thus a FFT does the job ('F6110_Fft' and 'F5110_Fft'). Additionally the cyclic prefix gets removed. After the subcarrier derandomization the data is picked out of the frame and arranged into the slot structure ('F6200_SelectMsParts' and 'F6100_SelectMsParts'). Additionally the pilots get separated for later use. If MIMO is active the corresponding decoder is called next, depending on the chosen mode ('F6250a_DecodeAlamouti' or 'F6250b_SeparateMIMOStreams' in downlink and 'F6210d_MimoZF' or 'F6210e_MimoMmse' in uplink). More on the topic of MIMO follows in a later deliverable. To be able to equalize the channel its coefficients have to be known, therefore channel estimation and interpolation is performed ('F6300a_ChnEstPusc', 'F6300a_ChnEstFusc' or 'F6300b_ChnEstAmc' in downlink and 'F6200a_ChnEstPusc' or 'F6200b_ChnEstAmc' in uplink). A more specific description of the channel estimation methods implemented follows in a later chapter. After multi antenna combining ('F6320_CombineMultiAntSig' and 'F6220_CombineMultiAntSig') if applicable, the channel gets equalized ('F6410a_EqlzPusc', 'F6410b_EqlzAmc23' or 'F6410c_EqlzFusc' and 'F6310a_EqlzPusc' or 'F6310b_EqlzAmc23' respectively) and the slot structure dissolved ('F6500_ExtractSymData' and 'F6400_ExtractSymData'). At this point a stream of noisy but equalized symbols is available.

Equal to the frame wrapper many changes have to be realized within the receiver to include the FBMC mode, starting with the transformation from time to frequency domain. Instead of a FFT an analysis filter bank is used ('F6110a_AnalysisFB' and 'F5110a_AnalysisFB'). A detailed description can be found in [1][6][7]. After acquiring the channel coefficients at the pilot positions ('F6300c_get_pilot_estimates' and 'F6200c_get_pilot_estimates'), carrier frequency estimation and correction follows ('F6301_cfo_correction' and 'F6201_cfo_correction'). Estimation of timing deviations ('F6302_ftd_estimation' and 'F6202_ftd_estimation') and of the channel coefficients for all subcarriers ('F6303_estimation' and 'F6203_estimation') is next. Using these estimates the channel gets equalized ('F6410d_equalization_FBMC' and 'F6310d_equalization_FBMC'). As workpackages are dedicated to the estimation and equalization topic and thus dedicated deliverables are available, it is desisted from a more detailed description at this point. As later functions expect complex QAM symbols, OQAM post processing is performed ('F6450_OQAMpostprocessing' and 'F6350_OQAMpostprocessing'). Up to this point processing was zone wise. Following functions like demodulation and forward error decoding expect symbol and bit streams respectively. Therefore slot structure is reconstructed ('F6460_SelectMSParts_FBMC' and 'F6360_SelectMSParts_FBMC'). Finally the stream of symbols is gathered ('F6400b_ExtractSymData_FBMC' and 'F6400b_ExtractSymData_FBMC').

Now a common point is reached. The following functions are usable independently of the chosen mode, be it WiMAX or FBMC. First repetition combining is performed via coherent combining of the received copies, if repetition coding was used ('F6510_RepCombining' and 'F6410_RepCombining'). As well optional is the use of HARQ (hybrid automatic repeat request). If HARQ is active and symbol combining is chosen, 'F6511_SymHarqCombining' and 'F6511_SymHarqCombining' respectively are called. HARQ in WiMAX is working on the basis of complete bursts within the frame. If a connection uses HARQ, a CRC16 (cyclic redundancy check using 16 bits) overhead is added to the complete burst within a single frame. This overhead is used at the receiver to check if a burst could be received correctly. If yes a so called 'ACK' (acknowledgment) is announced to the transmitter. In the case that no 'ACK' was received by the transmitter during a given frame period, the complete burst gets retransmitted. With symbol combining chosen, the receiver averages the symbols of all received copies of a single burst.

Forward error decoding is done on the basis of soft bits. Thus a soft bit demapper translates the symbols into bits (‘F6520_Dem’ and ‘F6420a_Dem’). Like in the transmitter processing is done on the basis of FEC blocks. After depuncturing (‘F6600b_DePunctureCtc’ or ‘F6610a_DePunctureCc’ and ‘F6500b_DePunctureCtc’ or ‘F6510a_DePunctureCc’) and deinterleaving (‘F6610b_DeInterleaverCtc’ or ‘F6600a_DeInterleaverCc’ and ‘F6510b_DeInterleaverCtc’ or ‘F6500a_DeInterleaverCc’) forward error decoding is done (‘F6630b_DecCtc’ or ‘F6630a_DecCc’ and ‘F6530b_DecCtc’ or ‘F6530a_DecCc’). Optionally, if HARQ is active and soft bit combining chosen, the received bit streams are chase combined before the actual error decoding step. Finally, after derandomization (‘F6640_DeRndz’ and ‘F6540_DeRndz’), the decoded byte stream (‘F6700_RecByteStream’ and ‘F6600_RecByteStream’) is ready to be compared to the originally transmitted one.

2.5.1 Channel estimation

WiMAX is an OFDM based transmission system. In such systems the available spectral band is divided into subbands each containing a so called subcarrier. If the delay spread of the channel and with it its frequency selectivity is not too high, it is approximately non-selective within the subbands as the coherence bandwidth is higher than the width of the subbands in this case. Due to this, equalization can be done subband per subband by a simple complex multiplication. The following figures show exemplary realisations of the implemented path delay models to illustrate this:

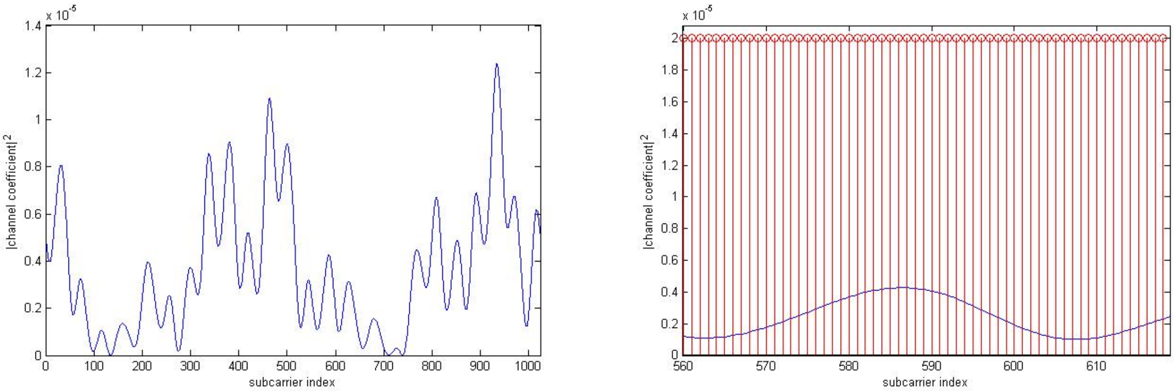

Figure 25: Exemplary channel realization in frequency domain, complete OFDM symbol and cut out (Ped B)

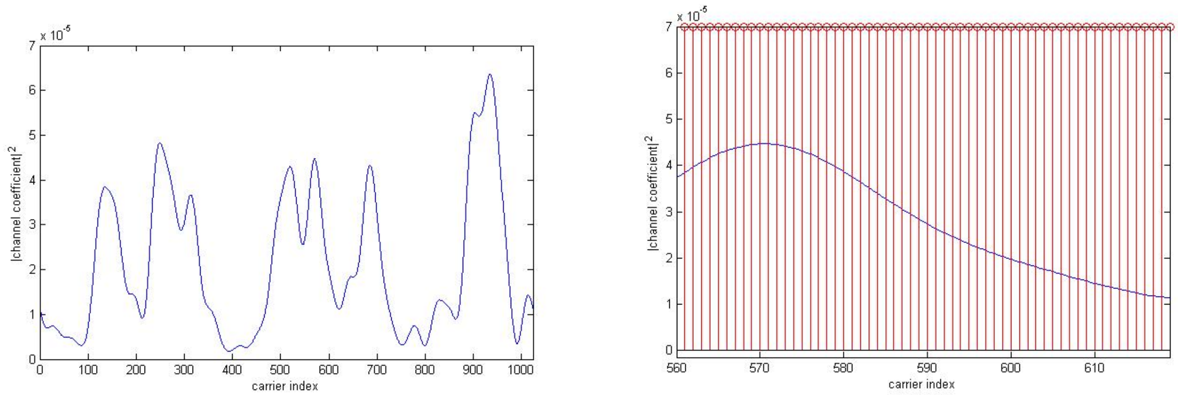

Figure 26: Exemplary channel realization in frequency domain, complete OFDM symbol and cut out (Veh A)

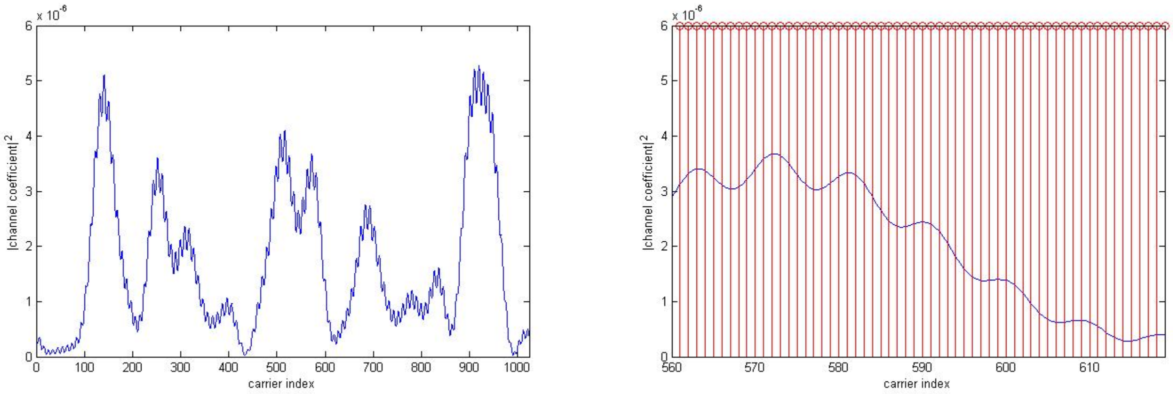

Figure 27: Exemplary channel realization in frequency domain, complete OFDM symbol and cut out (Veh A extended)

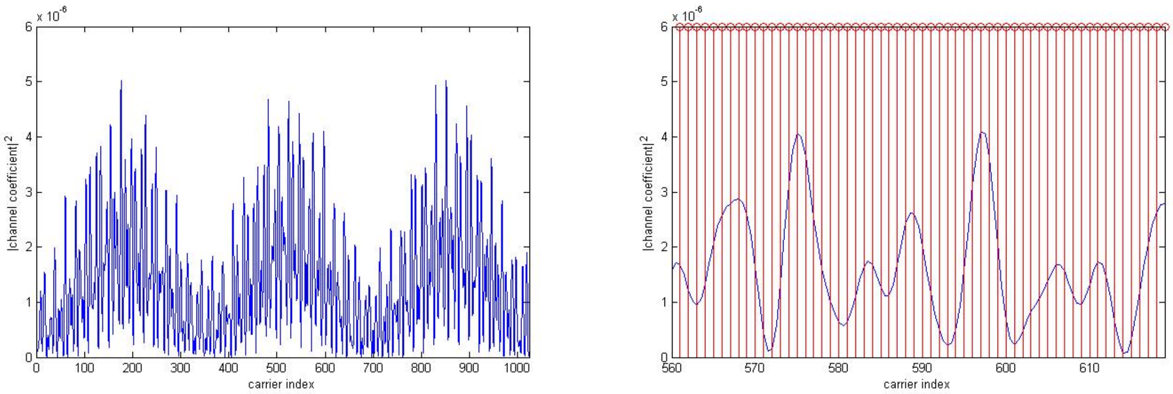

Figure 28: Exemplary channel realization in frequency domain, complete OFDM symbol and cut out (Veh B)

The figures on the left side show an exemplary realization of the respective channel over the complete range of an OFDM symbol. On the right cut outs are shown. The red lines indicate the borders between the subbands. With Ped B and Veh A the channel within a single subband obviously in fact is nearly constant. With Veh A extended and Veh B this is not the case anymore. Ped B and Veh A are describing the channel occurring in typical WiMAX scenarios. Veh A extended and Veh B are included into the simulator to be able to simulate extreme cases.

Next is a description of the channel estimators implemented for WiMAX mode. For FBMC mode descriptions can be found in [10]-[14]. Several ways to estimate these factors for all subcarriers are implemented, namely ideal estimation, pilot averaging and linear interpolation.

In the case of ideal estimation perfect channel knowledge is available at the output of the estimator. To get this knowledge the simulation is done once without noise, before the actual noisy runs. Additionally the estimator knows the originally transmitted frame in frequency domain. This way the fading coefficients can be estimated perfectly by a simple division per allocated subcarrier. Such an estimator naturally is not realizable. Nevertheless, if someone wants to know the theoretical achievable error rates, this estimator is needed.

Up to now the pilots were not needed for the estimation algorithms. This is not the case for pilot averaging and linear interpolation. There the channel estimations at the pilot positions are the basis. Thus the first step is to calculate the fading coefficients at the pilot positions. These estimations naturally are distorted by the noise. Let $y_{k,n}$ be the received value at a single pilot position within the frame:

$$y_{k,n} = x_{k,n} H_{k,n} + n_{k,n}$$

$x_{k,n}$ is the transmitted real pilot symbol, $H_{k,n}$ the complex fading coefficient and $n_{k,n}$ the complex noise sample. An estimate of $H_{k,n}$ can be determined as follows:

$$\tilde{H}_{k,n} = \frac{y_{k,n}}{x_{k,n}} = H_{k,n} + \frac{n_{k,n}}{x_{k,n}} = H_{k,n} + \tilde{n}_{k,n}$$

As the noise is a complex random variable with zero mean and variance σ_n^2 per dimension, the estimate $\tilde{H}_{k,n}$ is either a random variable with variance $\frac{\sigma_n^2}{x_{k,n}}$ per dimension. Its mean is the looked for fading coefficient $H_{k,n}$. The lower the SNR the worse gets the estimation. A way to reduce the error is to combine several estimations. A combination of j estimations leads to a reduction of the variance of the distortion by the factor j. Let's say we have 2 estimates $\tilde{H}_{k1,n}$ and $\tilde{H}_{k2,n}$ available. Averaging leads to:

$$\tilde{H}_{k,n} = \frac{1}{2}\left(\tilde{H}_{k1,n} + \tilde{H}_{k2,n}\right) = \frac{1}{2}\left(H_{k1,n} + \tilde{n}_{k1,n} + H_{k2,n} + \tilde{n}_{k2,n}\right)$$

The variance of the single estimates is $\frac{\sigma_n^2}{x_{k,n}}$, that of the averaged one:

$$Var\left[\tilde{H}_{k,n}\right] = E\left[\left(\frac{1}{2}\left(\tilde{n}_{k1,n} + \tilde{n}_{k2,n}\right)\right)^2\right] = \frac{1}{4}\left(E\left[\tilde{n}_{k1,n}^2\right] + E\left[\tilde{n}_{k2,n}^2\right] + E\left[\tilde{n}_{k1,n}\tilde{n}_{k2,n}\right]\right)$$

With independent noise

$$E\left[\tilde{n}_{k1,n}\tilde{n}_{k2,n}\right] = 0$$

holds. We then get:

$$Var\left[\tilde{H}_{k,n}\right] = \frac{1}{2}\sigma_n^2$$

This principle is taken advantage of when pilot averaging is used. Several adjacent pilots are averaged. This average is used as estimate for the surrounding carriers. Unfortunately, as the radio channel is both time variant and frequency selective, this averaging cannot be done infinitely, as the higher the distance of the pilots in time or in frequency are, the more their actual fading coefficient differ. Then the actual channel is not estimated accurately anymore. In downlink PUSC averaging is done using all pilots within a single cluster $(j = 4)$, in uplink PUSC within a tile $(j = 4)$ and in AMC zones within a slot $(j = 6)$. The calculated average is taken to approximate the coefficient of all carriers within that cluster/tile/slot. The dimensions in number of subcarriers are 2x14 (cluster), 3x4 (tile) and 3x18 (slot in AMC).

Finally linear interpolation is available to calculate the channel coefficients between the pilots. The same regions as with averaging are used in this case. Two different variants are implemented for downlink PUSC and AMC, linear regression and linear interpolation between the pilots. The regions to be interpolated always span more than a single OFDM symbol. With the interpolators described here, the assumption of a non changing channel on consecutive OFDM symbol is taken. Up to very high velocities this assumption holds. Therefore all pilots located within the interpolation region are jointly used for the interpolation process. After interpolation the calculated coefficients then are used for all OFDM symbols within the region. Figure 29 will clarify this:

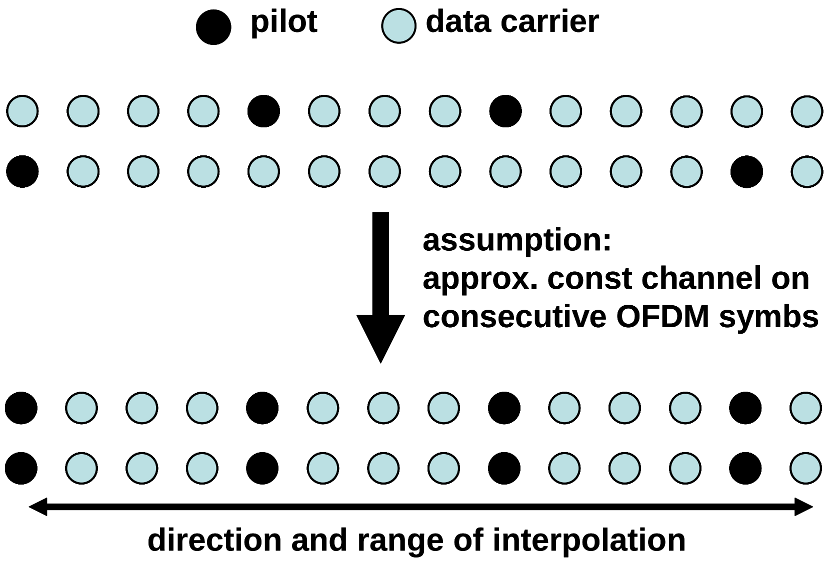

Figure 29: Linear interpolation in downlink PUSC (principle)

For AMC the procedure is similar.
To further clarify the different schemes Figure 30 shows an exemplary case:

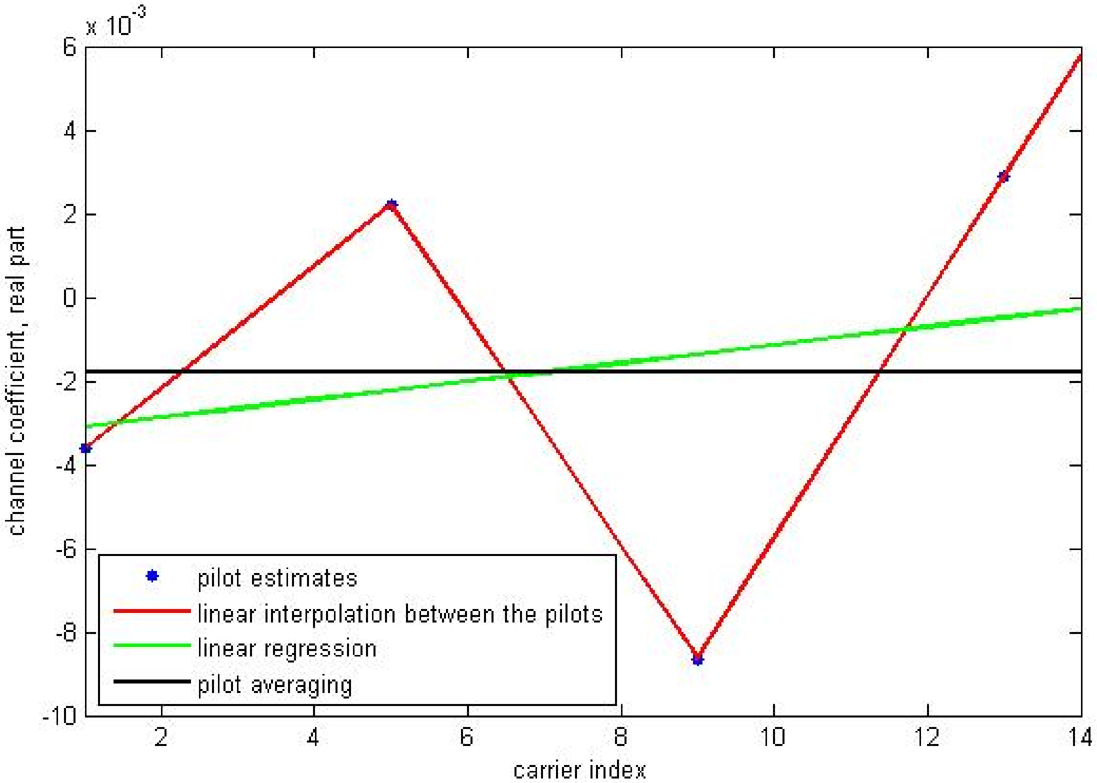

Figure 30: Linear interpolation in downlink PUSC (example)

With uplink PUSC interpolation is realized differently. 4 channel estimates are available per tile. Their relative positions within the frame shape a rectangle. Thus interpolation in this case is done in two steps, first in time then in frequency direction. Figure 31 depicts this:

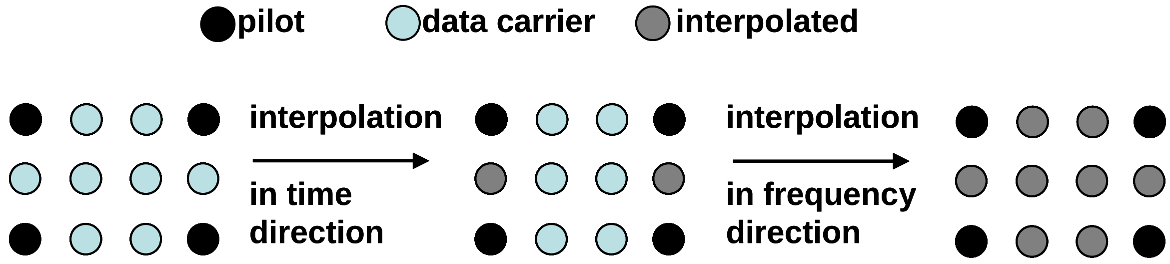

Figure 31: Linear interpolation in uplink PUSC (principle)

2.6 Evaluation

The actual purpose of the simulator is to generate performance measures, with which different settings can be compared. Data transmission in WiMAX is packet based. The bits to be transmitted are encapsulated into so called MAC PDUs (protocol data unit). Typically 64 bytes compose such a data packet. Therefore the most important measure is the so called packet error rate (PER), the ratio between the number of packets received erroneous and the quantity of originally transmitted ones. The PER is measured after the complete receiver chain including all error correction capabilities. Another measure generated by the simulator is the bit error rate (BER). Here the number of erroneous received bits relative to the number of transmitted bits is calculated. Again this measure is determined on the basis of the error corrected data stream. Additionally the so called raw bit error rate (rawBER) is determined. This measure does not take error correction into account. The bits checked are the ones fed out of the demodulator.

If HARQ is active additional statistics are obtained. These are the nack-rates, the residual nack-rate and the link-level throughput. The different HARQ retransmissions are distinguished with respect to the nack-rates. If a packet is transmitted up to 4 times, 4 nack-rates are calculated. E.g. the nack-rate for the third transmissions is the ratio between the number of nacks occurred after that third transmissions to the number of third transmissions occurred. The residual nack-rate connects the number of nacks unsolved (as the maximal number of transmissions is reached) to the total number of bursts transmitted. Finally the link-level throughput is the number of correctly transmitted bursts relative to the number of frames needed therefore.

3 Usage of the simulator

The configuration of the downlink simulator is done via 6 parameter files. The uplink simulator is configured via 7 parameter files. They all are assigned to a specific part of the chain. PAR0_SIM_... contains the basic simulation parameters such as SNR-range and number of frames to be simulated. With PAR1_Cell_... the frame specific characteristics can be decided. The structure of the frame (e.g. FFT length, number of OFDM symbols per frame …) is defined here. If the frame shall consist of several zones, their specification is done here, too.
Up to here the configuration of the uplink and the downlink simulators are comparable. This is changing now due to the slightly different architectures (broadcast channel in downlink ←→ multiple access channel in uplink). The next parameter file for the downlink simulator is PAR2_CELL_CHN_.... Here Channel specific parameters can be provided. In PAR4_MS_DATA_... data specific adjustments can be done. The location within the frame and the used modulation and encoding scheme of the data are two examples of these adjustments. It can be specified, if MIMO is used for this transmission. Finally the usage of HARQ is preset in here. The analogue attributes of the receiver are defined in PAR5_MS_ANA_.... The receiver algorithms are chosen in PAR6_MS_DIGI_.... They are for example MIMO detection algorithm and channel estimation method.
In uplink mobile specific settings are done in PAR2_MS_..., PAR3_MS_DATA_... and PAR4_MS_CHN_.... Channel and data related adjustments are done here. Receiver algorithms used and the analogue behaviour of the basestaion are regulated in PAR5_BS_ANA_... and PAR6_BS_DIGI_....

3.1 Running a simulation

If someone wants to simulate a given setting, he just can provide it using the parameter files. The simulator then is started using the script 'A_start.m'. Two different possibilities of calling this script do exist:

- A_start(ResultsDirString): Simulation results of a single run are gathered within a results directory. The string ResultsDirString can be used to differentiate different simulation runs. It is appended to the end of the directory name. Before the actual simulation a 'clear all' is performed.
- A_start(ResultsDirString, 0): Again the directory name is appended as above. The difference to the upper way of starting the simulation is that the 'clear all' is bypassed. This way breakpoints set for debugging purposes are not erased.

If someone wants to run a simulation to gather performance measures, mode one has to be selected. If 'clear all' is not executed persistent variables created in former runs may falsify the actual run. Mode two is the so called debugging mode, as here break points are not cleared.

If several different settings shall be simulated in a so called simulation campaign (such as e.g. different detection algorithms), then the overruling principle is to be used. There the common parameters are again provided by the parameter files. Additionally a so called overruling

('A_Overruling.m') file is to be made available. Here the differing parameters are provided to the simulator. Using the 'switch' command different cases can be defined. Each case has to have a unique numerical label. Then the start of the campaign can be done by using the script 'A_Runscript.m'. This script basically performs a for-loop over chosen cases, which are defined within the overruling file. The user has to include into the 'for' statement the numbers of those cases he wants to be simulated. These numbers agree with the numbers within the overruling file. For illustration a small example is given. First the essential part of the overruling file is shown:

```
switch (GL.SimNb)

    case{101}%Ped B 3 km/h
        C1.Cell{1}.CHN.MS{1}.PreDefModel = 'PEDB';
        C1.Cell{1}.CHN.MS{1}.Fad.Speed_kmh = 3;

    case{102}%Veh A 60 km/h
        C1.Cell{1}.CHN.MS{1}.PreDefModel = 'VEHA';
        C1.Cell{1}.CHN.MS{1}.Fad.Speed_kmh = 60;

    case{103}%Veh A 120 km/h
        C1.Cell{1}.CHN.MS{1}.PreDefModel = 'VEHA';
        C1.Cell{1}.CHN.MS{1}.Fad.Speed_kmh = 120;

end
```

Three cases are defined here. They differ in the chosen channel model. Case '101' uses Ped B with user speed 3 km/h, case '102' and '103' Veh A with 60 km/h and 120 km/h respectively. All other parameter settings are common and provided with the help of the parameter files. The relevant line of the runscript then is (if all three cases are to be simulated):

```
for iSim = [ 101 102 103 ]
```

or (if not all cases are to be simulated):

```
for iSim = [ 101 103 ]
```

Once the overruling file and the runscript are created, the simulation campaign is evoked via the call of 'A_Runscript'.

As already mentioned the simulator creates per simulation run a directory containing the results. 4 files are generated:

- '_workspace.mat': A snapshot of the workspace as it was at the end of the simulation.
- 'MSxx.txt', 'MSxx_BSyy.txt': A text file containing the results for mobile #xx using BS configuration #yy (the latter just in uplink). This file can be used to import the results into Excel.
- 'parameters.txt': A text file containing all parameters defined for this run. This way someone easily can reconstruct the simulation.
- 'simLog.txt': A text file containing all command line outputs during the simulation run.

Both the snapshot of the workspace and the text file containing the results, include both the adjusted SNR values and measured ones. Due to the fast fading behavior of the mobile channel, the actual

SNR value is a random variable. Therefore the predefined value may not be reached exactly. Hence during the simulation the effective SNR is measured. Thus, if performance curves are to be created, the measured SNR is to be used.

3.2 Parameter space of the simulator

As already stated earlier it was paid attention to keep the number of needed parameters as low as possible. Just general settings should be done via this parameter files to reduce complexity. Deducible ones are determined before the actual simulation starts. The following tables include the adjustable parameter space. Within these tables the first column contains the name of the specific parameter. If the selectable values are not obvious or somehow restricted, the second column carries examples and/or ranges. Remarks are located in the third column. Sometimes several parameters are grouped. That's the case if they share their purpose.

3.2.1 Downlink

3.2.1.1PAR0_SIM_... general simulation parameters

PAR0_SIM_... contains general simulation parameters such as termination conditions, SNRs to be simulated and general HARQ parameters.

Table 7: Simulation specific parameters (downlink)

Parameter name	Possible values	Remarks
C1.TermCond.NbFrms C1.TermCond.NbPackets C1.TermCond.NbPacketsErr C1.TermCond.NbBitErr	single value 1, 2, 3, …, Inf	Simulation termination conditions. If Inf is chosen, the respective condition is deactivated.
C1.TermCond.MinNbPackets	single value 0, 1, 2, …, Inf	Minimum number of packets to be simulated. Overrules the termination condition.
C1.FrmOfsStart	single value 0, 1, 2, …, Inf	Number of frames to be ignored by the evaluation functions at the beginning (e.g. to reach a steady state).
C1.InitRndSeed.UniDistributed C1.InitRndSeed.NormDistributed	single value 0, 1, 2, …, 2^{32-1}	Seeds for the random generators. Thereby results are reproducible.
C1.MsId	array of values integer numbers (>0)	Ids of the mobiles defined to be simulated.
C1.RxMsId	array of values integer numbers (>0)	Ids of the mobiles which actually decode their bursts. C1.RxMsId $\subseteq$ C1.MsId

C1.Snr_dB	array of values real numbers	SNRs to be simulated
C1.CellId	array of values integer numbers (>0)	Ids of the cells to be simulated. For each Id a frame is built for its own.
C1.Result.MinBitError C1.Result.MinPacketError C1.Result.MinHarqError	single value real number	Error rates smaller then this value are rounded up to this value (for drawing purposes).
C1.Harq.DelayInSubFrms	single value integer number (>0)	Number of frames the HARQ retransmission is delayed.
C1.Harq.MaxnbTrnsm	single value integer number (>0)	Maximum number of retransmissions for HARQ.
C1.TxAntSpace	array of values real numbers	Number of values = Number of antennas values = distance of antennas normalized to the wavelength with respect to the first one

Some of the defined mobiles may be just needed as interferers and thus their data is not needed to be detected. Then they are defined in C1.MsId but not in C1.RxMsId (e.g. C1.MsId = [1 2 3] and C1.RxMsId = [1 3] means, that three mobiles are present and get a burst allocated. However, just mobile #1 and #3 are evaluated. The data of mobile #2 just is transmitted, but not detected. Mobile #2 just is active as interferer.)

3.2.1.2PAR1_CELL_... cell specific parameters

PAR1_CELL_... contains cell specific parameters such as frame parameters (FFT size, length of CP, …) and zone parameters (permutation mode, length of zone, …). Additionally rectangular allocations can be defined here (e.g. to reserve parts of the frame for the FCH and the Maps). One file per defined cell (C1.CellId) must be provided to the simulator. These files must have the suffix xx with xx being the cell id (e.g. if C1.CellId = [1 13], two files have to be provided, one with the suffix '01' and one with '13'). The parameters within the file have the prefix C1_CELL…. During the simulation runs they are mapped to the structure C1.Cell{iC}… with iC being the cell id. The parameters starting with C1_CELL.Zone{iZ}… are defining the zones of the frame. They are mapped to the structure C1.Cell{iC}.Zone{iZ}…. iZ is the zone index starting with 1. Thus those parameters have to appear once per zone.

Table 8: Cell specific parameters (downlink)

Parameter name	Possible values	Remarks
C1_CELL.Frm.IdxPreamble	single number 2	Index of the preamble. Up to now just one preamble implemented.
C1_CELL.Frm.CycPrefix	single number	Relative length of the CP.

	any fraction	WiMAX compliant value: 1/8
C1_CELL.Frm.Fft	single number 1024	Size of the FFT, number of subcarriers (= M).
C1_CELL.Frm.OsymLen	single number 1, 2, 3, …	Number of OFDM symbols the DL subframe consists of. WiMAX compliant values: 26, 27, …, 35
C1_CELL.Frm.CrrFreq	single number arbitrary frequency	Center frequency. WiMAX compliant values [GHz]: 2.3-2.4, 2.496-2.69, 3.3-3.8
C1_CELL.Frm.ChnBandwidth	single number 10e6	Channel bandwidth [Hz].
C1_CELL.Frm.FCH.Diuc	single number 1	Choice of the modulation and coding scheme the DL frame prefix is to use. WiMAX: Just QPSK, CC 1/2 possible.
C1_CELL.Frm.DLMap. ActivateRndz	boolean value 'true', 'false'	Flag to activate/deactivate the data randomizer (DL map).
C1_CELL.Frm.DLMap.RepFac	single number 1, 2, 4, 6	Repetition coding used within the DL map.
C1_CELL.Frm.DLMap.Diuc	single number 1, 8	Choice of the modulation and coding scheme the DL Map is to use. WiMAX: Always QPSK, CC 1/2 and CTC 1/2 possible.
C1_CELL.Frm.DLMap.NbSlots	single number 1, 2, 3, …, number of available slots	Length of the DL map (number of slots).
C1_CELL.Frm.FBMC	boolean value 'true', 'false'	Flag to activate/deactivate FBMC mode.
C1_CELL.Frm.proto_opt	single number 1, 2	1: CNAM NPR 2: TUT PR
C1_CELL.Frm. overlapping_factor	single number 3 or 4, if proto_opt = 1 1, 2, 3, 4 or 5, if proto_opt = 2	= K
C1_CELL.Frm.p_boost	single number real value	Pilot boosting

C1_CELL.Frm.p_method	'AuxPilots', 'PoPilots'	Choice of the pilot method.
C1_CELL.Frm.excess_ratio	real number [0…1]	Ratio of the pre-/post-tail lengths before and after burst truncation
C1_CELL.Frm.rcos_ratio	real number [0…1]	Length of the raised cosine roll off.
C1_CELL.Zone{iZ}. MIMOUsed	boolean value 'true', 'false'	Flag to define if the zone #iZ is a MIMO zone. WiMAX: Zone 1 must not be a MIMO zone.
C1_CELL.Zone{iZ}. UseALLSC	single number 1	Flag to define if all subchannels shall be used. Up to now no segmentation supported.
C1_CELL.Zone{iZ}. PRBS_ID	single number 0, 1, 2	Zone specific parameter.
C1_CELL.Zone{iZ}. PermuMode	single string 'PUSC', 'AMC23', 'FUSC'	Mode of permutation.
C1_CELL.Zone{iZ}. OfsOsym	single number 1, 2, 3, …, C1_CELL.Frm.OsymLen - 1	Offset of the first OFDM symbol of the zone. Counted from the beginning of the frame. Preamble included.
C1_CELL.Zone{iZ}. OsymLen	single number 1, 2, 3, …, C1_CELL.Frm.OsymLen – C1_CELL.Zone{iZ}.OfsOsym	Length of the zone in number of OFDM symbols.
C1_CELL.Zone{iZ}. SegNr	single number 0	Number of the segment to be used. Up to now no segmentation supported.
C1_CELL.Zone{iZ}. SegSchn	array of numbers []	Which subchannels to be used for the segment. Up to now no segmentation supported.
C1_Cell.Zone{iZ}. DL_Permbase	single number 0, 1, 2, …, 31	Permutation base.
C1_CELL.Zone{iZ}. TXdivType	single number 0	Transmit diversity type used in the zone (iZ>1). 0 → STBC (FHDC not

		WiMAX compliant)
C1_CELL.Zone{iZ}. MIMO.MatrixInd	single number 0, 1	MIMO matrix indicator (iZ>1). 0 → Matrix A (STBC) 1 → Matrix B (SM)
C1_CELL.Zone{iZ}. RectAlloc{iR}.OfsOsym	single number 0, 1, 2, …, C1_CELL.Zone{iZ}.OsymLen - 1	Offset of the first OFDM symbol of the rectangular allocation. Counted from the beginning of the zone.
C1_CELL.Zone{iZ}. RectAlloc{iR}.OfsSchn	single number 0, 1, 2, …, number of subchannels available - 1	Offset of the beginning of the rectangular allocation in number of subchannels.
C1_CELL.Zone{iZ}. RectAlloc{iR}.TslotLen	single number 1, 2, 3, …, number of time slots available	Length of the rectangular allocation in number of time slots. (AMC:1 time slot = 3 OFDM symbols, PUSC:1 time slot = 2 OFDM symbols)
C1_CELL.Zone{iZ}. RectAlloc{iR}.SchnLen	single number 1, 2, 3, …, number of subchannels available	Length of the rectangular allocation in number of subchannels.

3.2.1.3PAR2_CELL_CHN_... channel parameters

PAR2_CELL_CHN_... contains channel specific parameters such as the Ids of the mobiles the frame shall be transmitted to and the channel model to be used. Again one file per defined cell (C1.CellId) must be provided to the simulator. The parameters within the file have the prefix C1_CELL_CHN…. During the simulation runs they are mapped to the structures C1.Cell{iC}.CHN… with iC being the cell id. The parameters starting with C1_CELL_CHN.MS{iM}… are defining the channel for the mobile with index iM. They have to appear once for all mobiles defined in C1_CELL_CHN.TxToMs. The indices used here must agree with those defined in C1.RxMsId.

Table 9: Channel specific parameters (downlink)

Parameter name	Possible values	Remarks
C1_CELL_CHN.IdxTxAnt	array of numbers depending on the defined antennas in C1.AntSpace	This parameter picks the antennas used for transmission of the frame out of the defined antenna space. [] → use all antennas defined in

		C1.AntSpace, [1 3] → use the first and third antenna…
C1_CELL_CHN.TxToMs	array of numbers depending on the defined mobiles in C1.RxMsId	This parameter defines to which mobile the frame shall be transmitted. For all here defined mobiles the following parameters have to be assigned. C1_CELL_CHN.TxToMs $\subseteq$ C1.RxMsId
C1_CELL_CHN.MS{iM}. ShortTermSNR	boolean value 'true', 'false'	Flag to determine if short term SNR shall be used. (SNR adjusted on frame basis → ideal power control) Should typically set to 'false'.
C1_CELL_CHN.MS{iM}. PreDefModel	single string 'AWGN', 'PEDB', 'VEHA', 'VEHAE', 'VEHB'	Cannel model to be used.
C1_CELL_CHN.MS{iM}. RxAntSpace	array of values real numbers	Number of values = Number of antennas values = distance of antennas normalized to the wavelength with respect to the first one
C1_CELL_CHN.Fad. SpecificChnPar	single number depending on the precalculated sequences	To define which set of precalculated fading sequence is to be used.
C1_CELL_CHN.MS{iM}. Fad.Speed_kmh	single number	Speed of the mobile [km/h]
C1_CELL_CHN.MS{iM}. Fad.MultiAntModel	single string 'CorrMatrix'	Multi-antenna model to be used.
C1_CELL_CHN.MS{iM}.Fad. CorrM.Type	single string 'Parameters', 'Uncorrelated', 'MaxCorrelated'	Type of correlation matrix to be used.
C1_CELL_CHN.MS{iM}.Fad. CorrM.CalcOption	single number 0, 1	Option for the calculation of the correlation matrix. 0 → use theoretical integral 1 → use approximation
C1_CELL_CHN.MS{iM}.Fad. CorrM.BsAod	array of numbers 0°…360°	Angel of departure at the basestation for each path.

C1_CELL_CHN.MS{iM}.Fad.CorrM.MsAoa	array of numbers 0°…360°	Angel of arrival at the mobile for each path.
C1_CELL_CHN.MS{iM}.Fad.CorrM.BsAngSpread	single number 0°…360°	Angular spread at the basestation.
C1_CELL_CHN.MS{iM}.Fad.CorrM.MsAngSpread	single number 0°…360°	Angular spread at the mobile.
C1_CELL_CHN.MS{iM}.FBMC.f_off	single number or 'r' [-0.5…0.5]	Frequency offset, in parts of a subcarrier bandwidth. r → random value
C1_CELL_CHN.MS{iM}.FBMC.ph_off	single number or 'r' [-π…π]	Phase offset in radians. r → random value
C1_CELL_CHN.MS{iM}.FBMC.fractional_time_delay	single number real value	Fractional time delay. Normalized with M/2.

3.2.1.4 PAR4_MS_DATA_... parameters concerning the data to be transmitted

PAR4_MS_DATA_... contains the parameters concerning the data transmission such as cell and zone index the burst shall be transmitted in, burst size and if FCH and DL map shall be ignored, detected or decoded. One file per defined mobile (C1.MsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_MS_DATA…. During the simulation runs they are mapped to the structures C1.MS{iM}.DATA… with iM being the mobile id.

Table 10: Data specific parameters (downlink)

Parameter name	Possible values	Remarks
C1_MS_DATA.IdxCell	single number depending on the defined cells in C1.CellId	In which cell the burst, defined in this file, shall be transmitted in.
C1_MS_DATA.IdxZone	single number depending on the defined zones in the chosen cell	Index of the zone the burst is located in.
C1_MS_DATA.CID	single number	Connection identifier.
C1_MS_DATA.Type	single number 1, 2, 3	Type of data to be transmitted (bytes). 1 → random data, 2 → just '1's 3 → 0, 1, 2, …, 15, 0, 1, …
C1_MS_DATA.Diuc	single number	This parameter chooses the modulation and coding scheme for

	[1, 2, …, 15]	the actual burst. The actual mapping follows after this table.
C1_MS_DATA.OfsTslots	single number 0, 1, … number of time slots within the current zone	Offset of the burst allocation in time slots. Counted from the beginning of the zone. (AMC:1 time slot = 3 OFDM symbs PUSC:1 time slot = 2 OFDM symbs)
C1_MS_DATA.NbTslots	single number 0, 1, … number of time slots within the current zone - C1_MS_DATA.OfsTslots	Length of the burst allocation in number of time slots. (AMC:1 time slot = 3 OFDM symbs PUSC:1 time slot = 2 OFDM symbs)
C1_MS_DATA.OfsSchn	single number 0, 1, … number of subchannels within the current zone	Offset of the burst allocation in number of subchannels.
C1_MS_DATA.NbSchn	single number 0, 1, … number of subchannels within the current zone - C1_MS_DATA.OfsSchn	Length of the burst allocation in number of subchannels.
C1_MS_DATA.NbSlots	single number 0, 1, … number of slots within the current zone	Number of slots allocated for the burst. = C1_MS_DATA.NbTslots * C1_MS_DATA.NbSchn
C1_MS_DATA.RepFac	single number 1, 2, 4, 6	Number of repetitions transmitted within the burst.
C1_MS_DATA.ActivateRndz	boolean value ‘true’, ‘false’	Flag to activate/deactivate the data randomizer.
C1_MS_DATA.Boosting	single number 0	Power boosting of the burst. 0 → 0 dB. WiMAX: no burst boosting in DL
C1_MS_DATA.PacketSize	single number or [] 0, 1, … depending on the size of the burst	Size of the packets in number of Bytes, the calculation of the PER is based on. [] → packet size = number of bytes within the burst
C1_MS_DATA. InfoBurstSize_Byte	single number or [] 0, 1, … depending on the	Number of info bytes to be transmitted within the burst.

	size of the burst	[] → number of bytes fitting into the burst.
C1_MS_DATA. FCHandMAPdecoding	single number 0, 1, 2	0 → ignore FCH and DL map 1→ detect FCH and DL map and calculate error rates 2 → detect and decode FCH and DL map, use the parameters decoded this way.
C1_MS_DATA.Harq.Type	single number 0, 1, 2	0 → no HARQ 1 → HARQ with softbit combining 2 → HARQ with symbol combining
C1_MS_DATA.Mimo.Used	boolean value 'true', 'false'	Flag to indicate if MIMO is used for the transmission of the data burst.
C1_MS_DATA. Mimo.MatrixInd	single number 0, 1	MIMO matrix indicator. 0 → Matrix A (STBC) 1 → Matrix B (SM)

Table 11: Available modulation and coding schemes (downlink)

	modulation scheme	coding scheme
1	QPSK	CC 1/2
2	QPSK	CC 3/4
3	16 QAM	CC 1/2
4	16 QAM	CC 3/4
5	64 QAM	CC 1/2
6	64 QAM	CC 2/3
7	64 QAM	CC 3/4
8	QPSK	CTC 1/2
9	QPSK	CTC 3/4
10	16 QAM	CTC 1/2
11	16 QAM	CTC 3/4
12	64 QAM	CTC 1/2
13	64 QAM	CTC 2/3
14	64 QAM	CTC 3/4
15	64 QAM	CTC 5/6

3.2.1.5 PAR5_MS_ANA_... parameters concerning the analogue part of the mobile

PAR5_MS_ANA_... contains the parameters concerning the analogue front end of the mobile (noise temperature and noise figure of the receive chain). One file per defined receiving mobile (C1.RxMsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_MS_ANA…. During the simulation runs they are mapped to the structures C1.MS{iM}.ANA… with iM being the mobile id.

Table 12: Parameters concerning the analogue front end of the mobile (downlink)

Parameter name	Possible values	Remarks
C1_MS_ANA.Temp	single number	Noise temperature.
C1_MS_ANA.NoiseFig_dB	single number	Noise figure.

3.2.1.6 PAR6_MS_DIGI_... parameters concerning the digital part of the mobile

PAR6_MS_DIGI_... contains the parameters concerning the digital part of the mobile such as channel estimator and number of iterations to be done while turbo decoding. One file per defined receiving mobile (C1.RxMsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_MS_DIGI…. During the simulation runs they are mapped to the structures C1.MS{iM}.DIGI… with iM being the mobile id.

Table 13: Parameters concerning the digital signal processing within the mobile (downlink)

Parameter name	Possible values	Remarks
C1_MS_DIGI. MIMOdecodeType	single number 0, 10, 40 11, 41, 12, 42	0 → Alamouti decoding with equal channel coeffs (AWGN, STBC) 10 → Alamouti decoding with pilot averaging clusterwise (STBC) 40 → Alamouti decoding with linear interpolation (STBC) 11 → ZF decoding with pilot averaging clusterwise (SM) 41 → ZF with linear interpolation (SM) 12 → MMSE decoding with pilot averaging clusterwise (SM) 42 → MMSE with linear interpolation (SM)
C1_MS_DIGI. ChnEstType	single number 2, 10, 40, 41	Estimation of the channel coefficients. 2 → ideal channel estimation (known transmit signal, no noise)

		10 → pilot averaging 40 → linear interpolation between the pilots 41 → linear regression
C1_MS_DIGI.FftType	single number 1, 11	1 → ideal FFT 11 → ideal FFT with quantisation of the output
C1_MS_DIGI.DemType	single number 1	1 → ideal demodulation with amplitude weights of the generated softbits
C1_MS_DIGI. CcDecType	single number 1	1 → ideal unquantised decoding
C1_MS_DIGI.BfMethod	single string 'MRC'	MRC → maximum ratio combining
C1_MS_DIGI.FBMC. CFO_correction	single string 'PCI', 'none', 'Estimation'	PCI → perfect channel information none → no estimation, no correction Estimation → offsets are estimated with the help of the pilots
C1_MS_DIGI.FBMC. FTC_estimation	single string 'PCI', 'Estimation'	PCI → perfect channel information Estimation → fractional time delay estimated, no correction up to now
C1_MS_DIGI.FBMC. estimation_method	single string 'PCI', 'Estimation'	PCI → perfect channel information Estimation → channel estimation with the help of the pilots
C1_MS_DIGI.FBMC. eq_case	single number 1, 2, 3	Number of taps the equalizer is using.
C1_MS_DIGI.FBMC. eq_criterion	single string 'ZF', 'MSE'	ZF → zero forcing MSE → mean squared error
C1_MS_DIGI.CTC. ScalBits	single number 1, 2, …	resolution for CTC decoding (recommended: 6 for QPSK and 16QAM 8 for 64 QAM)
C1_MS_DIGI.CTC. NbIteration	single number 1, 2, …	Number of iterations used for turbo decoding. (recommended: 4)

3.2.2 Uplink

3.2.2.1PAR0_SIM_... general simulation parameters

PAR0_SIM_... contains general simulation parameters such as termination conditions, SNRs to be simulated and general HARQ parameters.

Table 14: Simulation specific parameters (uplink)

Parameter name	Possible values	Remarks
C1.TermCond.NbFrms C1.TermCond.NbPackets C1.TermCond.NbPacketsErr C1.TermCond.NbBitErr	single value 1, 2, 3, …, Inf	Simulation termination conditions. If Inf is chosen, the respective condition is deactivated.
C1.TermCond.MinNbPackets	single value 0, 1, 2, …, Inf	Minimum number of packets to be simulated. Overrules the termination condition.
C1.FrmOfsStart	single value 0, 1, 2, …, Inf	Number of frames to be ignored by the evaluation functions at the beginning (e.g. to reach a steady state).
C1.InitRndSeed.UniDistributed C1.InitRndSeed.NormDistributed	single value 0, 1, 2, …, 2^{32-1}	Seeds for the random generators. Thereby results are reproducible.
C1.MsId	array of values integer numbers (>0)	Ids of the mobiles defined to be simulated.
C1.BsId	array of values integer numbers (>0)	Ids of different basestation configs.
C1.Snr_dB	array of values real numbers	SNRs to be simulated
C1.CellId	array of values integer numbers (>0)	Ids of the cells to be simulated. For each Id a frame is built for its own.
C1.Result.MinBitError C1.Result.MinPacketError C1.Result.MinHarqError	single value real number	Error rates smaller then this value are rounded up to this value (for drawing purposes).
C1.Harq.DelayInSubFrms	single value integer number (>0)	Number of frames the HARQ retransmission is delayed.
C1.Harq.MaxnbTrnsm	single value integer number (>0)	Maximum number of retransmissions for HARQ.
C1.AntSpace	array of values real numbers	Number of values = Number of antennas values = distance of antennas normalized to the wavelength with respect to the first one

3.2.2.2 PAR1_CELL_... cell specific parameters

PAR1_CELL_... contains cell specific parameters such as frame parameters (FFT size, length of CP, …) and zone parameters (permutation mode, length of zone, …). Additionally rectangular allocations can be defined here (e.g. to reserve parts of the frame for sounding). One file per defined cell (C1.CellId) must be provided to the simulator. These files must have the suffix xx with xx being the cell id (e.g. if C1.CellId = [1 13], two files have to be provided, one with the suffix '01' and one with '13'). The parameters within the file have the prefix C1_CELL…. During the simulation runs they are mapped to the structure C1.Cell{iC}… with iC being the cell id. The parameters starting with C1_CELL.Zone{iZ}… are defining the zones of the frame. They are mapped to the structure C1.Cell{iC}.Zone{iZ}…. iZ is the zone index starting with 1. Thus those parameters have to appear once per zone.

Table 15: Cell specific parameters (uplink)

Parameter name	Possible values	Remarks
C1_CELL.Frm.IdxPreamble	single number 2	Index of the preamble. Up to now just one preamble implemented.
C1_CELL.Frm.CycPrefix	single number any fraction	Relative length of the CP. WiMAX compliant value: 1/8
C1_CELL.Frm.Fft	single number 1024	Size of the FFT, number of subcarriers (= M).
C1_CELL.Frm.OsymLen	single number 1, 2, 3, …	Number of OFDM symbols the DL subframe consists of. WiMAX compliant values: 12, 13, …, 21
C1_CELL.Frm.CrrFreq	single number arbitrary frequency	Center frequency. WiMAX compliant values [GHz]: 2.3-2.4, 2.496-2.69, 3.3-3.8
C1_CELL.Frm.ChnBandwidth	single number 10e6	Channel bandwidth [Hz].
C1_CELL.Frm.FBMC	boolean value 'true', 'false'	Flag to activate/deactivate FBMC mode.
C1_CELL.Frm.proto_opt	single number 1, 2	1: CNAM NPR 2: TUT PR
C1_CELL.Frm. overlapping_factor	single number 3 or 4, if proto_opt = 1 1, 2, 3, 4 or 5, if proto_opt = 2	= K
C1_CELL.Frm.p_boost	single number real value	Pilot boosting
C1_CELL.Frm.p_method	'AuxPilots', 'PoPilots'	Choice of the pilot method.

C1_CELL.Frm.excess_ratio	real number [0…1]	Ratio of the pre-/post-tail lengths before and after burst truncation
C1_CELL.Frm.rcos_ratio	real number [0…1]	Length of the raised cosine roll off.
C1_CELL.Zone{iZ}. PermBase	single number 0, 1, …, 31	Permutation base..
C1_CELL.Zone{iZ}. PermuMode	single string 'PUSC', 'AMC23'	Mode of permutation.
C1_CELL.Zone{iZ}. OfsOsym	single number 1, 2, 3, …, C1_CELL.Frm.OsymLen - 1	Offset of the first OFDM symbol of the zone. Counted from the beginning of the frame.
C1_CELL.Zone{iZ}. OsymLen	single number 1, 2, 3, …, C1_CELL.Frm.OsymLen – C1_CELL.Zone{iZ}.OfsOsym	Length of the zone in number of OFDM symbols.
C1_CELL.Zone{iZ}. SegSchn	array of numbers []	Which subchannels to be used for the segment. Up to now no segmentation supported.
C1_CELL.Zone{iZ}. RectAlloc{iR}.OfsOsym	single number 0, 1, 2, …, C1_CELL.Zone{iZ}.OsymLen - 1	Offset of the first OFDM symbol of the rectangular allocation. Counted from the beginning of the zone.
C1_CELL.Zone{iZ}. RectAlloc{iR}.OfsSchn	single number 0, 1, 2, …, number of subchannels available - 1	Offset of the beginning of the rectangular allocation in number of subchannels.
C1_CELL.Zone{iZ}. RectAlloc{iR}.TslotLen	single number 1, 2, 3, …, number of time slots available	Length of the rectangular allocation in number of time slots. (AMC:1 time slot = 3 OFDM symbols, PUSC:1 time slot = 3 OFDM symbols)
C1_CELL.Zone{iZ}. RectAlloc{iR}.SchnLen	single number 1, 2, 3, …, number of subchannels available	Length of the rectangular allocation in number of subchannels.
C1_CELL.Zone{iZ}.	single number	0 → rectangular allocation is

RectAlloc{iR}.Uiuc	0, 11	CQICH FastFeedback area 11 → extended UIUC exists
C1_CELL.Zone{iZ}. RectAlloc{iR}.ExtUiuc	single number 4, 8	4 → rectangular allocation is a sounding zone 8 → rectangular allocation is HARQ-ACK transmit area

3.2.2.3PAR2_MS_... MS specific parameters

The sole use of PAR2_MS_... is to define if the mobile is processed within the receiver. If not its only purpose is to cause interference. One file per defined mobile (C1.MsId) must be provided to the simulator, again with the specific suffix. The parameter within the file has the prefix C1_MS…. During the simulation runs it is mapped to the structure C1.Ms{iM}… with iM being the mobile id.

Table 16: Mobile specific parameters (uplink)

Parameter name	Possible values	Remarks
C1_MS.SkipRec	boolean value 'true', 'false'	false → mobile is skipped within the receiver true → data of the mobile is decoded

3.2.2.4PAR3_MS_DATA_... parameters concerning the data to be transmitted

PAR3_MS_DATA_... contains the parameters concerning the data transmission such as cell and zone index the burst shall be transmitted in and burst size. One file per defined mobile (C1.MsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_MS_DATA…. During the simulation runs they are mapped to the structures C1.MS{iM}.DATA… with iM being the mobile id.

Table 17: Data specific parameters (uplink)

Parameter name	Possible values	Remarks
C1_MS_DATA.IdxCell	single number depending on the defined cells in C1.CellId	In which cell the burst, defined in this file, shall be transmitted in.
C1_MS_DATA.IdxZone	single number depending on the defined zones in the chosen cell	Index of the zone the burst is located in.
C1_MS_DATA.Type	single number 1, 2, 3	Type of data to be transmitted (bytes).

		1 → random data, 2 → just '1's 3 → 0, 1, 2, …, 15, 0, 1, …
C1_MS_DATA.Uiuc	single number [1, 2, …, 14]	This parameter chooses the modulation and coding scheme for the actual burst. The actual mapping follows after this table.
C1_MS_DATA.OfsTslots	single number 0, 1, … number of time slots within the current zone	Offset of the burst allocation in time slots. Counted from the beginning of the zone. (AMC:1 time slot = 3 OFDM symbs PUSC:1 time slot = 3 OFDM symbs)
C1_MS_DATA.OfsSchn	single number 0, 1, … number of subchannels within the current zone	Offset of the burst allocation in number of subchannels.
C1_MS_DATA.NbSlots	single number 0, 1, … number of slots within the current zone	Number of slots allocated for the burst. = C1_MS_DATA.NbTslots * C1_MS_DATA.NbSchn
C1_MS_DATA.RepFac	single number 1, 2, 4, 6	Number of repetitions transmitted within the burst.
C1_MS_DATA.ActivateRndz	boolean value 'true', 'false'	Flag to activate/deactivate the data randomizer.
C1_MS_DATA.PacketSize	single number or [] 0, 1, … depending on the size of the burst	Size of the packets in number of Bytes, the calculation of the PER is based on. [] → packet size = number of bytes within the burst
C1_MS_DATA.Harq.Type	single number 0, 1, 2	0 → no HARQ 1 → HARQ with softbit combining 2 → HARQ with symbol combining
C1_MS_DATA.Harq.Scal	single number 1	1 → float weight combining (only for softbit combining)
C1_MS_DATA.Mimo.Type	single number or [] 0, 2, 4, 5	Choice of MIMO mode. 0 or [] → no MIMO 2 → SU MIMO, 2 streams 4 → MU MIMO, 2 user, 2 streams 5 → MU MIMO, 4 user, 4 streams

C1_MS_DATA.Mimo. Id	single number 1, 2, 3, 4	Index to distinguish the mobiles if MU MIMO is active.
C1_MS_DATA.Sound. IdxRectAlloc	single number or [] depending on the defined rectangular allocations	0 or [] → no sounding zone any value else → index of the sounding zone
C1_MS_DATA.Sound SymOfs	single number 0, 1, 2	In which OFDM symbol the sounding pilots are to be placed.
C1_MS_DATA.Sound. ScrrOfs	single number 0, 1, … C1_MS_DATA.Sound. ScrrSpace-1	Offset in number of subcarriers for placing the sounding pilots.
C1_MS_DATA.Sound. ScrrSpace	single number 1, 2, 3, …	Periodicity of the pilot placement within the sounding symbol. (x → every xth subcarrier allocated for the actual mobile)

Table 18: Available modulation and coding schemes (uplink)

	modulation scheme	coding scheme
1	QPSK	CC 1/2
2	QPSK	CC 3/4
3	16 QAM	CC 1/2
4	16 QAM	CC 3/4
5	64 QAM	CC 1/2
6	64 QAM	CC 2/3
7	64 QAM	CC 3/4
8	QPSK	CTC 1/2
9	QPSK	CTC 3/4
10	16 QAM	CTC 1/2
11	16 QAM	CTC 3/4
12	64 QAM	CTC 1/2
13	64 QAM	CTC 2/3
14	64 QAM	CTC 3/4

64 QAM is available; however, WiMAX just uses QPSK and 16 QAM in uplink.

3.2.2.5PAR4_MS_CHN_... channel parameters

PAR4_MS_CHN_... contains channel specific parameters such as the channel model to be used and the speed of the mobile. One file per defined mobile (C1.MsId) must be provided to the simulator. The parameters within the file have the prefix C1_MS_CHN…. During the simulation runs they are mapped to the structures C1.MS{iM}.CHN… with iM being the mobile id.

Table 19: Channel specific parameters (uplink)

Parameter name	Possible values	Remarks
C1_MS_CHN. AddOnPowLev_dB	single value real number	Transmit power level of the mobile.
C1_MS_CHN.TxAntSpace	array of numbers depending on the defined antennas in C1.AntSpace	This parameter picks the antennas used for transmission of the frame out of the defined antenna space. [] → use all antennas defined in C1.AntSpace, [1 3] → use the first and third antenna…
C1_MS_CHN.PreDefModel	single string 'AWGN', 'PEDB', 'VEHA', 'VEHAE', 'VEHB'	Cannel model to be used.
C1_MS_CHN.Fad. SpecificChnPar	single number depending on the precalculated sequences	To define which set of precalculated fading sequence is to be used.
C1_MS_CHN.Fad.Speed_kmh	single number	Speed of the mobile [km/h]
C1_MS_CHN. Fad.MultiAntModel	single string 'CorrMatrix'	Multi-antenna model to be used.
C1_MS_CHN.Fad. CorrM.Type	single string 'Parameters', 'Uncorrelated', 'MaxCorrelated'	Type of correlation matrix to be used.
C1_MS_CHN.Fad. CorrM.CalcOption	single number 0, 1	Option for the calculation of the correlation matrix. 0 → use theoretical integral 1 → use approximation
C1_MS_CHN.Fad. CorrM.BsAod	array of numbers 0°…360°	Angel of arrival at the basestation for each path.
C1_MS_CHN.Fad.	array of numbers	Angel of departure at the mobile for

CorrM.MsAoa	0°…360°	each path.
C1_MS_CHN.Fad. CorrM.BsAngSpread	single number 0°…360°	Angular spread at the basestation.
C1_MS_CHN.Fad. CorrM.MsAngSpread	single number 0°…360°	Angular spread at the mobile.
C1_MS_CHN.FBMC.f_off	single number or 'r' [-0.5…0.5]	Frequency offset, in parts of a subcarrier bandwidth. r → random value
C1_MS_CHN.FBMC.ph_off	single number or 'r' [-π…π]	Phase offset in radians. r → random value
C1_MS_CHN.FBMC. fractional_time_delay	single number real value	Fractional time delay. Normalized with M/2.

3.2.2.6 PAR5_BS_ANA_... parameters concerning the analogue part of the basestation

PAR5_BS_ANA_... contains the parameters concerning the analogue front end of the basestation (noise temperature and noise figure of the receive chain). One file per defined receiving mobile (C1.BsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_BS_ANA…. During the simulation runs they are mapped to the structures C1.BS{iB}.ANA… with iB being the basestation id.

Table 20: Parameters concerning the analogue front end of the basestation (uplink)

Parameter name	Possible values	Remarks
C1_BS_ANA.Temp	single number	Noise temperature.
C1_BS_ANA.NoiseFig_dB	single number	Noise figure.

3.2.2.7 PAR6_BS_DIGI_... parameters concerning the digital part of the basestation

PAR6_BS_DIGI_... contains the parameters concerning the digital part of the basestation such as channel estimator and number of iterations to be done while turbo decoding. One file per defined basestation (C1.BsId) must be provided to the simulator, again with the respective suffix. The parameters within the file have the prefix C1_BS_DIGI…. During the simulation runs they are mapped to the structures C1.BS{iB}.DIGI… with iB being the mobile id.

Table 21: Parameters concerning the digital signal processing within the basestation (uplink)

Parameter name	Possible values	Remarks
C1_BS_DIGI.ChnEstType	single number	Estimation of the channel coefficients.

	2, 10, 40, 41	2 → ideal channel estimation (known transmit signal, no noise) 10 → pilot averaging 40 → linear interpolation between the pilots 41 → linear regression (just AMC)
C1_BS_DIGI.FftType	single number 1, 11	1 → ideal FFT 11 → ideal FFT with quantisation of the output
C1_BS_DIGI.DemType	single number 1	1 → ideal demodulation with amplitude weights of the generated softbits
C1_BS_DIGI.CcDecType	single number 1	1 → ideal unquantised decoding
C1_BS_DIGI.BfMethod	single string 'MRC'	MRC → maximum ratio combining
C1_BS_DIGI.MimoMethod	single string 'ZF', 'MMSE'	Method for stream separation
C1_BS_DIGI.IdxRxAnt	array of numbers depending on the defined antennas in C1.AntSpace	This parameter picks the antennas used for transmission of the frame out of the defined antenna space. [] → use all antennas defined in C1.AntSpace, [1 3] → use the first and third antenna…
C1_BS_DIGI.NoiseEstType	single number 1	Noise estimation type.
C1_BS_DIGI.FBMC. CFO_correction	single string 'PCI', 'none', 'Estimation'	PCI → perfect channel information none → no estimation, no correction Estimation → offsets are estimated with the help of the pilots
C1_BS_DIGI.FBMC. FTC_estimation	single string 'PCI', 'Estimation'	PCI → perfect channel information Estimation → fractional time delay estimated, no correction up to now
C1_BS_DIGI.FBMC. estimation_method	single string 'PCI', 'Estimation'	PCI → perfect channel information Estimation → channel estimation with the help of the pilots
C1_BS_DIGI.FBMC. eq_case	single number 1, 2, 3	Number of taps the equalizer is using.
C1_BS_DIGI.FBMC. eq_criterion	single string 'ZF', 'MSE'	ZF → zero forcing MSE → mean squared error

C1_BS_DIGI.CTC. ScalBits	single number 1, 2, …	resolution for CTC decoding (recommended: 6 for QPSK and 16QAM 8 for 64 QAM)
C1_BS_DIGI.CTC. NbIteration	single number 1, 2, …	Number of iterations used for turbo decoding. (recommended: 4)

4 Simulation results

Within this chapter first simulation results are presented. As the simulation phase just has started in PHYDYAS, the main focus today is the introduction of some of the features of the simulator. Advanced investigations will be presented in future deliverables.

The quality of a wireless link can vary significantly, mainly depending on the position of the mobile within the cell. Near the basestation the quality naturally is much better than at the border of the cell, both due to the smaller pathloss at the center of the cell and the stronger inter-cell interference at the border. To accommodate for that different modulation schemes are available. Additionally the amount of redundancy can be varied via the coderate. This way good links can be exploited for high data rates, while bad links still can be served.

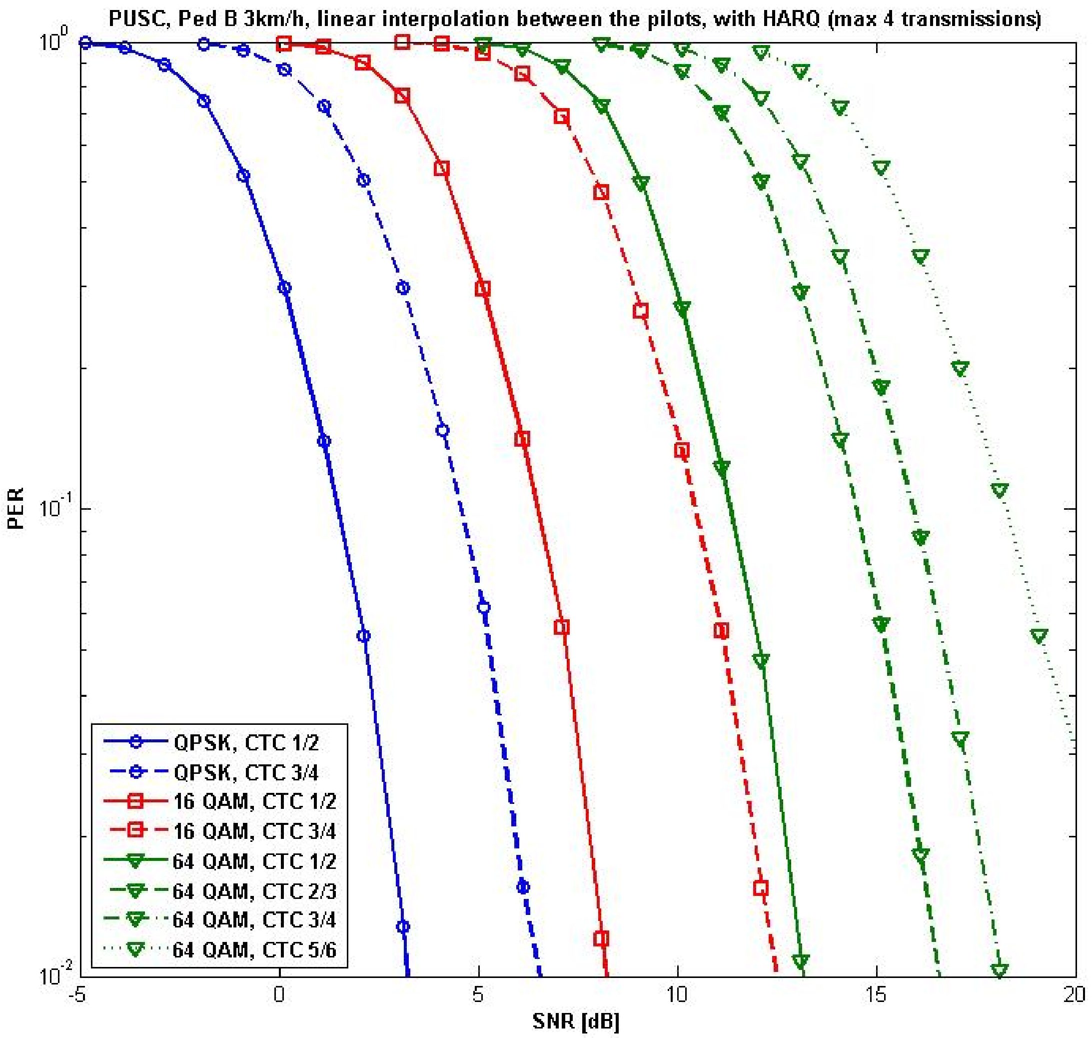

Figure 32: Comparison of modulation and coding schemes with varying robustness

No offsets (time, frequency and phase) are present. Fast fading and receiver noise are the only distortions (Ped B at 3 km/h). The used permutation scheme is downlink PUSC, FBMC is not active, channel estimation is done via linear interpolation between the pilots. HARQ is active with up to 3 retransmissions. The distance between the transmissions is 5 frames. CTC with 4 iterations is applied for error correction.

If HARQ is active, a CRC checksum is appended to the transmitted data burst. This way the receiver can perform error detection. If this detection fails no acknowledgement is sent to the transmitter. In that case the same burst is transmitted again. Combining all received replicas the probability of receiving the burst without errors is significantly increased:

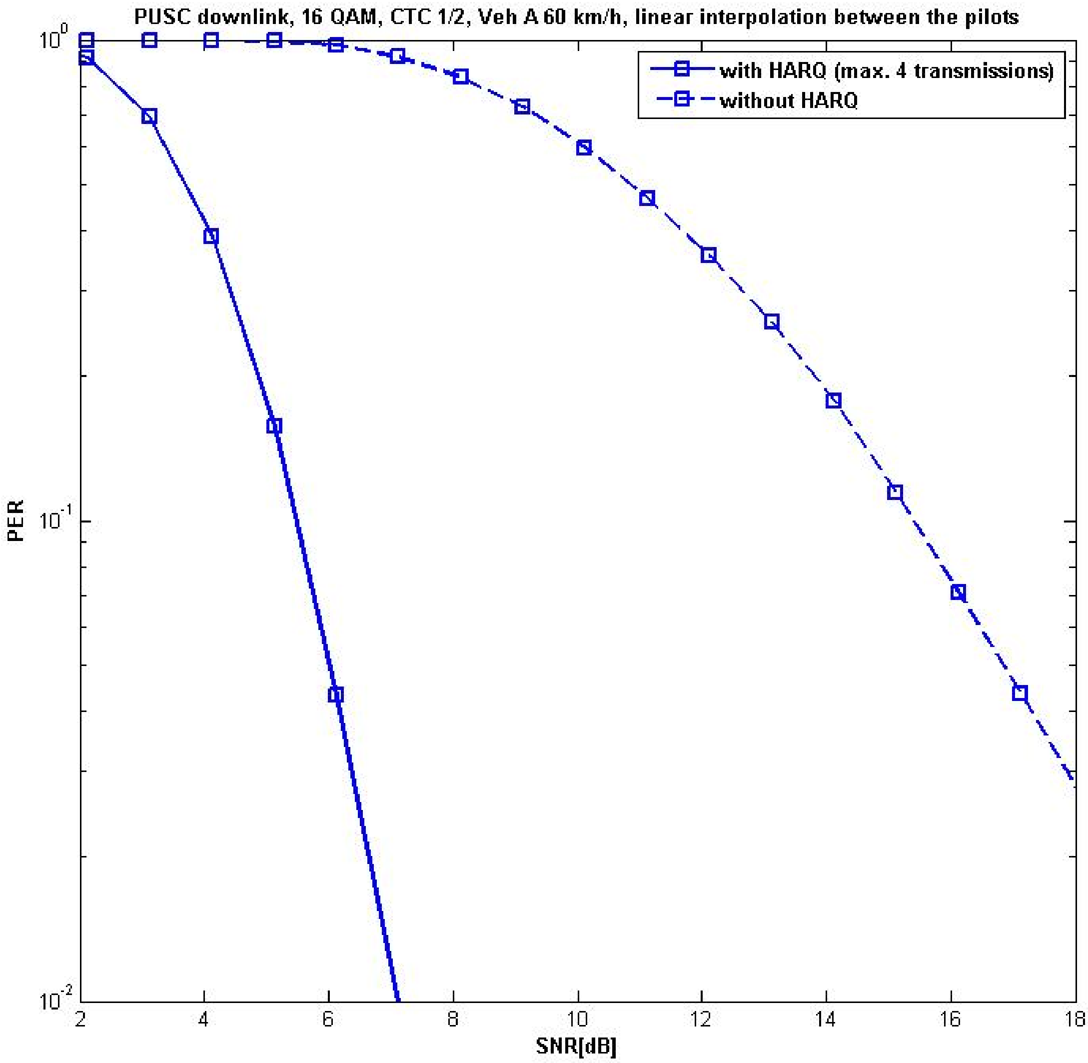

Figure 33: Influence of HARQ (PER)

Again fast fading and receiver noise are the distortions affecting the signal (Veh A at 60 km/h). The used permutation scheme is uplink AMC, FBMC is not active, linear interpolation between the pilots is used. The distance between the HARQ transmissions is 5 frames. Again CTC with 4 iterations is applied for error correction. The reason of the observable gain is twofold. On the one hand due to the coherent combining of the HARQ replicas (this effect is similar to the noise reduction when using pilot averaging for channel estimation described earlier) on the other hand due to the utilization of time diversity.
The normalized throughput is:

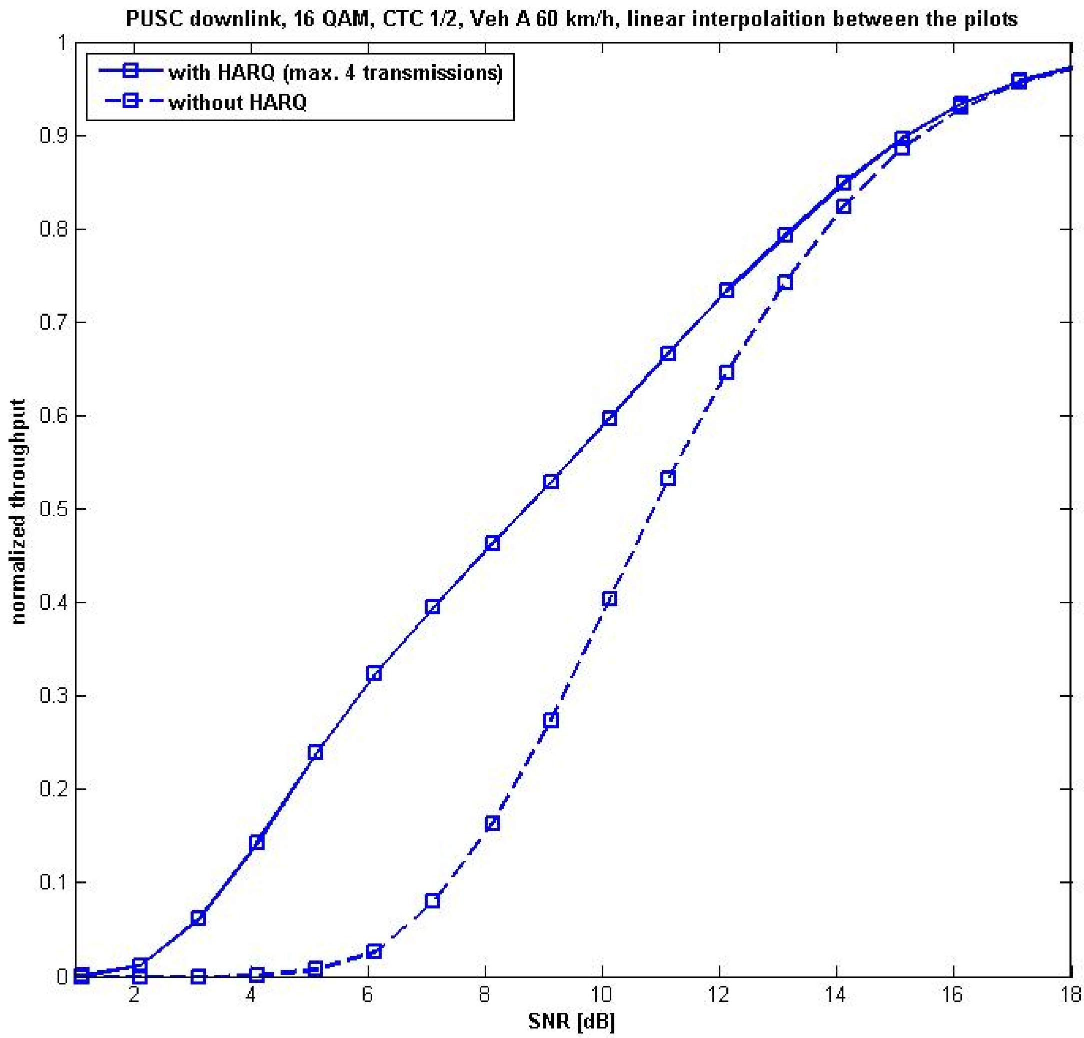

Figure 34: Influence of HARQ (throughput)

Let's say we want to ensure a packet error rate of 1% at the receiver. Looking into Figure 33 we see, that in this case a SNR of around 7 dB is needed. Looking at 7 dB in Figure 34 we see that due to HARQ the normalized throughput could be approximately increased by the factor 5.

As stated in an earlier chapter, different channel estimation algorithms are implemented:

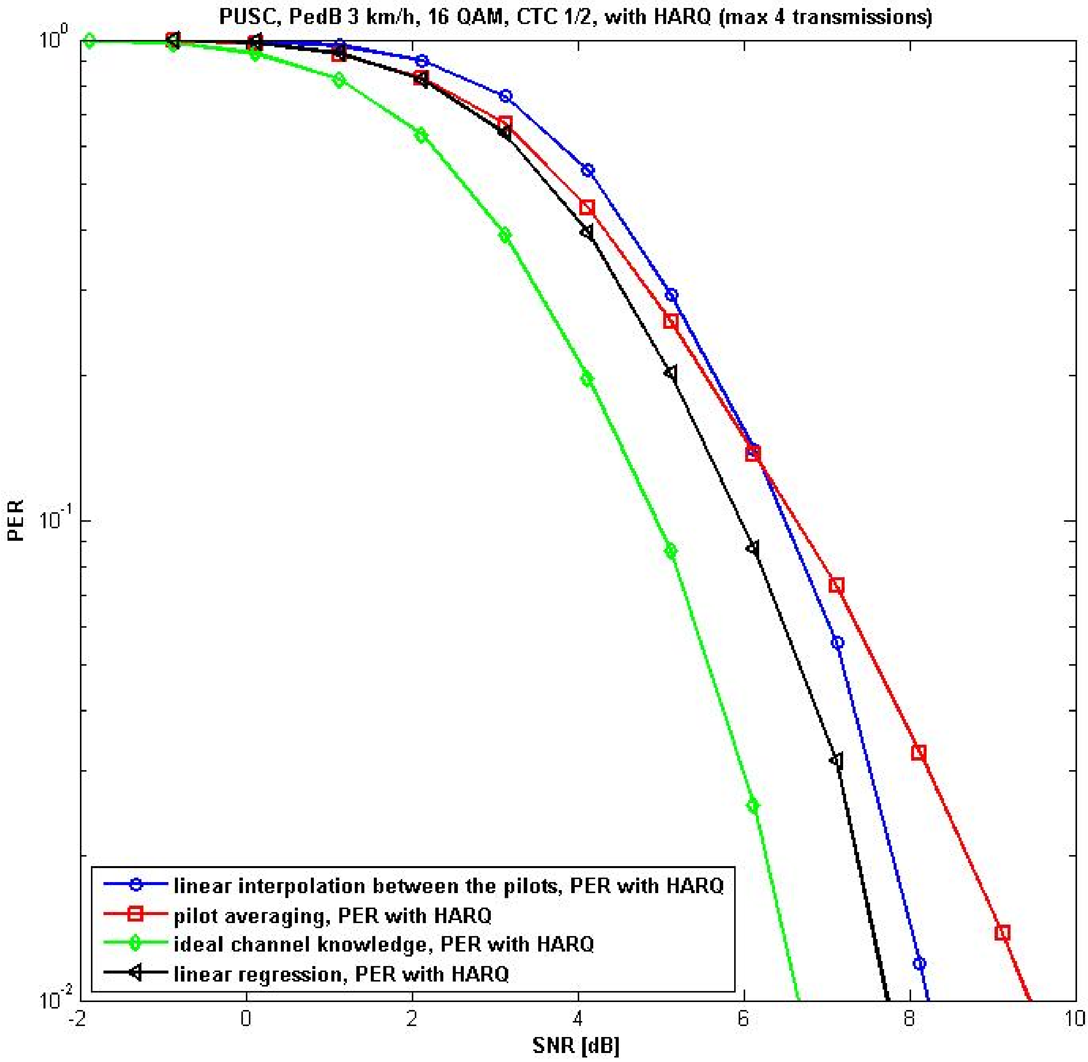

Figure 35: Comparison of diverse channel estimation methods

Fast fading and receiver noise are the distortions affecting the signal (Ped B at 3 km/h). The used permutation scheme is downlink PUSC, FBMC is not active. Again CTC with 4 iterations is applied for error correction. The estimation methods used are ideal channel estimation, pilot averaging and the two implemented linear interpolators. Naturally if perfect channel knowledge is present the best results can be observed. A comparison of pilot averaging with the linear interpolation between the pilots nicely shows the influence of the noise reduction present in the former. For low SNR values (< 6 dB) the system distortion due to noise exceeds the loss due to the poor replication of the channel in the case of pilot averaging. This is changing with increasing SNR. Now noise is low and the better reproduction of the channel advantages the linear interpolator. Linear regression is a compromise combining the advantages of pilot averaging and linear interpolation between the pilots.

Following is a comparison of the performance under different channel situations:

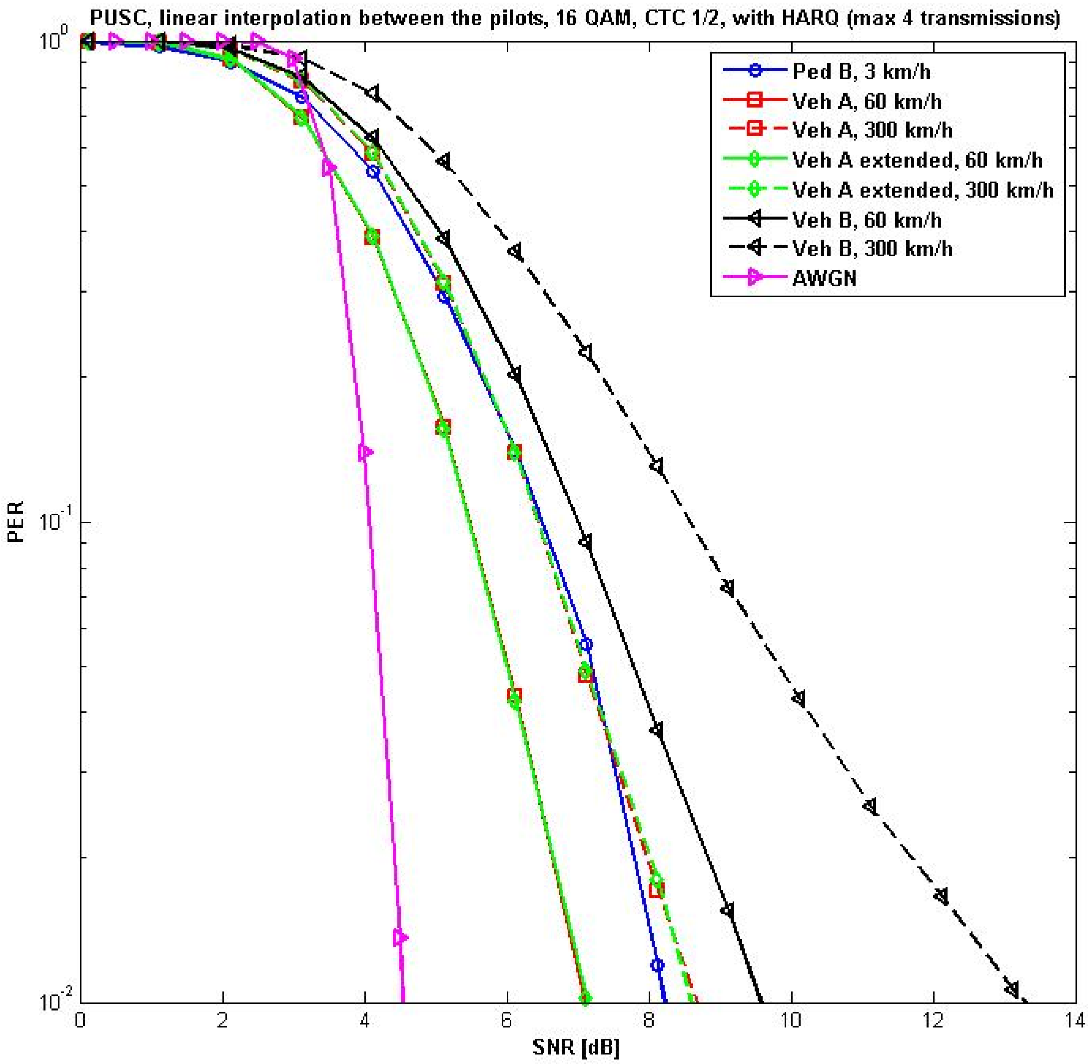

Figure 36: Comparison of different channels

The same system parameters are used as in the investigation of the channel estimators. The different curves correspond to different channels. Ped B at 3 km/h, Veh A, Veh A extended and Veh B at 60 km/h and 300 km/h each. Finally the simple AWGN case is shown. Up to now the simulator uses the same burst position within the frame for all frames. This disadvantages Ped B 3 km/h, as here the coherence time of the channel is higher than the distance between the retransmissions. This leads to high correlations between the retransmissions leading to worse performance, as diversity cannot be exploited fully. In reality the burst positioning from frame to frame typically changes. Thus the retransmissions would not be placed on the same subcarriers within the frame and thus would encounter independent channels. The simulator will be expanded in a way mimicking this behaviour. Obviously Veh B leads to the worst results. This is clear, as here the highest delay spread and with it the highest frequency selectivity is present.
Finally some exemplarily results for FBMC mode. Shown is the influence of fractional time delay:

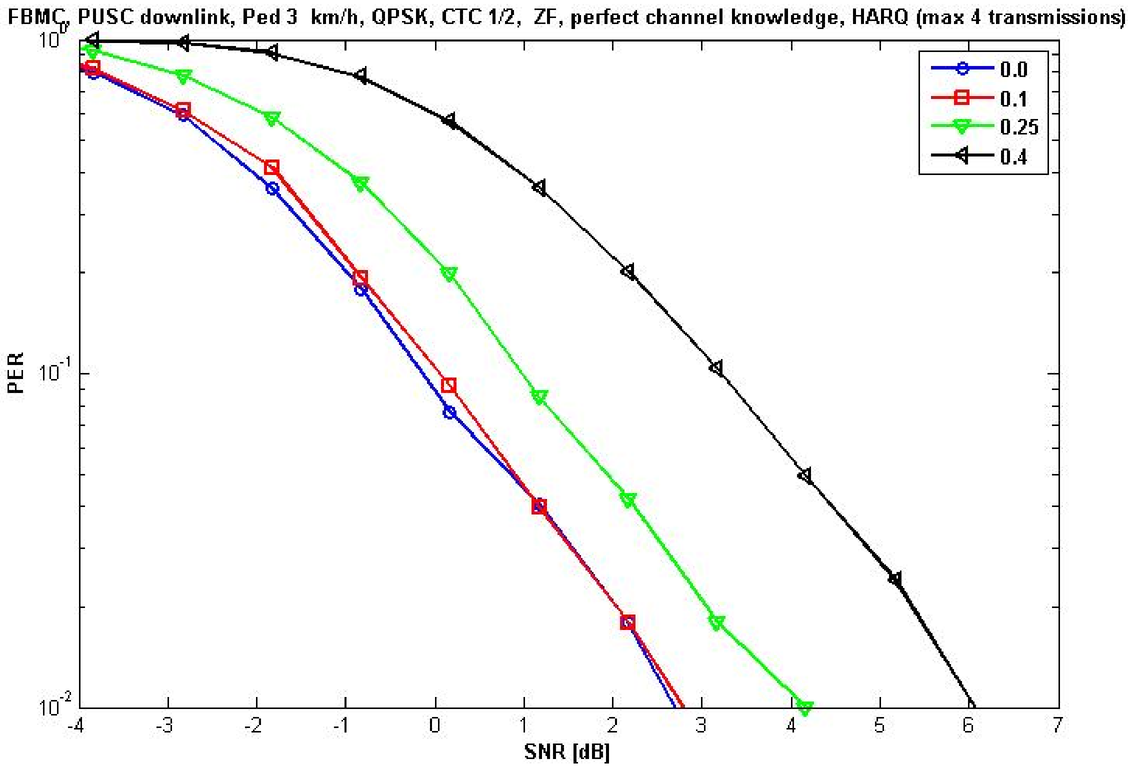

Figure 37: Sensitivity to fractional time delay if HARQ is active

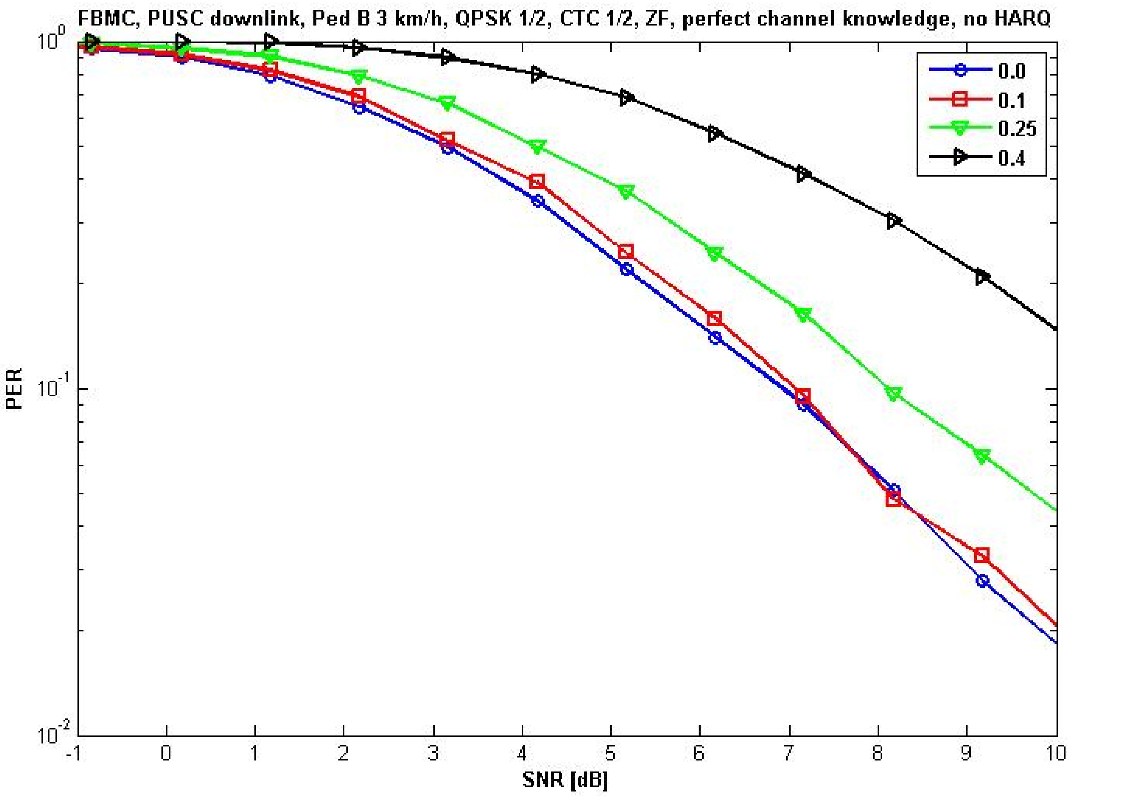

Figure 38: Sensitivity to fractional time delay if no HARQ is used

FBMC mode is active; the used permutation scheme is PUSC in downlink direction. Ped B3 at 3 km/h is the channel model. QPSK and CTC with coderate 1/2 are applied. The channel equalizer has perfect channel knowledge, zero forcing is used. The investigated distortion is the fractional time delay.

5 Outlook - simulation scenarios

Different simulation scenarios will be investigated. Those scenarios differ in terms of compatibility to WiMAX and with respect to their performance. Naturally the achievable performance depends on the grade of compatibility targeted, as if compatibility is forced compromises have to be accepted. To assess the performance FBMC can achieve, several simulation scenarios are defined:

- WiMAX TDD type of frame: this scenario maximizes the compatibility to WiMAX. Just the necessary adjustments needed for FBMC mode are made.
- Single user case with preamble based channel measurement (carrier frequency offset, timing offset and frequency response). During data transmission scattered pilot symbols may be used or not.
- Multiuser case: same as above but with multiple channels in the uplink (AMC-like data positioning). Synchronization techniques developed for groups of subchannels will be exploited.
- Multiuser case with distributed subcarriers (PUSC-like data positioning)

The scenario with maximized compatibility is fully implemented with exception of MIMO and uplink PUSC. Single user and multi user transmission generally can be applied. However, continuative elements, e.g. the introduction and the exploitation of a user preamble, still have to be provided and included into the simulator.

To have a common basis and to frame the simulation space some basic parameter settings are given in Table 22. They should be used in any case.

Table 22: Basic parameter settings

Parameter name	Parameter value
C1_CELL.Frm.Fft	1024
C1_CELL.Frm.CrrFreq	2.5e9 [Hz]
C1_CELL.Frm.ChnBandwidth	10e6 [Hz]
C1_MS_DATA.Diuc C1_MS_DATA.Uiuc	>7 to ensure that CTC is used
C1_CELL_CHN.MS{iM}.ShortTermSNR	false
C1_MS_DIGI.CTC.NbIteration C1_BS_DIGI.CTC.NbIteration	4

6 References

[1] M. Bellanger, "PHYDYAS reference filter bank," PHYDYAS internal report.

[2] IEEE 802.16e-2005, "Air Interface for Fixed and Mobile Broadband Wireless Access Systems", February 2006.

[3] WiMAX Forum, "Mobile System Profile Release 1.0 (Revision 1.6.1: 2008-04-01)", April 2008.

[4] M. Tanda, T. Fusco, M. Renfors, J. Louveaux, M. Bellanger, "Data-aided synchronization and initialization (single antenne)", ICT-211887, PHYDYAS deliverable D2.1, July 2008.

[5] J. Louveaux, L. Baltar, D. Waldhauser, M. Renfors, M. Tanda, C. Bader, E. Kofidis, "Equalization and demodulation in the receiver (single antenna)", ICT-211887 PHYDYAS deliverable D3.1.

[6] P. Siohan, C. Siclet, N. Lacaille, "Analysis and design of OFDM/OQAM systems based on filterbank theory" *IEEE Trans. Signal Processing*, pp 1170-1183, May 2002.

[7] M.G. Bellanger, "Specification and design of a prototype filter for filter bank based multicarrier transmission", in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing* (ICASSP '01), pp. 2417 – 2420, May 2001.

[8] T. Ihalainen, A. Viholainen, T. Hidalgo Stitz, M. Renfors, "On Effects of Transmission Burst Truncation in FBMC Systems", PHYDYAS internal report.

[9] F. Schaich, "Ranging, network entry", PHYDYAS internal report.

[10] M. Bellanger, "Pilot aided channel equalization and measurement – Direct approach," PHYDYAS internal report.

[11] E. Kofidis, D. Katselis, A. Rontogiannis, S. Theodoridis, "Preamble-based channel estimation methods for OFDM/OQAM systems," PHYDYAS internal report.

[12] C. Lele, J.-P. Javaudin, R. Legouable, A. Skrzypczak, P. Siohan, "Channel estimation methods for preamble-based OFDM/OQAM modulations", in *Proc. European Wireless 2007*, Paris, France, April 2007.

[13] T. Ihalainen, T. Hidalgo Stitz, M. Rinne, M. Renfors, "Channel equalization in filter bank based multicarrier modulation for wireless communications," EURASIP Journal on Advances in Signal Processing, vol. 2007, pp. Article ID 49 389, 18 pages, 2007, doi:10.1155/2007/49389.

[14] J.-P. Javaudin, D. Lacroix, A. Rouxel, "Pilot-aided channel estimation for OFDM/OQAM", in *Proc. Vehicular Technology Conf. (VTC 2003-Spring),* pp. 1581-1585, April 2003.