HORIZON 2020

Terahertz based Ultra High Bandwidth Wireless Access Networks

Deliverable ID:

D5.1

Preparation date:

14 September 2018

Milestone: Final Proposed

Title:

Initial PHY-Layer Model and Simulator

Editor/Lead beneficiary (name/partner):

Johannes Eckhardt / TUBS

Internally reviewed by (name/partner):

Noureddine Boujnah / WIT
Cyril Renaud / UCL

Approved by:

PSC tbc

Dissemination level		
PU	Public	
CO	Confidential, only for members of the consortium (including Commission Services)	

Revisions				
Version	**Date**	**Author**	**Organisation**	**Details**
0.1	03.04.18	Johannes Eckhardt	TUBS	Structure proposed
0.2	18.05.18	Johannes Eckhardt	TUBS	Structure approved, Content draft
0.3	02.07.18	Johannes Eckhardt	TUBS	Ready for internal review
0.4	14.09.18	Johannes Eckhardt	TUBS	Final proposed

The research leading to these results has received funding from the EU H2020 under grant agreement n° 761579.

Table of contents

List of figures

List of tables

Executive summary

The deliverable D5.1 gives an overview of the different simulators that are foreseen in work package 5 of the TERAPOD project. It outlines the major characteristics of each of these simulators while providing specific emphasis on the physical layer simulator and its characteristics.

1 Introduction

1.1 Summary

This deliverable deals with the simulation tools which will be used to model and simulate the THz communication for the TERAPOD project, within the individual layers of the OSI layer model. It gives a brief overview of each of the individual simulation tools as well as their interfaces to the other respective simulators. It also focuses on the physical layer simulator, outlining its general concept and functionalities.

1.2 Structure of this document

This document is structured as follows:

- Chapter 2 gives an overview of the whole work package 5 simulation strategy and will be repeated in every deliverable of this work package. It details the relationship and the interfaces between the individual simulators for each layer.
- Chapter 3 explains in more detail the functionality of the physical layer simulator which represents the main topic of this deliverable.

1.3 Relationships with other deliverables

This deliverable is the first deliverable of work package 5. The work presented in this document relates to the following deliverables:

- D2.1 – Initial Requirements and Scenario Specifications

1.4 Contributors

The following partners have contributed to this deliverable:

- Johannes Eckhardt / TUBS
- Noureddine Boujnah / TSSG
- Zdravko Bozakov / Dell EMC
- Fiach O'Donnell / Dell EMC

1.5 Acronyms and abbreviations

BDMA – Beam Division Multiple Access
BER – Bit Error Rate
CIR – Channel Impulse Response
CRC – Cyclic Redundancy Check
DLL – Data Link Layer
FDM – Frequency Division Multiplexing
FEC – Forward Error Correction
L3 – Layer 3, network layer
MAC – Medium Access Technique
NL – Network Layer
PHY – Physical Layer
SNR – Signal to Noise Ratio
TDM – Time Division Multiplexing
WP – Work Package

2 Overview of the simulation tools

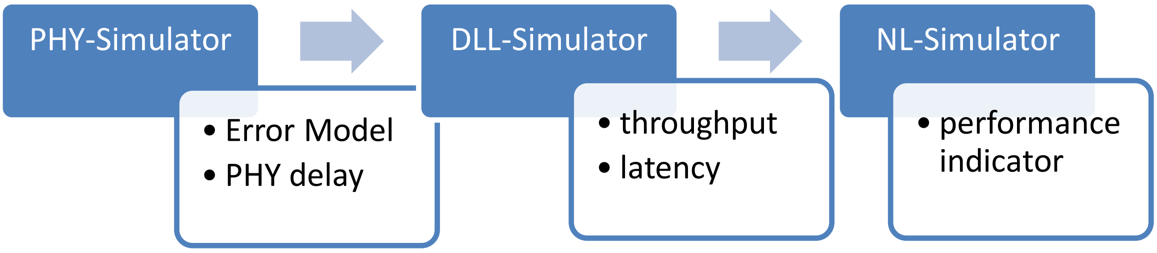

Figure 1: Overview of the simulation tools of work package 5

2.1 Physical Layer Simulator [TUBS]

2.1.1 Short description

The increasing demand for high data rates motivates the development of multi-gigabit communication systems in the THz frequency range. This frequency range shows propagation mechanisms which differ from the already developed and well-known systems. It is therefore mandatory to investigate the transmission of single bits over the physical layer to evaluate the transmission quality. A physical layer simulator models the transmission taking the channel coding, the modulation and the radio channel into account. Since the new devices created for THz communication may influence the transmission in a significant way, hardware impairments lie also in the scope of the TERAPOD project. The physical layer simulator will leverage the THz channel measurement and modelling from WP4 to correctly simulate the channel influence on the transmission. It is planned as a dedicated tool which fully represents the physical effects occurring in THz communication.

Two types of deployment can be imagined for the physical layer simulator. The first and principle type of deployment uses a random bit generator to create some significant error statistics for a certain input parameter set. Then an error pattern generator will be programmed which will emulate the physical layer characteristics. This will enable the integration of the physical layer characteristics to higher level simulators.

The second type of deployment consists in the utilization of real data provided by TSSG. The real data will be processed by the physical layer simulator and will be given back to TSSG after the transmission. In this way real data communication can be tested and evaluated.

2.1.2 Parameter inputs

The physical layer simulation depends on a finite set of configuration parameters and information input. Table 1 shows the different parameters and their suppliers.

Table 1: Input Parameters for the Physical Layer Simulator

Parameter	Origin	Comment
Coding Scheme	DLL	
Modulation Scheme	DLL	
Symbol duration / Nyquist bandwidth	DLL	
Transmit Power	WP4	SNR at the receiver is an equivalent value which can be used
CIR of the radio channel	WP4	Modelled by raytracing
Characteristics of RF hardware	WP4	

2.1.3 Output and Metrics

To evaluate the data transmission over the physical layer in dependence of the different input parameters the bit error rate BER will be measured. The BER is a common figure of merit that will act as a base of comparison for the different combinations of input parameters. For illustration reasons the theoretically possible data rate will also be calculated but the actual data throughput needs to be evaluated at higher layers. The BER itself does not represent all information of the error behaviour since correlation between error phenomena is possible. Therefore, an error pattern generator will be developed which on the one hand incorporates the whole error statistics and on the other hand enables the implementation of the statistical representation of the physical layer to higher layer simulators. This will be done by a compiled .dll file that can be integrated in any other piece of software.

Furthermore, the delay introduced by the physical layer will be evaluated in dependence of the input parameters. In the case that real data is processed by the physical layer simulator the processed data created during the simulation will be provided in an appropriate data format.

2.1.4 Requirements for other Layers

As the physical layer interacts with the DLL some agreements must be met to guarantee the accurate functionality.

One constraint for the DLL protocol concerns the frame length. Every time information is sent over the physical layer, the number of bits sent without an interruption need to be a multiple of the code word length and the modulation order.

In the physical layer simulator a time-continuous simulation will be realized. This enables the calculation of a representative statistic for a certain parameter input set and also enables a simulation of a transmission of real data. Therefore, a common data format between the DLL and the PHY layer needs to be defined.

To estimate the simulation time and the feasibility of the real data simulation, the active period and the total simulated time of the DLL simulator need to be dimensioned in cooperation with the PHY layer simulation resources.

2.1.5 Data Formats

At different interfaces. information needs to be exchanged between the simulators and their respective partners. Therefore, several data formats will be defined which enable the collaborative work flow.

For the sake of a reasonable data size, high volume simulation data will be exchanged in a machine-readable binary data file. For test and debugging reasons, a human-readable XML alternative for small data volumes will be used in parallel. The syntax of the binary data file will be defined by TUBS and TSSG.

The error pattern generator will be provided as a compiled .dll file, or if possible as a MEX file. The CTF created as an output of the channel modelling will be represented in a data format which will be developed on the base of the data format presented in the IEEE802.15.3d channel modelling document [1].

2.1.6 Simulation Cases

The simulation cases for the data centre scenario in terms of the geometry will conform to the use cases presented in the deliverable D2.1. Regarding the used modulation and coding schemes, the simulation will consider all schemes which will be supported by the hardware developed in work package three. Generally, all combinations of the input parameters can be simulated. Simulation cases with a special interest will be worked on over the course of the project.

2.2 Data Link Layer Simulator [TSSG]

2.2.1 Short description

The data link layer (DLL) receives packets from the upper layer and processes them. Packets are converted to frames with preambles and protection bits using a Cyclic Redundancy Check (CRC). The DLL is responsible for flow control and medium access. The data link layer receives packets from L3, where packets are converted into frames and buffered for transmission or re-transmission.

We can summarize the techniques and functionalities of the DLL as follows:

In the DLL, we start by framing where received packets from the network layer are mapped into frames with length L, and, for each frame, a CRC is computed using primitive polynomial and inserted in the frame tail. A frame remains in the buffer, for a given time, if no slot is available or if waiting for acknowledgement from the receiver, and if the transmission is successful it is discarded. Otherwise, it is re-transmitted. The DLL also performs frame aggregations: when the channel is available, N small frames are transmitted together in one super frame. Lost small frames will be retransmitted in the following block of frames using the correct order. At the receiving side frames are re-assembled. The aggregation enhances data throughput and also reduces errors. For harsh propagation channels, retransmission is required for lost frames (uncoded); a maximum number of retransmission should be set.

Medium Access Techniques (MAC) are algorithms used by the DLL to access the link. For terahertz communication, antennas are directional and phased array elements are used to steer beams. In this case possible MAC techniques that could be deployed are: Time Division Multiplexing (TDM), Frequency Division Multiplexing (FDM), possibly Frequency-Time Division multiplexing (F-TDM) and Beam Division Multiple Access (BDMA). Terahertz MAC techniques could also be assisted by lower band technologies.

The DLL interfaces with the physical layer where frame length should fit physical layer parameters, such as modulation order and coding rate and the available bandwidth. The DLL also interfaces with the network layer and exchanges information related to communication links and measurements, as well as buffers. In a network context, routing algorithms in the network layer will use DLL measurements.

In order to test DLL performances, we start by generating frames based on their traffic profile within the data center. Frame parameters are described in section 2.2.2. Some challenges in designing the real data link layer also need to be studied and addressed, such as beam switching and steering, and multi user access techniques.

2.2.2 Parameter inputs

Table 2: Input Parameters for the Data Link Layer Simulator

Parameter	**Designation**	**Comments**
Frame length	L	
Attached synch marker length	L_{asyn}	Length 7/11/13 for frame synchronization
Transmission time	T	Maximum one frame during T
Response frame length	L_{ack}	Each frame with Ack, possibly each group of frames with one ack
Switching time	T_{sw}	It depends on antenna technology
Active period	a(percentage)	100*L/T
Maximal number of retransmission	N_{max}	Depends on frame length and bit error probability
Error model	.dll/MEX file	Input raw frames , output frames with errors
CRC field	R bits	CRC field to each frame, primitive polynomial
Topology parameters	From DELL/EMC	Distance between nodes
Simulation Time	T_{sim}	1 mn

2.2.3 Output and Metrics

- Frame Error Rate: depends on error model
- Throughput: affected by re-transmission, bandwidth, channel and MCS
- Frame delay: processing time + waiting time + physical layer processing time
- Stochastic model of DLL: waiting time, number of frames in the queue

2.2.4 Requirements for other Layers

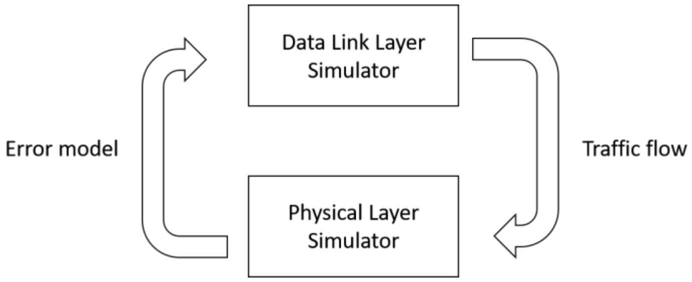

Figure 2: Relation between Physical and Data Link Layer Simulators

2.2.5 Data Formats

- Binary file of the traffic (sequence of bits, active period, number of bit per frame)

2.2.6 Simulation Cases

- Continuous transmission (for the simulation time)
- Communication using short frames
- Communication using long frames
- Re-transmission

2.3 Network Simulator [DER]

2.3.1 Short description

As today's network topologies grow increasingly more complex, and are prone to repeated change, the beneficial use of network simulation tools have never been more relevant. It can be extremely useful to monitor the behaviour and performance of a network under various conditions and modify the environmental attributes associated with it. Though the terms simulator and emulator are quite often used interchangeably in the context of network design and testing, it is important to distinguish the difference between the two and understand what each type of tool can bring to the TERAPOD project.

Network simulation, is a technique where a software program precisely models a network's behaviour and the interactions between its underlying network entities – switches, links, node and associated protocols. As a result, virtually arbitrary levels of fidelity can be realized given enough compute time and resources. On the other hand, network emulation, aims to increase the efficiency for simulating complex network system by trading off some degree of reproducibility and accuracy to reduce the time and resource required for analysis. Emulation approaches aim to realistically replicate large scale networks in a controlled environment, frequently utilizing existing stacks (e.g., TCP/IP) and technologies (e.g., virtualization) or close approximations thereof to test specific aspect of a networked system.

Both methods and application types are used primarily in the fields of research & development and academia to shed light on fundamental networking concepts such as standard network node modelling, queueing, traffic analysis, and techniques like software-defined mechanisms or parallel system integration. Both approaches come with support for the most popular technologies and networks – including, but not limited to, wireless sensor networks, wireless LANs, LTE-5G, ADHOC networks etc. For any projects or work concerning the design, development and testing of a network – being able to properly simulate or emulate said network is a crucial tool.

In the case of TERAPOD, network simulators/emulators are of vital importance to demonstrate and study the communication properties of ultra-high bandwidth wireless THz communication. At the network layer the above mentioned approaches will be used to assess the benefits of augmenting datacentre networks with dynamically configurable wireless links. Specifically, we will analyse how such links can be used to eliminate network hot-spots and improve utilization levels of the available network resources under typical datacentre traffic workloads. To this end, forwarding performance, queuing levels, and the responsiveness of SDN supported routing/forwarding approaches must be evaluated. Three different network simulation/emulation tools are considered for this purpose:

- *ns-3* – the third iteration of ns, a discrete event network simulator where events occur at a particular instance and mark state changes in the system. [2]
- *Mininet* – a network emulator that creates a virtual network of hosts, switches, controllers and links on a single Linux kernel. [3]
- *CORE* – Common Open Research Emulator, a tool for emulating networks on one or more machines that can also be connected to live networks. [4]

2.3.2 Parameter inputs

Table 3: Input Parameters for the Network Simulator

Parameter	Designation	Comments
DC topology	G	Logical DC topology graph subject to geometrical constraints
Wireless link capacity	C	Subject to geometry constraints
Wireless link latency	D	Total link latency

Parameter	Designation	Comments
Wireless link setup time	T_s	
MTU	M	Length in bytes
Simulation Time	T_{sim}	
Traffic matrix/traces	F	Including statistical traffic characteristics

2.3.3 Output and Metrics

Generally, we expect our simulations to yield the following outputs:

- Total utilization: given equal network conditions, the overall utilization of the network is a key target metric. The goal of the work is to demonstrate a higher utilization.
- Number of network hotspots: network events, where the link utilization and/or queuing delays exceed a predefined threshold for a specific amount of time.
- Convergence time: time necessary for SDN infrastructure to respond to a network change by bringing up a new wireless link and the time necessary for network protocols to adapt to the resulting topology changes.

The above list will be refined and extended upon once the experimental environment has been finalized.

2.3.4 Requirements for other Layers

The physical and DL layers must provide the selected simulation tool with a high-level description of the terahertz wireless links, subject to the physical constraints of the developed datacentre. From the perspective of the DC operator, the links are characterized by a constant capacity (rate) over a specific time duration, as well as the delay (latency) experienced by data traversing the link. Thus, a simple latency rate curve representation, which has long been used for performance analysis from network calculus, will provide a suitable starting point for further evaluations. If necessary these can be extended to probabilistic equivalents at later stages of the work, to account for variability of the wireless links.

In addition, the setup latency must be provided (e.g., total time needed to provision a new wireless link).

2.3.5 Data Formats

Text based data formats will be used to encode the used network topology and associated component parameters, encoded e.g., using XML, JSON, depending on the utilized simulation tool.

2.3.6 Simulation Cases

We plan to simulate a fat-tree and spine-leaf datacentre topologies. In addition the scenarios will use pre-generated traffic traces (e.g. using DCTgen2) which will be injected into the topologies. As a result, the setup will allow us to reproducibly compare the defined performance metrics with or without terahertz wireless links.

3 PHY-Layer Model and Simulator

3.1 Overview

The purpose of the physical layer simulator is the prediction of the physical layer's behaviour and its effect on the information transmission. This is done by modelling the whole chain of transmission from the channel encoder via the modulator and the channel itself to the detector and the channel decoder. This method allows for evaluation whether or not the information is transmitted successfully. Figure 3 shows the typical model of the physical layer and its components. First the discrete bit stream is coded with a certain channel coding. Then the coded bit stream is mapped to a symbol depending on the modulation scheme. The symbol is shaped in time domain by the transmit filter and then sent over the channel. The receiver samples the received signal with the symbol period and the detector decides which symbol has been received. At the end the received signal is decoded and it is checked whether there has been a transmission error.

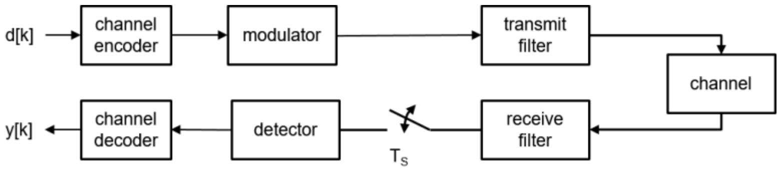

Figure 3: Scheme of the Physical Layer Simulator

The physical layer simulator developed in the TERAPOD project will be integrated in the in-house Simulator for Mobile Networks (SiMoNe) of the Technische Universität Braunschweig. SiMoNe enables raytracing in the THz range and provides the infrastructure for channel modelling which is a significant part of the physical layer simulation [5]. In addition, the programming language C# is highly performant to simulate the target BER of 10^{-12} which is defined in D2.1. To obtain some significant statistics for the reached BER, a simulation of 10^{13} to 10^{14} bits is necessary. This equals a real time of data transmission of around 17 minutes with a data rate of the striven 100 Gbps.

3.2 Concept

In the physical layer simulator, a block concept is used. The block concept makes it possible that simulations with a focus on different aspects are performed in a modular way. Having a limited computing capacity in mind, multiple simulations on different levels of abstraction can show all aspects of the physical layer communication within the same simulation platform.

A standard block may have some inputs such as the configuration parameters or the states from other blocks. It is triggered by a Caller Block in a round robin manner and provides various states to other blocks. These states contain the processed information. Figure 4 shows a basic block scheme.

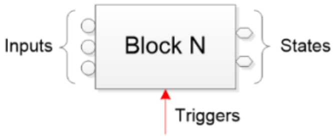

Figure 4: Standard Block Scheme

The working principle is as follows. Once a block is triggered it first saves the data as its inputs in a cache. This ensures that no data is lost in the processing chain. Then it processes the available date and puts it to its outputs. These outputs are called states because only the block itself has a writing permission for its states. Other blocks which are linked to the block of interest can only read the data. This technique guarantees multiple access on the output of one block.

The blocks of the physical layer simulator can generally be classified in blocks which work in the bit domain and blocks which work in the time domain. Bit domain blocks process bits and are located in front of the modulator or after the detector. Time domain blocks are located between the modulator and the detector and process samples.

3.3 Blocks

There are different classes of blocks which are explained more in detail in the following section.

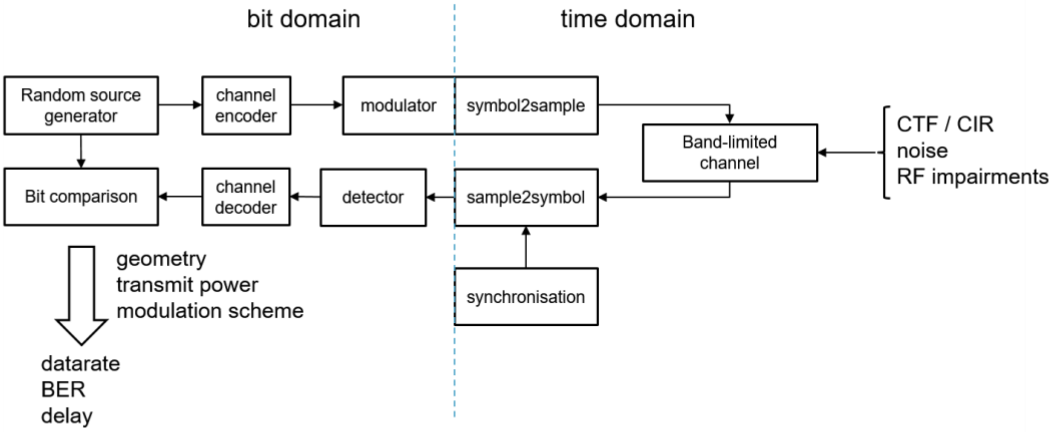

Figure 5: Scheme of the PHY Simulator

Random Source Block

The random source block creates a pseudo-random sequence of 0 and 1 with an equal probability.

Channel Encoder and Channel Decoder

The Channel Encoder realises a forward error correction (FEC). The targeted coding schemes which will be implemented at first are the Hamming Code, the Reed Solomon Code and LDPC 11/15 and 14/15 Codes.

Modulator and Detector

The modulator maps a sequence of bits to a symbol. Linear modulation schemes are in the focus of the simulation. At first On-Off-Keying (OOK), Binary Phase Shift Keying (BPSK), Quadrature PSK (QPSK), and Quadrature Amplitude Modulation with 16 symbols (QAM16), 32symbols (QAM32) and 64 symbols (QAM64), respectively.

Based on the sampled value the detector decides which symbol is received. It is possible to implement simple algorithms like the Most Likelihood Estimator (MLE) or more advanced estimators like the Most Likelihood Sequence Estimator (MLSE). Even very performant adaptive estimators (Forward-

Backward-Estimator, etc.) can be implemented but the high data rate and ~~though~~ the available computational capacity must be considered.

Symbol2Sample and Sample2Symbol

The blocks Symbol2Sample and Sample2Symbol are the interfaces between the bit domain and the time domain. In the first case the symbol2sample block covers the function of the transmit filter and extends the discrete symbol in time. For the reason of a discrete and therefore band-limited representation, the actual pulse shape form will be considered in the channel.

The sample2symbol block realises the sampling with the symbol period. In dependence of a synchronisation algorithm the time discrete representation of the received signal is interpolated and sampled. These sampled values are fed into the detector which identifies the received symbol. In the first phase of the project the synchronisation is assumed as known. Later synchronisation algorithms based on training sequences can be introduced.

Band-limited channel

The band-limited channel is the core part of the physical layer simulator because it distinguishes the simulation of THz communication from other simulations at lower frequencies. The simulation is performed in the equivalent base band system. Hence, the carrier frequency does not directly influence the simulation. Only the bandwidth and the propagation characteristics in the targeted frequency range represent the differences in the THz communication. Based on the scenario data, the material parameters and the antenna diagrams, a ray tracing propagation model is deduced. Together with the band-limited transmit pulse, the band-limited impulse response of the propagation channel is obtained by convolution and can be represented by an FIR filter. This quantity is well suited for a discrete processing in the simulator because it is a full representation of the propagation mechanism in the selected band. Noise can be added to the transmitted signal as well as an interference term or RF impairments which are introduced by the THz hardware.

The type of RF impairments which will have an influence on the transmission will be figured out together with work packages three and four. Possible impairments are phase noise, IQ imbalance of the analogue-to-digital converter, nonlinearities of the hardware or oscillators which leak into the signal path.

At first the additive white Gaussian noise (AWGN) channel will be implemented to theoretically validate the physical layer simulator. The AWGN channel provides a reference for the resulting BER and therefore serves as a verification for the obtained simulation results.

3.4 Inputs

The physical layer simulation depends on a set of input parameters which enable the adaptation of the data transmission to the environment. The input parameters are presented in 2.1.2 and will be explained in more detail in this section.

Modulation Scheme

The Modulation Scheme determines how many bits will be modulated in one symbol. Since the available bandwidth in the THZ range is very high, the project will at first focus on simple modulation schemes. Depending on the electronic devices used in the THz transmitter and receiver, information can also be transmitted in the signals phase and therefore phase shift keying and quadrature amplitude modulation becomes possible.

Coding Scheme

The FEC reduces the errors occurring during the data transmission. Through redundancy and correlation added to the signal, a certain number of errors can be corrected unless the BER is not too high. As given in the standard IEEE 802.15.3d the PHY simulator will focus on the three coding schemes Hamming Code, Reed Solomon code and Low Density Parity Check LDPC Code [1].

Symbol Duration / Nyquist Bandwidth

The symbol duration T_S which is related to the Nyquist Bandwidth B_N by

$$T_\mathrm{S} = \frac{1}{2B_\mathrm{N}} \tag{3.1}$$

sets the maximum possible Data Rate. Together with the used transmit pulse, it also defines the necessary bandwidth. In the standard IEEE 802.15.3d a minimum bandwidth of 2.16 GHz and a maximum bandwidth of 69.12 GHz as well as a maximum spectral power density profile are defined. The spectral power density profile restricts the possible transmit pulses. The simulator will use a root raised cosine filter to shape the transmitted pulses.

Transmit Power

The higher the transmit power the better the signal-to-noise ration SNR at the receiver. The simulation will show which transmit power is needed to transmit the information over a certain distance with a certain data rate.

CIR of the radio channel

The CIR of the radio channel is dependent on the geometry and the materials of the environment. The channel will be measured and modelled in WP 4.

Characteristics of RF hardware

The RF hardware can influence the signals in a significant way. The hardware impairments become even more important when a cutting-edge technology is used. To quantify the influence of the electronic components on the data transmission and the BER, the PHY simulator will take various hardware impairments into account. On the basis of the hardware characterisation of T4.1 in WP4, the simulator will model effects like phase noise, IQ imbalances of the analogue to digital converter ADC, nonlinearities and oscillator leakage effects.

3.5 Outputs

As already briefly described in 2.1.3, the physical layer simulator will have as an output the bit error rate BER and the PHY latency. These performance indicators globally show the quality of the data transmission in the physical layer. To integrate the characteristics of the physical layer to higher layer simulators, a bit error pattern generator will be programmed. In dependence of the prior evaluated statistics of the data transmission it will manipulate a certain input bit stream by adding bit errors in that way that the characteristics of the PHY layer are represented. TSSG and TUBS will together define the shape of the bit error generator to guarantee its envisaged behaviour.

3.6 Data Formats

A short introduction to the different data formats is given in 2.1.5. The definition of data formats used to exchange data between the different tasks in WP5 is of major interest. Therefore the focus of coordinated work between TSSG and TUBS will lie in the definition of the shape of the bit error generator and the data format for data exchange on the bit level.

4 Conclusion/Further work

The deliverable D5.1 briefly shows the major characteristics of the physical layer simulator, the link layer simulator and the network simulator. It names the interfaces between the simulators and the requirements. Consecutively, the main structure of the physical layer simulator is given, providing details on the concept of the simulation, the different parts and components and the simulation parameters.

The next step will be the realisation of a simplified model as a proof of concept. After this, the presented concept with more blocks and a higher complexity will be realised. The focus of the simulation studies always lies on the simulation of data transmission over a THz channel.

References

[1] IEEE 802.15.3d IEEE Standard for High Data Rate, Wireless Multi-Media Networks, Amendment 2: 100 Gb/s Wireless, Switched Point-to-Point Physical Layer

[2] ns-3 discrete-event network simulator for research and educational use - https://www.nsnam.org/

[3] Mininet network emulator, creating virtual networks on a single machine - http://mininet.org/

[4] CORE, Common Open Research Emulalator for network emulation on multiple machines - https://www.nrl.navy.mil/itd/ncs/products/core

[5] D. M. Rose, J. Baumgarten, S. Hahn and T. Kurner, "SiMoNe - Simulator for Mobile Networks: System-Level Simulations in the Context of Realistic Scenarios," *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, Glasgow, 2015, pp. 1-7.