ulm university universität
uulm

Robust Wireless Communications Through the Air-Ground Channel in C-Band

DISSERTATION

zur Erlangung des akademischen Grads eines

DOKTOR-INGENIEURS
(DR.-ING.)

der Fakultät für Ingenieurwissenschaften,
Informatik und Psychologie der Universität Ulm

von

Daniel M. Mielke
aus Kiel

Gutachter:	Prof. Dr.-Ing. Robert Fischer
	Dr.-Ing. Michael Walter
Amtierender Dekan:	Prof. Dr. rer. nat. Matthias Tichy

Ulm, 30. Juni 2025

Abstract

Daniel M. Mielke

Robust Wireless Communications Through the Air-Ground Channel in C-Band

From a technical point of view, much of the communication infrastructure in today's aviation dates back to the beginning of the aviation age: Analogue voice radio and very few digital systems are still the main drivers in air traffic management. While this lack of modern high-throughput, robust, and cyber-secure data links is a significant, but not a fundamental problem in traditional manned aviation, it makes further automation of aviation or even unmanned aviation impossible. Thus, to make unmanned aviation a reality, the development of new data links – called command and control links – is necessary.

The present dissertation contributes to this development as follows: To get a good understanding of the air-ground communication channel, a comprehensive channel measurement campaign has been carried out in C-band using a jet aircraft. The collected channel sounding data have been processed with a focus on the multi-path components, where, unlike some existing approaches, the detection and the tracking of multi-path components is split into two separated steps to save processing time. A novel multi-path component tracking approach that is based on finding paths within a graph has been developed. The outcome of this multi-path component tracking is used to validate the applied processing chain by estimating the position of the corresponding reflectors. To represent the temporal evolution of the tracked multi-path components, a compact form has been developed, whose statistics are analysed by estimating the kernel density. This kernel is then used to generate instances of a channel model. Eventually, this channel modelling approach is used to simulate the transmission of several waveform configurations based on orthogonal frequency division multiplex, which could be used as a command and control link for unmanned air vehicles.

Acknowledgements

I would like to thank Prof. Dr.-Ing. Uwe-Carsten Fiebig, may God bless his soul, and Prof. Dr.-Ing. Robert Fischer for supervising this dissertation. For tragic reasons that no one could have foreseen, Prof. Fiebig was unable to accompany this project to the end; I am all the more grateful to Prof. Fischer for taking on this task.

This work would not have been possible without the expertise of and the many discussions with Dr.-Ing. Michael Walter, to whom I am greatly indebted.

I further want to thank my (former) colleagues from DLR's Institute of Communications and Navigation – especially Dr.-Ing. Michael Schnell, Dr. techn. Thomas Gräupl, Dr.-Ing. Nicolas Schneckenburger, Dr.-Ing. Christian Gentner, Dr.-Ing. Miguel Ángel Bellido Manganell, and Dr.-Ing. Stephan Sand as well as the head of institute, Dr.-Ing. Florian David, for supporting me with their expertise, for sharing their scientific experience, and for their emotional support.

I would like to thank Prof. Dr.-Ing. Peter A. Höher for the excellent education he gave me at the Christian Albrechts University of Kiel and for awakening my interest in the fields of information theory and wireless communications.

This acknowledgement would of course not be complete without giving thanks to the most important people in my life: I want to thank my dear parents Gudrun and Erich for their unconditional support – and my love Charlotte.

Contents

In Memory of
Prof. Dr.-Ing. Uwe-Carsten Fiebig
* 10/07/1962 - † 09/05/2024

Chapter 1

Introduction

1.1 Aviation – Unthinkable without Wireless Radio

The time period between the late 19th and early 20th century was – until then – one of the most innovative periods in human history. Two of the many inventions that were made during this time should be mentioned here: The art of wireless transmission of information through electromagnetic waves[1] and the art of flying[2].

It is doubtful whether aviation could have achieved its triumph without the invention of wireless radio technology, as aviation is unthinkable without *communication*, *navigation*, and *surveillance*. Apart from few exceptions at the beginning of the aviation age, like communications through optical signals and celestial navigation, all of these so called CNS *(Communication, Navigation, Surveillance)* systems make use of electromagnetic waves.

The value of *communication* and *navigation* in aviation can already be seen in the fact that the most basic formula that every pilot learns during training is: aviate, navigate, communicate [19]. *Surveillance*, eventually, is essential for ATM *(Air Traffic Management)* to guarantee safe and efficient flight operations.

1.1.1 Current Situation of CNS Systems in Civil Aviation

As we can see, aviation – not to mention *modern* civil aviation with its high aircraft densities especially in the area around main hubs – is unthinkable without wireless radio technology in general and wireless communications in particular – two research fields that have seen a remarkable evolution within the last decades. However, the long product lifecycles and the high level of regulation in aviation led to a situation, where the actually deployed communications technology is several generations behind the communications technology applied in mass-market communications standards: While well established standards like LTE *(Long Term Evolution)*, 5G, or WiFi apply modern digital communication technology, analogue voice radio "is currently and will be for the foreseeable future the primary means for pilots to communicate with different entities" [91] like other pilots and air traffic controllers. Although this quotation is from the year of 2007, the main message still applies and the transition from analogue to modern digital communications in aviation is far from complete.

[1] First wireless radio transmission of information by MARCONI in 1895.

[2] First flight of a manned heavier-than-air aircraft by the WRIGHT brothers in 1903.

In addition to analogue voice radio, implemented using either double or single sideband amplitude modulation in VHF *(Very High Frequency)* band or HF *(High Frequency)* band, respectively, very few digital communication services exist in aviation: The most prominent digital data service in aviation is CPDLC *(Controller-Pilot Data Link Communications)* allowing the exchange of simple text messages – it is therefore sometimes compared to the SMS *(Short Message Service)* in cell networks [84]. The CPDLC service is usually implemented by a data link called VDL *(VHF Data Link)* Mode 2 or a network called FANS *(Future Air Navigation System)*[3], that uses both terrestrial and satellite based narrowband communication links.

For the current situation of aeronautical communications, the following three problems can be identified:

- **Congestion of Spectrum**: The 2019 report of the *Frequency Management Group* of ICAO *(International Civil Aviation Organization)* anticipates a congestion of VHF radio channels in some parts of the world, most notably in South-Eastern Europe [99]. The proposed countermeasure is the reduction of the applied channel bandwidth down to 8.33 kHz in the respective areas, as it is already implemented in other areas. Reducing the bandwidth of analogue voice channels, however, comes at the cost of degraded audio quality. Furthermore, the error-prone and distracting radio channel switching, which is still performed manually by the aircrew, becomes even more complex.
- **Lack of Broadband Data Links**: The *Data Communications Infrastructure Working Group* of ICAO states in [10] that the present data links "cannot provide broadband services [required for advanced ATM] now or in the future, due to the lack of available spectrum".
- **Lack of Security**: An analysis of the cybersecurity measures in present aeronautical data links has drawn the authors of [12] to the conclusion, that the current situation allows attackers "to seriously disrupt air traffic by exploiting data link flaws". Besides, analogue voice radio does not provide *any* security measures (neither authentication nor encryption nor integrity checks); its whole security is based on the human-in-the-loop principle and manual plausibility checks.

The need to modernise the CNS infrastructure was recognised, and large programmes were launched to foster the deployment of more modern communication standards in civil aviation: Alongside the American NextGen *(Next Generation Air Transportation System)* programme, there is the European SESAR *(Single European Sky ATM Research)* programme, a public-private partnership that aims to develop technologies to modernise the operation of manned and unmanned air vehicles of all types.

SESAR promotes LDACS *(L-Band Digital Aeronautical Communication System)*, a terrestrial digital data link for CNS in manned civil aviation, developed under the leadership of DLR *(German Aerospace Center)*. As of now, LDACS is the first integrated CNS system, as it, although initially designed for communications, can be used for APNT *(Alternative Positioning,*

[3]Even if the name suggests otherwise, FANS allows communication as well; the actual implementations are called either FANS-1 or FANS-A, depending on whether the aircraft was manufactured by BOEING or AIRBUS.

Navigation, and Timing) [5] and surveillance [21] as well. LDACS does not only provide much higher data rates than all currently used communication systems [86], but also includes state-of-the-art cybersecurity measures like a modern authentication procedure making use of a PKI *(Public Key Infrastructure)* [54]. Also, the implementation of post-quantum security measures in LDACS has been demonstrated [59].

A comprehensive cost/benefit and business case analysis of several FCI *(Future Communications Infrastructure)* concepts for European airspace came to the result that a multilink deployment – thus including both terrestrial links like LDACS as well as satellite communications – is expected to be the "best performing and most flexible choice" as it "provides the required capacity improvement in the long term, while reducing technical and economic risks" [35].

1.1.2 Unmanned Aviation

The term UA *(Unmanned Aircraft)* is a hypernym that includes several types of air vehicles like drones and RPA *(Remotely Piloted Aircraft)*. There is no single, clear distinction between drones and RPAs in the literature: While [4] tries to distinguish them based on their degree of autonomy – where a drone is assumed to have a higher degree of autonomy than a RPA –, ICAO takes the level of airspace integration into account: According to [49], an RPA is an "equal partner in the civil aviation system, able to interact with ATC *(Air Traffic Control)* and other aircraft on a real-time basis".

In this thesis, if not denoted otherwise, we will use the term UA when referring to all kinds of recoverable[4] unmanned air vehicles that are generally operated BVLOS *(Beyond Visual Line of Sight)*. In practice, this definition is close to ICAO's definition of an RPA, as BVLOS operations may require some sort of airspace integration in many cases.

Like many technologies, unmanned aviation has its roots in military applications where UAs were (and still are) used for reconnaissance and combat [96]. Nowadays, UAs play a more important role in military operations than ever before, as the latest big conflicts have shown: The "use of drones and loitering munition [was] instrumental" [108] in the Second Nagorno-Karabakh War (started in 2020) [50] and at the time of writing not a day goes by without reports of drone attacks during the Russo-Ukrainian War (started in 2022) [71], [92], that might be the "first large-scale, high intensity war where both sides have extensively deployed military and commercial drones" [44].

But not only military applications for UA are on the rise. Multiple areas of operation are already covered, mostly by smaller UAs: Drones are being used at sports events [41], for private and public surveillance [64], in agriculture [100], for the inspection of civil infrastructure such as power lines [25], and the delivery of medical supplies [23], to name just a few examples. However, the vision of using large UAs for unmanned freight operations[5], as it has been investigated in [97], has not become reality yet.

[4] An example for a non-recoverable, i.e. expendable, unmanned air vehicle would be a cruise missile [46].
[5] A perfect example for ICAO's definition of an RPA.

The use of UAs is expected to increase even further: A study presented in [24] predicts a compound annual growth rate of 12.23 % for the global UA market until 2,027 resulting in an overall market volume of USD 25.13 billion.

1.1.3 Communications in Unmanned Aviation

Regulators and literature agree that the operation of UAs in general and the integration of UAs in non-segregated airspace in particular requires a communication link between the UA and some controlling and/or monitoring authority, see e.g. [20], [34], [40]. The data that is transmit over this link includes flight trajectories, telemetry and status information, video feed(s), and an ATC voice relay [78]. We call this data C2 *(Command and Control)* data – and distinguish it from *payload data* which is defined as all data that is *not* immediately required for the operation of an UA.

By its very nature, automation plays a significant role in unmanned aviation due to the absence of a human aircrew onboard the UA. The lack of a human pilot onboard who – in nowadays aviation – acts as a control instance for incoming information has direct consequences on the requirement of the applied communication technology, since "automated and autonomous processes require trust in the communication entities and transferred data" [48].

The foreseen required data throughput[6] caused by mandatory applications like video feeds, and the requirement for a high level of data integrity and authenticity show that the communication infrastructure currently used in aviation is not sufficient for the safe and secure operation of civil[7] UAs. As even modern CNS systems developed for manned aviation, such as LDACS, are unlikely to meet the aforementioned requirements, there is a demand for a new data link technology that can be used as a reliable C2 link for UAs.

As it is common practice in aviation that mission critical communications uses exclusive frequency bands dedicated to aviation by the ITU *(International Telecommunication Union)*, a C2 link for UAs shall also be deployed in such protected spectrum [40]. The official CONOPS *(Concept of Operation)* on RPAS *(Remotely Piloted Aircraft System)* published by ICAO suggests the usage of spectrum in C-band[8] from 5.030 GHz to 5.091 GHz [36].

The objective of this thesis is to design a waveform suitable for a reliable terrestrial[9] C2 link deployed in C-band, allowing the save and secure operation of UAs.

[6] A comprehensive study of the expected applications and the resulting throughput requirements of a C2 link for the operation of UAs has been published in [78].

[7] The communication infrastructure armed forces can use for the operation of UA is not only unavailable, but also mostly inapplicable to civil applications.

[8] Using the letter designation of the IEEE *(Institute of Electrical and Electronics Engineers)* according to [37]; not to be confused with NATO C-band.

[9] In aviation, a terrestrial communication system, is set up using multiple GSs *(Ground Stations)*. Each GS provides coverage for a certain area called *radio cell*. The communication with the clients, i.e. aircraft, is therefore carried out through the ground-to-air and the air-to-ground channels, respectively.

1.2 Development of Wireless Waveforms

To design of a new waveform, it is essential to understand the characteristics of the channel between the transmitter and the receiver. Furthermore, the knowledge of the channel characteristics can be used to develop a channel model which is a useful tool to evaluate the performance of a newly designed waveform.

Performing channel measurements is a common way to gain knowledge on the channel characteristics: The basic idea is to continuously transmit a periodic signal through the channel, while recording the received signal in the receiver. By "comparing" the transmitted signal with the received signal in post-processing, it is possible to extract knowledge on the measured channel's characteristics.

For the aeronautical AG *(Air-Ground)* channel, which is the relevant channel for a terrestrial C2 link, the acquisition of useful channel measurement data inevitably requires the implementation of a flight measurement campaign that ideally covers all relevant phases of a flight, e.g. take-off, en-route, and landing. It is known from literature [30], that there are considerable differences in the channel characteristics of the AG channel for the different phases of a flight. As we assume that the C2 link should be available during all phases of a flight, all these different channel characteristics must be taken into account when designing the waveform; the same also applies to the channel model.

1.3 Contributions of the Dissertation

The contributions of this dissertation can be split up into the following parts:

1. Design, implementation, and execution of a flight measurement campaign to measure the AG wireless channel in C-band [58], [61].
2. Design and implementation of algorithms required for pre-processing the data collected during the measurement campaign [61], including a novel concept for the handling of measurement data using a central data base server.
3. Design and implementation of a novel two-stage approach for the detection and tracking of multipath components is presented, exploiting the ability of parallel processing to enable fast data processing [62].
4. The findings from the tracking of the detected multipath components are used to establish a compact representation of the evolution of the tracked multipath components' parameters [60].
5. A novel approach to analyse and reproduce the statistics of the set of these compact representations of the evolution of the tracked multipath components' parameters [60].
6. A concept to use this approach as a basis for channel modelling [60].

1.4 Structure of the Dissertation

This dissertation is structured as follows:

Chapter 2 provides a general introduction in the topic of radio wave propagation and channel modelling in AG channels. We explain the phenomenon of multipath propagation and how this phenomenon can be handled in channel modelling by using the concept of MPCs *(Multipath Components)*. The introduction also comes with an overview of the existing channel modelling approaches in this area.

Chapter 3 is devoted to the flight measurement campaign that has been performed to collect channel measurement data for the AG channel in C-band. In this chapter we describe the general campaign concept, the hardware setup of both the transmitter and the receiver, the applied synchronisation concept, and the flight tracks and manoeuvres that were performed.

The processing of the measurement data that is required in advance to the actual data processing is covered in **Chapter 4**: This chapter consists of two parts: The computation of the so called GT *(Ground Truth)* and, based on this GT data, the pre-processing of the data samples received during the flight measurement campaign. The latter step is required to bring all collected data into a common domain.

In **Chapter 5**, the component of the most dominant signal path, the *dominant component* is discussed. The processing steps to compute the received power of this component is explained before this processing is applied to actual measurement data. Both the evolution and the statistical distribution of this power during several flight scenarios is investigated.

Chapter 6 focuses on the processing of MPCs *(Multipath Components)*. This chapter introduces a novel, two-staged approach for the separated detection and tracking of MPCs. Two different approaches for the MPC *detection* are presented before a graph structure is introduced that is used to represent the detected MPCs. An algorithm for the tracking of these MPCs within the graph is presented and several options for the representation of the detected mutual connections within the MPCs are discussed. Eventually, the introduced processing steps are applied to measurement data from the flight measurement campaign. This includes the application of reflector localisation that is performed based on the results from the MPC tracking.

In **Chapter 7**, the findings from the previous chapter are taken up and developed further: A compact representation of the detected MPC evolution is introduced and it is shown how it can be used to create a channel model. The statistics of the model realisations are then compared to the extracted statistics of the measurement data to validate the presented model.

The thesis concludes with comprehensive simulations in **Chapter 8**, where the transmission of several configurations of an OFDM *(Orthogonal Frequency-Division Multiplex)* waveform – as it could be used for a C2 link – through several instances of the proposed channel model is simulated.

The structure of the thesis is also visualised in Figure 1.1, where a block diagram is used to give the reader a better understanding of how the structure of the thesis represents the steps of the overall processing chain that is used to achieve the results presented in this work.

FIGURE 1.1: Processing chain oriented visualisation of the structure of this dissertation in a block diagram; used abbreviations: MPC *(Multipath Component)*, OFDM *(Orthogonal Frequency-Division Multiplex)*.

Chapter 2

Wave Propagation and Channel Modelling

This chapter summarises some general information about the physical phenomenon of radio wave propagation and the task of modelling this phenomenon. It is also intended to give an overview of other channel modelling approaches, particularly in the field of aeronautical air-ground channels.

2.1 Radio Wave Propagation

When a stone is thrown into still water, concentric waves propagate from the point of impact, which can be thought of as a *transmitter*, to the outside. The following observations can be made: The height, i.e. the amplitude, of these waves becomes weaker with increasing travelling distance. Furthermore, if a wave front is hitting an obstacle, it gets reflected. If we chose an arbitrary point on the water surface as the *receiver*, we will see that it will first get hit by parts of the original concentric wave fronts and then by the wave fronts reflected from the obstacles. Furthermore, the reflected wave fronts are likely to have a different angle of arrival in the *receiver* than the first wave front.

This example of waves on a water surface can be interpreted as a simplified two-dimensional projection of radio wave propagation in three-dimensional space where similar effects can be observed. In this section, the effects of radio wave propagation relevant to AG communications are explained in detail.

2.1.1 Multipath Components

As C-band AG communications in aviation is discussed in this dissertation, we assume far field conditions if not denoted otherwise: For the discussed setup, we estimate the FRAUNHOFER Distance to be in the range of 0.4 m; the distances between transmitter and receiver and between (close to all) reflectors and the receiver are obviously *much* greater considering the geometries in AG channels. The assumption of far field conditions and the application of high-frequency assumptions on the MAXWELL equations [110] allow us to assume plane wave fronts for the propagating radio waves. It is therefore possible to model the propagation of radio waves as rays travelling along a path.

Similarly to what is described in the introductory example above, propagating radio waves are reflected by obstacles and, depending on the geometry, these reflected signals may arrive at the receiver together with the signal that travelled directly from the transmitter to the receiver. This effect is called multipath propagation. The individual components of such a multipath propagation environment are called MPCs *(Multipath Components)* and indexed by the MPC index $l \in \mathbb{N}$.

2.1.1.1 Parameters of MPCs

In the field of channel modelling, it is common [82] to describe the parameters of an MPC using the following parameters:

Amplitude The received power of a radio signal is of great importance in communication systems as it by definition affects the SNR *(Signal-to-Noise Ratio)* in the receiver and therefore, due to the SHANNON-HARTLEY theorem, the communication channel's capacity. The signal power is proportional to the absolute-squared of the (generally complex) *amplitude* of the signal, thus the phase information of the amplitude is lost. To allow a representation of the phase information, we therefore use the MPC's (complex) amplitude α instead of its power as a parameter.

Delay Since nothing – including radio waves – can travel at infinite speed, but only at limited speed $c \leq c_0$, where c_0 denotes the speed of light in vacuum, the travelling of a radio wave from the transmitter to the receiver takes some time. In the case of radio wave propagation in the AG channel, we assume a constant[1] propagation speed of $c_{\text{air}} = 2.997,025,47 \times 10^8\,\text{m}\,\text{s}^{-1}$ if not denoted otherwise. This travelling time is called *delay* and is denoted by τ. As we will see later, the relative delay between different MPCs is of higher interest than the absolute delay of an individual MPC in some cases.

Doppler Shift In case of a moving receiver and/or transmitter, another effect comes into account: The *Doppler* effect. While the delay results in a signal shift in time domain, the Doppler effect results in a signal shift in frequency domain. We denote this frequency shift by ν.

In case a signal with a carrier frequency f_c is transmitted from a static origin through the AG channel to a moving receiver travelling at speed $\vec{v} \in \mathbb{R}^3$, the frequency f_{Rx} of the signal arriving at the receiver is given by

$$f_{\text{Rx}} = f_c \left(1 + \frac{\langle \vec{v}, \vec{e}_{\text{orig}} \rangle}{c_{\text{air}}}\right), \tag{2.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product in the respective space – here: $\mathbb{R}^3$ – and $\vec{e}_{\text{orig}} \in \mathbb{R}^3$ is a unit vector describing the direction of the direct path from the receiver to the origin of the

[1] Since the refractive index of the air in the Earth's atmosphere depends on the air's temperature and pressure, the actual lightspeed is not constant either. Nevertheless, we consider the simplification of assuming a constant lightspeed to be sufficiently precise for the topics discussed in this dissertation.

radio wave, see e.g. [26], [85]. The origin of the radio wave is either the actual transmitter in the GS or, in case of a reflected signal, the corresponding reflection point.

Using (2.1), the Doppler shift $\nu = f_{\mathrm{Rx}} - f_{\mathrm{c}}$ is given by

$$\begin{aligned}\nu &= \frac{f_{\mathrm{c}}}{c_{\mathrm{air}}} \left\langle \vec{v}, \vec{e}_{\mathrm{orig}} \right\rangle \overset{\substack{\text{definition of}\\ \text{inner product}}}{=} \frac{f_{\mathrm{c}}}{c_{\mathrm{air}}} |\vec{v}| \overbrace{\left|\vec{e}_{\mathrm{orig}}\right|}^{=1} \cos\{\overbrace{\angle\left(\vec{v}, \vec{e}_{\mathrm{orig}}\right)}^{=\varphi_\nu}\} \\ &= \frac{f_{\mathrm{c}}}{c_{\mathrm{air}}} |\vec{v}| \cos\{\varphi_\nu\}. \end{aligned} \tag{2.2}$$

Thus, the actual Doppler shift corresponds to the carrier frequency f_{c} scaled by the resulting speed along the direct path between the origin and the receiver, and the wave propagation speed.

According to [7] the Doppler shift of an MPC can also be understood as the carrier frequency scaled by the negative change rate of the MPC's delay over time t:

$$\nu = -\frac{\mathrm{d}\tau}{\mathrm{d}t} f_{\mathrm{c}}. \tag{2.3}$$

From (2.3), it becomes obvious that it is impossible to actually measure an MPC's Doppler shift without observing it for a certain amount of time. As a matter of consequence, it is impossible to determine an MPC's momentary Doppler shift with an arbitrarily high time resolution, as the actual Doppler shift may have changed during the observation time.

This circumstance can also be explained using (2.2), as it shows that the Doppler depends on an object's – here: the receiver's – speed with respect to the transmitter. However, it is only possible to measure an object's momentary speed by observing the object's position within the reference area for a certain time period greater than zero.

2.1.1.2 Representation of MPCs

The discussion above can be wrapped up as follows: An MPC is described by

- its delay $\tau \in \mathbb{R}$,
- its Doppler shift $\nu \in \mathbb{R}$,
- and its amplitude $\alpha \in \mathbb{C}$.

A useful mathematical concept for a joint representation of such a structure is a *tuple*. Although it would be formally correct to combine real-valued and complex-valued values in a tuple, we decide to use a real-valued tuple to represent an MPC by splitting α into its (real-valued) amplitude $|\alpha|$ and its (real-valued) argument $\arg\{\alpha\}$. Thus, in the following of this dissertation, an MPC is represented by a quadruple $\xi \in \mathbb{R}^4$:

$$\xi = \left(\tau,\ \nu,\ |\alpha|\,,\ \arg\{\alpha\} \right). \tag{2.4}$$

Using the polar representation of α in favour of the CARTESIAN representation allows a straightforward computation of the MPC's power when necessary.

The real-valued representation as given above also simplifies the representation of MPCs in software: A common way to represent a tuple in a computer is an array. Arrays are data structures consisting of multiple elements of the same memory size; in most implementations, this means that all elements of an array must be of the same data type [29]. It is more memory efficient to store four real-valued floating point numbers than three complex-valued floating point numbers[2].

In many cases more than just a single MPC is discussed. Thus, in the following, MPCs are indexed by the MPC index defined above, so that ξ_l refers to the l-th MPC within a certain set of MPCs denoted by $\mathcal{M}$. The same nomenclature applies for the parameters of an MPC, e.g. the delay of ξ_l is given by τ_l.

2.1.2 Description of Propagation Mechanisms

Propagating radio waves are subject to certain propagation mechanisms. In the following, we describe these mechanisms on the basis of [110], taking into account the finding from the previous section that propagating radio waves can be understood as rays.

2.1.2.1 Line-of-Sight Propagation

One special MPC is the component travelling along the LOS *(Line-of-Sight)* path: We call this propagation mechanism LOS propagation. The corresponding LOS signal is defined as the signal component that arrives at the receiver after travelling the direct path, i.e. the path that can be travelled in the least time. Due to atmospheric effects like tropospheric bending, this path is not necessarily a perfectly straight line, i.e. the shortest path, although this is often assumed in practice [82]. The theory behind this effect is explained by FERMAT's principle of least time [110].

The distance of this LOS path is called the LOS distance denoted by d_{LOS}. The time period between transmission and reception along the LOS path is the LOS delay denoted by $\tau_{\mathrm{LOS}} = d_{\mathrm{LOS}}/c_{\mathrm{air}}$. Following this nomenclature, we denote the Doppler shift of the LOS component by ν_{LOS} and its complex amplitude as α_{LOS}.

Analogous to the definition of the LOS distance above, we define the total distance that the signal of an MPC indexed by l travels from the transmitter to its corresponding reflector and from this reflector to the receiver by d_l. Due to the definition of the LOS component, it is $d_{\mathrm{LOS}} < d_l \forall\, l$. From the assumption of a limited and constant propagation speed (here: c_{air}), it follows that the LOS component, if present, is always the first signal component to arrive at the receiver: $\tau_{\mathrm{LOS}} < \tau_l' \forall\, l$, where $\tau_l' = d_l/c_{\mathrm{air}}$ is the total travelling time of the MPC indexed by l. If not denoted otherwise, the delays are normed with respect to the LOS component in the remainder of this thesis: $\tau_l = \tau_l' - \tau_{\mathrm{LOS}}$.

Depending on the channel conditions, the LOS component is either present in the receiver or not. In the latter case, the LOS path is blocked by obstacles like buildings, walls, other vehicles, the aircraft's fuselage, or terrain. Identifying the true LOS component in the receiver

[2] Under the assumption of common IEEE 754 floating point numbers of double precision the real-valued case consumes $4 \times 8\,\mathrm{B} = 32\,\mathrm{B}$, whereas the complex-valued case consumes $3 \times (8\,\mathrm{B} + 8\,\mathrm{B}) = 48\,\mathrm{B}$.

is impossible without any side information: Since a blockage of the LOS component is possible, it would be inaccurate to simply declare the first received signal component as the LOS component. At the same time, it would be inaccurate to always declare the strongest received signal component as the LOS component.

Unlike many other wireless channels, such as indoor wireless channels used in WiFi or mobile radio, the LOS component is present in many air-ground communications scenarios and therefore plays a very important role in the aeronautical channel, as will be shown in the remainder of this dissertation.

2.1.2.2 Reflection and Refraction

Both the effect of reflection and the effect of refraction occur at the boundary of two media with different refractive indices. These media can be entirely different media, like air and a concrete wall of a building, or the same medium with different conditions, like warm air and cold air.

The energy of the ray travelling from medium A to medium B is split into two parts: One part travels back to medium A – where the angle of reflection corresponds to the angle of incidence – and one part propagates through medium B. The first effect is called reflection, the second effect is called refraction; both mechanisms can be explained using Snell's law.

According to the law of conservation of energy, the power of the reflected ray is less or (only in theory) equal than the power of the incoming ray. The power, and therefore also the amplitude, of the reflected ray as well as its phase depend on the (relative) permittivity ϵ_r and (relative) permeability μ_r of the involved media [77]. According to Maxwell, both of these physical quantities are directly linked to a material's refractive index $n = \sqrt{\epsilon_\mathrm{r}\mu_\mathrm{r}}$, see e.g. [75].

Besides the LOS component mentioned above, these reflected rays are discussed in detail in this dissertation. The effect of refraction, however, is not discussed any further, since the refracted part of a ray does not contribute to the received signal in the scenarios discussed in this dissertation.

2.1.2.3 Diffraction

Diffraction is the effect when an incoming ray is not only reflected, but when multiple new rays are launched from the point where the incoming ray has entered another medium.

In practice, this means that a ray "bends around an object" [26]. The result is the counter-intuitive phenomenon that a signal can be received even in case the direct LOS component and other reflected components are blocked by obstacles. This phenomenon is often modelled using the Fresnel knife edge model [26].

2.1.2.4 Scattering

Scattering appears on rough surfaces of (usually) larger sizes, like forests or water surfaces. As described in [82], scattering "often occurs simultaneously with reflection", since – depending on the surface conditions – a part of the signal is reflected in a certain direction, whereas the

remaining part is scattered in multiple directions. From the law of conversation of energy follows that the reflected component is considerably weaker than the incoming ray, because part of its energy is distributed among the scattered components [63].

2.2 Channel Description

2.2.1 Deterministic Channel Description

2.2.1.1 Weight Function and Channel Impulse Response

The IR *(Impulse Response)* is the reaction of a system excited by a DIRAC impulse and is of central importance in the field of system theory. In the general case, such a system is considered as time-variant – as a matter of consequence, its IR is also considered as time-variant. The IR is given by $h(t', t)$, where t denotes the time of *observation* and t' denotes the time of *excitation* of the system.

A (wireless) communication channel is also a system, thus its behaviour can also be described by its IR, called the CIR *(Channel Impulse Response)*. The signal received at a time instant t, after it has travelled through a time-variant channel described by its time-variant CIR $h(t', t)$, is given by

$$s_{\mathrm{Rx}}(t) = \int_{-\infty}^{\infty} s_{\mathrm{Tx}}(t')\, h(t', t)\, \mathrm{d}t' \;+\; n(t), \tag{2.5}$$

where $s_{\mathrm{Tx}}(t)$ denotes the transmitted signal and $n(t)$ denotes a noise term.

As pointed out in [31], it is desirable to describe a time-variant channel's behaviour independently of its time of excitation t'. By substituting $\tau = t - t'$, we get the definition of the *weight function* [31], also known as *input delay spread function* [8]

$$w(\tau, t) = h(t - \tau, t), \tag{2.6}$$

which only depends on the time of observation t and the delay τ, thus the "time until the excitation is eliminated" [31] compared to the time-variant CIR. The weight function therefore "may be interpreted as the response at time t to a unit impulse input τ seconds in the past" [8]. Using this weight function, we rephrase the statement given in (2.5) and get

$$s_{\mathrm{Rx}}(t) = \int_{-\infty}^{\infty} s_{\mathrm{Tx}}(t - \tau)\, w(\tau, t)\, \mathrm{d}\tau \;+\; n(t). \tag{2.7}$$

Despite the different definitions, much of the literature does not differentiate between the weight function and the time-variant CIR, see e.g. [15], [82].

The drawing in Figure 2.1(a) shows a typical scenario given in wireless aeronautical communications where we can safely apply the assumption from Section 2.1.1 i.e. that propagating radio waves can be modelled as rays. A signal $s_{\mathrm{Tx}}(t)$ is emitted from the transmitter on the ground. The radio waves onto which the signal is modulated propagate and are subject

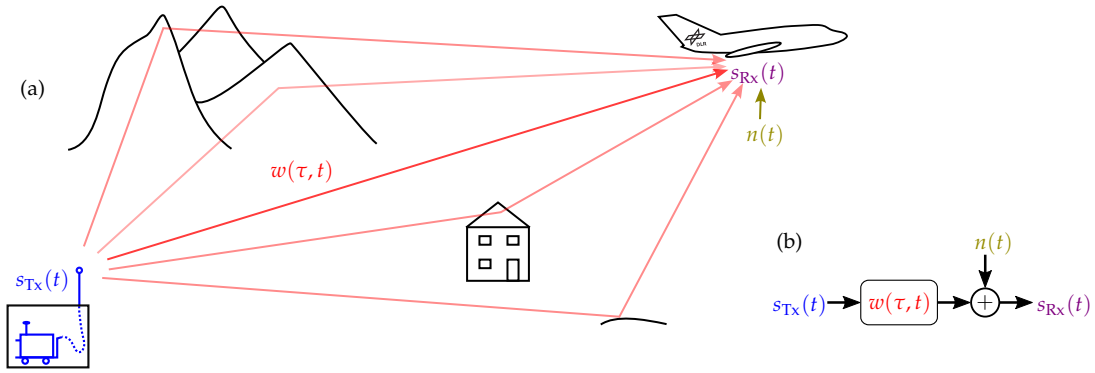


FIGURE 2.1: A typical scenario in AG communications, not true to scale: (a): Sketch of the real-world setup. A signal – $s_{Tx}(t)$ given in blue – is emitted, propagates through the air while being subject to propagation effects like reflections – given in red – and arrives at the receiver where it is superimposed with noise $n(t)$ given in olive. The sum of all of these multipath components and the noise form the received signal $s_{Rx}(t)$ given in purple. (b): System theoretic description of the scenario based on [15]. The colour code helps the reader to map the elements of the system theoretic description in (b) to their real-world counterparts in (a).

to the propagation mechanisms discussed in Section 2.1.2. All effects of these propagation mechanisms are – in the most general case – represented by the weight function $w(\tau, t)$. The signal components arrive at the receiver located in the aircraft and are summed up. These summed up signal components together with some noise form the received signal $s_{Rx}(t)$ that is finally processed in the receiver.

The simple block diagram in Figure 2.1(b) shows a system theoretic description of the same scenario. The block diagram's message corresponds to the message of (2.7).

In case the channel is considered as an LTI *(Linear Time Invariant)* system, the CIR as well as the weight function become time-invariant, thus their behaviour is now independent of the time of observation t. The need to differentiate between these two functions is no longer given and we only consider the time-invariant CIR given by $h(\tau)$. Under the assumption of a time-invariant channel, the theoretical description of the signal transmission given in (2.7) corresponds to the convolution of the CIR with the transmitted signal and a subsequent addition of the noise term:

$$\begin{aligned} s_{Rx}(t) &= \int_{-\infty}^{\infty} h(\tau)\, s_{Tx}(t-\tau)\, \mathrm{d}\tau \;+\; n(t) \\ &= \int_{-\infty}^{\infty} h(t-\tau)\, s_{Tx}(\tau)\, \mathrm{d}\tau \;+\; n(t) \end{aligned} \tag{2.8}$$

Assuming a constant CIR, at least for a certain time period, is common practice and necessary when processing measurement data as it will be shown in later chapters of this dissertation.

2.2.1.2 Alternative Representations of a Time-Variant Channel

In [8], BELLO investigates other system functions besides the weight function to describe a filter in general and a communication channel in particular. The investigated system functions and their relations are listed in the following.

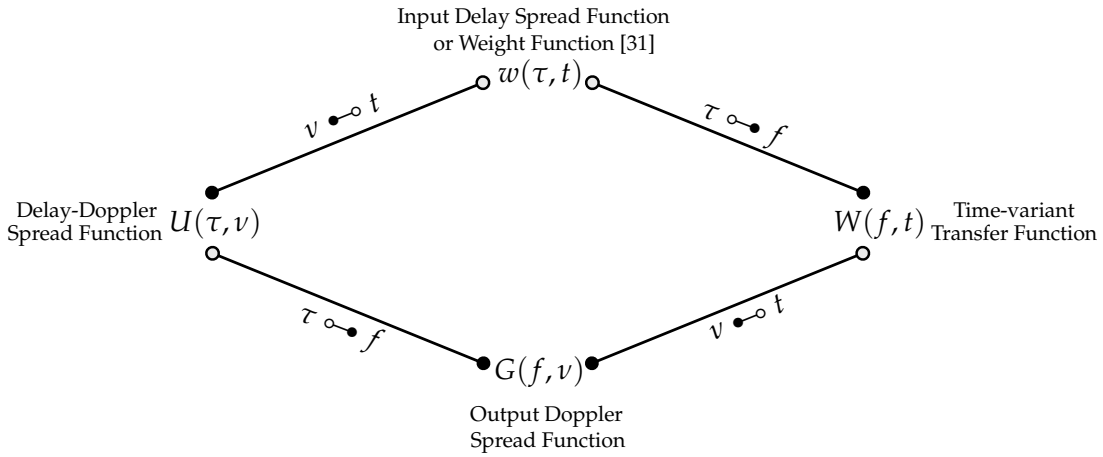


FIGURE 2.2: Relationship between kernel functions for describing a channel with respect to the FOURIER transform according to BELLO. $A \circ\!\!-\!\!\bullet B$ denotes the FOURIER transform where A represents the time domain and B represents the frequency domain, see Appendix B.1. Figure based on [8].

Delay-Doppler Spread Function Applying the FOURIER transform (see Appendix B.1) with respect to the observation time t on the weight function results in the *delay-Doppler spread function*

$$U(\tau, \nu) = \mathcal{F}\{w(\tau, t)\}_{\mathrm{d}t} \,. \tag{2.9}$$

This function is probably the most intuitive one of the system functions introduced in this section due to its dependency of the delay and the Doppler – two important parameters of MPCs as we know from (2.4).

Time-variant Transfer Function The *time-variant transfer function*, is achieved by applying the FOURIER transform with respect to the delay τ on the weight function:

$$W(f, t) = \mathcal{F}\{w(\tau, t)\}_{\mathrm{d}\tau} \,. \tag{2.10}$$

Output Doppler Spread Function Finally, applying the FOURIER transform with respect to the observation time t on the time-variant transfer function results in the *output Doppler spread function*:

$$G(f, \nu) = \mathcal{F}\{W(f, t)\}_{\mathrm{d}t} \,. \tag{2.11}$$

Alternatively, it is also possible to derive this function by applying the FOURIER transform with respect to the delay τ on the delay-Doppler spread function:

$$G(f, \nu) = \mathcal{F}\{U(\tau, \nu)\}_{\mathrm{d}\tau} \,. \tag{2.12}$$

The relationship between the weight function and the equations (2.9) to (2.12) is also illustrated in Figure 2.2.

2.2.2 Statistical Channel Description

Analysing the statistics of the deterministic channel description as described in the above section leads to the *statistical* channel description. The ACFs *(Autocorrelation Functions)* are of particular interest in this context. Taking the example of the delay-Doppler spread function as given in (2.9), the corresponding ACF is given by

$$R_{UU}(\tau,\tau',\nu,\nu') = \mathbb{E}\{\, U(\tau,\nu)\, U(\tau',\nu')^* \,\}, \tag{2.13}$$

where $\square^*$ denotes the conjugate complex of $\square$ and $\mathbb{E}\{\square\}$ denotes the expectation value of $\square$. Under the assumption of *weak stationarity* (also known as *wide-sense stationarity*) and under the assumption of *uncorrelated scattering*, the above equation and the so-called *scattering function* have the following relation [31], [82]:

$$\underbrace{R_{UU}(\tau,\tau',\nu,\nu')}_{(2.13)} = \underbrace{\delta(\nu-\nu')}_{\substack{\text{follows from}\\ \text{wide-sense}\\ \text{stationarity}}}\ \underbrace{\delta(\tau-\tau')}_{\substack{\text{follows from}\\ \text{uncorrelated}\\ \text{scattering}}}\ \underbrace{S(\tau,\nu)}_{\substack{\text{scattering}\\ \text{function}}}. \tag{2.14}$$

The above conditions that need to be fulfilled for the above simplification of $R_{UU}(\tau,\tau',\nu,\nu')$ are cumulated under the term WSSUS *(Wide-Sense Stationary Uncorrelated Scattering)*. According to [3], the WSSUS conditions – most notable the stationarity – are "never satisfied exactly" in practice. Thus, it is common practice to assume stationarity for a certain time period (and a certain frequency band of interest).

Another interesting tool to describe a channel is the PDP *(Power Delay Profile)* that can be derived from the scattering function [31] as defined above:

$$P(\tau) = \int_{-\nu_{\max}}^{\nu_{\max}} S(\tau,\nu)\, \mathrm{d}\nu, \tag{2.15}$$

where $\nu_{\max}$ denotes the maximum Doppler shift that is expected in the observed scenario – which is usually the case when the cosine in (2.2) is equal to 1.

The PDP shows the distribution of the received signal power along the delay axis. In Appendix C, practical time-discrete definitions of the PDP and the scattering function are discussed.

The PDP can be used as a base for the computation of two central measures that quantify important properties in wireless fading channels as explained in the following.

2.2.2.1 Delay Spread

Continuous RMS-DS The RMS *(Root Mean Square)* DS *(Delay Spread)* is a common and important property of a channel as it reflects the power distribution along the delay axis at a certain time instant in a single number. The higher the delay spread of a channel, the more relevant is the multipath propagation. Following the definition used in literature (e.g. in [82]),

the continuous RMS-DS is given by

$$\sigma_\tau = \sqrt{\frac{\int_0^\infty (\tau - \mu_\tau)^2 \, P(\tau) \, \mathrm{d}\tau}{\int_0^\infty P(\tau) \, \mathrm{d}\tau}}, \tag{2.16}$$

where the mean energy delay μ_τ is given by

$$\mu_\tau = \frac{\int_0^\infty \tau \, P(\tau) \, \mathrm{d}\tau}{\int_0^\infty P(\tau) \, \mathrm{d}\tau}. \tag{2.17}$$

The lower limit of the integrals in the above equations is explained by the fact that no MPC can arrive in the receiver before the actual signal has been transmitted ($T = 0$), i.e. the observed system is causal.

Discrete RMS-DS It is also possible to compute the RMS-DS using the MPCs that are present at a certain time instant, represented by the set $\mathcal{M}$. Combining the definition from literature, e.g. [52], with our definition of an MPC and its parameters (2.4), the discrete RMS-DS is given by:

$$\sigma_\tau = \sqrt{\frac{\sum_{\forall \xi_l \in \mathcal{M}} |\alpha_l|^2 \tau_l^2}{\sum_{\forall \xi_l \in \mathcal{M}} |\alpha_l|^2} - \mu_\tau^2}, \tag{2.18}$$

where, analogous to the continuous case, the mean energy delay μ_τ is given by

$$\mu_\tau = \frac{\sum_{\forall \xi_l \in \mathcal{M}} |\alpha_l|^2 \tau_l}{\sum_{\forall \xi_l \in \mathcal{M}} |\alpha_l|^2}. \tag{2.19}$$

2.2.2.2 K-Factor

The K-factor reflects the ratio of the power of the LOS component and the power of all other components in a channel with multipath propagation [80]. Using the notation introduced in Section 2.2.2.1, the (linear) K-factor at a certain time instant is given by

$$K = \frac{|\alpha_{\mathrm{LOS}}|^2}{\sum_{\forall \xi_l \in \mathcal{M} \setminus \{\xi_{\mathrm{LOS}}\}} |\alpha_l|^2}, \tag{2.20}$$

where α_{LOS} denotes the amplitude of the MPC identified as the LOS component and the sum in the denominator iterates over all (discrete) MPCs *except* for the MPC identified as the LOS component.

It is common practice to use the logarithmic representation of the K-factor, as this not only allows a convenient representation of larger ranges, but also indicates whether the LOS component is stronger (positive sign) or weaker (negative sign) than the sum of all other components.

The K-factor is also commonly used when modelling fading processes by a RICE distribution: As shown in Appendix E, this K-factor – sometimes also called "RICE factor" [31] – can be used to scale the RICE distribution according to the desired power ratio of the LOS component and the remaining multipath components within the modelled process.

2.3 Channel Modelling

Channel modelling is the attempt to describe the relevant effects that occur during the propagation of radio waves in a model. Depending on the model, it can later be used to develop and to evaluate a radio waveform that is designed for communication and/or ranging (which is the basis for localisation applications).

Many channel modelling approaches have been developed over time. Also, multiple approaches to categorise those channel modelling approaches can be found in literature, see e.g. [3], [15], [31], [82]. In the following, we adapt the main categorisation from [82] to classify selected sets of channel models. Additionally, their applicability on aeronautical AG modelling is assessed, including state-of-the-art examples.

2.3.1 Deterministic Channel Modelling

In [82], the term of *deterministic* channel modelling is used as a synonym for *physical* channel modelling, whereas the term of *phenomenological* channel modelling is used in [31]. In all cases, the idea is to derive the behaviour of the channel from the geometrical and physical conditions of a given scenario. This is possible since the propagation mechanisms of radio waves are quite well understood and can be expressed in mathematical formulae that can be solved numerically. However, this approach is subject to certain limitations that are common for numerical approaches: The model can only be as good as its input parameters – and those are usually just known up to a certain level of detail. Also, a numerical approach always comes with quantisation effects, and simplifications as those are required to limit the model's computational complexity. Eventually, such a deterministic model requires a detailed knowledge on the simulated environment (e.g. exact geometry, information reflection coefficients) [15].

The modelling approaches introduced here have in common, that the actual propagation of a radio wave is assumed to be modelled by rays, see Section 2.1.1. Analysing the propagation of radio waves by using the MAXWELL equations is usually not applied to larger setups and only used in antenna theory, a topic that is not investigated any further in this dissertation.

2.3.1.1 Basic Models

Besides the basic AWGN *(Additive White Gaussian Noise)* channel, the simplest thinkable channel model is the *space scenario* [31], thus a model without any obstacles and/or reflectors.

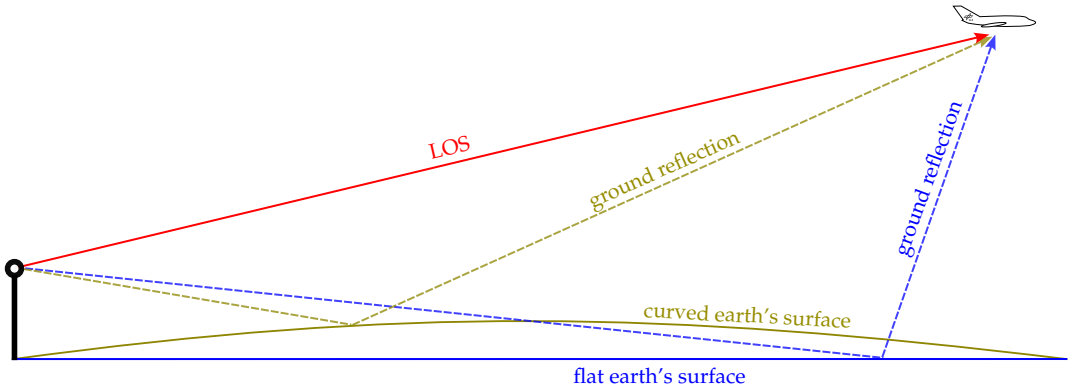


FIGURE 2.3: Sketch showing two classes of two-ray models as they are used in aeronautical AG communications, not true to scale: The flat-earth two-ray model and the curved-earth two-ray model. In both cases, one ray refers to the LOS component and one ray refers to the component caused by the ground reflection. The LOS component is given in red in the sketch. For the flat-earth two-ray model the ground reflection is given by a dashed blue line, for the curved-earth two-ray model the ground reflection is given by a dashed olive line. The assumed curvature of the earth's surface is given by a solid line of corresponding colour.

As such, only one single ray – the LOS component – is considered in this model. Only the FSPL *(Free Space Path Loss)*, the limited radio wave propagation speed (resulting in a delay), and the Doppler shift are considered.

The category of *two-ray* models take, as the name suggest, another ray into account. These models are commonly used in aeronautical AG channel modelling and cover both the LOS component and the signal component caused by the ground reflection[3]. Especially in case of the en-route scenario, these simple models can achieve a satisfying performance [51], [61], [82], since the LOS component as well as the ground reflection component are by far the most relevant components in this scenario.

As the sketch in Figure 2.3 shows, there are two relevant subclasses of two-ray models: Some assume a flat earth surface, e.g. [66], [67], more advanced models like the CE2R *(Curved-Earth Two-Ray)* model take the actual earth curvature into account, e.g. [51]. The article series [51], [53], [93] shows how the simple concept of a two-ray model can be expanded and, by taking environmental parameters like permeability into account, achieve results that match real-world AG measurements surprisingly well.

2.3.1.2 More Sophisticated Models

The more realistic a model tries to simulate the propagation of radio waves, the more complex it becomes: On the one hand, more rays need to be considered, on the other hand, more physical effects like those listed in Section 2.1.2 need to be taken into account. An approach where the geometric and physical properties of an environment and their effect on the radio wave propagation can be simulated in almost arbitrary detail is the GTD *(Geometrical Theory of Diffraction)* approach, where the radio waves are modelled as travelling rays – which explains the more popular name of *ray-tracing* [82]. It is usually applied to larger setups like indoor wireless or 5G/6G applications. Since these scenarios typically fulfil the conditions to

[3] In [66], it is suggested to call this ground reflection component the *specular component* to emphasise the fact it may consist of multiple non-resolvable components.

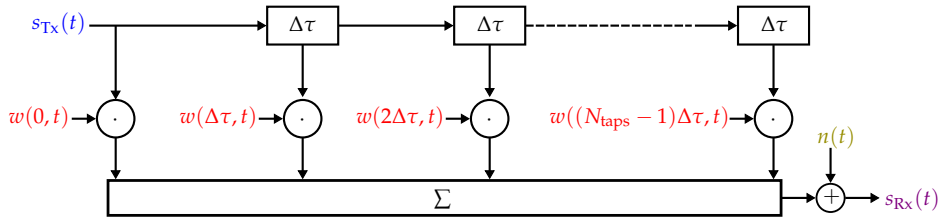


FIGURE 2.4: General case of a Tapped Delay Line with N_{taps} taps and an additional noise process: From Section 2.2.1.1, we know that it is more convenient to use the weight function $w(\tau, t)$ instead of the time-variant CIR $h(t', t)$ at the taps of the TDL. The reader may compare the colour coded elements of the figure to those given in Figure 2.1.

consider the radio wave propagation as rays, see Section 2.1.1, the concept of ray-tracing can be applied [110].

The technology of ray-tracing plays an important role in the field of computer graphics, where it has gained a lot of attention within the last years. This trend was enabled by the increased availability of powerful hardware (in terms of graphics cards) that allows ray-tracing-based rendering in real-time. Although video games are the main driver for the development of such powerful graphics cards, projects like SIONNA [32], that is developed by the well-known manufacturer of graphics cards NVIDIA, show that this hardware can also be used for ray-tracing in wireless channel modelling.

Despite some urban drone scenarios, there is virtually no literature on the application of true ray-tracing in the field of aeronautical AG channel modelling. The main reason for this circumstance is most likely the size of a typical geometry in this environment, as deterministic channel models are generally not applicable to "large scale simulations" [15]: The distances between transmitter, potential reflectors, and the receiver are much larger compared to many other wireless communication scenarios. This results in a considerable increase in the required amount of rays to achieve representative results and eventually in a tremendous increase in computational complexity.

2.3.2 Statistical Channel Modelling

In a statistical channel model, the simulated channel properties are derived from certain statistical distributions that are parametrised such that the desired channel properties are achieved. As such, it does not aim to provide a precise prediction of temporal and spatial information in a certain scenario, but it "provides information about mean values" [31].

A well-known channel model is the TDL *(Tapped Delay Line)* as sketched in Figure 2.4 where the weights of the taps are distributed according to a certain distribution. The RICE and the RAYLEIGH distribution are commonly used here to model RICIAN fading or RAYLEIGH fading, respectively, of the simulated channel [31].

An example for the application of statistical channel modelling in aviation is given in [30], where a set of simple statistical channel models for several air-ground scenarios in the VHF band is provided. Some of the approaches from this publication have been used and adapted to aeronautical C-band in [57].

2.3.3 Hybrid Channel Modelling

In [15], the term *hybrid* channel model is used to describe channel models that combine elements from both deterministic and statistical channel models. One subcategory of a hybrid model is a geometry-based stochastic model. A geometry-based stochastic model also tries to model the physical properties of a wireless channel and the radio wave propagation in it, however it implies simplifications compared to the pure deterministic models: A realistic modelling of the true environment is not necessary, instead, reflectors and scatterers are randomly distributed within the simulated scene based on certain statistical properties. In many cases, the actual CIR is then generated by the evaluation of a simplified ray-tracing approach [3].

Geometry-based stochastic models are more common in aeronautical AG channel modelling: In [82] an approach for such a channel model is introduced including a design suggestion for a regional airport. The described model is based on data from a measurement campaign in the aeronautical L-band. An approach based on an information-filter is applied to the measurement data to track MPCs. The findings from this tracking are then used to determine the properties of the statistical distributions that are used to locate the reflectors in a model realisation.

A geometry-based stochastic model for urban drone scenarios is introduced in [111]. However – in contrast to [82] where MPCs components are a central part of the model – this approach only discusses the LOS component and the probability of its occurrence.

The channel modelling approach proposed in this dissertation can also be considered as a hybrid channel model: As it will be shown in Chapter 7, the main focus is on the statistics; however, the findings from Section 6.3 – where the relationship between the geometry and the MPC delay and Doppler shift is discussed – are used to bring a geometrical aspect into the model.

Chapter 3

Measurement Campaign

The purpose of this chapter is to give the reader an understanding of how the flight measurement campaign was designed and performed. Less detailed descriptions of the campaign have been published in [58] and [61].

3.1 Motivation

Any modelling process, regardless of whether the resulting model is intended to be used in engineering, physics, economics, or social studies, requires a data set on which to base the model. For the modelling of radio wave propagation in aeronautical channels, the base of such a model is channel sounding data that is collected during several phases of a flight.

A model can only by as good, as the data it is based on.

3.1.1 Previous AG Measurement Campaigns in C-Band

Channel sounding campaigns for the C-band terrestrial ground-air channel have already been performed. The results of a large L- and C-band measurement campaign have been presented in [51], [53], [93], [94]: While [51] presents the general campaign setup and findings on the channel behaviour in over-water scenarios, [93] focuses on the wave propagation in hilly and mountainous terrain. Suburban and near-urban scenarios have been covered in [53]. Finally, the effect of airframe shadowing during flight has been investigated and modelled in [94].

3.1.2 Discussion of the Measured Channel

The motivation of channel sounding has been described in the previous chapter. But what is the *channel* that is actually sounded? Obviously, the *desired* channel that we are interested in is *only the wireless channel* between the transmitting and the receiving antennas. However, it is impossible to *only* measure this channel, since a number of hardware effects influence the signals on both the transmitting and the receiving side. An often applied strategy of minimizing the impact of these effects onto the final channel model is to perform reference measurements. They allow isolating and compensating these hardware effects by appropriate signal processing. Details on how this compensation is applied are given in Section 4.2.

3.2 Channel Sounding Signal

3.2.1 Center Frequency

Although the anticipated frequency band for a C2 link is 5,030 MHz to 5,091 MHz [36], the centre frequency f_c used in this campaign is 5,200 MHz. This decision was made due to hardware availability; however, no significant channel deviations are expected between these two bands.

3.2.2 Waveform Design

The waveform used in the campaign is a so-called multi-tone signal. Multi-tone signals are a popular type of waveform; technologies like OFDM are a well-known subclass of these type of signals. They are also popular in channel-sounding as the very diverse – in terms of envisioned application, applied scenarios, and used frequency band – channel sounding campaigns in [6], [45], [82], [95] show.

3.2.2.1 Peak-to-Average Power Ratio

The literature, e.g. [112], suggests to calculate the PAPR *(Peak-to-Average Power Ratio)* – usually given in dB – of a time-continuous signal $s(t)$ according to

$$\mathrm{PAPR}\{\boldsymbol{x}\}|_{\mathrm{dB}} = 10\log_{10}\left\{\frac{\max\{|s(t)|^2\}}{\mathbb{E}\{|s(t)|^2\}}\right\}, \tag{3.1}$$

where $\max\{\square\}$ denotes the maximum value of a function or sequence $\square$. The PAPR definition for discrete signals represented by a vector $\boldsymbol{x}$ is analogous to (3.1): The division of the vector's maximum power $\max\{|\boldsymbol{x}|^2\}$ and its average power $\mathbb{E}\{|\boldsymbol{x}|^2\}$ is converted to dB-scale.

Like all multi-carrier signals, multi-tone signals suffer from a (potentially) high PAPR. As the name suggests, the PAPR describes the ratio of the peak power of the signal and the average power of the signal. The peak power of a multi-tone signal is achieved once a significant subset of its sub-carriers are in-phase at the same time [31]. This effect becomes more likely and more intense, once the number of sub-carriers increases.

In communications, a signal with a high PAPR usually requires the HPA to operate at a significant back-off resulting in an imperfect operating point to avoid distortions or even hardware damage. The connection between the chosen operating point of the HPA and the effect of non-linear distortions is sketched in Figure 3.1: The figure shows the typical shape of an HPA's transfer function. In the linear region (highlighted in grey), the amplitude af the incoming signal (blue) is amplified without any non-linear distortions. It is therefore desired to chose the operating point of the HPA such that the signal's amplitude is not intruding the non-linear region of the HPA by adding a back-off with respect to the transition from the linear to the non-linear-region[1]. This is the case in part (a) of the figure: The amplified signal (red) shows no distortions.

[1] In practice, this position is usually given by the HPA's manufacturer as the GCP *(Gain Compression Point)*, usually denoted by $P_{1\mathrm{dB}-\mathrm{GCP}}$ or $P_{3\mathrm{dB}-\mathrm{GCP}}$, respectively

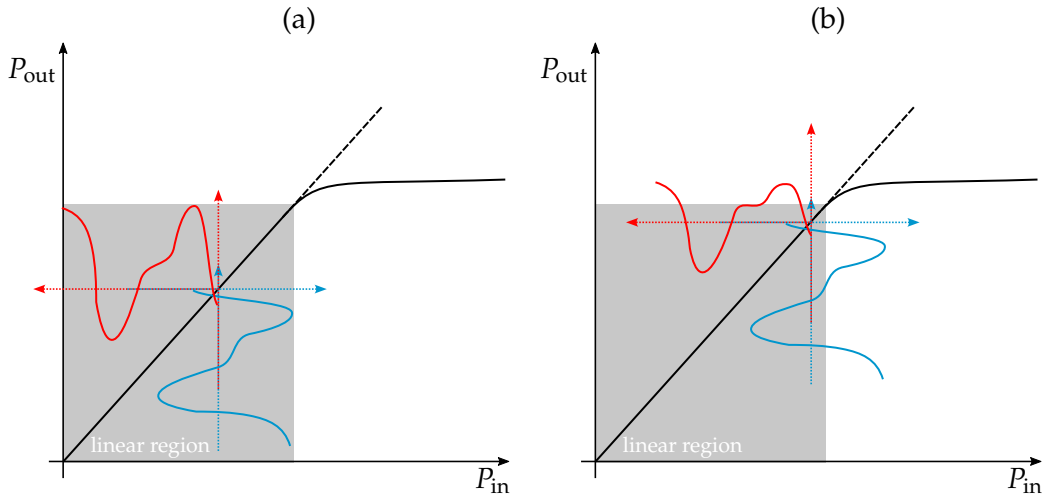


FIGURE 3.1: Sketch of the transfer function of a HPA *(High Power Amplifier)* and its effect on the amplified signal: The incoming signal is denoted in blue, the linear region is denoted by a grey background, the dashed black line shows the theoretical interpolation of the slope from its linear region. In (a), the operating point is chosen with a sufficient back-off such that the output signal (red) is linearly amplified and not distorted. In (b), the operating point is too close to the non-linear region of the slope – the peaks of the amplified signal are heavily distorted.

In (b), however, the back-off with respect to the beginning of the non-linear-region has been chosen too small: The amplitudes of the amplified signal (red) are distorted.

A signal having a low PAPR therefore requires a smaller back-off and allows a greater output power – which is desirable in the case of channel sounding, as the range over which the channel sounding signal can be received gets increased.

3.2.2.2 Waveform Generation

The (discrete) time-domain channel sounding signal represented by a vector $\boldsymbol{s}_{\text{CS}} \in \mathbb{C}^N$, where $N = 2{,}048$ denotes the length of the sequence given in samples, is generated in multiple steps:

1. First, a vector $\boldsymbol{S}_{\text{CS},0} = [S_0, S_1, \ldots, S_n, \ldots, S_{N-1}] \in \mathbb{C}^N$ is created. As we consider this vector to be given in frequency-domain, we use a capital symbol. The n-th element of this vector is given by:

$$S_n = \begin{cases} 0, & \text{if } \underbrace{n < N_{\text{G}}}_{\substack{\text{lower}\\ \text{guards}}} \vee \underbrace{n > N - N_{\text{G}}}_{\substack{\text{upper}\\ \text{guards}}} \vee \underbrace{n = N/2}_{\substack{\text{DC}\\ \text{carrier}}}, \\ \mathrm{e}^{\mathrm{j}\varphi_n}, & \text{else}, \end{cases} \tag{3.2}$$

   where $N_{\text{G}} = 5$ denotes the amount of guard carriers. These guard carriers ensure that the signal can be received without a degraded spectrum even in case of high Doppler shifts. Since we assume a symbol duration[2] of $T_{\text{symb}} = 20\,\text{ns}$, the sub-carrier spacing is given by $\Delta f = (NT_{\text{symb}})^{-1} \approx 24.41\,\text{kHz}$. Thus, the waveform provides guard bands of around $\Delta f N_{\text{G}} \approx 122.07\,\text{kHz}$.

[2] A symbol corresponds to one element of the vector that represents the respective signal.

The phases of the non-zero elements in (3.2) are distributed as NEWMAN Phases according to

$$\varphi_n = \pi \frac{(n - N/2 - 1)^2}{N}. \tag{3.3}$$

Signals having a phase distribution according to (3.3) show a low – in fact "very close to the minimal achievable" [11] – PAPR and outperform other approaches like RUDIN-SHAPIRO phases. Thus, they are commonly used in applications where a low PAPR is desirable – like channel sounding, see e.g. [45], [82].

2. The frequency domain vector gets transformed into time-domain using the IDFT *(Inverse Discrete Fourier Transform)*:

$$\boldsymbol{s}_{\mathrm{CS},0} = \mathrm{DFT}^{-1}\left\{\mathrm{DSHFT}^{-1}\left\{\boldsymbol{S}_{\mathrm{CS},0}\right\}\right\}. \tag{3.4}$$

The reader may note the application of the inverse DFT-shift as defined in Appendix B.2 *before* the IDFT is applied. This is required since the assignment of the elements of $\boldsymbol{S}_{\mathrm{CS},0}$ in (3.2) is performed with the zero-frequency at $n = N/2$, whereas the IDFT expects the zero-frequency at $n = 0$.

3. To reduce the PAPR of the signal even further, an approach based on ICF *(Iterative Clipping and Filtering)* is applied. ICF is one possible strategy commonly used for the PAPR reduction of multi-tone signals like OFDM signals [105], [112]. During each iteration of the loop – indexed by $k \in \mathbb{N}$ – the current signal's peak value is scaled according to

$$\begin{aligned} \boldsymbol{s}'_{\mathrm{CS},k} &= \boldsymbol{s}_{\mathrm{CS},k-1} \\ n_{k,\max} &= \underset{n\in\{0,\ldots,N-1\}}{\operatorname{argmax}} \left\{|\boldsymbol{s}'_{\mathrm{CS},k}[n]|\right\} \\ \boldsymbol{s}'_{\mathrm{CS},k}[n_{k,\max}] &= a\, \boldsymbol{s}'_{\mathrm{CS},k}[n_{k,\max}], \quad \text{with } a \in \mathbb{R}, 0 < a < 1. \end{aligned} \tag{3.5}$$

An alternative notation for the addressing of vector elements was used in the above equation for the sake of clearness: $\boldsymbol{x}[n]$ corresponds to the n-th element of $\boldsymbol{x}$, thus x_n. During the generation of the channel sounding signal, the scaling factor was set to $a = 0.95$.

In a next step, $\boldsymbol{s}'_{\mathrm{CS},k}$ is filtered using an LP *(Low-Pass)* filter such that the guard carriers are set to zero. This operation corresponds to an element-wise multiplication with a BOXCAR window in discrete frequency-domain and is therefore straightforward to implement:

$$\begin{aligned} \boldsymbol{s}_{\mathrm{CS},k} &= \mathrm{DFT}^{-1}\left\{\mathrm{DFT}\{\boldsymbol{s}'_{\mathrm{CS},k}\} \cdot \mathrm{DSHFT}^{-1}\left\{\boldsymbol{H}_{\mathrm{BC}}\right\}\right\}, \\ \text{with } \boldsymbol{H}_{\mathrm{BC}} &= [H_{\mathrm{BC},0}, H_{\mathrm{BC},1}, \ldots, H_{\mathrm{BC},n}, \ldots H_{\mathrm{BC},N-1}], \\ \text{where } H_{\mathrm{BC},n} &= \begin{cases} 0, & \text{if } n < N_{\mathrm{G}} \vee n > N - N_{\mathrm{G}} \\ 1, & \text{else.} \end{cases} \end{aligned} \tag{3.6}$$

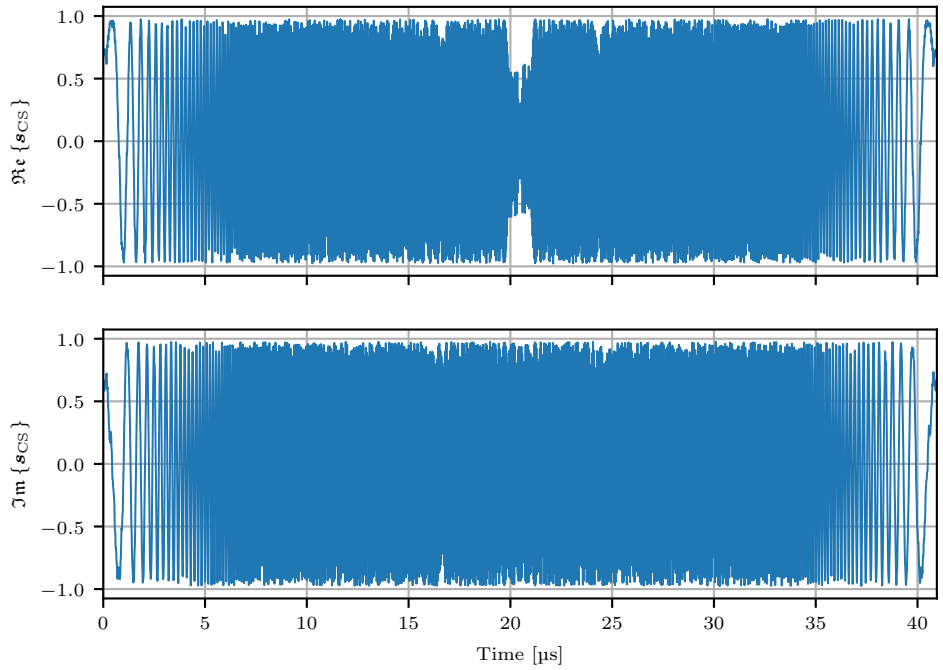


FIGURE 3.2: Real and imaginary part of the generated channel sounding signal s_{CS} in time domain. No obvious signal peaks can be identified that would increase the PAPR.

TABLE 3.1: Properties of the channel sounding sequence and the channel sounding signal.

Description	Symbol	Value	Unit
Symbol Duration	T_{symb}	20	ns
Signal Bandwidth	W	49.78	MHz
Sequence Length	N	2,048	Smpl
Sequence Length	T_{CSS}	40.96	µs
Carrier Frequency	f_c	5.2	GHz
Wave Length	λ_c	5.764	cm
Spacial Resolution	Δr_{min}	5.994	m

The ICF loop is terminated after $K = 80{,}000$ iterations, as no more significant improvement in PAPR reduction could be observed.

4. When the loop has finished, the channel sounding signal has been finalised: $\boldsymbol{s}_{CS} = \boldsymbol{s}_{CS,K}$. The plots in Figure 3.2 show the real and the imaginary part of $\boldsymbol{s}_{CS}$ in time domain.

5. In a last step, the signal gets upsampled by a factor of $f_{up} = 2$; we denote this upsampled signal by $\boldsymbol{s}_{CS,up} = \mathrm{UPS}_{f_{up}}\{\boldsymbol{s}_{CS}\}$. The spectrum of $\boldsymbol{s}_{CS,up}$ computed using the DFT *(Discrete Fourier Transform)* as given in Appendix B.2 is given in Figure 3.3.

Using (3.1), the PAPR of the discrete channel sounding signal – i.e. the passband signal – has been determined to be around 0.285 dB. The reader may compare this comparatively low PAPR value with the time domain plot of the signal in Figure 3.2.

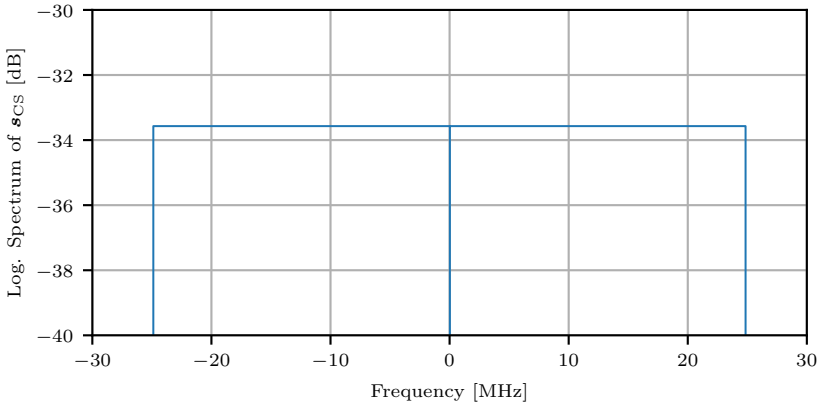


FIGURE 3.3: Logarithmic spectrum of the perfect channel sounding (baseband) signal $\boldsymbol{s}_{\mathrm{CS,up}}$. The spectrum has almost the shape of a BOXCAR window.

3.2.2.3 Time-Continuous Signals

The conversion of the time-discrete signal to the time-continuous baseband signal $s_{\mathrm{CS}}(t)$ can be described by the WHITTAKER-SHANNON interpolation formula [31] according to

$$s_{\mathrm{CS}}(t) = \sum_{n=-\infty}^{\infty} \boldsymbol{s}_{\mathrm{CS,up}}[\underset{Nf_{\mathrm{up}}}{\mathrm{mod}}\{n\}]\; g_{\mathrm{Tx}}\left(\frac{t - nT'_{\mathrm{symb}}}{T'_{\mathrm{symb}}}\right) \tag{3.7}$$

where $\underset{D}{\mathrm{mod}}\{a\}$ denotes the modulo operation on a to basis D and $g_{\mathrm{Tx}}(t)$ denotes the impulse response of the transmission filter, i.e. the characteristics of the AWG *(Arbitrary Waveform Generator)*'s DAC *(Digital-to-Analog Converter)* which corresponds to the normalised sinc-function $\mathrm{sinc}\, x = \frac{\sin \pi x}{\pi x}$ in the ideal case[3].

The time-discrete signal $\boldsymbol{s}_{\mathrm{CS,up}}$ is replayed in an continuous loop on the AWG during the campaign; this is represented by the limits of the sum in (3.7). The symbol duration of the upsampled signal used here is set to $T'_{\mathrm{symb}} = T_{\mathrm{symb}}/f_{\mathrm{up}}$ such that the resulting bandwidth of the generated signal is the same as the bandwidth of $\boldsymbol{s}_{\mathrm{CS},}$. The purpose of this oversampling of the discrete signal before playing it on the AWG is to avoid possible signal distortions caused by the internal resampling filter of the AWG. This is the procedure recommended by the manufacturer of the AWG [74].

The baseband signal $s_{\mathrm{CS}}(t)$ eventually gets modulated onto a carrier signal with frequency $f_{\mathrm{c}} = 5.2\,\mathrm{GHz}$ and is then transmitted. We denote this passband signal by $s_{\mathrm{Tx}}(t)$.

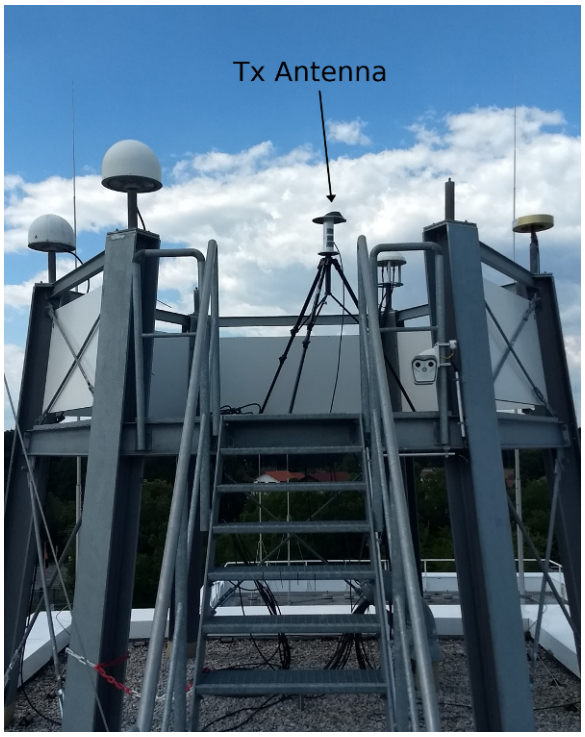
All parameters of the channel sounding sequence and the transmitted passband signal, respectively, are given in Table 3.1.

3.3 Ground Station Hardware Setup

The campaign involved one single ground-based transmitting station, that will be called GS *(Ground Station)* in the following. Since the reference measurement procedure described in Section 3.5 requires the operation of the GS at two different locations, the GS was designed

[3]The sinc-function is also referred to as the si-function in German literature [31].

(A) Ground Station



(B) Transmitting Antenna

FIGURE 3.4: (A) The portable Ground Station. (B) Position of the transmitting antenna on the rooftop of the building of IKN *(Institut für Kommunikation und Navigation)*. The antenna's altitude is $h_{\mathrm{Tx}} \approx 651.2\,\mathrm{m}$ given in the WGS84 *(World Geodetic System 1984)* system.

as a portable platform as shown in Figure 3.4a. The individual devices of the GS and the signal flow between these devices are presented in Figure 3.5. In the following, the involved devices are explained in more detail.

3.3.1 Rb *(Rubidium)*-Clock and GNSS Receiver

The GS consists of an atomic clock[4] having an Rb *(Rubidium)* oscillator and a GNSS *(Global Navigation Satellite System)* receiver[5]. Together, these two devices form a GNSS-disciplined oscillator and build the time base of the GS. This is a common approach to benefit from both the short-term clock-stability of Rb-clocks and the long-term clock-stability of the GNSS system at the same time. GNSS data is logged during all measurements and converted into the common RINEX data format afterwards.

3.3.2 Arbitrary Waveform Generator

The pre-generated channel sounding signal $s_{\mathrm{CS,up}}$ is loaded into the Arbitrary Waveform Generator[6] and played at 100 MHz in an infinite loop during measurements as described in (3.7). The AWG uses the 10 MHz reference signal of the Rb-clock as a time base and outputs the channel sounding signal with an average power of $-5.5\,\mathrm{dBm}$.

[4] SPECTRATIME LNRCLOK-1500
[5] JAVAD Delta3
[6] ROHDE & SCHWARZ SMBV100

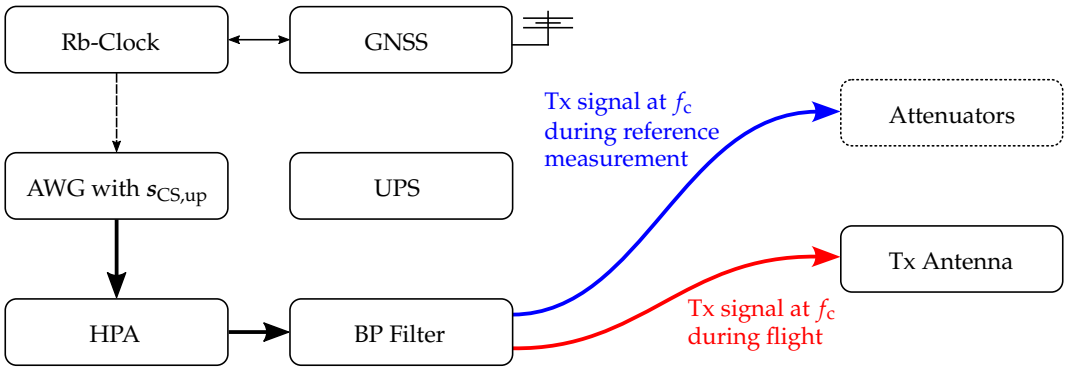


FIGURE 3.5: Block diagram of the Ground Station. Bold arrows denote the flow of the channel sounding signal, dashed arrows denote the 10 MHz reference signal, other arrows denote data lines. As the attenuators are only used for the reference measurement, see Section 3.5, they are not considered as an actual part of the GS and are given with a dotted border. The remaining devices are explained in detail in Section 3.3.

3.3.3 High Power Amplifier and Band-Pass Filter

The HPA[7] amplifies the AWG's output signal by an average gain of 53.5 dB. An additional BP *(Band-Pass)* filter (center frequency at 5.2 GHz, pass-band bandwidth of ~400 MHz) is connected to the output of the HPA to reduce out-of-band radiation. Due to the losses implemented by the BP filter, cables, and connectors, the average output power of the channel sounding signal coming out of the filter is $P_{\text{out}} \approx 47.5\,\text{dBm}$.

According to the manufacturer, the HPA's 1 dB GCP is at $P_{\text{1dB-GCP}} = 49.5\,\text{dBm}$. Considering the low PAPR of the channel sounding signal, the resulting back-off is assumed to be sufficient to avoid non-linear distortions of the signal.

3.3.4 Uninterruptible Power Supply

All active devices of the GS, except for the HPA, are connected to the UPS *(Uninterruptible Power Supply)*[8] to allow a mobile operation of the GS for about 12 min.

3.3.5 Transmitting Antenna

During the measurement flights, a transmitting antenna was used having an average gain of $G_{\text{Tx}} \approx 5\,\text{dBi}$. The antenna characteristics are part of the campaign results. Vertical polarization is used, as this is the common polarization in other aeronautical air-ground communication systems [86], [2].

3.4 Air Station Hardware Setup

Besides the transmitting GS, the campaign involved one single receiver located onboard of a DASSAULT FALCON 20E aircraft owned and operated by the DLR *(German Aerospace Center)*. The aircraft is equipped with a pressurized cabin and is powered by two jet turbines. These

[7] MICROWAVE AMPS AM60 Series
[8] EATON 9130

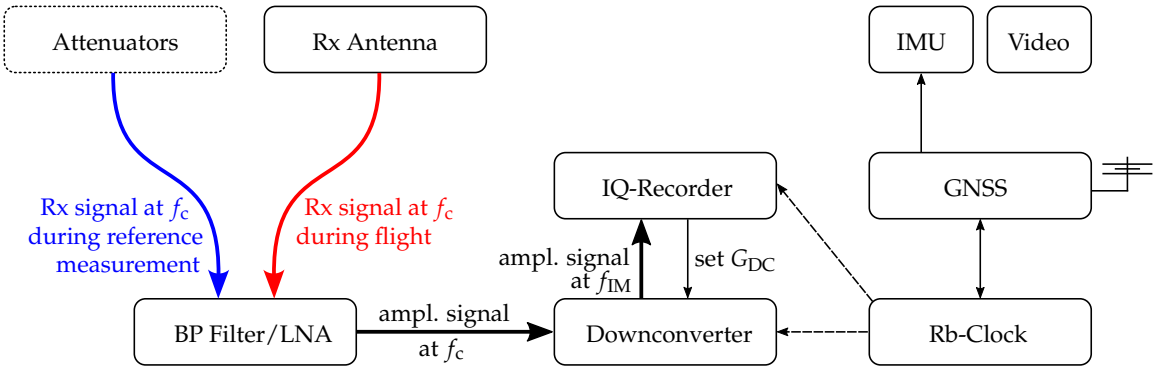


FIGURE 3.6: Block diagram of the Air Station. Bold arrows denote the flow of the channel sounding signal, dashed arrows denote the 10 MHz reference signal, other arrows denote data lines. As the attenuators are only used for the reference measurement, see Section 3.5, they are not considered as an actual part of the AS and are given with a dotted border. The remaining devices are explained in detail in Section 3.4.

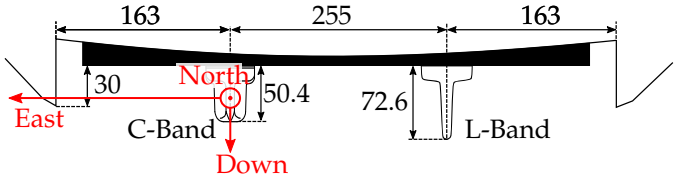


FIGURE 3.7: Front view sketch of the receiving antenna mounted at the bottom of the aircraft, not true to scale. The C-Band antenna is used to receive the channel sounding data, the L-Band antenna is used for another mission. All measurements are given in mm. The axes of the receiver-centred NED *(North-East-Down)* coordinate system are marked in red.

features allow flight profiles equivalent to those of conventional aircraft in terms of velocity and flight altitude.

The setup of this AS *(Air Station)*, as it is called in the following, is explained in this section.

3.4.1 Rb-Clock and GNSS Receiver

Similar to the GS, these devices form a GNSS-disciplined oscillator to provide a stable clock source through a 10 MHz reference signal. GNSS data is logged during all measurements and converted into the common RINEX data format afterwards.

3.4.2 Receiving Antenna

A vertical polarized receiving antenna[9] is used. According to the manufacturer's data-sheet, it provides an omnidirectional antenna pattern with an antenna gain of $G_{Rx} \approx 4$ dBi. A more precise description of the receiving antenna's pattern is not available. The receiving antenna is mounted at the bottom of the aircraft as shown in Figures 3.7 and 3.8. This location is determined by the architecture of the research aircraft.

3.4.3 BP Filter and Low Noise Amplifier

The incoming signal is filtered and amplified by a two-stage LNA *(Low Noise Amplifier)* and corresponding BP filters. The overall gain including cable and filter losses is ~14.1 dB.

[9] ANTCOM 2B-C-BANDV-XT-2 Rev A

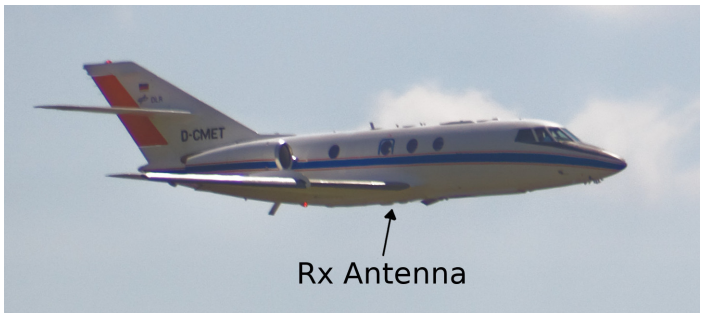


FIGURE 3.8: Position of the receiving antenna on the bottom of the FALCON aircraft.

3.4.4 Downconverter

Since the RF *(Radio Frequency)* frontend[10] of the device to record the received signal only supports frequencies up to 3.6 GHz, the incoming signal at $f_c = 5.2\,\text{GHz}$ must be shifted to the appropriate frequency range using a mixer. The frequency the signal is shifted to is called intermediate frequency and is set to $f_{IM} = 80\,\text{MHz}$ in the given setup. The mixer is tuned by the 10 MHz signal provided by the GNSS-disciplined oscillator.

In fact, this whole down-conversion process involves more than just a mixer, but also multiple filters, an LNA, and a variable attenuator. We will summarise these devices and refer to them as downconverter. The total gain of the downconverter is denoted by G_{DC} and can be set manually by the operator of the AS using the LABVIEW software running on the IQ-recorder. Providing a variable gain for the signal amplification extends the dynamic range of the AS.

3.4.5 IQ-Recorder

In the IQ-recorder's[11] RF frontend, the signal is further downmixed in a multistage mixer setup. Eventually, the IQ-recorder stores the baseband signal at a sample rate of $f_{sr} = 50\,\text{MHz}$. The device stores the raw values of the ADC *(Analog-to-Digital Converter)* which provides a resolution of 14 bit for each the real and the imaginary part.

Sets of 98,304 kSmpl – corresponding to 48,000 complete instances of the channel sounding sequence – are stored in a binary file, thus one binary file contains exactly 1,966.08 ms of the recorded signal. The filename of the binary file contains an auto-incremented integer called *bin_id*, which allows the file to be uniquely identified. Each binary file contains a header including the current GNSS-timestamp, the current value of the IQ-recorder's sample counter ρ, and the current value of the gain control of the downconverter G_{DC}.

During recording, the ADC's data stream of $2 \times 14\,\text{bit} \times f_{sr} \approx 175\,\text{MB}\,\text{s}^{-1}$ needs to be stored. This high data rate motivates the usage of an external hardware RAID *(Redundant Array of Independent Disks)*[12]. In order to achieve maximum write speeds, a RAID0 configuration is used. For the sake of clearness, the RAID is considered as part of the "IQ-Recorder" block in Figure 3.6.

[10] NATIONAL INSTRUMENTS PXIe-5667
[11] NATIONAL INSTRUMENTS PXI Series running WINDOWS 7 with LABVIEW
[12] NATIONAL INSTRUMENTS HDD8265

Similar to the downconverter, the IQ-recorder also uses the 10 MHz signal of the GNSS-disciplined oscillator as a time reference. It is used to tune the radio frontend and as a clock basis the internal sampling process which ensures a stable counting of the recorded samples.

3.4.6 Video and IMU

An IMU *(Inertial Measurement Unit)*[13] is used to record data on the orientation and three-dimensional acceleration of the aircraft. All data is logged including GNSS-timestamps. For documentation, a video camera is filming all flights through the window over the left wing of the Falcon aircraft.

3.5 Measurement Procedure

In the beginning of this chapter it was discussed that it is impossible to just measure the actual wireless channel, but that hardware effects like the transfer function or the group delay, are always part of the measurements. However, it is possible to compensate parts of these hardware effects by performing a reference measurement. Only the effects of hardware that is involved in *both* the reference measurement *and* the actual channel measurement can be compensated. This condition is fulfilled in the given setup for *all* hardware, *except* for:

- The Tx antenna of the GS, as it cannot be part of the reference measurement but is obviously essential for the actual channel sounding.
- The Rx antenna of the AS, for the same reason.
- The cascade of attenuators that is required for the reference measurement to reduce the power of the channel sounding signal such that the receiver is not damaged. A reduction of the transmission power would have spoiled the measurement result, as the transfer function of a device like an HPA is likely to depend on the signal power.
- The cable connecting the GS' BP filter output with the Tx antenna. For structural reasons, a permanent installation of this cable was necessary in the transmitting location and a different cable had to be used during the reference measurement to connect the GS' BP filter output and the attenuators. However, both cables had the same length and were taken from the same batch of the same manufacturer. Measurements showed, that the relevant parameters of these two cables are practically identical.

Reference measurements were performed before and after each flight. Thus, each measurement flight consisted of the following steps:

3.5.1 Pre-Flight Reference Measurement

3.5.1.1 State

The aircraft is on the apron with active APU *(Auxiliary Power Unit)*. The receiver station onboard the aircraft runs on the aircraft's APU power; the Rb-clock is heated up, the GNSS

[13] XSENS MTi-100 Series

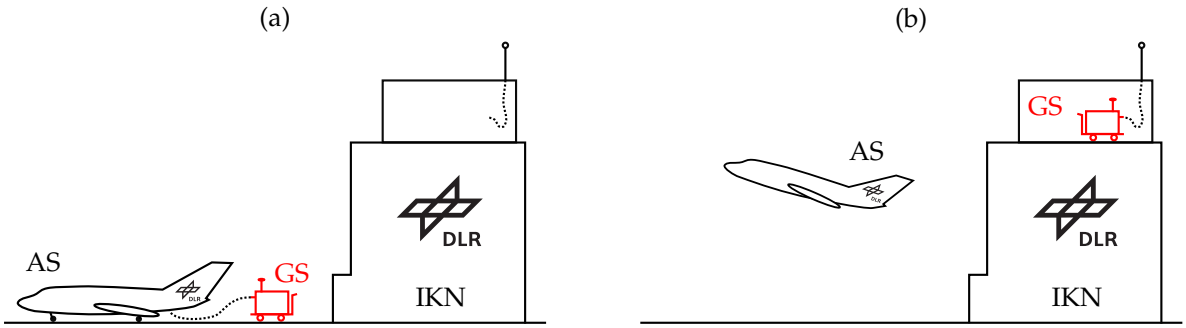


FIGURE 3.9: Schema of the two locations of the GS *(Ground Station)* marked in red: (a) The GS is located next to the AS *(Air Station)* on the apron for the reference measurement before and after each flight. (b) The GS is located in its transmission location on top of the IKN building providing a similar setting as if it was located on an airport tower. The AS is moving – either in the air or on the ground.

receiver has a valid fix, and IQ-sample recording is prepared but paused. The recording of GNSS and IMU data has been started.

The GS is located next to the aircraft on the apron as shown in Figure 3.9(a) and is connected to the local power supply network. The GS' Rb-clock is heated up and its GNSS receiver has a valid fix.

The filter output of the GS is connected to the antenna cable of the receiving antenna using a cascade of attenuators, as it is indicated by the blue arrows in Figure 3.5 and Figure 3.6, respectively. The attenuators provide an overall attenuation of $A_{\mathrm{dB,ref}} = 90\,\mathrm{dB}$, corresponding to an FSPL of $\sim$0.15 km for the given carrier frequency f_c.

3.5.1.2 Action

First, the Rb-clocks of the GS and the AS are synchronized according to the manufacturer's specifications. Then, the actual pre-flight reference measurement is performed: The GS starts transmitting the channel sounding signal, the AS starts the IQ-sample recording for about 10 s.

3.5.2 Preparation of Air Station and Relocation of Ground Station

3.5.2.1 State

The cable between the GS and the AS is removed, the Rx antenna is connected to the BP filter as indicated by the red arrow in Figure 3.6. The IQ-recorder is on idle: Its internal sample counter is running, but no data is recorded.

The GS is not connected to the power supply network; its AWG, Rb-clock, and GNSS receiver are now running on UPS power.

3.5.2.2 Action

The GS is moved from the apron into its transmission location on the rooftop of the institute's building. Once the transmitting location is reached as depicted in Figure 3.9(b), the station is connected to the local power supply network; it is waited until the GS' GNSS receiver has a valid fix. The filter output is connected to the cable of the Tx antenna as indicated by the red

arrow in Figure 3.5 and the HPA is switched on: The transmission of the channel sounding signal is started.

The aircraft is starting its engines; once they are running, the power supply of the devices of the AS is *seamlessly* switched from APU power to the power generated by the aircraft engines.

3.5.3 Actual Channel Measurement

3.5.3.1 State

The GS is in its transmission location and is emitting the channel sounding sequence. In the beginning, the aircraft is still in parking position on the apron but ready for take-off. It then approaches the runway, takes-off, follows the dedicated flight pattern, lands, and returns to its parking position.

3.5.3.2 Action

The IQ-sample recording is started once the aircraft starts taxiing towards the runway. The operator of the AS observes the received power of the channel sounding signal and adjusts G_{DC} accordingly during taxi and flight. The recording of IQ-samples is stopped once the aircraft has returned to its parking position. The IQ-Recorder, however, remains in idle mode so that its sample counter continues to count.

3.5.4 Relocation of Ground Station and Post-Flight Reference Measurement

3.5.4.1 State

The GS has terminated its transmission, the HPA is switched off. The GS is disconnected from the power supply network; again, the GNSS-receiver, the Rb-clock, and the AWG are running on the UPS.

The aircraft has seamlessly switched the power supply of the AS from turbines to the APU.

3.5.4.2 Action

The GS is moved to the apron where the aircraft is waiting. The setup as sketched in Figure 3.9(a) is restored and a reference measurement is performed as described in Section 3.5.1.

Once this post-flight reference measurement is finished, both the AS and GS stop the recording of IMU and GNSS data and are shut down.

3.6 Flight Overview

The campaign took place in July 2018 and consisted of four flights. All flights started and ended at the EDMO airport in Oberpfaffenhofen, Germany, thus close to the location of the GS. While the second flight was a long range flight with a LOS distance of up to 600 km, the remaining flights took place in the area around EDMO (< 60 km). The flight routes of all flights are plotted in Figures 3.10 and 3.11; the flight altitude (above mean sea level) is colour coded. Flight dates and flight durations are presented in Table 3.2.

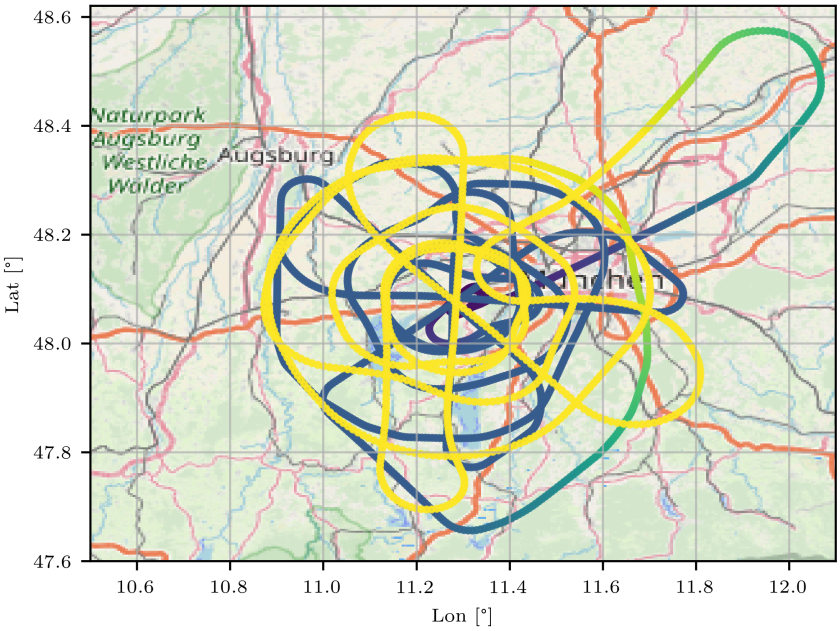
TABLE 3.2: Basic Flight Statistics

#	Date	Duration	# sequences
I	2018/07/09	3:25h	617.4×10^9
II	2018/07/10	2:10h	379.8×10^9
III	2018/07/10	3:10h	572.2×10^9
IV	2018/07/12	1:35h	281.8×10^9

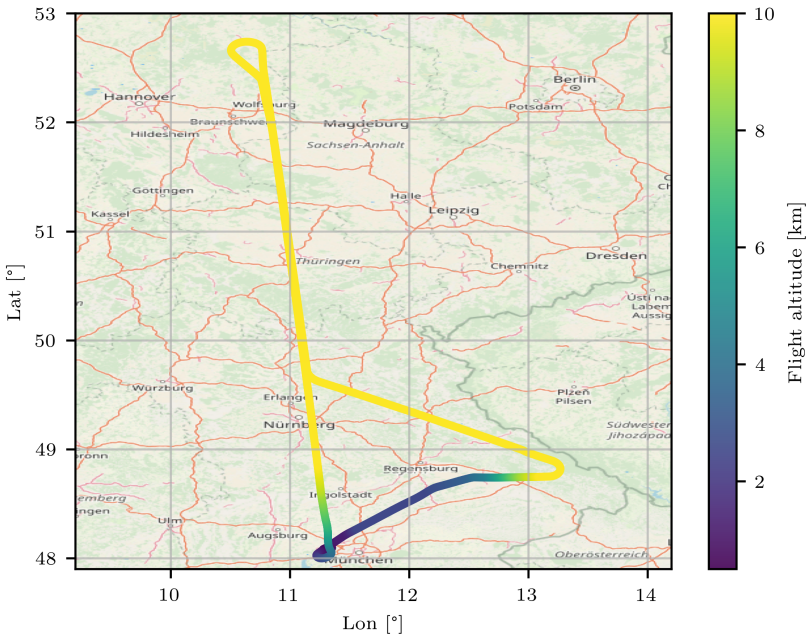
All flights ended with a few go-arounds before the actual landing. In aviation, a go-around is initiated when a landing is aborted[14]. A go-around may happen during final approach "whenever landing conditions are not satisfactory" [1]. The go-around scenario, especially the part where the aircraft is above the runway, is dominated by strong multipath components besides the LOS path together with a comparatively high speed of the aircraft resulting in rapidly changing reflections, Doppler shifts, and Doppler spreads.

Furthermore, manoeuvres with rolling angles up to $\pm 50^\circ$ were flown to investigate the effect of airframe shadowing during banking.

[14] "Go-around" is sometimes mistakenly used equivalently to "missed approach", although the latter term describes an aborted instrument approach.

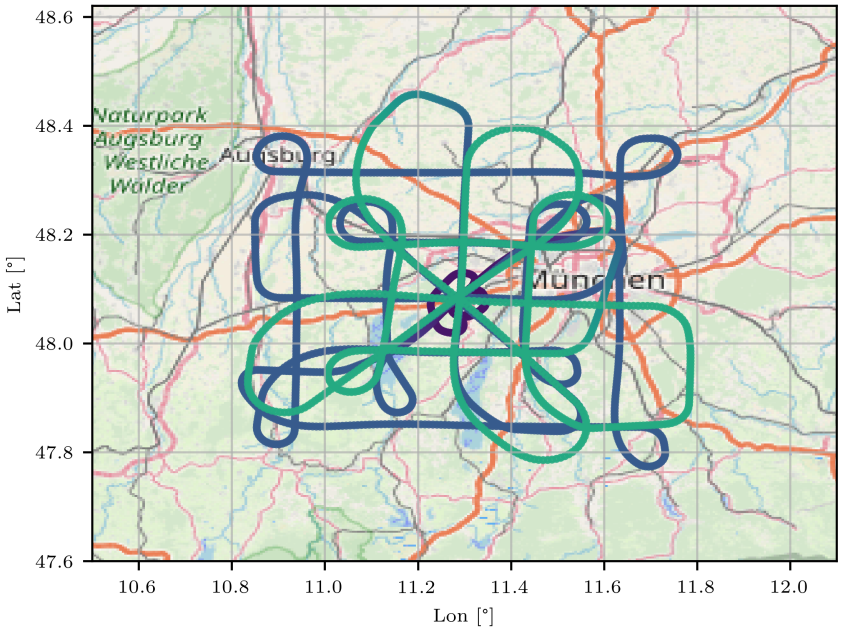


(A) Flight I

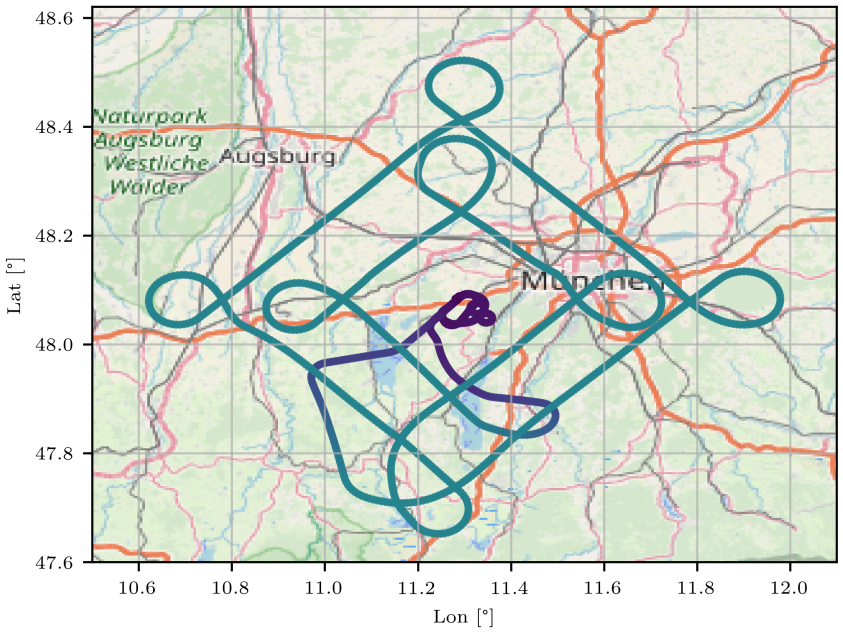


(B) Flight II

FIGURE 3.10: Flight routes of the first two flights. The flight altitude is colour coded; the colour key for all maps is given in Figure 3.10b. Except for flight II, the routes were located in the area around the transmitter located at the EDMO airport in Oberpfaffenhofen, west of Munich. Background image: ©OpenStreetMap, https://openstreetmap.org/copyright



(A) Flight III



(B) Flight IV

FIGURE 3.11: Flight routes of the remaining two flights. The flight altitude is colour coded; the colour key for all maps is given in Figure 3.10b. Background image: ©OpenStreetMap, https://openstreetmap.org/copyright

Chapter 4

Ground Truth and Pre-Processing

4.1 Ground Truth Data

Whenever scientists perform measurements, they try to reduce the amount of unknowns. Ideally, the remaining unknowns correspond to the variables and effects that motivated the measurement in the first place. An example for a measurement environment where most of these variables are known is an anechoic chamber as it used for antenna measurements: All relative and absolute distances are known, as well as the architecture, the location and orientation of potential reflectors, temperature, humidity, and the composition of the air. Taking the knowledge on the environmental conditions into account increases the precision and the generality of the measurement results: It becomes possible to distinguish whether an observation is caused by the environmental conditions or if it is actually an effect of the device under test.

In the scenario of a flight measurement campaign, however, these environmental conditions are usually neither controllable nor completely measurable. Nevertheless, it is at least possible to estimate and track the location of the transmitter and the receiver over time using GNSS *(Global Navigation Satellite System)*. The same applies to the orientation of these entities using an IMU *(Inertial Measurement Unit)*. We call this estimation of an entity's position and orientation based on side information like GNSS and IMU data at a certain time instant, as well as other information closely related to that, the GT *(Ground Truth)*. The GT is an important component of the presented data processing chain, as the knowledge on position and orientation is necessary to determine if a LOS connection was available at a certain time instant and, if so, what the LOS distance was.

4.1.1 Raw Data Sources

The following sections describe all data sources that are used to compute the actual GT data. This list includes both constant and time-varying data.

4.1.1.1 GNSS Data of Air Station

As presented in Section 3.4, a high-performance GNSS *(Global Navigation Satellite System)* receiver was used in the AS that recorded the AS' timestamped position in time-discrete domain to a RINEX *(Receiver Independent Exchange Format)* file. The receiver evaluated the signals

of GPS *(Global Positioning System)*[1], GALILEO[2], and GLONASS *(Global'naya Navigatsionnaya Sputnikovaya Sistema)*[3]. All collected GNSS data was later processed by the PPP *(Precise Point Positioning)* service of the Geodetic Survey of Natural Resources Canada [98] to determine time drifts and compensate inaccuracies caused by atmospheric effects.

4.1.1.2 IMU Data of Air Station

An IMU *(Inertial Measurement Unit)* is a device to measure physical forces. It is therefore possible to determine accelerations and, based on the Earth's gravitation, an object's orientation. As shown in Section 3.4, an IMU, that records its timestamped current orientation, was also part of the setup onboard the aircraft. The IMU contains its own GNSS receiver that is used as a source for reliable timing information in the given setup. All IMU data was recorded in the time-discrete domain and is later used to determine the aircraft's roll, pitch, and yaw angles.

4.1.1.3 Antenna Offset in Air Station

The GNSS receiver in the aircraft determines the current position of the GNSS antenna that is attached to the GNSS receiver. However, the current position of the C-band antenna is of interest in the later processing; it is therefore necessary to compensate the offset between these two antennas. This offset is determined by evaluating the plans of the aircraft used in the campaign. While fixing this offset is trivial in receiver-centred coordinate systems, it becomes more complex in Earth-centred or transmitter-centred coordinate systems as the current roll, pitch, and yaw angles of the aircraft come into account. This antenna correction is performed as suggested in [82] and applied to the GT data where necessary.

In the following, we always refer to the actual position of the receiving C-band antenna onboard the aircraft whenever the position of the receiver or aircraft is mentioned, if not denoted otherwise.

4.1.1.4 GNSS Data of Ground Station

The GNSS receiver of the GS – see Section 3.3 – was used in two ways:

- Before the actual flight, the later position of the transmitting C-band antenna was determined by installing the GNSS antenna in the exact same location[4] the transmitting C-band antenna will be installed during the measurements. This positioning measurement was kept running for about 20 min; the results were processed by the PPP and averaged over time.
- During the flight measurements, the GS' GNSS receiver was also recording positioning and timing information, similar to the AS' GNSS receiver. All collected GNSS data was

[1] Probably the most prominent GNSS, operated by the United States.
[2] The GNSS developed and operated by the European Union.
[3] The GNSS developed and operated by the Soviet Union and the Russian Federation, respectively.
[4] The reference points of the two different antennas are actually their individual phase centres.

again processed by PPP. Please note that the measured positioning information is not used for the GT computations, as only the timing information is relevant here.

4.1.2 Synchronisation and Sampling

Both the recorded GNSS data and the recorded IMU data, as well as the samples recorded by the IQ-recorder, are stored with an absolute timestamp derived from the current GPS time. It is therefore possible to synchronise this data based on this timing information. However, both the GNSS and the IMU data were recorded in the time-discrete domain. The sampling rates and the actual sampling timestamps differ for all devices involved. A cubic spline interpolation is used to estimate the GNSS data and the IMU data, respectively, for an arbitrary time instant within each measurement flight. It is therefore possible to determine an estimate of the transmitter's and the receiver's current location and orientation for each IQ sample recorded by the IQ-recorder.

4.1.3 Clock Drifts

The environment of an oscillator, like the oscillators in the involved Rb clocks, influences its clock-stability over time. The environmental conditions commonly referred to in this circumstance are temperature and humidity. It is therefore common practice to mount the oscillator in a chamber with controlled humidity and to equip this chamber with a self-regulating heater so that these conditions remain constant for the time of interest. Even more significantly varying environmental conditions come into account when the oscillator is located on a mobile platform in general – and on board an aircraft in particular. According to [27], these additional effects include vibrations and acceleration, as well as changes of the gravitation field and the magnetic field.

All of these environmental changes have the potential to temporarily detune the oscillator, resulting in clock drifts and ultimately incorrect timing information. As these timing deviations add up over the duration of a measurement flight, significant errors in e.g. the delay estimation of the MPCs may occur.

4.1.3.1 Estimation

As emphasised in [82], the underlying effects resulting in drifts of the Rb clocks are impossible to predict, however it is possible to estimate the actual clock drift in offline processing by comparing the output of the GS' and the AS' clocks with a common time source like the global GPS time. The algorithm applied to the recorded GNSS data during the PPP processing takes the GPS time into account and is able to provide information on the drifts of the involved clocks. These estimated drifts, denoted by $\tau_{\mathrm{GS}}(t)$ and $\tau_{\mathrm{AS}}(t)$, are plotted in Figure 4.1 for all four flights of the campaign introduced in Chapter 3. For easier comparison, the drifts are given with respect to the time instant of the corresponding reference measurement T_{ref}, which will be used later for the data processing. See Section 4.2.2.2 for an exact definition of T_{ref}.

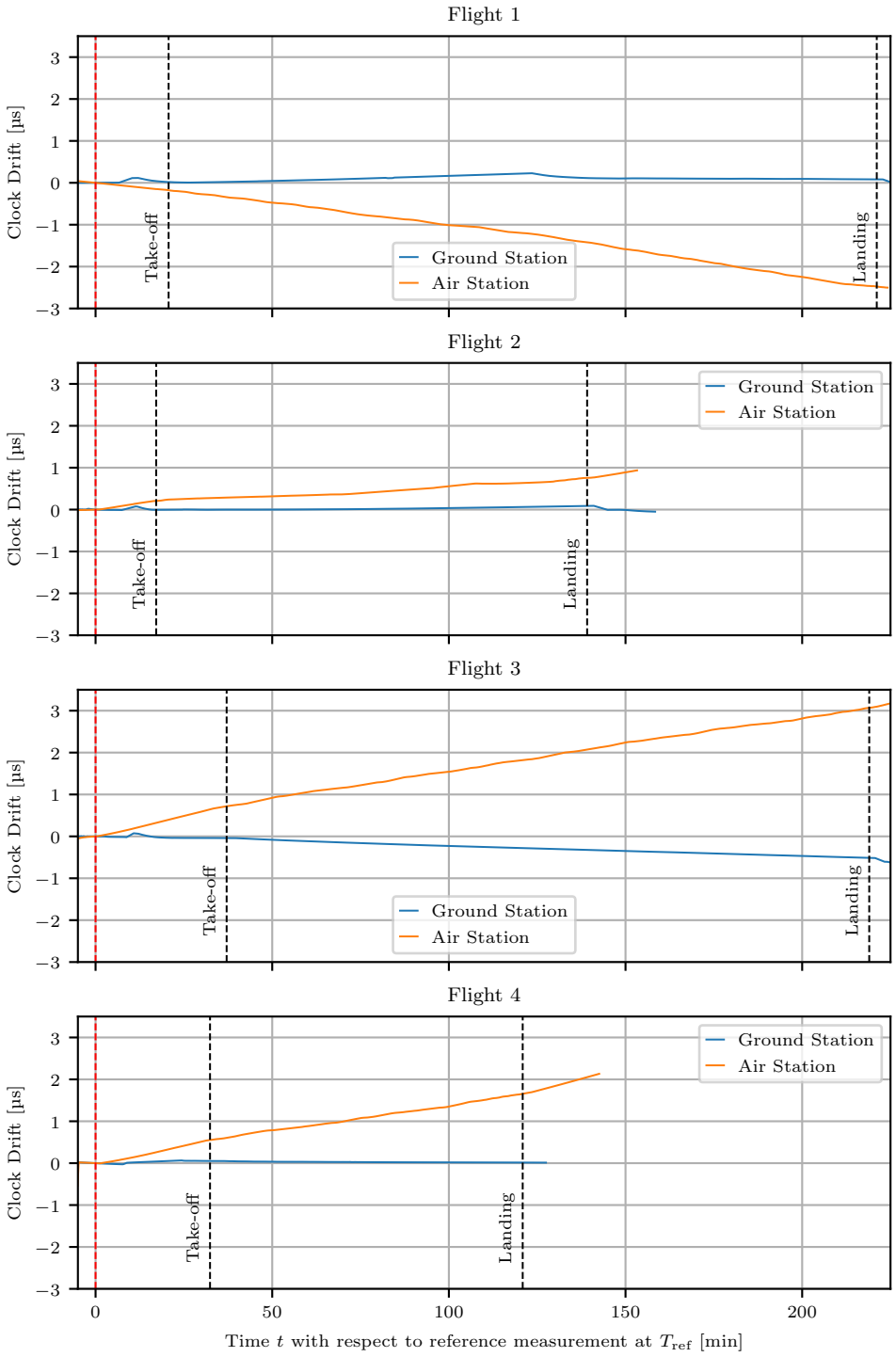


FIGURE 4.1: Estimated drifts of the clocks in the GS ($\tau_{\text{GS}}(t - T_{\text{ref}})$, blue) and in the AS ($\tau_{\text{AS}}(t - T_{\text{ref}})$, orange), respectively. Please note that the drifts are given with respect to T_{ref}, thus the time of the reference measurement of the specific measurement flight, which is marked by a dashed red line at $t = 0$ min. The other two dashed lines mark the (relative) time of take-off and landing, respectively.

The plots show that in all cases the absolute drift of the GS clock is less than the absolute drift of the AS clock. This phenomenon matches the expectations, since the Rb clock in the AS is subject to much more challenging environmental conditions than the Rb clock of the GS. Further, it can be observed that the drifts of both clocks are close to zero for $t = 0\,\text{min}$, which is not surprising, as the procedure described in Section 3.5.1 includes a synchronisation of the Rb clocks right before the reference measurement is performed.

It can also be observed that there is a small peak of the GS clock's drift between the reference measurement and the take-off in all four cases. We assume that the reason for the increase of the drift towards this peak is a loss of the GNSS signal during the the relocation of the mobile GS, see Section 3.5.2. Once the GS is in its transmitting position on the rooftop of the institute's building, GNSS can be received again and the drift decreases.

4.1.3.2 Compensation

The detected clock drifts need to be compensated during the processing of the channel sounding data. At time instant t, the overall clock drift compensation $\tau_{\text{drift}}(t)$ is computed according to

$$\tau_{\text{drift}}(t) = \overbrace{\left(\tau_{\text{AS}}(t) - \tau_{\text{AS}}(T_{\text{ref}})\right)}^{\text{AirStation}} - \overbrace{\left(\tau_{\text{GS}}(t) - \tau_{\text{GS}}(T_{\text{ref}})\right)}^{\text{GroundStation}}. \tag{4.1}$$

4.1.4 Provided Ground Truth Data

The GT data that is actually needed for the later processing tasks differs from the raw data mentioned above. For example, there is no use-case for the raw GNSS positioning information of the aircraft without the applied antenna correction. Furthermore, depending on the processing task, certain coordinate systems are more convenient for the calculations than others.

It was therefore decided to not just store the raw GT data, but to store data that is derived from this raw GT and can be directly used during the later processing. This data includes:

- Current position of the receiving antenna in WGS84 coordinates
- Current position of the receiving antenna in ECEF *(Earth Centered, Earth-Fixed Cartesian Coordinate System)*
- Current position of the receiving antenna in TxCCCS *(Transmitter Centered Cartesian Coordinate System)* and TxCPCS *(Transmitter Centered Polar Coordinate System)*
- Current position of the transmitting antenna in RxCCCS *(Receiver Centred Cartesian Coordinate System)* and RxCPCS *(Receiver Centred Polar Coordinate System)*
- Current velocity above ground of the aircraft and estimated LOS Doppler shift
- Current LOS distance d_{LOS} and estimated FSPL between the transmitting and the receiving antenna

- Current correction values for the involved Rb clocks to compensate time drifts as explained in Section 4.1.3

4.1.5 Management of Ground Truth Data

The computation of the GT data – even a single GT sample – requires the loading of the whole data source files for the IMU data of the AS, the GNSS data of the AS, and the GNSS data of the GS. These files sum up to several gigabytes and even a modern computer system needs a notable amount of time[5] to load this high amount of data into its memory; the required synchronisation and interpolation of these data for the requested timestamp takes even more time. At the same time, all later data processing, requiring a pre-processing based on GT data, is performed in blocks of a certain size, as it will be shown in the following sections.

The size of a block, i.e. its length in time, does not change during the processing, thus all timestamps the GT data are required for are known in advance[6]. This circumstance motivates a pre-computation of all GT data.

Instead of dumping the pre-computed GT values into local files[7], it was decided to write them into a relational data base hosted on a central server using the widespread SQL *(Structured Query Language)* technology.

4.1.5.1 Benefits of Using an SQL Data Base

Using an SQL server for hosting the GT data has the following benefits:

- An SQL server software is optimised to handle huge amounts of data. It provides fast searching algorithms as well as efficient storage and caching capabilities.
- Modern SQL implementations allow the efficient storage of floating point values of high precision. These implementations outperform text-based storage systems like CSV *(Comma Separated Values)* files in terms of precision and speed.
- If the SQL server is accessible on the network as suggested in Figure 4.2, clients can request GT data from the server – without the need to store all GT data locally.
- All relevant programming languages provide an interface to communicate with an SQL server. The PYTHON library NUMPY [29], that is mostly used during the later processing, allows an efficient handling of floating point data received over an SQL connection.
- Even if the grid of timestamps of the pre-calculated GT data points does not fit the client's needs, the client still has the option to interpolate the required values based on the data looked up in the data base. In most cases, this is still faster than loading, synchronising, and interpolating the raw GT data.

[5] On the workstation used for the data processing the loading of the raw data takes about 40 s to 60 s, depending on the length of the flight.

[6] For the GT data, the timestamp of a block is always given with respect to its centre.

[7] A performant implementation would require a distribution over multiple files to allow quick access by avoiding the loading of huge files; such an implementation would require the development of a complex file management.

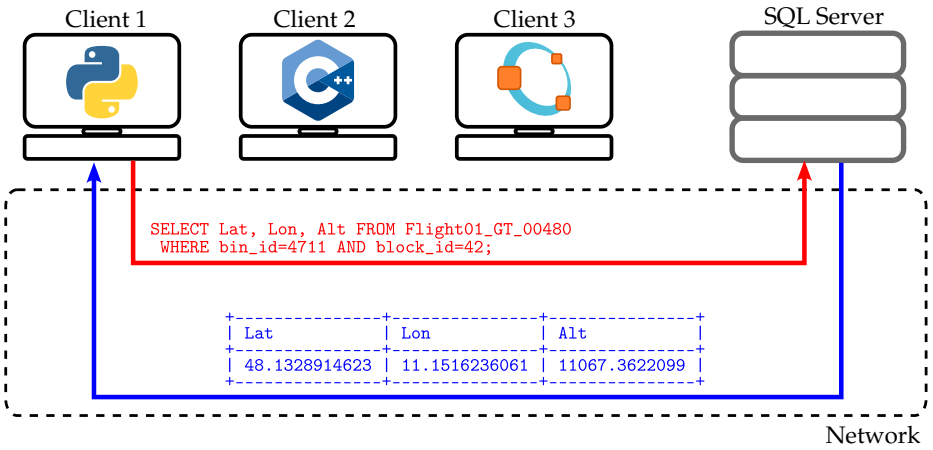


FIGURE 4.2: Scheme of the SQL server hosting the Ground Truth data, and three example clients running applications in PYTHON, C++, and OCTAVE. Red: Request from Client 1. Blue: Reply from SQL Server. Logos: via WIKIMEDIA Commons[8]

- The same SQL data base can also be used to store other data like processing results. Especially when using table joins, this allows a very convenient and efficient handling of data.

4.1.5.2 Structure of the Data Base

The data base contains multiple tables of the same structure. For each of the four flights, an individual table for each block size that is of interest for the later processing is created. Splitting up the GT data over multiple tables results in less entries per table and a more efficient primary key handling which allows faster look-ups.

4.1.5.3 Primary Key

Looking up data in a table of a relational data base can be speeded up by defining a primary key that acts as a unique identifier for each entry of the specific table. Using the timestamp for this purpose would result in bad performance since the floating point data type that is required to represent the timestamp would be an inappropriate choice of data type for a primary key.

Instead, the fact that all IQ data is stored in binary files having a unique *bin_id* (see Section 3.4.5) and the fact that the data in the files is processed in blocks having a *block_id* is exploited: Both the *bin_id* and the *block_id* are represented by integer values and are therefore an appropriate choice of data type for a primary key. Since the *block_id* is unique only inside of one binary file, a tuple of *bin_id* and *block_id* is required to allow a one-to-one identification within one table. This tuple is therefore used as the primary key in each of the GT data base tables.

The example request in Figure 4.2 shows how data is requested using the table's primary key.

[8] PYTHON: www.python.org, GPL; C++: Jeremy Kratz, Public domain; OCTAVE: Developer of GNU Octave: John W. Eaton, GPL

4.2 Radio Signal Pre-Processing

4.2.1 System-Theoretical Description of Received Signal

Using (2.8) and following the nomenclature from Section 3.2.2, the channel sounding signal at the receiver can be described as

$$s_{\mathrm{Rx}}(t) = h(t) * s_{\mathrm{Tx}}(t) \; + \; n(t), \tag{4.2}$$

where $*$ denotes the convolution, $n(t)$ describes a noise term, and $h(t)$ describes the (unknown) CIR of the channel between the transmitter and the receiver. The noise is considered as white – i.e. it has a uniform power spectral density – and GAUSSian – i.e. its amplitudes follow a (complex) GAUSSIAN distribution. Both of these assumptions are usually not *perfectly* fulfilled in a real-world measurement setup, but the deviations – e.g. the noise spectrum may be influenced by internal filtering – are considered as negligible as it is common practice, see e.g. [82].

In the case of the reference measurement, the CIR is dominated by the attenuators, in case of the actual channel measurements, it is dominated by the actual wireless channel; however, influences of both the transmitting and receiving antenna are expected.

The time-discrete signal $\boldsymbol{s}_{\mathrm{Rx}} = [s_{\mathrm{Rx},0}, s_{\mathrm{Rx},1}, \ldots]$ is created by sampling according to

$$s_{\mathrm{Rx},k} = \boldsymbol{s}_{\mathrm{Rx}}[k] = \left(s_{\mathrm{Rx}}(t) * g_{\mathrm{Rx}}(t)\right) \left(\frac{k}{f_{\mathrm{sr}}}\right), \; k \in \mathbb{N}, \tag{4.3}$$

where $g_{\mathrm{Rx}}(t)$ is the filter of the ADC *(Analog-to-Digital Converter)* and f_{sr} denotes the sample rate.

4.2.2 Pre-processing of Received Radio Signal

4.2.2.1 Conversion of ADC Values to Amplitudes

As mentioned in Section 3.4.5, the IQ-recorder stores the samples of the recorded signal as raw ADC values. Several parameters are required to convert these values to absolute amplitudes. These parameters include the value of G_{DC} (see Section 3.4.4): While the remaining parameters are constant and were determined by separate calibration measurements, the value of G_{DC} is variable and is read from each binary file's header. The benefit of taking G_{DC} already into account when converting the ADC values to actual float amplitudes is that a later power compensation based on G_{DC} is not necessary any more.

Binary files that have been recorded during an update of G_{DC} are not considered in the further processing for the following reason: While changing the value of G_{DC} has an immediate effect on the power of the signal at the ADC of the IQ-recorder, the updated value for G_{DC} is recorded in the header of the next binary file and *not* in real time. It is therefore impossible to determine the correct amplitudes of the samples of the entire binary file that was recorded while the value for G_{DC} was changed, since the *exact* time of the G_{DC} update is unknown.

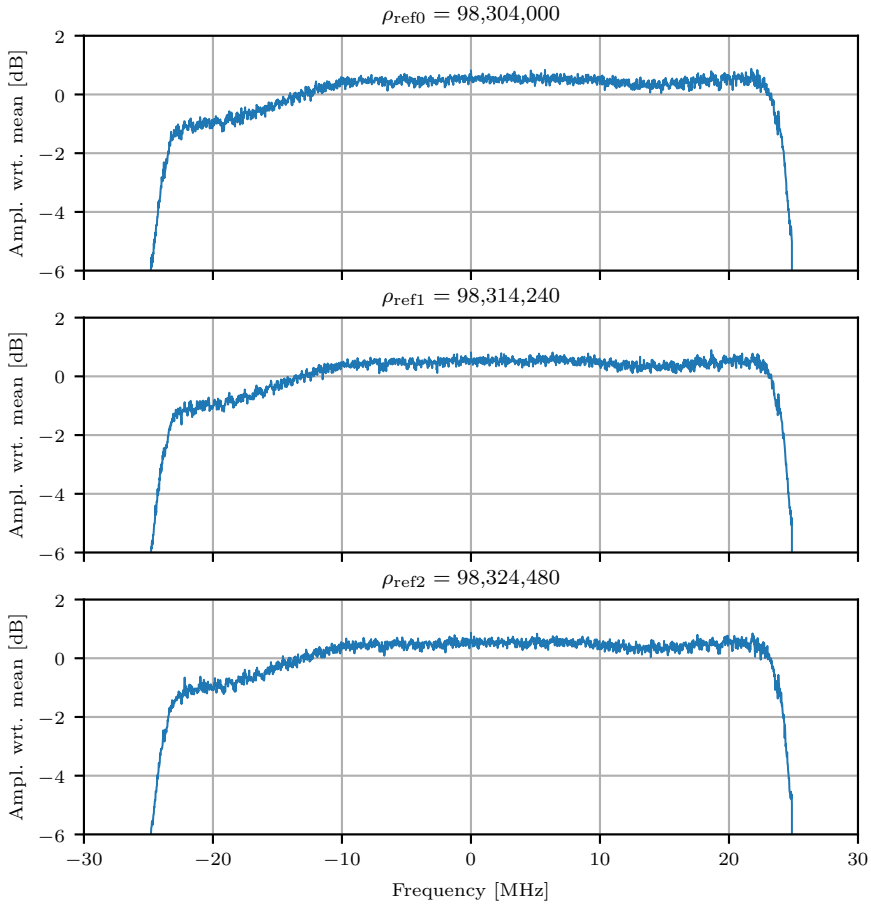


FIGURE 4.3: Logarithmic spectra of three arbitrarily chosen reference signals. The three spectra of the different reference signals show only very small deviations for several time instants.

The amount of binary files recorded during the described campaign that cannot be processed for this reason is in the range of 0.1 %.

4.2.2.2 Extraction of Reference Signal

From the properties of the transmission signal (3.7) and the recording sample rate it follows that a vector of $N = 2{,}048$ samples of the received IQ data is guaranteed to contain *exactly* one (circularly shifted) instance of the channel sounding signal $\boldsymbol{x}_{\text{SC}}$. Thus, an arbitrary vector of length N can be chosen from the data recorded during the reference measurement starting at sample number ρ_{ref} as the reference signal used for evaluation:

$$\boldsymbol{x}_{\text{ref}} = \left[\boldsymbol{x}_{\text{Rx}}[\rho_{\text{ref}}], \boldsymbol{x}_{\text{Rx}}[\rho_{\text{ref}} + 1], \ldots, \boldsymbol{x}_{\text{Rx}}[\rho_{\text{ref}} + N - 1]\right]. \tag{4.4}$$

The logarithmic spectra of three different instances of the reference signal $\boldsymbol{x}_{\text{ref}}$ with arbitrarily chosen start sample numbers $\{\rho_{\text{ref0}}, \rho_{\text{ref1}}, \rho_{\text{ref2}}\}$ are plotted in Figure 4.3. The spectra are computed using the DFT as defined in Appendix B and scaled by their individual mean power.

Apparently, the deviations between these three different instances are extremely small and are therefore considered as negligible.[9] Thus, the actual value of ρ_{ref} is free of choice; however, this chosen value of ρ_{ref} is later required to determine the time shift when comparing the extracted reference signal with the measurement signal. We define T_{ref} as the time instance of the recording of the sample indexed by ρ_{ref}.

4.2.2.3 Normalisation of Data Blocks to the LOS Component

Loading a Block All further processing of the received channel sounding data is performed in blocks of BN samples, thus each block has the length of $B \in \mathbb{N}$ complete channel sounding sequences. A block b is created by cutting out BN consecutive samples of the received signal:

$$\begin{aligned} \boldsymbol{x}''^{(b)} &= \left[\boldsymbol{s}_{\text{Rx}}[\rho^{(b)}], \boldsymbol{s}_{\text{Rx}}[\rho^{(b)}+1], \ldots, \boldsymbol{s}_{\text{Rx}}[\rho^{(b)}+BN]\right] \\ &= \left[\boldsymbol{s}_{\text{Rx}}[\rho^{(b)}], \boldsymbol{s}_{\text{Rx}}[\rho^{(b)}+1], \ldots, \boldsymbol{s}_{\text{Rx}}[\rho^{(b+1)}-1]\right], \end{aligned} \tag{4.5}$$

where $\rho^{(b)}$ denotes the value of the IQ-recorder's sample counter corresponding to block b. This notation, in which the block index is given in round brackets as a superscript number, is used throughout the rest of this dissertation where applicable.

We call $B \in \mathbb{N}$ the *block size*, as it defines how many consecutive channel sounding sequence instances of length N are processed together in one block. A larger B increases the Doppler resolution during processing, however, this increase comes at the cost of time resolution: If B is chosen large, it is possible that relevant fast fading channel effects are vanished out and thus are not modelled appropriately later. This effect becomes even more likely considering the comparatively high velocity of an aircraft.

Compensating for the LOS Delay Based on the GT data, $\boldsymbol{x}''^{(b)}$ is shifted such that the current LOS delay $\tau_{\text{LOS}}^{(b)}$ is compensated using an approach similar to the approach presented in [82]. For the sake of simplicity, we assume the time drift compensation as introduced in (4.1) is considered as part of $\tau_{\text{LOS}}^{(b)}$.

In the general case, the LOS delays the individual instances of the channel sounding sequence within block b are shifted by are *not* the same, since the receiver in the aircraft is moving and the resulting delay is therefore changing over time. For the m-th instance of the channel sounding sequence within $\boldsymbol{x}''^{(b)}$, this resulting delay is given by

$$\tau_{\text{LOS},m}^{(b)} = \tau_{\text{LOS}}^{(b)} + \overbrace{\left(\underbrace{-\frac{\nu_{\text{LOS}}^{(b)}}{f_{\text{c}}}}_{\substack{\text{see} \\ (2.3)}} \underbrace{\frac{N}{f_{\text{sr}}}\left(m - \frac{B-1}{2}\right)}_{\substack{\text{Time shift w. r. t.} \\ \text{block centre}}} \right)}^{\substack{\text{Delay caused by} \\ \text{Doppler shift}}}. \tag{4.6}$$

To compensate for these time shifts, we exploit the shift theorem of the DFT as stated in Appendix B. The B instances of the channel sounding sequence are first individually

[9] This statement implies the assumption of a constant frequency response of the hardware during one measurement flight.

transformed to frequency-domain

$$\forall\, m \in \{0, \ldots, B-1\}: \quad \boldsymbol{C}''^{(b)}_m = \mathrm{DFT}\left\{\left[\boldsymbol{x}''^{(b)}[mB], \boldsymbol{x}''^{(b)}[mB+1], \ldots, \boldsymbol{x}''^{(b)}[(m+1)B-1]\right]\right\}. \quad (4.7)$$

In a next step, the time shift correction as determined in (4.6) is applied to each instance according to

$$\forall\, m \in \{0, \ldots, B-1\}: \quad \boldsymbol{C}'^{(b)}_m[k] = \boldsymbol{C}''^{(b)}_m[k] \mathrm{e}^{-\mathrm{j}2\pi \frac{k f_{\mathrm{sr}}}{N} \tau^{(b)}_{\mathrm{LOS},m}},\ \forall\, k \in \{0, \ldots, N-1\}. \quad (4.8)$$

Eventually, the individual channel sounding sequences are transformed back to time-domain

$$\forall\, m \in \{0, \ldots, B-1\}: \quad \boldsymbol{c}'^{(b)}_m = \mathrm{DFT}^{-1}\left\{\boldsymbol{C}'^{(b)}_m\right\} \quad (4.9)$$

and then are concatenated to a single vector $\boldsymbol{x}'^{(b)} \in \mathbb{C}^{BN}$ where the elements are given by

$$\forall\, m \in \{0, \ldots, B-1\}: \quad \boldsymbol{x}'^{(b)}[mB+n] = \boldsymbol{c}'^{(b)}_m[n],\ \forall\, n \in \{0, \ldots, N-1\}. \quad (4.10)$$

Compensating for the LOS Doppler Shift In a next step, we compensate for the LOS Doppler shift $\nu^{(b)}_{\mathrm{LOS}}$ of the received signal as we have estimated it when computing the GT in Section 4.1. As stated in Appendix B, a frequency shift by f_{shft} – thus a shift in frequency-domain – corresponds to a multiplication by an exponential term $\mathrm{e}^{\mathrm{j}2\pi f_{\mathrm{shft}} t}$ in time-domain. Applied to the elements of the (time-discrete) data block $\boldsymbol{x}'^{(b)}$ and inverting the sign of the exponent to actually perform a compensation, this corresponds to

$$\boldsymbol{x}^{(b)}[n] = \boldsymbol{x}'^{(b)}[n] \cdot \mathrm{e}^{-\mathrm{j}2\pi \nu^{(b)}_{\mathrm{LOS}} \frac{n}{f_{\mathrm{sr}}}},\ \forall\, n \in \{0, 1, \ldots, BN-1\}. \quad (4.11)$$

Normalising the Power In the following, we further assume that the power levels of $\boldsymbol{x}_{\mathrm{ref}}$ and $\boldsymbol{x}^{(b)}$ are matched, such that the attenuation $A_{\mathrm{dB,ref}}$ used during the recording of the reference signal (see Section 3.5.1) and the FSPL of the LOS component during the recording of $\boldsymbol{x}^{(b)}$, denoted by $\mathrm{FSPL}^{(b)}$, are compensated. This is achieved by the application of

$$P^{(b)}_{\mathrm{dB}} := P'^{(b)}_{\mathrm{dB}} - A_{\mathrm{dB,ref}} + \mathrm{FSPL}^{(b)}_{\mathrm{dB}}, \quad (4.12)$$

where $P'^{(b)}_{\mathrm{dB}}$ denotes the power of $\boldsymbol{x}^{(b)}$ before the compensation and $P^{(b)}_{\mathrm{dB}}$ denotes the power of $\boldsymbol{x}^{(b)}$ after the compensation. We call $P^{(b)}_{\mathrm{dB}}$ the calibrated received power of block b with respect to the FSPL. The logarithmic FSPL of the LOS component of block b is computed according

to

$$\begin{aligned} \mathrm{FSPL}_{\mathrm{dB}}^{(b)} &= 10\log_{10}\left\{\left(\frac{4\pi d_{\mathrm{LOS}}^{(b)} f_{\mathrm{c}}}{c_{\mathrm{air}}}\right)^2\right\} \\ &= 20\log_{10}\left\{d_{\mathrm{LOS}}^{(b)}\right\} + 20\log_{10}\left\{f_{\mathrm{c}}\right\} + \underbrace{20\log_{10}\left\{\frac{4\pi}{c_{\mathrm{air}}}\right\}}_{\approx -147.55}, \end{aligned} \tag{4.13}$$

where $d_{\mathrm{LOS}}^{(b)}$ denotes the distance between the transmitter and the receiver (see Section 2.1.2.1) at the time instance when block b was recorded.

4.2.2.4 Parallelisation of Data Block Pre-processing

The processing of the data blocks as shown in the previous subsection is fully parallelisable: Thanks to the absence of temporal data dependencies, block b can be processed without any data related to block $b-1$. This property is also made possible by the fact that the required GT data used for the processing described in Section 4.2.2.3 is fully precomputed for each b and can be loaded by a lookup in the SQL data base as described in Section 4.1.5.

Chapter 5

Processing the Dominant Component

We do not consider resolvable multipath propagation effects in this chapter, rather we focus on the *dominant signal component* and evaluate its received power. We define the dominant component as the signal component with the strongest power. This does not necessarily correspond to the LOS component, as this component may be subject to blockage or is reduced in power due to non-resolvable scattering effects.

5.1 Power Calculation

The power of the dominant component P_{dom} is computed in blocks of B complete instances of the channel sounding sequence. The processing steps are as follows [61]:

1. The pre-processing as described in Section 4.2.2 is applied. Thus, the remaining processing steps are applied to blocks $\boldsymbol{x}^{(b)}$ that are normed to the delay, the Doppler shift, and the power of the LOS component as described in the referenced section.

2. The coherent PDP is computed using the corresponding reference signal $\boldsymbol{x}_{\text{ref}}$ as defined in Section 4.2.2.2 and an upsampling of $f_{\text{up}} \in \mathbb{N}$ according to the definition given in Appendix C.3:

$$\boldsymbol{s}_{\text{coh}}^{(b)} = \text{PDP}_{\text{coh}}\{\boldsymbol{x}^{(b)}\}. \tag{5.1}$$

3. According to our definition from the top of this section, we denote the maximum of the PDP as the received power of the dominant signal component:

$$P_{\text{dom}}^{(b)} := \max\left\{\boldsymbol{s}_{\text{coh}}^{(b)}\right\} \tag{5.2}$$

   We call $P_{\text{dom}}^{(b)}$ the calibrated received power of the dominant signal component of block b with respect to the FSPL.

Just as with the pre-processing, there are no temporal data dependencies in the presented processing; thus it is fully parallelisable.

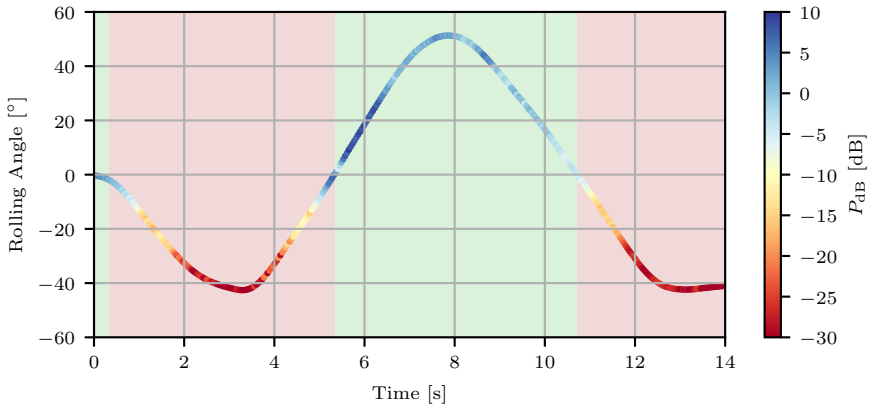


FIGURE 5.1: Colour coded received power during strong banking. The background colour represents the presence of a LOS path that is determined from the polar angle of a receiver-centred NED *(North-East-Down)* coordinate system and the aircraft geometry, see Figure 3.7 and [61]; red: "LOS path is blocked", green: "LOS path is present".

5.2 Evolution of P_{dom}

The following results were computed with $B = 100$ if not denoted otherwise. This B corresponds to a block length of 4.096 ms. The power of the dominant component P_{dom} is given in a logarithmic scale and denoted by P_{dB}.

5.2.1 Banking Angle

The behaviour of the received power computed according to Section 5.1 during strong banking is given in Figure 5.1 providing the rolling angle (positive rotation around the axis that defines the travelling direction, see [61]) on the y-axis. Raising the left wing and lowering the right wing indicates a positive rolling angle and vice versa.

By the time the given snapshot was recorded, the LOS distance was 4.6 km at a flight altitude of 1.01 km (given in WGS84). Thus, the difference in altitude between transmitter and receiver was just 358 m resulting in a large polar angle $\Theta_{\mathrm{ENU}} \approx 85.5°$ in the spherical counterpart to the transmitter-centred ENU *(East-North-Up)* coordinate system. At the same time, this results in a large polar angle in the spherical counterpart to the receiver-centred NED coordinate system (see Figure 3.7), even in case the rolling angle is zero.

The aircraft passed the transmitter from southeast, heading northeast; thus, seen from aircraft, the transmitter was on the left side. Observing the transmitter's position given in the receiver-centred NED coordinate system allows a precise declaration if airframe shadowing is present (LOS path not available) or not present (LOS path available). In Figure 5.1, the background colour indicates if the LOS path is available (light green colour) or not available (light red colour). The presence of the LOS path is determined evaluating the NED angles (in case of banking, the polar angle is of significant interest) and the airframe architecture around the receiving antenna as sketched in Figure 3.7. For the sake of completeness, it is pointed out that no obstacles around the transmitter are blocking the LOS path for the observed period of time.

The presented rolling angle in Figure 5.1 indicates, that the first bank (0.2 s to 5.3 s) leads to a loss of the LOS path since the received power P_{dB} drops down to -30 dB. This statement holds a comparison with the expected presence of the LOS path based on the NED angles. However, the figure shows that the dramatic drop in received power does not happen suddenly, but that a transition from strong reception to low reception is happening. This observation can be explained by knife edge diffraction caused by the airframe, most likely the guide rails.

The second bank (5.3 s to 11 s) indicates a bank in the opposite direction, such that the receiving antenna has a direct view to the transmitter without any shadowing due to the airframe or similar. Consequently, the received power is up to 40 dB higher compared to the power received during the first bank. The third bank starting at 11 s shows the same behaviour as the first bank.

5.2.2 Take-off

The received power during take-off is shown in Figures 5.2 and 5.3. All flights started and ended at the EDMO airport, the starting direction was the same for all four flights (heading: south east). The position of the runway and the transmitter is shown in Figure 5.6.

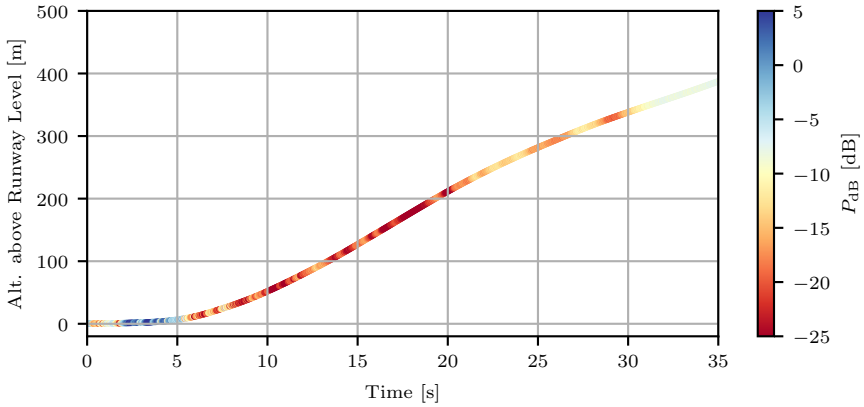
While P_{dB} is mostly in the range of -2 dB to 4 dB as long as the aircraft is on the runway, one can observe an intense drop of up to 27 dB during take-off and climbing. This observation applies to all four take-offs in a very similar manner, however, the exact position of the strongest drops during climbing varies.

To understand the behaviour of the received power, it helps to recall that the receiving antenna is located at the bottom of the aircraft and that the aircraft is heading away from the transmitter during take-off. This results in a complete block of the LOS path as long as the aircraft is climbing with a strong pitch angle in the given direction. Furthermore, no reflector close to the receiving antenna, which could result in a stronger received signal, is present. Nevertheless, the received power varies a bit and was observed to take on values as high as -12 dB. The moderate variations may be caused by signals reflected from buildings and obstacles located behind the runway when seen from the transmitter.

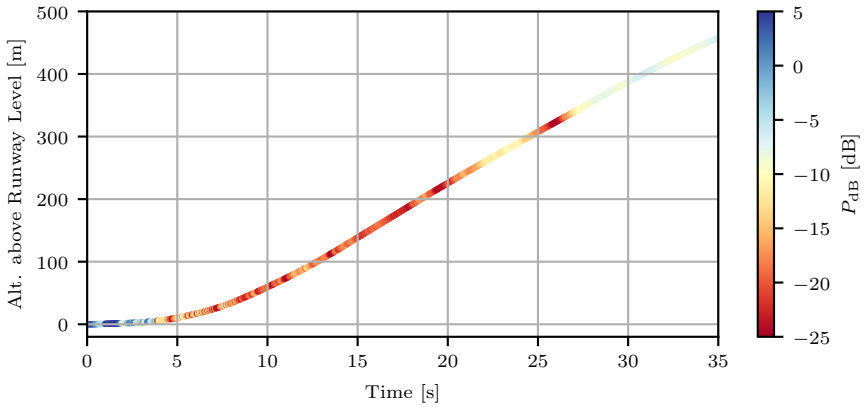
5.2.3 ENR *(En-route)*

Figure 5.4 shows the received power during a part of flight II, i.e. the long range flight, vs. the LOS distance. The data shown in Figure 5.4 was recorded during a northbound flight at an altitude of 9.3 km, following a radial course away from the transmitter. One can observe a periodically changing received power within a dynamic range of 5 dB. The period is in the range of 11 s, corresponding to a delta of about 1.5 km of LOS distance.

These oscillations can be interpreted as large-scale fading and are explained by the CE2R model, see Section 2.3.3. This reflection model describes the behaviour of the received signal which is composed of two signals via superposition: The LOS signal and the signal reflected off the ground. In our case, this composed signal (mostly) defines the dominant component as introduced above. The theoretical signal power based on the CE2R model is also given in Figure 5.4. Apparently, the model and the measured received power match quite well.

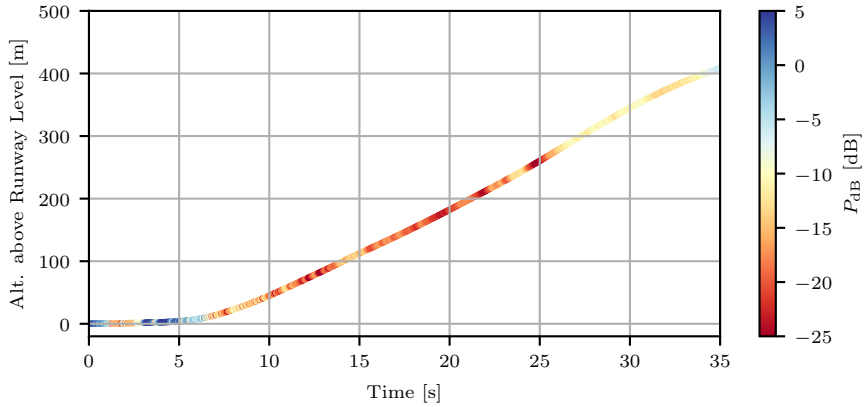


(A) Flight I

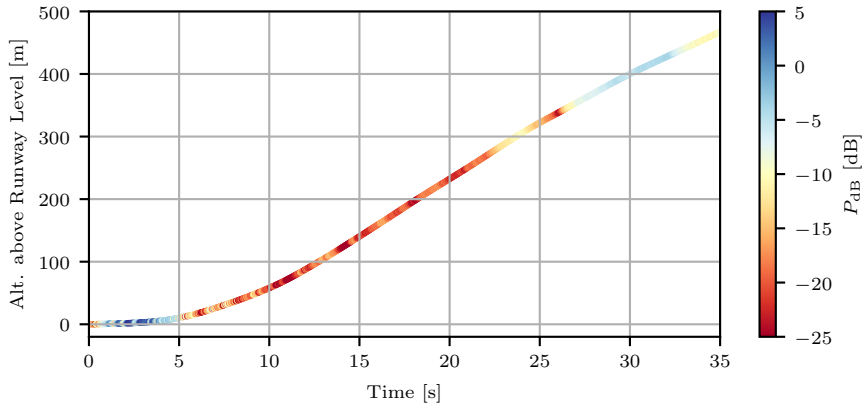


(B) Flight II

FIGURE 5.2: Colour coded receiving power during take-off. The same runway with the same heading during take-off was used for all flights.



(A) Flight III



(B) Flight IV

FIGURE 5.3: Colour coded receiving power during take-off. The same runway with the same heading during take-off was used for all flights.

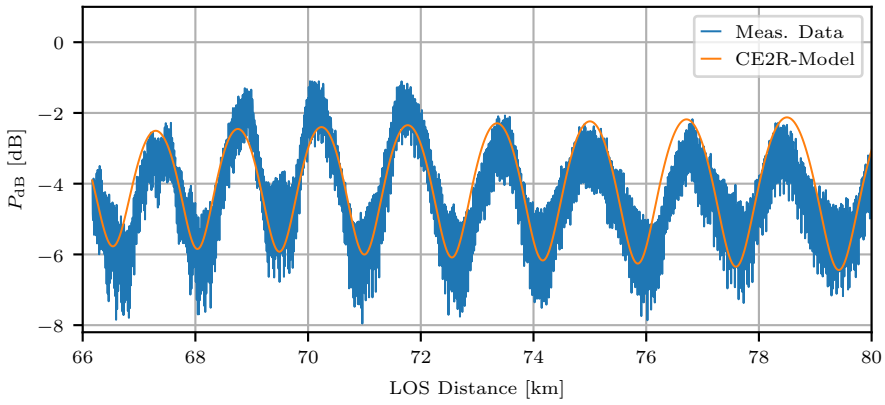


FIGURE 5.4: Received power during en-route vs. the distance between the transmitting and the receiving antenna. The aircraft is following a radial course away from the transmitter. Both the received power computed from the measurement data and the result of the CE2R *(Curved-Earth Two-Ray)* model are plotted.

The deviations are attributed to different permittivities at the reflection point and a different relative height, as they are not taken into account in the applied model.

The CE2R model as introduced in [51] can be applied only under the condition that the LOS path and the ground wave path are not resolvable at the receiver, thus that their path lengths differ by less than Δr_{min}. It has been verified that this condition is fulfilled for the geometry observed in Figure 5.4.

5.2.4 Go-around

Figure 5.6 shows the received power during eleven go-arounds (see Section 3.6) flown at different altitudes above the runway of the EDMO airport.

The displayed parts of the go-around all have a "U"-shaped flight altitude profile, where the bottom part is flown with different altitudes over the runway ("runway part") as can be seen in Figure 5.6a. During the runway part, the roll, pitch and yaw angles of the aircraft were all kept around 0°. The received power is comparatively strong ($> -5\,\mathrm{dB}$), except for two sections ("drop section") labelled A and B where the received power partially drops down to $-22\,\mathrm{dB}$.

The position of these drop sections with respect to the transmitter can be identified better in Figure 5.6b, where the received power is plotted in dependency on the azimuth angle φ and the polar angle θ according to the ENU *(East-North-Up)* coordinate system as defined in [61].

The dependency of the received power on the azimuth angle is clearly visible. The two drop sections A and B are highlighted in both plots and are found to be in the range of $89° < \varphi < 94°$ and $114° < \varphi < 122°$, respectively. Although the antenna pattern of the transmitter is not considered during data processing, we do not affiliate these significant drops to the antenna, as we do not see a similar power drop at similar azimuth angles in other flight scenarios. As the go-arounds were flown along both directions of the runway, we also consider it as unlikely that the drops can be explained by the receiver's antenna pattern. An analysis of the NED angles and the aircraft's architecture shows that we can also rule out airframe shadowing as the reason for the drops in received power. Therefore, we assume the

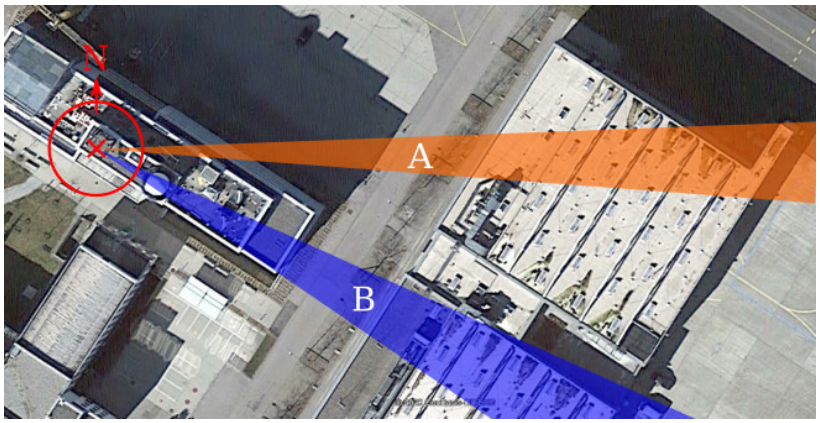


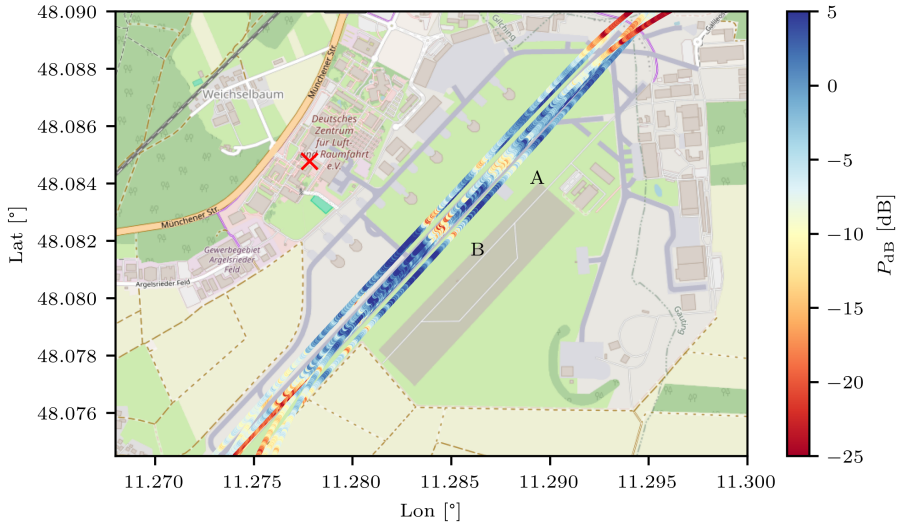
FIGURE 5.5: Bird view of the transmitter (marked by a red cross) and its immediate environment. A and B mark the angle ranges where significant drops in received power are observed during go-arounds. Please note that neither the roof northwest of the transmitter nor the construction crane north of the transmitter were present during the measurement flights. Background image: ©2020 GeoBasis-DE/BKG, Google Earth

power drops in A and B are caused by obstacles between the transmitter and the receiver that cause interfering rays (non-resolvable scatter components) that degrade the computed received power of the dominant component.

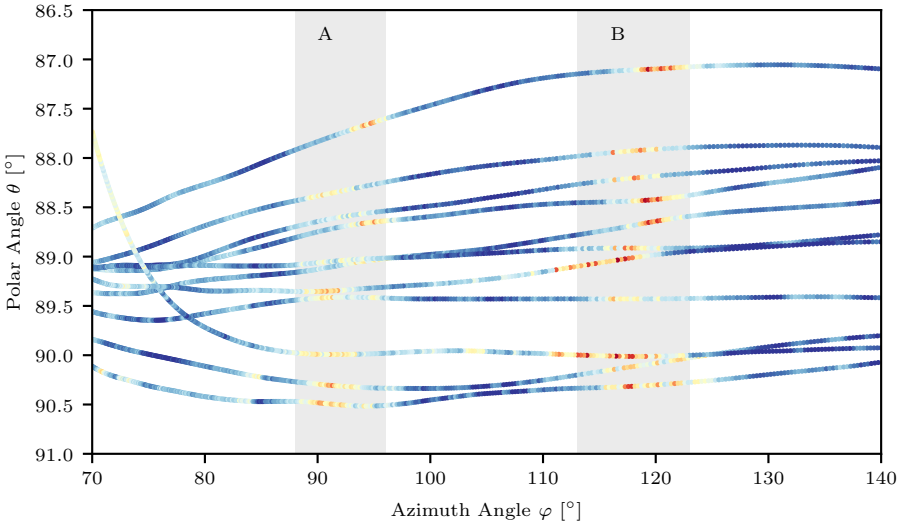
Let us first have a look on the situation in drop zone A: Figure 5.5 shows that parts of the roof of a hangar building intersect with the ground projection of the LOS path for the corresponding angle range of $89° < \varphi < 94°$. The authors have verified that this roof does not block LOS path between the transmitter and the receiver for any of the flights observed here. We therefore assume, that these drops are caused by non-resolvable signal components reflected off this roof and reflected off obstacles located on this roof. Apparently, these components are contributing to the dominant component destructively. The condition for a non-resolvable signal path is that the distance between the length of the LOS path and the length of the reflected path is less than Δr_{min}. Considering the location and height of the hangar building, we were able to find reflection points on this roof fulfilling this condition for the specific angle range for all flights. Drop zone A is bounded by higher obstacles located on the roof blocking the path of the reflected signals for greater values of φ as it can be verified in Figure 5.5.

For drop zone B, the lower reception power can be explained in a different way: Figure 5.5 reveals that a small radome is located about 15 m southeast of the transmitter. The radome is not that high that it blocks the LOS path for any of the flights, however, its top is high enough to disrupt the estimated first FRESNEL Zone of the link between the transmitter and the receiver at least partially for $114° < \varphi < 122°$. We assume that the radome, therefore, causes the recorded power drops in zone B.

We understand these findings as a hint that the immediate environment of the ground antenna has a strong impact on the resulting channel conditions, especially in critical flight manoeuvres like go-arounds. We therefore suggest to carefully choose the ground antenna's location.



(A) Map of the runway, some airport buildings, and the transmitter (red cross); Background image: ©OpenStreetMap, https://openstreetmap.org/copyright



(B) Dependency on ENU *(East-North-Up)* angles

FIGURE 5.6: Received power during eleven go-arounds recorded during all four flights. The two "drop sections" are marked with A and B, respectively. While A shows not that significant drops, the received power in B drops by up to 22 dB.

Chapter 6

Processing the Multipath Components

In contrast to Chapter 5, we focus on all (resolvable) MPCs in this chapter. Again, the processing is performed block-wise. Following the notation introduced in Section 2.1.1, the set of detected MPCs of block b is denoted by $\mathcal{M}^{(b)}$; the amount of detected MPCs in block b is denoted by $L^{(b)} = \left|\mathcal{M}^{(b)}\right|$. The MPCs in $\mathcal{M}^{(b)}$ are again indexed by the MPC index l.

6.1 Detection of Multipath Components

6.1.1 MPC Detection Based on Delay

The basic idea of this approach is to detect peaks within either the coherent PDP *(Power Delay Profile)* or the incoherent PDP of a data block. Each of these peaks represents either a single MPC or a cluster of MPCs; the position of the peaks can be used to estimate the underlying MPCs' delay. However, no Doppler information can be extracted (directly) following this approach.

Power Delay Profile The first step is to compute the PDP of the current data block $\boldsymbol{x}^{(b)}$ by applying the processing described in Appendix C.3. For this processing, the reference signal $\boldsymbol{x}_{\text{ref}} \in \mathbb{C}^N$ and a the upsampling factor $f_{\text{up}} \in \mathbb{N}$ are used as it has already been introduced in Section 5.1.

While the (coherent) PDP has the advantage of a lower noise floor, short term MPCs may vanish out if the block size B is chosen too large. On the other hand, if B is chosen too small, signals of some reflectors may vanish in the noise floor and will remain undetected. The smallest possible block size, thus $B = 1$, corresponds to the case when the absolute squared of a single IR is used for detection.

In the following, we refer to the current block's PDP by $\boldsymbol{s}^{(b)} \in \mathbb{R}^{Nf_{\text{up}}}$ and to the current block's averaged CIR as defined in Appendix C.2 by $\bar{\boldsymbol{h}}^{(b)} \in \mathbb{C}^{Nf_{\text{up}}}$, where f_{up} denotes the upsampling factor.

We also define the corresponding logarithmic vector by

$$\boldsymbol{s}_{\log}^{(b)} = 10 \log_{10} \left\{ |\boldsymbol{s}^{(b)}| \right\}, \tag{6.1}$$

where the mathematical operations are applied element-wise.

Noise Floor The power of the noise floor for the current block is estimated based on areas of $\boldsymbol{s}^{(b)}$ where the appearance of measurable strong MPCs or measurement artifacts is very unlikely. If we denote the threshold delay up to which significant MPCs are expected by $\tau_{\max}$, this area is described by $\{\tau_\sigma | \tau_{\max} < \tau_\sigma < N/f_{\mathrm{sr}}\}$. We denote the part of $\boldsymbol{s}^{(b)}$ where this condition holds by $\boldsymbol{s}_\sigma^{(b)}$. Thus, the noise floor of block b is determined by ${\sigma^{(b)}}^2 = \mathbb{E}\{\boldsymbol{s}_\sigma^{(b)}\}$.

Peak Detection The next task is to detect local maxima ("peaks") in $\boldsymbol{s}_{\log}^{(b)}$. A peak must fulfil the following two conditions:

- The peak's value must exceed the determined noise floor by a certain power P_{ex}. If we denote the index of the potential peak within the PDP by $n \in \mathbb{N}$, this condition is given by $\boldsymbol{s}_{\log}^{(b)}[n] \geq {\sigma^{(b)}}^2|_{\mathrm{dB}} + P_{\mathrm{ex}}|_{\mathrm{dB}}$.
- The peak must have a certain prominence defined by a minimum distance to other detected peaks. This minimum distance is chosen based on the resolution along the delay axis of the PDP.

All detected $L^{(b)}$ peaks' positions n_l inside of $\boldsymbol{s}_{\log}^{(b)}$ are stored in $\mathcal{P}^{(b)}$, where $n_l \in \mathbb{N}$ denotes the index of the l-th peak.

Determination of the Delay The delay of the MPC can be determined using the delay resolution of the PDPs as given in (C.1) and the index found in the previous step according to

$$\tau_l = n_l \Delta_\tau. \tag{6.2}$$

Determination of the Amplitude The complex amplitude is given by $\alpha_l = \bar{\boldsymbol{h}}^{(b)}[n_l]$. In order to complete the four elements of the MPC as defined in (2.4), the Doppler shift of MPC l is set to $\nu_l = 0$. Then, ξ_l is added to $\mathcal{M}^{(b)}$.

6.1.2 MPC Detection Based on Delay and Doppler Shift

The basic idea of this approach is very similar to the previous one, however, instead of using the PDP, the DDSF *(Delay-Doppler Spread Function)* is used. The DDSF provides not only information on the delay, but also on the Doppler shift of an MPC. Thus, the processing steps are basically the same as for the PDP case described above; however, all processing is performed on two-dimensional data instead of one-dimensional data.

Again, peaks in the DDSF may represent either a single MPC or a cluster of MPCs, as it is not possible to resolve such a cluster into its individual components at this point.

Delay-Doppler Spread Function The detection process starts with the computation of the DDSF of the current data block b according to Appendix C.4:

$$\boldsymbol{S}^{(b)} = \mathrm{DD}\{\boldsymbol{x}^{(b)}\}. \tag{6.3}$$

We further define a logarithmic representation of the DDSF:

$$\boldsymbol{S}_{\log}^{(b)} = 20\log_{10}\left\{|\boldsymbol{S}^{(b)}|\right\}, \tag{6.4}$$

where the all mathematical operations are applied element-wise. Again, the reference signal $\boldsymbol{x}_{\mathrm{ref}} \in \mathbb{C}^N$ and the upsampling factor $f_{\mathrm{up}} \in \mathbb{N}$ are used for the computation.

The structure of the returned matrix $\boldsymbol{S}^{(b)}$ is visually explained in Figure 6.1.

Noise Floor To compute the power of the noise floor, a similar approach as introduced in Section 6.1.1 is used: The power of the noise floor for the current block ${\sigma^{(b)}}^2$ is estimated based on areas of $\boldsymbol{S}^{(b)}$ where the appearance of measurable strong MPCs or measurement artifacts is very unlikely. These areas are given by all elements of $\boldsymbol{S}^{(b)}$ where $\{\tau_\sigma | \tau_{\max} < \tau_\sigma < N/f_{\mathrm{sr}}\}$ and $\{\nu_\sigma | \nu_{\min} < |\nu_\sigma| < \nu_{\max}\}$ hold. If we define a matrix $\boldsymbol{S}_\sigma^{(b)}$ that contains all values of $\boldsymbol{S}^{(b)}$ where the two conditions hold, the noise floor is given by ${\sigma^{(b)}}^2 = \mathbb{E}\{|\boldsymbol{S}_\sigma^{(b)}|^2\}$. In Figure 6.1, the areas used to compute the noise floor are highlighted in yellow.

Peak Detection The next task is to detect peaks in $\boldsymbol{S}_{\log}^{(b)}$. Similar to the peak detection within the PDP, we define two related conditions a potential peak has to fulfil:

- The peak's value must exceed the determined noise floor by a certain power P_{ex}. If we denote the index of the potential peak within the DDSF by $(n, m) \in \mathbb{N}^2$, this condition is given by $\boldsymbol{S}_{\log}^{(b)}[n, m] \geq {\sigma^{(b)}}^2|_{\mathrm{dB}} + P_{\mathrm{ex}}|_{\mathrm{dB}}$.
- The peak must have a certain prominence. This is ensured by defining an elliptical search area, called *footprint*, that is used to detect local maxima. Using an elliptical footprint instead of a circular footprint – which would be the straightforward two-dimensional pendant to the distance measure used for the one-dimensional peak detection within the PDP – is advantageous since the footprint's dimensions can be chosen taking the different resolution along the delay and the Doppler axis of $\boldsymbol{S}^{(b)}$ into account.

Finding local maxima within a two-dimensional matrix is a common task in the field of digital image-processing and therefore well understood. In this dissertation, the peak detection approach by the authors of [102] has been used.

All detected $L^{(b)}$ peaks' coordinates inside of $\boldsymbol{S}^{(b)}$ are stored pairwise (n_l, m_l) in $\mathcal{P}^{(b)}$, where n_l denotes the row index and m_l denotes the column index of the l-th peak. Since the DFT of a finite, non-periodical signal usually shows the spectral leakage effect, leading to side lobes that are likely to be mistaken for peaks, windowing is applied as described in Appendix C.5.

Determination of the Delay and the Doppler Shift Based on the resolution of the DDSF given by (C.1) and (C.9), the index pairs found in the previous step can be translated to a delay and Doppler value. For the delay, this mapping is done according to (6.2), for the Doppler, the mapping is done according to

$$\nu_l = \left(m_l - \frac{B}{2}\right)\Delta_\nu. \tag{6.5}$$

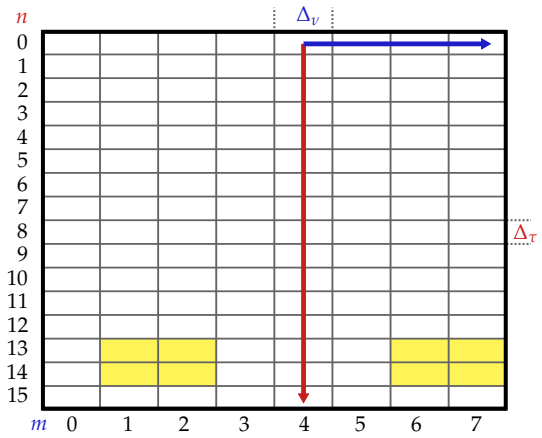


FIGURE 6.1: Matrix returned by the discrete DDSF *(Delay-Doppler Spread Function)* with $N = 16$, $B = 8$ and no upsampling. The row indices n can be converted to delays using the delay resolution Δ_τ (red) using (6.2); correspondingly, the column indices m can be converted to Doppler shifts using the Doppler resolution Δ_ν (blue) using (6.5). Due to the pre-processing, the LOS component is always expected in $(0, B/2)$. The areas highlighted in yellow are used to determine the noise floor.

Figure 6.1 provides a visual explanation of this translation process.

Determination of the Amplitude Two alternative approaches can be used to determine the remaining element of the detected MPCs, the amplitude.

- The straightforward way is to take the complex values of the individual peaks from the linear representation of the DDSF given in (6.3), thus $\alpha_l = \boldsymbol{S}^{(b)}[n_l, m_l]$. This approach is fast and robust to imperfect estimates of the delays and the Doppler shifts of the individual MPCs. However, if the peaks detected in the DDSF are reduced in height, either by the spectral leakage effect or as a result of the windowing to reduce the impact of the spectral leakage effect, the weights will be estimated to be too low.

- The more sophisticated way is – once all MPCs have been detected – to use an LS *(Least Squares)* optimisation to find values for all α_l, such that the squared EUCLIDEAN distance between a synthesised version of the current data and the actual (pre-processed) measurement data is minimised. This approach is independent of the spectral leakage effect, however, it highly depends on a precise delay and Doppler shift estimation: If one of these parameters has been estimated incorrectly for MPC l, the corresponding absolute value of the amplitude $|\alpha_l|$ may tend towards zero during the following LS *(Least Squares)* optimisation.

  1. The basis of the synthesised signals is the reference signal: We introduce a row vector $\boldsymbol{x}_{\text{ref}}^{(B)} \in \mathbb{C}^{1\times BN}$ containing B concatenated instances of the reference signal. We further introduce a function $\mathfrak{F}_\tau^\nu : \mathbb{C}^{1\times BN} \mapsto \mathbb{C}^{1\times BN}$ that applies a delay shift by τ and a Doppler shift by ν to a vector representing a discrete signal. This function's processing is analogous to the compensation of the LOS delay and Doppler shift explained in Section 4.2.2.3.
  2. Using the above definitions, a matrix $\boldsymbol{Y}^{(b)} \in \mathbb{C}^{BN\times L^{(b)}}$ is created, whose l-th column contains a modified version of the reference signal, shifted in time and frequency

according to τ_l and ν_l, respectively, thus:

$$\boldsymbol{\Upsilon}^{(b)} = \begin{pmatrix} \mathfrak{F}_{\tau_0}^{\nu_0}\left\{\boldsymbol{x}_{\text{ref}}^{(B)}\right\} \\ \mathfrak{F}_{\tau_1}^{\nu_1}\left\{\boldsymbol{x}_{\text{ref}}^{(B)}\right\} \\ \vdots \\ \mathfrak{F}_{\tau_{L^{(b)}-1}}^{\nu_{L^{(b)}-1}}\left\{\boldsymbol{x}_{\text{ref}}^{(B)}\right\} \end{pmatrix}^T. \tag{6.6}$$

3. Now, an approximation for the column vector $\boldsymbol{\alpha}^{(b)} \in \mathbb{C}^{L^{(b)}\times 1}$, where the l-th item represents the amplitude of the l-th MPC α_l, needs to be found for the equation

$$\boldsymbol{\Upsilon}^{(b)}\boldsymbol{\alpha}^{(b)} = \boldsymbol{x}^{(b)}. \tag{6.7}$$

Under the assumption that ${\boldsymbol{\Upsilon}^{(b)}}^T\boldsymbol{\Upsilon}^{(b)}$ is invertible, the LS approach can be applied to (6.7) to find an approximation for $\boldsymbol{\alpha}^{(b)}$ according to

$$\hat{\boldsymbol{\alpha}}^{(b)} = \left({\boldsymbol{\Upsilon}^{(b)}}^T\boldsymbol{\Upsilon}^{(b)}\right)^{-1} {\boldsymbol{\Upsilon}^{(b)}}^T\boldsymbol{x}^{(b)}. \tag{6.8}$$

4. Besides the desired approximation for $\boldsymbol{\alpha}^{(b)}$, the LS algorithm also returns a residual $\epsilon^{(b)}$ that we understand as a measure of precision of the estimated MPCs' parameters. The residual is computed according to

$$\epsilon^{(b)} = \sum_{BN} \underbrace{||\boldsymbol{x}^{(b)} - \boldsymbol{\Upsilon}^{(b)}\hat{\boldsymbol{\alpha}}^{(b)}||^2}_{\in\mathbb{R}^{BN}}, \tag{6.9}$$

which corresponds to the sum over all elements of the vector that results from the squared EUCLIDEAN distance between the measured signal block and the weighted synthesised signals.

5. By combining τ_l, ν_l, and the absolute value and phase of the l-th item of $\hat{\boldsymbol{\alpha}}^{(b)}$, all elements of MPC ξ_l, as defined in (2.4), are given.

Now that all elements of all MPCs are determined, the MPCs are added to $\mathcal{M}^{(b)}$.

6.1.3 Application on Measurement Data

The processing explained in the previous section has been applied to measurement data collected during the flight measurement campaign introduced in Chapter 3. The results of this processing are introduced and discussed in the following.

6.1.3.1 Take-Off

Figure 6.2 shows a colour-coded representation of the DDSFs of three consecutive blocks $b \in \{0, 1, 2\}$ computed according to the steps described in Section 6.1.2. The underlying data were recorded during take-off and processed with a block size of $B = 480$, corresponding to a block duration of $BN/f_{\text{sr}} = 19.6\,\text{ms}$. The detected MPCs are highlighted by red crosses.

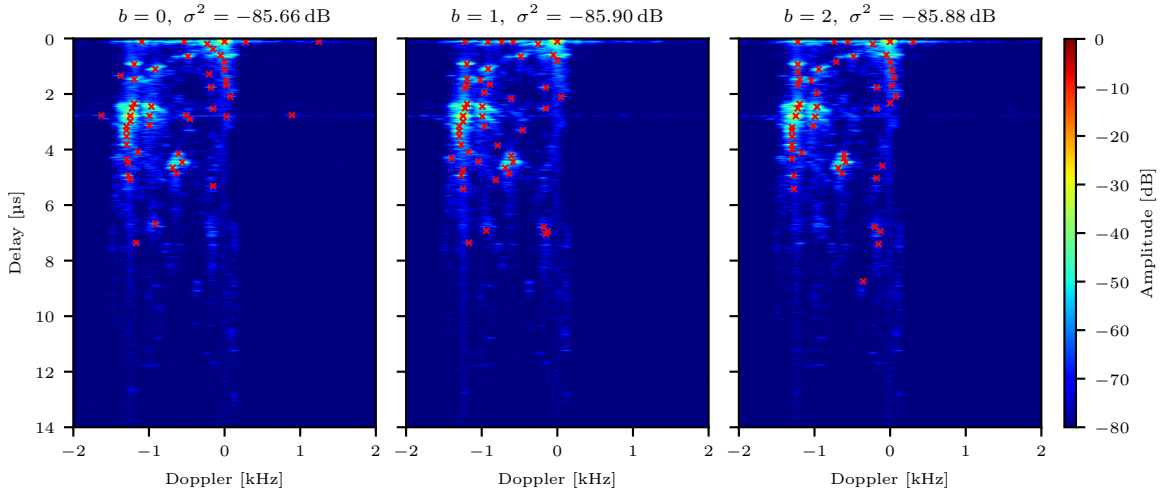


FIGURE 6.2: Colour-coded representation of the logarithmic DDSFs of three consecutive blocks recorded during take-off. The strongest 50 MPCs that also exceed a power 15 dB above the respective noise floor σ^2 are highlighted by red crosses. All blocks have been processed individually without any time dependent data dependency.

All three figures show two main reflector areas: One that corresponds to the LOS path and its close environment and one that consists of multiple MPCs with a Doppler shift of around -1.1 kHz and a delay of 2.1 µs to 3.5 µs. A third, less intense reflector area can be seen for a Doppler shift of around 0.8 kHz and a delay of 4.0 µs to 4.3 µs. Only very few MPCs with a positive Doppler shift can be observed.

It can be observed, that most of the MPCs detected in block $b = 0$ are also detected for $b = 1$ and $b = 2$ and vice versa. However, this does not apply for some MPC: Especially those having a delay of more than 5.8 µs tend to be missed in some blocks during detection. At the same time, it can be observed that these MPCs not only have a comparatively huge delay, but are also comparatively weak in power.

6.1.3.2 Flyover

Figure 6.3 also shows a colour-coded representation of the DDSFs of three consecutive blocks $b \in \{0, 1, 2\}$. The underlying data were recorded just seconds before a direct flyover of the AS over the GS in an altitude of around 3.2 km. All processing parameters were the same as for the take-off scenario described above.

First, it can be observed that the noise floor is about 7.3 dB stronger compared to the take-off scenario. We explain this by the applied gain setting G_{DC}, see Sections 3.4.4 and 4.2.2.1, during recording: As the LOS distance – and consequently the FSPL – was higher than during the recording of the data shown in Figure 6.2, a higher pre-amplification of the received signal was required, including an increase of the noise floor.

It can also be observed, that the LOS component is blurred along the Doppler axis. We explain this by the fact, that the aircraft was moving, while the received data are processed in blocks. Theoretically, block processing as applied here requires a perfect snapshot of the received data where *all* parameters that affect the measurement remain *exactly* the same for the length of one block. For obvious reasons this is not possible in a real-world scenario,

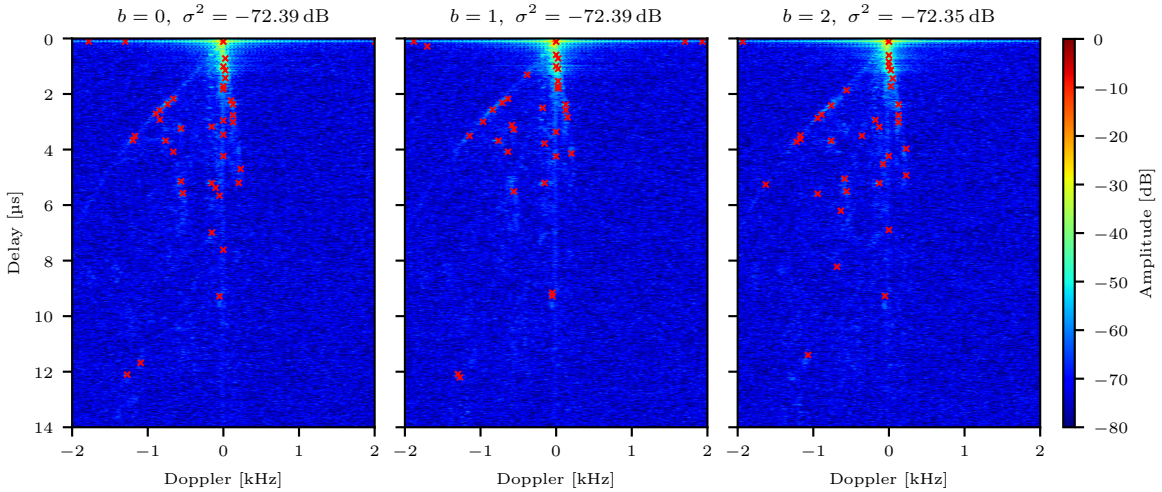


FIGURE 6.3: Colour-coded representation of the logarithmic DDSFs of three consecutive blocks recorded during a flyover over the GS. The strongest 50 MPCs that also exceed a power 8 dB above the respective noise floor σ^2 are highlighted by red crosses. All blocks have been processed individually without any time dependent data dependency.

especially when an aircraft is moving at cruising speed. However, to extract any kind of Doppler information from measurement data, some sort of block processing is required as the Doppler can be described as the time derivative of the delay. Thus, a trade-off between available Doppler resolution and time precision has to be resolved.

6.1.4 MPC Detection Improvements

The MPC detection approach presented so far can be called a *coarse* detection process, as the presented processing results are limited in precision due to the following two aspects:

- First, the resolution of the underlying data structure that is used as a basis for the peak detection, so either the PDP or the DDSF.
- Second, clusters of MPCs that are so close together such that they can only be detected as a single MPC cannot be resolved any further.

Although these aspects may be irrelevant in many cases, we will now discuss approaches to address these limitations for those cases where these aspects are of interest. Without loss of generality, the approaches are explained based on the two dimensional MPC detection, thus based on the DDSF.

6.1.4.1 Increasing Detection Resolution

The potential problem of the approach described in Section 6.1.2 is, that its precision is limited by the resolution of the discrete DDSF. Therefore, the exact position of an MPC inside of a tile of the DDSF matrix – and consequently its exact delay and Doppler shift – cannot be determined any further. We address this issue by applying an optimisation algorithm that tries to adjust the delay and Doppler shift of every detected MPC in the current block b.

Selection of Optimisation Algorithm The optimisation algorithm required to solve this problem shall have the following properties:

- The algorithm shall be applicable to multi-dimensional problems as all MPCs of a block b shall be optimised as a whole and each MPC shall be optimised along both the delay and the Doppler axis.
- The algorithm shall work on a bounded optimisation space for each optimisation dimension as this increases the probability of finding an actual global optimum and not just a local optimum.
- The algorithm shall be free of derivatives as those cannot be provided for the given problem.

The BOBYQA *(Bound Optimization by Quadratic Approximation)* algorithm [72] is an optimisation algorithm fulfilling these requirements and is chosen to be applied to the given problem using the implementation provided by [38]. The BOBYQA algorithm has also been applied to a similar problem in [82].

Prerequisites We assume the coarse detection as described in Section 6.1 has been completed for block b and the parameters of the detected $L^{(b)}$ MPCs are stored in the set $\mathcal{M}^{(b)}$.

Dimensionality We understand that in case these MPCs are taken from the DDSF, the optimisation problem has $2|\mathcal{M}^{(b)}| = 2L^{(b)}$ dimensions.

Initialisation The optimisation algorithm is initialised by the delay and the Doppler shifts of the MPCs given in $\mathcal{M}^{(b)}$.

Optimisation Bounds The optimisation bounds are defined for each individual MPC $\zeta_l^{(b)} \in \mathcal{M}^{(b)}$ as follows: When Δ_τ and Δ_ν denote the resolution of the DDSF along the delay and the Doppler axis, respectively, as defined in (C.1) and (C.9), the LB *(Lower Bound)* and the UB *(Upper Bound)* for the delay optimisation of MPC l are given by:

$$\tau_l^{(\{\mathrm{LB,UB}\})} = \max\left\{ 0,\ \tau_l \mp \frac{\Delta_\tau}{2} \right\} \tag{6.10}$$

and, correspondingly, for the Doppler optimisation of MPC l, the bounds are given by

$$\nu_l^{(\{\mathrm{LB,UB}\})} = \nu_l \mp \frac{\Delta_\nu}{2}. \tag{6.11}$$

Optimisation Process The algorithm basically executes the LS optimisation described in (6.6) to (6.9) in an iterative loop indexed by $i \in \mathbb{N}$:

First, a matrix $\tilde{\boldsymbol{Y}}_i^{(b)}$ is generated using the delay and Doppler shifts of the MPCs given in the current (hypothetical) set of MPCs $\tilde{\mathcal{M}}_i^{(b)}$ similar to (6.6). Then, an equation similar to (6.7)

$$\tilde{\boldsymbol{Y}}_i^{(b)} \tilde{\boldsymbol{\alpha}}_i^{(b)} = \boldsymbol{x}^{(b)} \tag{6.12}$$

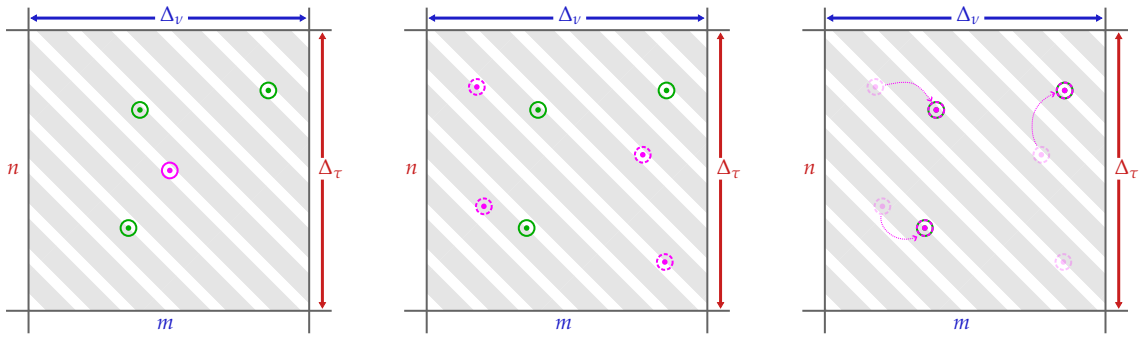


FIGURE 6.4: Resolving a cluster of three MPCs (marked by a green circle) within tile (n, m) of the DDSF (marked by a diagonal pattern): (a) The peak detection within the DDSF delivers one single MPC at position (τ_l, ν_l) (parental MPC, marked by a pink circle), however, the actual three MPCs are located somewhere within the tile (b) Applying (6.14) and (6.15) using a split factor of $\Psi = 4$ results in four derived MPCs, randomly distributed within the tile (marked by a dashed pink circle) – these MPCs are used for the initialisation of the optimisation (c) The optimisation process is finished, the position of three out of the four derived MPCs has been found, the remaining one is omitted.

is solved for $\tilde{\boldsymbol{\alpha}}_i^{(b)}$ – again using the LS approach.

Thus, the overall objective of the optimisation problem is to minimise the residual $\epsilon_i^{(b)}$ returned by the LS algorithm applied to (6.12). Using the definition given in (6.9), this leads to

$$\mathcal{M}_{\text{fine}}^{(b)} = \underset{\tilde{\mathcal{M}}_i^{(b)}}{\operatorname{argmin}} \left\{ \underbrace{\sum_{BN} \left|\left| \boldsymbol{x}^{(b)} - \tilde{\boldsymbol{Y}}_i^{(b)} \tilde{\boldsymbol{\alpha}}_i^{(b)} \right|\right|^2}_{\epsilon_i^{(b)}} \right\}. \tag{6.13}$$

Termination The execution of the optimisation is terminated when the result does not improve more than a certain threshold from one iteration to the next: $\epsilon_{i-1}^{(b)} - \epsilon_i^{(b)} < \epsilon_{\text{thresh}}$.

6.1.4.2 Resolving MPC Clusters

Although the approach presented in Section 6.1.4.1 has the ability to determine the parameters of the MPCs even below the resolution of the DDSF, it cannot resolve multiple MPCs whose mutual distances in delay and/or Doppler are lower than the corresponding DDSF resolution. In case this is not acceptable, we suggest the following approach to resolve the individual MPCs. The sub-figures in Figure 6.4 visualise the steps of the described optimisation approach for a single tile of the DDSF (thus a potential cluster of multiple MPCs).

- It is assumed that the coarse detection process as defined in Section 6.1.2 has been completed for the current block b; thus there is a set $\mathcal{M}^{(b)}$ with $L^{(b)}$ MPCs.
- For each element of $\mathcal{M}^{(b)}$, thus for each MPC, $\Psi \in \mathbb{N}$ copies are created. A random delay and random Doppler shift are added to each of these copies individually according

to

$$\forall\, l \in \{0, 1, \ldots, L^{(b)} - 1\}:$$
$$\forall\, \psi \in \{0, 1, \ldots, \Psi - 1\}:$$
$$\tau_{l,\psi} = \tau_l + \left(\mathcal{U}_{[0,1]} - 0.5\right) \Delta_\tau \tag{6.14}$$
$$\nu_{l,\psi} = \nu_l + \left(\mathcal{U}_{[0,1]} - 0.5\right) \Delta_\nu, \tag{6.15}$$

where τ_l and ν_l correspond to the delay and Doppler shift of $\xi_l \in \mathcal{M}^{(b)}$ and $\mathcal{U}_{[v,w]}$ denotes a uniform distribution on the interval $[v, w]$. These newly created MPCs are denoted by $\xi_{l,\psi}$ and are called *derived* MPCs; the MPCs in the "original" MPC set $\mathcal{M}^{(b)}$ are called *parental* MPCs. The derived MPCs of this block b are pooled in the set of $\tilde{\mathcal{M}}_\Psi^{(b)}$.

In Figure 6.4(a), the parental MPC of the DDSF tile at (n, m) is represented by a pink circle. The step of creating the derived MPCs for the DDSF tile at (n, m) is visualised in Figure 6.4(b) using a split factor of $\Psi = 4$; the derived MPCs are represented by dashed pink circles.

- To get estimates of the underlying MPCs, the following problem, here given in a compact representation, needs to be solved:

$$\mathcal{M}_\Psi^{(b)} = \underset{\tilde{\mathcal{M}}_\Psi^{(b)}}{\operatorname{argmin}} \left\{ \sum_{BN} \left|\left| \boldsymbol{x}^{(b)} - \tilde{\boldsymbol{Y}}^{(b)} \tilde{\boldsymbol{\alpha}}^{(b)} \right|\right|^2 \right\}. \tag{6.16}$$

  Apparently, this problem has the same structure as the problem given in (6.13).

- Due to the same structure of the problem to solve, the algorithm presented in Section 6.1.4.1 is applied on $\tilde{\mathcal{M}}_\Psi^{(b)}$ in a slightly modified form: The algorithm is initialised using $\mathcal{M}_\Psi^{(b)}$. For each derived MPC $\xi_{l,\psi}$, the optimisation bounds are the same as for the corresponding parental MPC ξ_l. The amount of optimisation dimensions is $2\Psi L^{(b)}$.

- Once the termination criterion is fulfilled (see above), the resulting MPCs are given in $\mathcal{M}_\Psi^{(b)}$.

During the initialisation phase of the optimisation – thus when the derived MPCs are created – the described approach alleges that *every* element in $\mathcal{M}^{(b)}$ is in fact an MPC cluster. In case one or more of these derived MPCs do not exist in the actual measurement data, the optimisation algorithm will not be able to find a reasonable solution for their delay and their Doppler shifts. As a matter of consequence the LS approach for finding the corresponding amplitudes is expected to return a result tending towards zero for the respective MPCs in order to minimise the deviation between the synthesised data and the actual measurement data represented by the residual. Thus, those initially incorrectly alleged MPCs do not carry any (relevant) weight. This process is sketched in Figure 6.4(c), where the derived MPC located in the bottom right of the tile is considered as unnecessary.

Since the split factor Υ determines the behaviour of the approach within one tile, its choice depends – among other factors – on the size of a tile and, as such, on the resolution of the DDSF which follows from the measurement parameters of the processed data. Theoretically, the split factor can be set arbitrarily high since the subsequent processing is expected to zero-out incorrectly alleged MPCs anyway. However, a high split factor leads to a considerable increase of computational effort and is unlikely to significantly improve the accuracy of the overall result in practice, i.e. when processing real-world measurement data that is always subject to noise and imperfect pre-processing. For a setup with $f_{\mathrm{sr}} = 50\,\mathrm{MHz}$, $N = 2{,}048$ and $B = 480$, a split factor of 4 was applied.

6.2 Tracking of Multipath Components

Tracking of MPCs describes the behaviour of the components over time. This is often realised by filter-based approaches, e.g. in [82]. However, path-based approaches have also been discussed in literature, see e.g. [33], [88]. In the following section, another path based approach to track MPCs over time is introduced. This MPC *tracking* approach – referred to as PABATRACK– is directly linked to the MPC *detection* approaches presented in the previous section as it is based on the detected sets of MPCs.

6.2.1 Graph Structure

Based on the sets of MPCs $\mathcal{M}^{(b)}$ detected using one of the approaches given in Section 6.1, a directed graph $G = (\mathcal{M}, \mathcal{D})$ as given in Figure 6.5 is created. The graph, whose structure may remind the reader of a trellis structure known from the decoding of certain FEC *(Forward Error Correction)* codes (see e.g. [31]), is explained in the following.

6.2.1.1 Nodes

The nodes $\mathcal{M}$ of the graph are given by the union of all MPC sets: $\mathcal{M} = \bigcup_b \mathcal{M}^{(b)}$. For the sake of clearness, the example depicted in Figure 6.5 shows only a small graph where the nodes are taken from just five MPC sets identified by $\mathcal{M}^{(\{0,1,2,3,4\})}$. In a real-world application, the amount of blocks is in the range of dozens or even hundreds, depending on the duration of the processed measurement data. Also, the amount of detected MPCs per block is usually higher than in the given example.

As the figure suggests, MPCs of a block b are not mixed with MPCs of other blocks. The MPCs of a block b are sorted in descending order based on the absolute value of their amplitude:

$$|\alpha_l^{(b)}| \geq |\alpha_{l+1}^{(b)}|. \tag{6.17}$$

6.2.1.2 Edges

The edges $\mathcal{D}$ of the MPC graph G are given by the set of all distances between the MPCs of consecutive blocks. These distances are defined as follows: The distance from the l_{s}-th MPC

(start) of the b-th block to the l_e-th MPC (end) of the $(b+1)$-th block is given by

$$d_{l_\mathrm{s}\to l_\mathrm{e}}^{(b\to b+1)} = \mathfrak{D}\left\{\tilde{\xi}_{l_\mathrm{s}}^{(b)}, \tilde{\xi}_{l_\mathrm{e}}^{(b+1)}\right\}, \tag{6.18}$$

where $\mathfrak{D} : \mathbb{R}^4 \times \mathbb{R}^4 \mapsto \mathbb{R}$ is a function expressing the distance between two MPCs by a real value. In the following, a weighted EUCLIDEAN distance function given by

$$\begin{aligned}\mathfrak{D}\left\{\xi_1, \xi_2\right\} =& \Big(\underbrace{a_\tau(\tau_1' - \tau_2')^2}_{\text{Delay}} + \underbrace{a_\nu(\nu_1' - \nu_2')^2}_{\text{Doppler}} + \underbrace{a_\alpha(|\alpha_1|' - |\alpha_2|')^2}_{\text{Weight}} \Big)^{\frac{1}{2}}, \\ & \text{with weight factors } a_\tau + a_\nu + a_\alpha = 1 \text{ and } \{a_\tau, a_\nu, a_\alpha\} \in [0,1]\end{aligned} \tag{6.19}$$

is used, where the weight factors allow the user to adjust the contribution of the individual elements of the MPCs to the MPC distance. The reader may note that the argument arg $\{\alpha\}$ of the involved MPCs is not used here.

The primes in (6.19) indicate a feature scaled representation of the respective element of an MPC ξ according to the popular min-max scaling

$$x' = \frac{x - x_\mathrm{min}}{x_\mathrm{max} - x_\mathrm{min}}, \quad \forall\, x \in \xi, \tag{6.20}$$

where $x_{\{\mathrm{min,max}\}}$ denote a minimum/maximum value, respectively, for each of the elements that are used for the scaling of the respective element. $x_{\{\mathrm{min,max}\}}$ are chosen such that all appearing values of the specific element among the processed scope are mapped to a value within $[0, 1)$, thus a defined interval. Also, the feature scaling transforms the delay, usually given in very small numbers in the range of 1×10^{-6}, and the Doppler shift, usually given in large number in the range of 1×10^3, into a comparable range.

All determined distances between two consecutive blocks b and $b+1$ are aggregated in a distance matrix $\boldsymbol{D}^{(b)} \in \mathbb{R}^{L^{(b)} \times L^{(b+1)}}$, where the entry of the l_s-th row and the l_e-th column is set to $d_{l_\mathrm{s}\to l_\mathrm{e}}^{(b\to b+1)}$.

The MPC graph G as shown in Figure 6.5 is used to describe an evolution over time. From the figure, this representation of the temporal evolution emerges from the ascending block indices b from left to right. Furthermore, all edges of G are directed; as mentioned above, the direction of these edges is always from a block b to the next block $b+1$ and *not* vice versa. Thus, G is a directed graph, but not a symmetric directed graph [9, Definition 6].

6.2.2 Representing the Temporal Evolution of an MPC in a Graph

6.2.2.1 General Definition of a Path

In the field of graph theory, a *path* within a graph is defined as a "sequence of edges" of this graph [9, Definition 7]. According to the same definition, the corresponding distinct sequence of nodes is the "vertex[1] sequence of the path". Thus, a path cannot intersect itself since its assigned nodes are distinct. If the edges are distinct, but the vertices are not, the sequence of

[1] *Vertex* is a commonly used synonym to *node*.

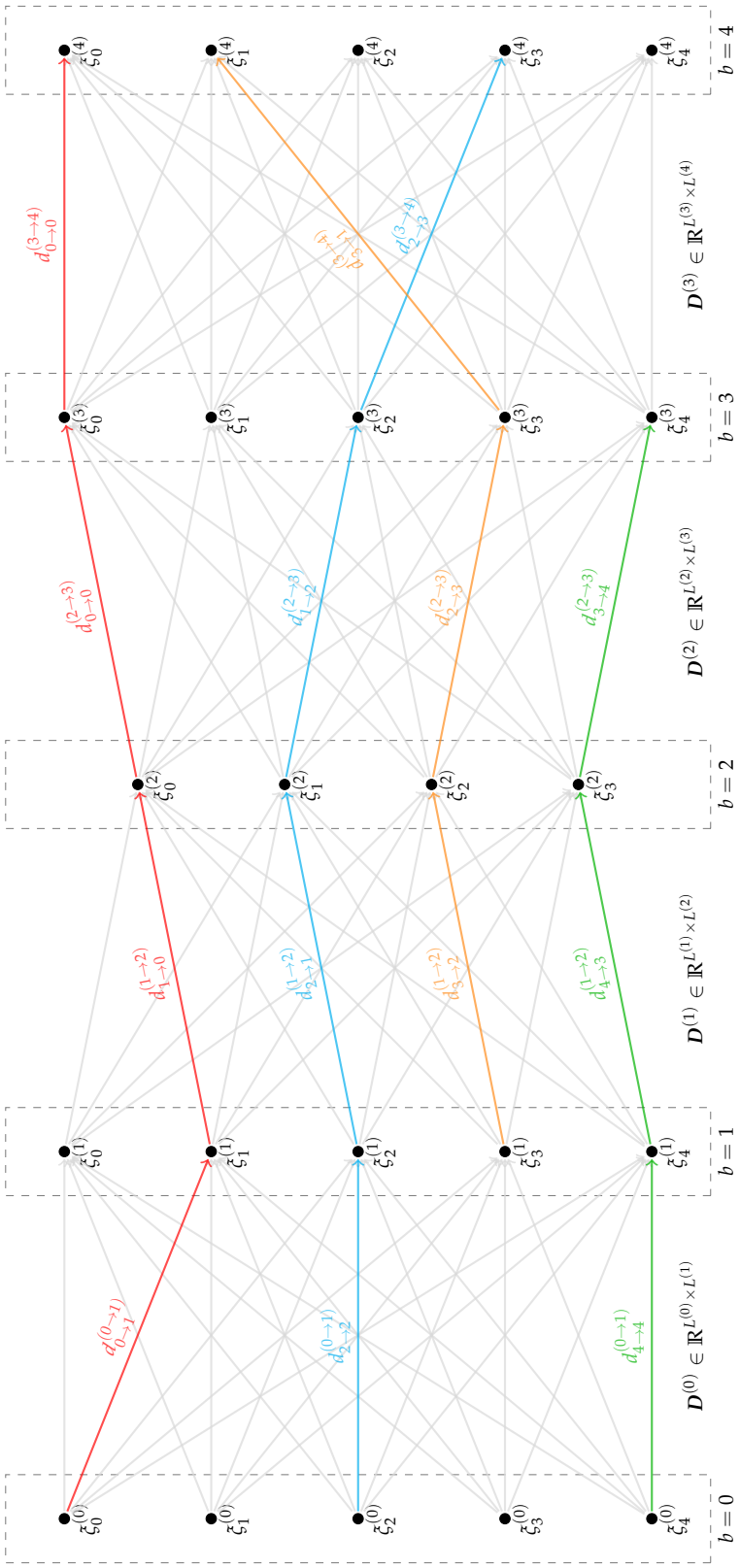


FIGURE 6.5: Graph representation of the MPCs $\xi_l^{(b)}$ detected in five consecutive blocks indexed by b; MPCs are represented by a black dot, blocks are represented by a dashed-line rectangle: While only four MPCs have been detected in block $b = 2$, five MPCs have been detected in the other blocks. The arrows connecting the MPCs represent the distance metrics $d_{l_s \to l_e}^{(b \to b+1)}$ between all MPCs of consecutive blocks; however, for the sake of clearness, only few distance metrics are given in the figure. All distance metrics between block b and block $b + 1$ are given in $\boldsymbol{D}^{(b)}$.

edges is called a *trail*. If neither the edges, nor the vertices are distinct, the sequence of edges is called a *walk*.

6.2.2.2 Paths in the MPC Graph

To represent the temporal evolution of an MPC – known as "tracking" – individual MPCs of consecutive blocks of G are connected using the edges defined above. The structure of G as given in Figure 6.5 and the structure of the tracking algorithm introduced later show that the tracking will always result in a set of distinct nodes and distinct edges – which fulfils the definition of a *path* (and not those of a *trail* or a *walk*). The meaning of the word *path* as it is used in this context must not be confused with the meaning of *path* as it is used in the context of radio wave propagation in Chapter 2! In the remainder of this chapter the word *path* is always used in the context of graph theory if not denoted otherwise. To highlight this, we will refer to these paths within the MPC graph as *MPC paths*.

Although the above definition describes a path as a sequence of *edges*, a path definition based on its *nodes* is used in the following. This is considered to be more intuitive, as these nodes correspond to the MPCs that should be tracked by the presented approach. Due to the structure of G, this still leads to an unambiguous description of an MPC path.

Each node of G – thus each MPC $\xi_l^{(b)}$ of G – can be uniquely addressed by its block index b and its MPC index within the corresponding block l; thus each MPC can be identified by a tuple $(b, l) \in \mathbb{N}^2$. Using the notation from Appendix A.2, an MPC path in G is therefore defined as a sequence of these tuples:

$$\gamma_G = \left(\, (b,l)_i \, \right)_{i=0}^{\beta-1}, \tag{6.21}$$

where $\beta \in \mathbb{N}$ denotes the length of the specific MPC path. The above definition of an MPC path uses indices referring to MPC data in an MPC graph G rather than containing actual MPC data. To emphasise this, the MPC graph this reference-based MPC path belongs to is given in the subscript.

In case it is necessary, an MPC path index $\rho \in \mathbb{N}$ is added as a superscript to avoid ambiguities: $\gamma_G^{(\rho)}$. The root – i.e. the very first element – of an MPC path $\gamma_G^{(\rho)}$ is given by the tuple $(b_\rho^{\triangleright}, l_\rho^{\triangleright})$, the last element of an MPC path $\gamma_G^{(\rho)}$ is given by the tuple $(b_\rho^{\square}, l_\rho^{\square})$.

The set of all MPC paths detected in G is denoted by $\mathcal{P}$.

6.2.3 Path Detection

To describe the MPC path detection process, pseudo code is used. Pseudo code is a common tool to describe the logic of an algorithm since it makes "algorithms more readable" [56]. For the sake of simplification, the description of the PABATRACK algorithm is split into three parts: the initialisation given in Listing 6.1, the *inner* part given in Listing 6.2, and the *outer* part given in Listing 6.3.

6.2.3.1 Initialisation

The pseudo code of the initialisation phase of the algorithm is given in Listing 6.1.

```
1 # initialise set where all detected MPC paths are stored
2 set P = {}
3
4 # iterate over all blocks along time
5 for int b in range(0, num_blocks-1):
6
7     # initialise integer vector indicating assignment status of MPCs
8     # of current block b
9     global vector(int) Γ^(b) = 0_{L^(b)×1}
```

LISTING 6.1: Pseudo code of the initialisation part of the MPC path detection algorithm. Comments are marked with a leading # sign.

The set $\mathcal{P}$ that is used to store all MPC paths detected during the processing is initialised as an empty set in Line 2. When an MPC path is fully recognized during the subsequent processing, it is added to this set.

To indicate whether an MPC of a block b has been assigned to an MPC path already, a vector $\Gamma^{(b)} \in \mathbb{N}^{L^{(b)}}$ is defined for each block: If its value in the l-th position is 0, the l-th MPC of this block is not assigned to an MPC path yet, if the vector's value in the l-th position is set to 1, the l-th MPC is assigned to a path. In the beginning, no MPC of any block is assigned to a path; thus all $L^{(b)}$ elements of all vectors $\Gamma^{(b)} \forall b$ are set to zero – in Line 9 this is denoted by the $L^{(b)} \times 1$ dimensional all-zero matrix $\mathbf{0}_{L^{(b)}\times 1}$. These vectors are declared as global variables in the given pseudo code to emphasise that they must be accessible from all scopes of the algorithm.

6.2.3.2 Outer Part

The purpose of the outer part (see Listing 6.2) of the algorithm is to detect the root node of a new path. Once a root node has been found, the inner part, represented by the subroutine `detect_path`, is called to find the remaining items of the path. In the following, we explain the outer part of the algorithm in detail:

The loop defined in Line 2 iterates over the blocks starting at the first block. Considering the graph in Figure 6.5, this loop iterates over the blocks from left to right.

The loop in Line 5 ensures that the algorithm keeps processing on the current block b, until all of its MPCs have been assigned to a path.

The loop defined in Line 8, finally, iterates over the MPCs of the current block b. Unless the current MPC l has been assigned to an MPC path yet (check in Line 11), the actual MPC path detection subroutine `detect_path`, corresponding to the inner part of our approach, is called in Line 18. The arguments passed to this call are the current block index b and MPC index l as they are used as the root node of the new path. Once `detect_path` returns a new MPC path γ_G, it is added to the set representing the MPC path pool in Line 24.

Please note that the update of the global variable $\Gamma^{(b)}$ is performed inside of `detect_path`. Due to this so called side-effect, `detect_path` is not called a *function*, but a *subroutine* to highlight the lack of idempotence.

```
1  # iterate over all blocks along time
2  for int b in range(0, num_blocks-1):
3
4      # run until all MPCs of current block are assigned to a path
5      while Γ^(b) != ones(L^(b)):
6
7          # iterate over all MPCs of current block
8          for int l in range(0, L^(b)-1):
9
10             # check if l-th MPC is already assigned to an MPC path
11             if Γ^(b)[l] == 1:
12
13                 # it is → continue with next MPC
14                 continue
15
16             # call subroutine to detect MPC path with root in (b,l) and
17             # store returned MPC path in γ_G
18             list γ_G = detect_path(
19                 root_block_index = b,
20                 root_mpc_index = l
21             )
22
23             # add MPC path to MPC path set
24             P.add(γ_G)
```

LISTING 6.2: Pseudo code of the outer part of the MPC path detection algorithm. Comments are marked with a leading # sign. See Listing 6.3 for the `detect_path` subroutine.

6.2.3.3 Inner Part

The inner part of the algorithm is given in the subroutine `detect_path` in Listing 6.3. As the arguments passed to this subroutine describe the root of a new MPC path inside of the graph, this tuple is added as a first element (Line 10) to the ordered list representing the MPC path γ_G. An ordered list is a common data structure used in computer science to represent a finite sequence, as it is used to describe an MPC path as introduced in (6.21).

During this subroutine, the MPC index of the last element of the MPC path is represented by l; in the beginning, this corresponds to the MPC index of the root node (Line 15).

Similar to the block loop in Listing 6.2, the block loop of the subroutine iterates from "left" to "right" over the blocks – but not necessarily starting with the 0-th block, but starting with the root block of the current MPC path (Line 18).

Within the loop, the distances from the current MPC l to all MPCs of the next block that are not yet assigned to an MPC path are computed and the minimum is determined (Line 30). Please note, that the actual call of $\mathfrak{D}$ can be substituted by a lookup in the corresponding distance matrix $\boldsymbol{D}^{(b)}$.

In case either no unassigned MPCs can be found (check in Line 24) or the determined minimum distance exceeds a threshold d_{thresh} (check in Line 33), the loop stops – corresponding to a termination of the current path. The threshold $d_{\text{thresh}} \in \mathbb{R}$ defines a maximum distance two consecutive MPCs are allowed to have within a path.

If none of these conditions is fulfilled, the index of the MPC with the minimum distance is

assigned to l (Line 40) and the new node is appended to the MPC path γ_G (Line 44). Finally, the detected MPC is marked as assigned (Line 49).

Once the loop has terminated, no matter what criterion caused the loop's termination, the subroutine returns the detected MPC path γ_G.

6.2.4 Path Improvements

The algorithm presented in Section 6.2.3 has properties that may result in a degraded MPC tracking under certain conditions. We suggest the following approaches to address these issues.

6.2.4.1 MPC Path Merging

In case an MPC remains undetected for the duration of a single (or more) block(s), e.g. because it does not exceed the detection threshold, an actual MPC path is detected as multiple individual MPC paths. To detect and connect these MPC path segments, we evaluate the distances between the last element of all MPC paths and the root elements of all other MPC paths:

We first set a threshold $\beta_{\Delta\max} \in \mathbb{N}$ that defines the maximum length of interruptions given in blocks that we consider as acceptable. We then iterate over all permutations of all MPC paths in $\mathcal{P}$ represented by $\gamma_G^{(m)}$ and $\gamma_G^{(n)}$ to detect all MPC path combinations, where

$$0 < b_n^{\triangleright} - b_m^{\square} < \beta_{\Delta\max}, \tag{6.22}$$

holds. The MPC paths $\gamma_G^{(m)}$ and $\gamma_G^{(n)}$ are merged if

$$d_{l_m^{\square} \to l_n^{\triangleright}}^{(b_m^{\square} \to b_n^{\triangleright})} < d_{\text{thresh}} \tag{6.23}$$

is fulfilled. In case this condition holds for multiple MPC path pairs, the one with the lowest distance is chosen.

It is obvious, that the merging of two MPC paths, as it is described above, leads to MPC paths having undefined values at $b_m^{\square} < b < b_n^{\triangleright}$. For this case, the popular cubic spline interpolation[2] can be used to generate the missing values. However, adding such generated values to the MPC paths is a modification of the original measurement data. Especially the amplitude, which is in most cases the reason why an MPC got "missed" during the MPC detection process, is likely to get overestimated when applying the suggested interpolation. Nevertheless, the decision whether to artificially generate the amplitudes of missing MPC path segments using an interpolation approach depends on the probability of missing an MPC during the MPC detection process.

[2]In case the MPC path contains of less than 4 elements, a cubic spline interpolation cannot be applied, thus a linear interpolation might be used as a fall back.

```
1  sub detect_path(
2          int root_block_index,
3          int root_mpc_index
4      ) → list:
5
6      # initialize ordered list for new path
7      list γ_G = ()
8
9      # add tuple defining MPC path root
10     γ_G.append(
11         (root_block_index, root_mpc_index)
12     )
13
14     # init MPC index for loop
15     int l = root_mpc_index
16
17     # iterate over all subsequent blocks
18     for int b in range(root_block_index,num_blocks-2):
19
20         # get list of unassigned MPCs in next block
21         list avail_indices = index( Γ^(b+1) == 0 )
22
23         # check if all MPCs are already assigned
24         if empty(avail_indices):
25
26             # they are → break loop
27             break
28
29         # find minimum distance by applying (6.18)
30         float d_min = min_{∀ l̃∈avail_indices} { D { ξ_l^(b), ξ_l̃^(b+1) } }
31
32         # check if min. distance is above threshold
33         if d_min > d_thresh:
34
35             # it is → break loop
36             break
37
38         # get index of MPC with min. distance and use it as new MPC index
39         # for next iteration
40         l = argmin_{∀ l̃∈avail_indices} { D { ξ_l^(b), ξ_l̃^(b+1) } }
41
42         # add tuple describing node position within the graph
43         # to the current path
44         γ_G.append(
45             (b+1, l)
46         )
47
48         # mark MPC as assigned
49         Γ^(b+1)[l] = 1
50
51     # return the list describing the path
52     return γ_G
```

LISTING 6.3: Pseudo code of the `detect_path` subroutine called by Listing 6.2. Comments are marked with a leading # sign.

6.2.4.2 Short MPC Path Elimination

MPCs that appear for the duration of just one block are likely to be the results of a misdetection. In this case, the algorithm in Section 6.2.3 assigns these single MPCs to individual MPC paths of length one. To only keep track of MPCs that have been detected for a longer duration, a threshold $\beta_{\text{min}} \in \mathbb{N}$ can be set, that defines a minimum length of MPC paths. Consequently, all MPC paths that do not fulfil

$$\beta^{(\rho)} \geq \beta_{\text{min}} \tag{6.24}$$

are then eliminated.

If the proposed MPC tracking approach is used for channel modelling, those MPCs that have been assigned to distinct MPC paths which have not been eliminated can be modelled by a process that generates MPC paths. However, the MPCs assigned to one of the eliminated MPC paths should not be ignored: We interpret these MPCs as a result of diffuse scattering and suggest to model them by a separate process. Thus, β_{min} is understood as a parameter that has an influence on the amount of MPCs that are modelled by the process that generates MPC paths. The higher this amount, the smaller the amount of MPCs that are modelled by the separate process.

6.2.4.3 Low-Pass Filtering

In the case of measurement data collected in a setup as it is was used in the measurement campaign described in Chapter 3, sudden "jumps" in the position of the reflectors causing the detected MPCs are very unlikely: First, both the transmitter and (most) of the reflectors are fixed. Second, a civil aircraft follows a smooth flight trajectory. So, the time evolution of the relative geometry can be considered as smooth.

Based on this assumption, it is unlikely that the actual evolution of the location dependent parameters of an MPC caused by a reflector in a fixed position – the delay and the Doppler shift, see Section 6.3 – shows sudden jumps either. Nevertheless, as it will be seen later when applying the processing chain to measurement data, these jumps can be observed. Following the above reasoning, they can be considered as processing artifacts due to

- the limited resolution of the DDSF, mostly along the Doppler axis, or
- a value for d_{thresh} that is unsuitable (i.e. too large) for the given data.

In the latter case, the algorithm assigns MPCs to a path, that are in fact too "far away" from the remaining elements of the MPC path and should be assigned to a separate MPC path instead.

Applying a low-pass filter to the MPC parameters of an MPC path that that are subject to jumps that are considered as processing artifacts helps smoothing the corresponding curve. An example for the application of this filtering is given in Section 6.2.6.

6.2.5 Alternative Path Notation

6.2.5.1 Graph-Indepentent MPC Path Notation

So far, a detected MPC path is defined as a sequence of tuples providing the position of each MPC of the MPC path within the corresponding MPC graph. While this reference-based notation is helpful for explaining the MPC path detection process, it is more convenient to have a graph-independent description of the detected MPC paths for the following processing steps. Therefore, an alternative notation as proposed in [60] is now introduced.

To allow a straightforward addressing of the items of the sequence representing an MPC path indexed by ρ, a path-specific index called *local* block index $b^{(\rho)}$ is introduced. The conversion between this *local* block index and the *global* block index b that is already familiar to the reader is performed by

$$b = b_\rho^{\triangleright} + b^{(\rho)}, \tag{6.25}$$

where $b_\rho^{\triangleright}$ denotes the global block index of the very first block of MPC path $\gamma_G^{(\rho)}$.

For example, the orange MPC path in Figure 6.5 starts at the global block index $b = 1 = b_{\text{orange}}^{\triangleright}$. However, within the scope of the orange path, this element is the zeroth one, thus $b^{(\text{orange})} = 0$.

The local block index can now be used to define an MPC path as a sequence of MPCs. Using the alternative notation for finite sequences from Appendix A.2, an MPC path is now given by:

$$\gamma^{(\rho)} = \left(\xi^{(b^{(\rho)})} \right)_{b^{(\rho)}=0}^{\beta^{(\rho)}-1}, \tag{6.26}$$

which is, in contrast to the definition in (6.21), a graph-independent definition of an MPC path. As long as the corresponding global start block index $b_\rho^{\triangleright}$ is known, it is always possible to paste an MPC path of the description given above into a global setting (i.e. like an MPC graph) by applying (6.25).

6.2.5.2 Evolution of a Single MPC Parameter

The MPC path notation introduced in (6.26) allows the description of the evolution of all parameters of the respective MPC as a whole. In case the evolution of just a single parameter of an MPC path is of interest, the following notation is used: The evolution of the delay of an MPC path indexed by ρ delay is denoted by

$$\gamma_\tau^{(\rho)} = \left(\tau^{(b^{(\rho)})} \right)_{b^{(\rho)}=0}^{\beta^{(\rho)}-1} \tag{6.27}$$

and the evolution of this path's Doppler shift is denoted by

$$\gamma_\nu^{(\rho)} = \left(\nu^{(b^{(\rho)})} \right)_{b^{(\rho)}=0}^{\beta^{(\rho)}-1}. \tag{6.28}$$

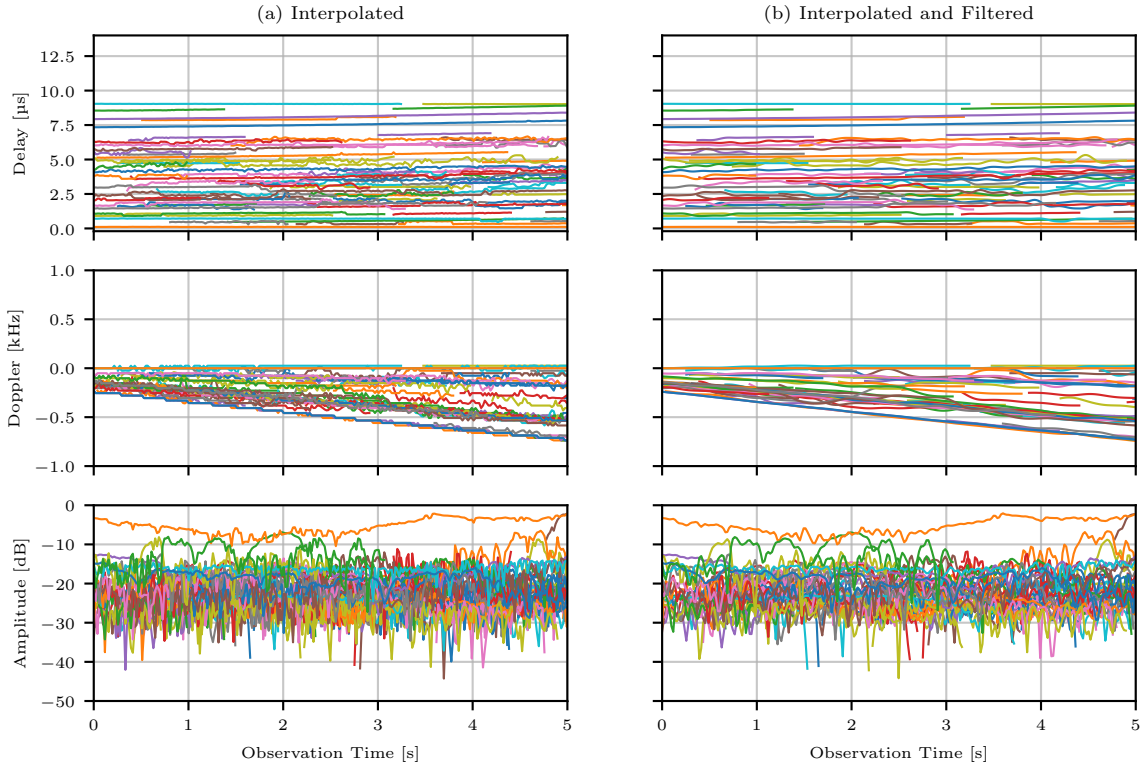


FIGURE 6.6: A set of MPC-paths showing the evolution of delay (top), Doppler (middle), and the absolute value of the weight (bottom) of the detected MPCs during take-off. In both (a) and (b) missing values were interpolated using a cubic spline interpolation where possible. In (b), an additional low-pass filter has been applied to the data of (a) to smooth the evolution and to reduce the impact of outliers. For the sake of clearness, the plots show only paths with a minimum length of 50 blocks.

6.2.6 Application on Measurement Data

The MPC tracking explained in the previous section has been applied to measurement data collected during the flight measurement campaign introduced in Chapter 3. The results of this processing are introduced and discussed in the following.

6.2.6.1 Raw MPC Path Data

Figure 6.6 shows a 5 s cutout of the evolution of the MPC components represented as MPC paths. The MPCs were detected using the approach shown in Section 6.1.2; the MPC paths were found using the algorithm presented in Section 6.2.3 using a feature scaling as presented in (6.20) and the distance function given in (6.19) using $a_\tau = 0.5$, $a_\nu = 0.5$, and $a_\alpha = 0$. The underlying data were recorded while the aircraft was on the runway and accelerating for take-off. For the sake of clearness, short MPC paths having a length less than 50 blocks have been excluded from the plots.

6.2.6.2 Interpolated Data Points

Figure 6.6(a) shows the MPC paths as they were detected by the algorithm including the application of the MPC path improvements presented in Section 6.2.4. Due to the *path merging*,

MPC paths with missing blocks are part of the detected MPC path set. Here, a cubic spline interpolation was applied to estimate the values of these missing blocks.

Most of the Doppler shifts increase in value over time by becoming more negative. This is caused by the increasing speed (and therefore higher absolute Doppler shift) of the aircraft during take-off and by the relative movement of the aircraft with respect to the reflectors like airport buildings, trees and fences.

The figure shows no MPC paths with a positive Doppler shift. This matches our observations from Figure 6.2, where no strong MPCs with a positive Doppler shift were detected. In both cases, the underlying data was recorded during the same take-off; however, the data shown in Figure 6.2 was recorded shortly after the data presented in Figure 6.6.

The Doppler shifts, especially the shifts of the MPC paths with the lowest Doppler shifts, show a step-wise unsteady evolution. From the plot this step size can be determined as 25.4 Hz which corresponds to the Doppler resolution of the DDSFs the underlying MPCs got extracted from. We therefore consider these steps not as a natural effect, but a processing artifact that will be addressed below.

The amplitudes of the detected components, including the LOS component given in orange, show a typical small scale fading behaviour. We assume this fading of the individual components is caused by non-resolvable MPCs that add up either constructively or destructively.

6.2.6.3 Filtered Data Points

Figure 6.6(b) shows low-pass filtered versions of the MPC paths presented in Figure 6.6(a). The filtering was applied for the reasons explained in Section 6.2.4: Especially the unsteady evolution of the Doppler shifts due to the limited Doppler resolution in Figure 6.6(a) motivates the application of the filtering.

The impact of the filtering can be clearly seen in the plot of the Doppler shifts: The step-wise leaps that dominated the plotted MPC paths in the unfiltered case are now gone and the curves look much smoother.

The effect on the delay and the amplitudes, respectively, is not as strong as for the Doppler shifts; however, the MPC paths now look smoother than before.

6.3 Reflector Localisation

6.3.1 Theoretical Description

The delay and Doppler information of an MPC can be used to estimate the corresponding reflector's location relative to the transmitter and the receiver:

- The *delay* can be used to estimate the joint distance the radio signal travelled from the transmitter to the reflector and from the reflector to the receiver. This allows to estimate the position of the reflector to be on the surface of a prolate spheroid[3] having the transmitter and the receiver in its focal points [103], [104] in the three-dimensional

[3] A prolate spheroid can be constructed by rotating an ellipse around its major axis.

case. In the two-dimensional case, the reflector is located on the border of an ellipse having the transmitter and the receiver in its focal points.

- The *Doppler shift*, however, can be used to estimate the angle of arrival of the reflected signal at the receiver[4]. In the three-dimensional case, the reflector is then located on a nappe (one half of a conical surface) having the receiver as its apex, the heading direction of the moving receiver as its main axis, and the angle of arrival as its angle. In the two-dimensional case, the reflector is located on one of two lines intersecting in the receiver's position and having the angle of arrival as their angle relative to the heading direction of the receiver.

By combining the prolate spheroid and the nappe in the three-dimensional case (or the ellipse and the lines in the two-dimensional case, respectively) it is possible to reduce the uncertainty about the reflector's position: The position is expected to be within the intersection of the two geometrical objects defined by the corresponding delay and Doppler shift, respectively.

In case the geometry between transmitter, receiver, and reflector is changing, e.g. in case the receiver is moving, it is possible to reduce the uncertainty about the reflector's position even further by tracking the corresponding MPC over time and superimposing the resulting position estimates. In the following, the reflector localisation for the two-dimensional case is explained in detail under the assumption that the MPC detection and MPC tracking approaches from the previous sections have been applied in the first place.

6.3.2 Realisation Using Point Sets

A realisation of the reflector localisation based on point sets is discussed in this section. An advantage of using point sets is that both the three-dimensional case and its projection on a plane (thus, the two-dimensional case) are covered at the same time. In the following, an application on CARTESIAN coordinates either in two dimensions or in three dimensions is assumed[5].

6.3.2.1 Exploiting Delay Information

The set of potential positions of the reflector causing MPC $\xi_l^{(b)}$ can be described using the speed of light in air c_{air}[6] and the MPC's delay τ_l by

$$\mathcal{E}_{\tau,l}^{(b)} = \left\{ P \mid d(\text{Tx}, P) + d(P, \text{Rx}) = c_{\text{air}} \tau_l' \right\}, \quad \text{with } \tau_l' = \tau_l + \tau_{\text{LOS}}^{(b)}, \tag{6.29}$$

where Tx and Rx denote the position of the transmitter and receiver, respectively, $\overrightarrow{XY}$ denotes a vector from a point X to a point Y, and $d(X, Y) = ||\overrightarrow{XY}||$ denotes the EUCLIDEAN distance between a point X and a point Y – depending on the case of application either in $\mathbb{R}^2$ or in $\mathbb{R}^3$. Please note, that the LOS compensation introduced in Section 4.2.2.3 must be reverted

[4] A fixed transmitter and a moving receiver is assumed here.

[5] The required inner product and the required norm is defined for both the two- and the three-dimensional EUCLIDEAN space over $\mathbb{R}$.

[6] Again, the speed of light is assumed as constant.

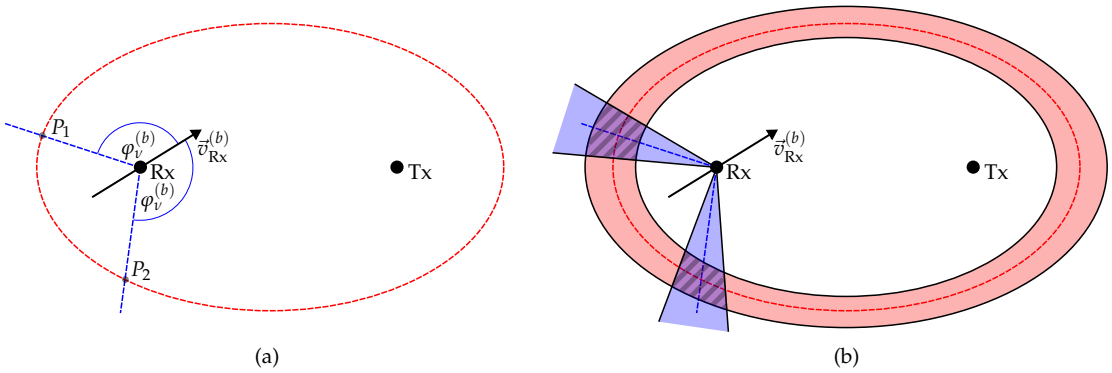


FIGURE 6.7: Visualisation of the set-based reflector localisation for the two-dimensional case: (a) General principle assuming perfect MPC estimates (b) Taking uncertainties due to limited resolution into account.

by adding the estimated delay of the LOS component for the data block MPC l has been detected in.

In the two-dimensional case, the point set $\mathcal{E}_{\tau,l}^{(b)}$ corresponds to an ellipse whose focal points are given by the position of the transmitter and the receiver. This ellipse is visualised by a dashed red contour in the sketch given in Figure 6.7(a).

6.3.2.2 Exploiting Doppler Information

Using the definition given in (2.2), the resulting Doppler shift ν_l of an MPC $\xi_l^{(b)}$ under the assumption of a static reflector located in P and a receiver located in Rx moving along $\vec{v}_{\mathrm{Rx}}^{(b)}$ – a value derived from the ground truth data, see Section 4.1 – is given by

$$\nu_l = \frac{f_\mathrm{c}}{c_\mathrm{air}} \left\langle \vec{v}_{\mathrm{Rx}}^{(b)}, \vec{e}_P \right\rangle - \nu_{\mathrm{LOS}}^{(b)} = \frac{f_\mathrm{c}}{c_\mathrm{air}} \left| \vec{v}_{\mathrm{Rx}}^{(b)} \right| \cos \left\{ \overbrace{\angle \left(\vec{v}_{\mathrm{Rx}}^{(b)}, \vec{e}_P \right)}^{=\varphi_\nu^{(b)}} \right\} - \nu_{\mathrm{LOS}}^{(b)}, \tag{6.30}$$

with the unit vector describing the direction from the receiver to the point P:

$$\vec{e}_P = \frac{\overrightarrow{\mathrm{Rx}P}}{||\overrightarrow{\mathrm{Rx}P}||}. \tag{6.31}$$

This equation can be resolved for the angle of arrival $\varphi_\nu^{(b)}$:

$$\varphi_\nu^{(b)} = \arccos \left\{ \frac{c_\mathrm{air} \nu_l'}{f_\mathrm{c} |\vec{v}_{\mathrm{Rx}}^{(b)}|} \right\}, \quad \text{with } \nu_l' = \nu_l + \nu_{\mathrm{LOS}}^{(b)}. \tag{6.32}$$

The location of the corresponding reflector is then in the set given by

$$\mathcal{E}_{\nu,l}^{(b)} = \left\{ P \,\middle|\, \angle \left(\vec{v}_{\mathrm{Rx}}^{(b)}, \vec{e}_P \right) = \varphi_\nu^{(b)} \right\}. \tag{6.33}$$

The reader may note the non-resolvable ambiguity in this definition caused by the nature of the inverse cosine function used to compute $\varphi_\nu^{(b)}$: Since the inverse cosine function's value

range is $[0, \pi]$, see e.g. [13], it is not possible to get an unambiguous result – not even for the two-dimensional case.

Evaluating the sign of ν_l' can help to verify the previous calculations: In case the sign is positive, Rx is travelling towards the reflector, in case it is negative, Rx is travelling away from the reflector.

In the two-dimensional case, this set corresponds to two lines intersecting in the location of the receiver having an angle of $\pm\varphi_\nu^{(b)}$ with respect to the travelling direction of the receiver. As mentioned above, it is not possible to resolve this ambiguity any further at this stage. The resulting two lines correspond to the dashed blue lines in Figure 6.7(a).

6.3.2.3 Localisation by Information Combining

By combining the information gathered from the delay and the Doppler shift of the respective MPC, the uncertainty about the position of the corresponding reflector can be reduced: The reflector's position is in the intersection

$$\mathcal{E}_l^{(b)} = \mathcal{E}_{\tau,l}^{(b)} \cap \mathcal{E}_{\nu,l}^{(b)}. \tag{6.34}$$

In the two-dimensional case, this corresponds to the two intersection points highlighted by grey circles labelled P_1 and P_2, respectively, in Figure 6.7(a).

By taking side information, like knowledge on the given geometry (position of buildings, mountains,...), into account, ambiguities can be resolved even further. However, this is considered as beyond of scope of this section.

6.3.3 Point Sets and Limited MPC Parameter Resolution

The approach in the previous section assumed a very high precision of the parameter estimates of the MPC. However, it is unlikely that this high level of precision can actually be achieved in real-world measurements, e.g. due to imperfect GT data. Also, the resolution during the detection step of the MPCs is limited. In the following, an approach that is more robust against these limitations is presented.

6.3.3.1 Exploiting Delay Information

In case the coarse MPC detection approach as shown in Section 6.1.2 is used, the limited delay resolution Δ_τ caused by the DDSF comes into account. This aspect can be addressed by using two prolate spheroids/ellipses sharing the same focal points: One *inner* figure representing the shorter delay $(\tau_l' - \Delta_\tau/2)$ and one *outer* figure representing the larger delay $(\tau_l' + \Delta_\tau/2)$. The reflector's position is then in the point set defined by

$$\begin{aligned}\mathcal{E}_{\tau,l}^{(b)} &= \overbrace{\left\{P \;\middle|\; d(\mathrm{Tx},P) + d(P,\mathrm{Rx}) < d_{\mathrm{out}}\right\}}^{\text{outer figure}} \setminus \overbrace{\left\{P \;\middle|\; d(\mathrm{Tx},P) + d(P,\mathrm{Rx}) < d_{\mathrm{in}}\right\}}^{\text{inner figure}} \\ &= \left\{P \;\middle|\; d_{\mathrm{in}} \leq d(\mathrm{Tx},P) + d(P,\mathrm{Rx}) < d_{\mathrm{out}}\right\}.\end{aligned} \tag{6.35}$$

with the fixed distances

$$d_{\text{out,in}} = c_{\text{air}} \left(\tau_l' \pm \frac{\Delta_\tau}{2} \right). \tag{6.36}$$

For the two-dimensional case, thus when the figures refer to ellipses, the resulting point set corresponds to the light-red shape in Figure 6.7(b).

6.3.3.2 Exploiting Doppler Information

Analogous to the definition of the point set derived from the delay – while taking the limited delay resolution into account –, the point set derived from the Doppler shift – while taking the limited Doppler resolution Δ_ν into account – is given by the set

$$\mathcal{E}_{\nu,l}^{(b)} = \begin{cases} \left\{ P \;\middle|\; \varphi_{\nu,\text{B}}^{(b)} \leq \angle \left(\vec{v}_{\text{Rx}}^{(b)}, \vec{e}_P \right) < \varphi_{\nu,\text{A}}^{(b)} \right\}, & \text{if } \varphi_{\nu,\text{A}}^{(b)} > \varphi_{\nu,\text{B}}^{(b)} \\ \left\{ P \;\middle|\; \varphi_{\nu,\text{A}}^{(b)} \leq \angle \left(\vec{v}_{\text{Rx}}^{(b)}, \vec{e}_P \right) < \varphi_{\nu,\text{B}}^{(b)} \right\}, & \text{if } \varphi_{\nu,\text{A}}^{(b)} < \varphi_{\nu,\text{B}}^{(b)} \\ \left\{ P \;\middle|\; \angle \left(\vec{v}_{\text{Rx}}^{(b)}, \vec{e}_P \right) = \varphi_{\nu,\text{A}}^{(b)} \right\}, & \text{else} \end{cases} \tag{6.37}$$

with the fixed angles

$$\varphi_{\nu,\{\text{A,B}\}}^{(b)} = \arccos \left\{ \frac{c_{\text{air}} (\nu \pm \frac{\Delta_\nu}{2})}{f_{\text{c}} |\vec{v}_{\text{Rx}}^{(b)}|} \right\}. \tag{6.38}$$

In the two-dimensional case, this set corresponds to two segments of a circle with an infinite radius. In Figure 6.7(b), these segments are represented by the light-blue shapes.

6.3.3.3 Localisation by Information Combining

The final position of the reflector is lying in the intersection of the sets $\mathcal{E}_l^{(b)} = \mathcal{E}_{\tau,l}^{(b)} \cap \mathcal{E}_{\nu,l}^{(b)}$.

In the two-dimensional example given in Figure 6.7(b), this intersection is highlighted by a bar pattern.

6.3.3.4 Exploiting Temporal Evolution

The motivation for the MPC tracking presented in Section 6.2 was to track a certain MPC over time. It is assumed here that this MPC is caused by the same reflector for the time it was trackable. At the same time it is assumed that this specific reflector is static – which is the case for buildings and other obstacles around an airport –, that the transmitter is static, and that the receiver was moving during the recording of the data the MPC tracking is applied to. Thus, the relative geometry is changing over time, while the absolute position of the reflector is assumed to be constant. This circumstance can be exploited to reduce the uncertainty about the position of the reflector by performing the above reflector localisation for each block $b^{(\rho)}$ of an MPC path $\gamma^{(\rho)}$. Under the condition that a common coordinate system was used for the block-wise reflector localisation, the intersection of all resulting point sets contains the

position of the reflector:

$$\mathcal{E}_{\boldsymbol{\gamma}^{(\rho)}} = \bigcap_{\forall b^{(\rho)} \in \boldsymbol{\gamma}^{(\rho)}} \mathcal{E}^{(b^{(\rho)})}. \tag{6.39}$$

6.3.4 Practical Implementation

In a practical implementation for the three-dimensional case, the point sets from above are implemented using multidimensional arrays [42] in $\mathbb{R}^{X \times Y \times Z}$, where $X, Y, Z \in \mathbb{N}$.

The arrays are a quantised representation of the observed scenario in CARTESIAN coordinates. Since the transmitter in the scenario discussed here is static, it is convenient to use a transmitter-centred ENU coordinate system to describe the geometry of this scenario. Thus, the transmitter is located at $(0, 0, 0)_{\mathrm{ENU}}$ which corresponds to the arbitrarily chosen array element at $(x_{\mathrm{Tx}}, y_{\mathrm{Tx}}, z_{\mathrm{Tx}})$. The ENU coordinates of the array element at (x, y, z) are thus given by $((x - x_{\mathrm{Tx}})\Delta_{\mathrm{E}}, (y - y_{\mathrm{Tx}})\Delta_{\mathrm{N}}, (z - z_{\mathrm{Tx}})\Delta_{\mathrm{U}})_{\mathrm{ENU}}$, where Δ_{E}, Δ_{N}, and Δ_{U} denote the chosen spacial resolution along the East-, North-, and Up-axis, respectively.

The application of (6.35) and (6.37), respectively, correspond to an element-wise check whether the corresponding condition is fulfilled for an array element or not. If the condition is fulfilled, the corresponding element is set to 1, if it is not, it is set to 0. These computations are performed in ENU coordinates using the conversion introduced above.

For an MPC $\xi_l^{(b)}$, the point set defined by the delay (6.35) is implemented as the multidimensional array $\boldsymbol{M}_{\tau,l}^{(b)} \in \mathbb{R}^{X \times Y \times Z}$ and the point set defined by the Doppler shift (6.37) is implemented as the multidimensional array $\boldsymbol{M}_{\nu,l}^{(b)} \in \mathbb{R}^{X \times Y \times Z}$.

The processing step corresponding to the procedure in (6.34) – thus the intersection of the two point sets – is performed by an element-wise multiplication of the two arrays:

$$\boldsymbol{M}_l^{(b)} = \boldsymbol{M}_{\tau,l}^{(b)} \cdot \boldsymbol{M}_{\nu,l}^{(b)}. \tag{6.40}$$

The operation given by (6.39) is implemented by computing the element-wise average over all block-specific arrays of the current MPC path:

$$\boldsymbol{M}_{\boldsymbol{\gamma}^{(\rho)}} = \frac{1}{\beta^{(\rho)}} \sum_{\forall b^{(\rho)} \in \boldsymbol{\gamma}^{(\rho)}} \boldsymbol{M}^{(b^{(\rho)})}, \text{ with } \boldsymbol{M}_{\boldsymbol{\gamma}^{(\rho)}} \in \mathbb{R}^{X \times Y \times Z}. \tag{6.41}$$

Using the average instead of performing a binary decision has the benefit that the processing becomes more robust against outliers and other artifacts appearing in practical measurement data.

The practical implementation for the two-dimensional case is analogous to the three-dimensional case: Instead of three-dimensional arrays, two-dimensional arrays (matrices) are used, e.g. $\boldsymbol{M}_{\tau,l}^{(b)} \in \mathbb{R}^{X \times Y}$ is used to implement the concept of the point set defined in (6.35). Consequently, the Up-dimension of the ENU coordinates is ignored.

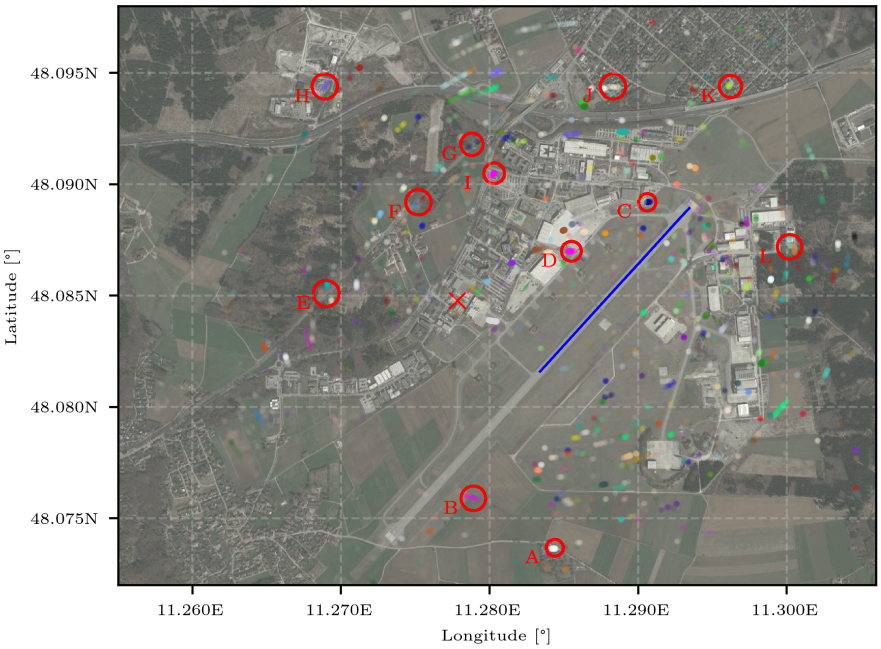


FIGURE 6.8: Map of the EDMO airport with estimated positions of reflectors: The red cross marks the position of the transmitter, the blue line on the runway shows the track of the aircraft during the recording of the displayed data. Each reflectors' position was estimated by evaluating the nodes of the corresponding MPC path as given in Figure 6.6(b). Only paths of a minimum length of 50 blocks were considered. The locations of a subset of the most prominent reflectors are highlighted by red circles. Background image: ©2020 GeoBasis-DE/BKG, Google Earth, plot powered by [55].

6.3.5 Application on Measurement Data

Figure 6.8 shows a map of the EDMO airport. The position of the transmitter of the channel sounding signal is marked by a red cross. The track the aircraft has travelled while the data that got evaluated for the reflector localisation has been recorded is denoted by a blue line; the aircraft started heading south-west. The MPC paths that were used to perform the reflector localisation correspond to those shown in Figure 6.6(b).

The MPC path based reflector localisation works by superimposing the results of the reflector localisation for each individual MPC of an MPC path as described above. The longer an MPC path is, i.e. the more persistent an MPC is, the better will be the result of the estimation, as outliers and even ambiguities are averaged out due to the superimposition.

In the figure, the certainty about the position of a reflector is represented by the data point's opacity. A colour code is used to let the reader distinguish, what data points belong to a common path. However, as the amount of detected paths exceeds the amount of available colours by far, the colours are not exclusively mapped to a specific path.

While the comparatively high delay resolution allows a pretty precise estimation of the ellipse the reflector is located on, the estimation of the angle suffers from the comparatively low Doppler resolution. Although the superimposition helps to improve the precision of the angle estimation, the shape of the data point clusters shows the lack of precision of the angle estimation. However, in most of the cases the algorithm was able to resolve the ambiguities caused by the Doppler-to-angle conversion: The actual reflector's position has been identified, while the mirrored reflector's position got averaged out.

In the following, we want to discuss a few of the most prominent reflectors highlighted in the map.

The reflector at *A* in the south has been identified as a shelter with a metal roof. The reflectors around *B* have been identified as a fence that is part of the airfield barrier.

Many reflectors have also been found close to the aircraft along the runway: The reflectors both in *C* and *D* are very persistent and have been identified as airport service buildings.

The reflectors in *E*, *F*, and *G* are most likely related to the railway lines along which they have been detected. Their origin might be the power lines along the railway, the rails or even a train that passed by during recording.

A few reflectors have been detected in the mining area north of the highway at *H*; the reflector in *I* is located between a parking lot and an intersection with traffic lights.

The reflectors in *J* and *K* are located in the suburbs north of the highway and are most likely some higher buildings.

East of the runway we have highlighted the building in *L*, as it is also the origin of a quite persistent reflector.

Chapter 7

A C-Band Channel Model for the Aeronautical Channel

7.1 Introduction

Before the actual channel model is presented, the approach that is used for modelling the channel is explained. The concept discussed in this dissertation is based on the MPC paths introduced in Section 6.2.2: So far, these MPC paths were used to represent the temporal evolution of MPCs that have been detected in either the PDP or the DDSF. For the channel modelling, this processing chain is executed in reverse direction: First of all, a set of MPC paths is generated. This set's MPC paths are then assembled to a structure similar to the MPC graph with its detected MPC paths. In a last step, this structure is then evaluated block by block: For each block, an impulse response is generated according to the definition of this block's MPCs. As all this is synthetic data, the block rate can – theoretically – be set as high as required, thus it is possible to sample impulse responses from this structure with an arbitrary high rate.

It is clear from the above paragraph that the key question is how these MPC paths can be generated. This problem can be described as the generation of

- four-dimensional
- time-series data
- of varying length.

The *four dimensions* follow from the MPC definition given in (2.4). We talk about *time-series data*, since the MPC paths represent the temporal evolution of values. As we have seen in the previous chapter, the MPC paths are likely to have a *varying length*, since the "lifetime" of a reflector's signal component varies.

The main idea of the approach presented here is to train a model using the MPC paths detected during the processing of measurement data recorded during a certain flight scenario. The model is then used to generate new MPC paths with similar statistical properties to the MPC paths detected in the measurement data, while the desired level of similarity is determined by the user.

Using time-series data of varying length for the training of a model and using this model for the generation of time-series data of varying length is not a straight forward task since many training approaches expect all data to have the same dimensions, see e.g. [28].

In the following, a few approaches of how such data can still be used to train a model are briefly discussed.

Padding A common procedure to equalise the length of a set of time-series data is a simple padding, either using zeros ("zero-padding") or, to avoid sudden jumps that may spoil the data set, by using the corresponding edge value(s). These padded data vectors can than be used to train a model that expects data vectors of the same dimensions.

Scaling Scaling is a common pre-processing step to equalise the dimensions of the input data during the training phase of a model. It is extremely common to scale two-dimensional data like images before the actual processing (e.g. the training), see e.g. [28]. However, an image-scaling-like pre-processing step is not applicable to the time-series data we face here.

Recurrent Neural Networks RNNs *(Recurrent Neural Networks)* are a common tool to process time-series data and are also capable of handling data of varying length, as e.g. shown in [69]. In this publication, a GAN *(Generative Adversarial Network)* is created using two RNNs; a GAN can be used to generate artificial data that matches the statistical properties of the training data. In the author's opinion, this general concept of forming a GAN from two RNNs to generate time-series data is a promising approach to the problem at hand. However, an approach based on neural networks is beyond the scope of this dissertation and is therefore considered as future work.

Compact Representation A close analysis of the temporal evolution of the MPC parameters over time and considering the geometrical and physical background that explains this evolution suggests the discussion of a compact representation, that uses a fixed set of parameters to describe the relevant aspects of the parameters evolution. Thus, each MPC path can be represented by a vector with fixed dimensions, independently of the actual length of the corresponding MPC path. The set of these representation vectors is then further processed: It is possible to analyse its statistics and use these findings later for the generation of new data, i.e. it can be used to train a model. In contrast to the other approaches, this approach is not a generic one that can be applied to arbitrary time-series data, but rather an approach tailored to the described use case. It is further elaborated in the next section.

7.2 Channel Modelling Using MPC Paths

7.2.1 Compact MPC Path Representation

In this section, it is investigated how the temporal evolution of MPC parameters can be represented in a compact form. This compact form, as it was initially proposed in [60], plays

an important role in the later channel modelling. As a certain loss of precision is considered as acceptable, this can be understood as a lossy compression.

7.2.1.1 Delay and Doppler Shift Evolution

A simplified representation of the temporal delay evolution has been discussed in literature already: For example, the authors of [33] try to model the delay evolution using a linear approach, thus a polynomial of first degree. However, according to (2.3), the Doppler shift can be understood as the negative change rate of the delay over time multiplied by the carrier frequency used for transmission; thus there is a linear relationship between the Doppler shift and the delay's temporal derivative. Following this assumption and using a polynomial of first degree (i.e. a linear line) as in [33] to represent the delay's evolution results in a constant Doppler shift over time.

From the results presented in the previous chapters (e.g. Figure 6.6), it is known that most of the Doppler shifts of the MPCs detected in the measurement data are not constant, but are evolving over time. Consequently, to allow non-constant Doppler shifts, another approach is introduced that does not represent the evolution of an MPC's delay by a linear line, but by a polynomial of second degree that we denote by $p_\tau^{(\rho)} : \mathbb{R} \mapsto \mathbb{R}$. The definition of this polynomial is

$$p_\tau^{(\rho)}(t) = p_2^{(\rho)} t^2 + p_1^{(\rho)} t + p_0^{(\rho)}. \tag{7.1}$$

The units of the polynomial coefficients are as follows: $p_2^{(\rho)}$ is given in Hz, $p_1^{(\rho)}$ is unit-less, and $p_0^{(\rho)}$ is given in s. We also introduce a vector representation in $\mathbb{R}^3$ of this polynomial: $\boldsymbol{p}_\tau^{(\rho)} = [p_2^{(\rho)},\ p_1^{(\rho)},\ p_0^{(\rho)}]$.

To find the polynomial that models the Doppler evolution, (2.3) is applied to (7.1):

$$\begin{aligned} p_\nu^{(\rho)}(t) &= -\frac{\mathrm{d}}{\mathrm{d}t} p_\tau^{(\rho)}(t)\ f_\mathrm{c} \\ &= -\frac{\mathrm{d}}{\mathrm{d}t} \left(p_2^{(\rho)} t^2 + p_1^{(\rho)} t + p_0^{(\rho)} \right)\ f_\mathrm{c} \\ &= -\left(2 p_2^{(\rho)} t + p_1^{(\rho)} \right)\ f_\mathrm{c}. \end{aligned} \tag{7.2}$$

In the following it is investigated how the polynomial coefficients are determined.

Straight Forward Determination of Polynomial Coefficients A common approach to determine polynomial coefficients by fitting to some existing data is the LS approach that minimises the squared error between the polynomial (here: $p_\tau^{(\rho)}(t)$) and the data it is supposed to fit to (here: $\gamma_\tau^{(\rho)}$). Using the substitution $t = b^{(\rho)} T_\mathrm{B}$, where $T_\mathrm{B} = \frac{BN}{f_\mathrm{sr}}$ denotes the duration of a processing block, the optimisation problem is given by

$$\boldsymbol{p}_{\tau}^{(\rho)} = \operatorname{argmin} \left\{ \sum_{b^{(\rho)}=0}^{\beta^{(\rho)}-1} \left| \underbrace{p_{\tau}^{(\rho)}\left(b^{(\rho)} T_{\mathrm{B}}\right)}_{(7.1)} - \underbrace{\gamma_{\tau}^{(\rho)}[b^{(\rho)}]}_{(6.27)} \right|^2 \right\}. \tag{7.3}$$

The well-established NELDER-MEAD approach [65] as implemented in [101] is used to solve this optimisation problem.

The reader may note that the fitting of the polynomial in (7.3) is performed using the *local* block index $b^{(\rho)}$ and not the *global* block index b, see Section 6.2.5. This is beneficial, as this approach provides a time-independent polynomial representation of the delay evolution which is a requirement for the later processing.

Improved Determination of Polynomial Coefficients The approach in (7.3) only takes the information of the MPC path delay into account when determining the polynomial coefficients. In case this fitting approach is applied to measurement data, that are always subject to inaccuracies and mis-detections, the fitting is influenced by these degrading measurement artifacts. This erroneous fitting has been observed to result in numerical instabilities, especially in the case of the derived Doppler polynomial: The multiplication by the carrier frequency – usually a large number in the range of millions or even billions – makes this equation vulnerable to these effects.

A numerically more robust solution can be found by taking both the delay information $\gamma_{\tau}^{(\rho)}$ as given in (6.27) and – if present – the Doppler information $\gamma_{\nu}^{(\rho)}$ as given in (6.28) into account when finding the polynomial coefficients. This approach is also based on minimising squared errors, here denoted by $\boldsymbol{\epsilon}_{\tau}^{(\rho)} \in \mathbb{R}^{\beta^{(\rho)}}$ for the error of the delay evolution fitting and $\boldsymbol{\epsilon}_{\nu}^{(\rho)} \in \mathbb{R}^{\beta^{(\rho)}}$ for the error of the Doppler shift evolution fitting.

The squared error between the polynomial fit on the path's delay evolution as given in (7.1) and the path's actual delay evolution is basically given by the term inside of the summation in (7.3). Thus, the $b^{(\rho)}$-th element of the error vector is given by

$$\boldsymbol{\epsilon}_{\tau}^{(\rho)}[b^{(\rho)}] = \left| \underbrace{p_{\tau}^{(\rho)}\left(b^{(\rho)} T_{\mathrm{B}}\right)'}_{(7.1)} - \underbrace{\gamma_{\tau}^{(\rho)}[b^{(\rho)}]'}_{(6.27)} \right|^2, \tag{7.4}$$

where $\square'$ denotes a feature scaled[1] representation of $\square$. Correspondingly, the $b^{(\rho)}$-th element of the vector of the squared error between the polynomial fit on the path's Doppler evolution and the path's actual Doppler evolution is given by

$$\boldsymbol{\epsilon}_{\nu}^{(\rho)}[b^{(\rho)}] = \left| \underbrace{p_{\nu}^{(\rho)}\left(b^{(\rho)} T_{\mathrm{B}}\right)'}_{(7.2)} - \underbrace{\gamma_{\nu}^{(\rho)}[b^{(\rho)}]'}_{(6.28)} \right|^2, \tag{7.5}$$

[1]Without feature scaling, the later optimisation would actually be performed on the Doppler values only, as the Doppler error is typically several orders of magnitude larger than the delay error. A min-max scaling as given in (6.20) is applied if not denoted otherwise.

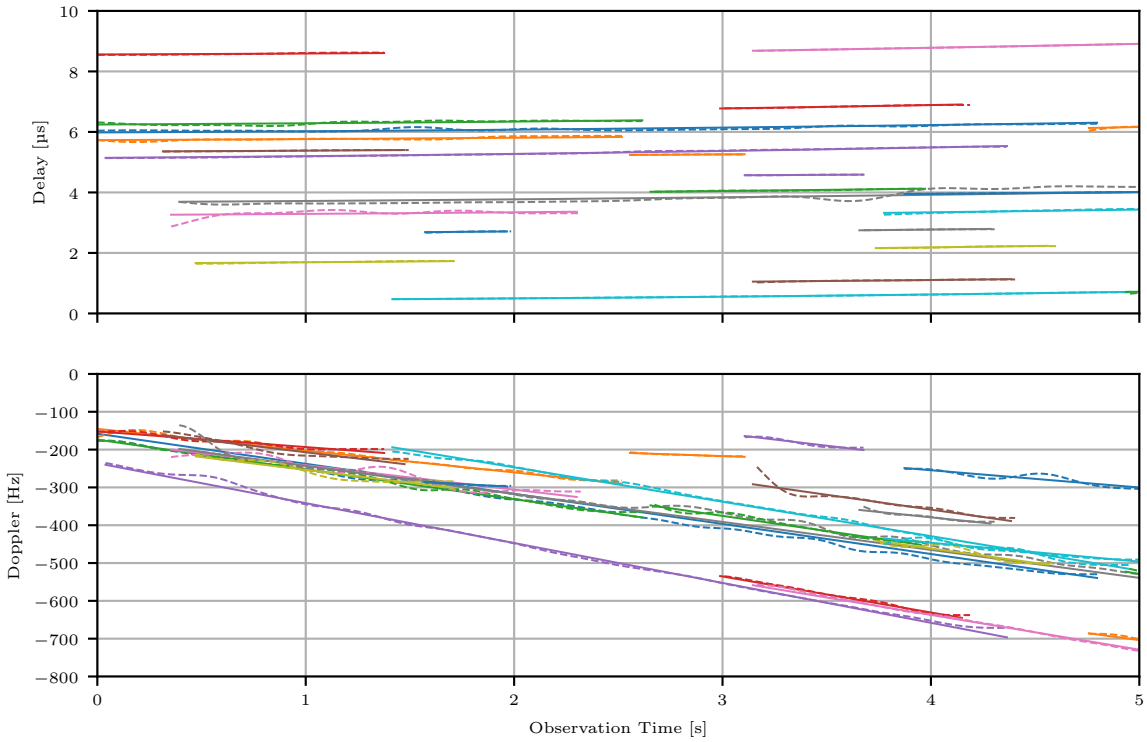


FIGURE 7.1: Delay and Doppler evolution of selected paths taken from Figure 6.6: A dashed line represents the data as it has been detected by the MPC processing chain presented in Chapters 4 and 6. A solid line of the same colour represents the evaluation of the corresponding polynomial $p_\tau^{(\rho)}(t)$ or $p_\nu^{(\rho)}(t)$, respectively, as it has been determined using the approach given in (7.6).

where – analogous to the delay error computation in (7.4) – feature scaling is applied for the same reason. The optimisation problem from (7.3) is now expanded to

$$\boldsymbol{p}_\tau^{(\rho)} = \operatorname{argmin} \left\{ \sum_{b^{(\rho)}=0}^{\beta^{(\rho)}-1} w_\tau \underbrace{\boldsymbol{\epsilon}_\tau^{(\rho)}[b^{(\rho)}]}_{(7.4)} + w_\nu \underbrace{\boldsymbol{\epsilon}_\nu^{(\rho)}[b^{(\rho)}]}_{(7.5)} \right\}, \tag{7.6}$$

where $w_{\{\tau,\nu\}} \in \mathbb{R}$ are weight factors that allow a weighting of the influence of the delay and the Doppler shift during the optimisation process. Since feature scaling was applied when computing $\boldsymbol{\epsilon}_\tau^{(\rho)}$ and $\boldsymbol{\epsilon}_\nu^{(\rho)}$, respectively, we do not see a motivation for an unequal weighting and thus set the values to $w_\tau = w_\nu = 0.5$. Again, the NELDER-MEAD approach is used to solve the optimisation problem.

The graphs in Figure 7.1 show the result of applying the described polynomial fitting approach with respect to the delay and the Doppler shift evolution to paths detected in actual measurement data. The paths the approach was applied to were taken from the data visualised in Figure 6.6; the amount of displayed paths was reduced for the sake of clearness. A dashed line is used to represent the actual data from the detected MPC path, a solid line of the same colour is used to show the evaluation of the corresponding polynomial. Both the delay and the Doppler shift evolution of a given MPC path are plotted with the same colour to allow the reader to easily compare the graphs of the two figures.

Apart from minor inaccuracies, that cannot be represented by polynomials of such low

degree and are likely to be caused by mis-detections anyway, the polynomial fits of both the evolution of the delay and the Doppler shift match the measurement data. Since the connection between delay and Doppler shift evolution is derived from established theory – i.e. BELLO's equation (2.3) – this high quality of the match is another indicator that the measurement campaign presented in Chapter 3 as well as the processing chain presented in Chapters 4 and 6 produced valid results.

7.2.1.2 Amplitude and Phase Evolution

The evolution of the MPC path amplitudes is way more chaotic and less deterministic than the evolution of delay and, consequently, the Doppler shift. Using polynomials to represent this evolution in an appropriate manner does not succeed: Polynomials of lower degree are unable to adequately represent the amplitude evolution, the application of polynomials of higher degree results in numerical instabilities. As an alternative, the statistical moments of the amplitude evolution within the measured data are determined under the assumption of RICE fading. This is motivated by the circumstance that it is very likely that the detected MPC in fact is the result of multiple individual scatterers that have not been resolved any further. Using the RICE distribution to model these fading effects is fairly common in wireless communications (see e.g. [31], [68]).

To find the parameters $(\varrho^{(\rho)}, \sigma^{(\rho)})$ of the RICE distribution describing the statistics of the amplitude evolution of an MPC path $\boldsymbol{\gamma}^{(\rho)}$, the method of moment-based parameter estimation as described in Appendix E is applied[2].

Phase Evolution The analysis of the phase evolution of an MPC path is not of interest here, since the phases will later be generated altogether with the amplitudes. Thus, the phases are not part of the compact representation.

7.2.1.3 Joint Representation

For the following processing, it is convenient to have a single vector for each MPC path that holds all information to represent the respective MPC path according to our approach described above. This includes:

- The global start time $T_{\triangleright}^{(\rho)} = b_{\rho}^{\triangleright} T_{\mathrm{B}}$ of MPC path $\boldsymbol{\gamma}^{(\rho)}$ given in seconds.
- The duration $T_{\leftrightarrow}^{(\rho)} = (b_{\rho}^{\square} - b_{\rho}^{\triangleright}) T_{\mathrm{B}}$ of MPC path $\boldsymbol{\gamma}^{(\rho)}$ given in seconds.
- The parameters of the polynomial defining the evolution of the delay over time $\boldsymbol{p}_{\tau}^{(\rho)}$ from above.
- The shape and scale parameters $(\varrho^{(\rho)}, \sigma^{(\rho)})$ of the distribution used to represent the path's amplitude, here: the RICE distribution.

[2] In those cases where the moment-based parameter estimation fails due to a lack of numerical stability, an ML *(Maximum Likelihood)* approach as implemented in [101] is applied.

Thus, the whole representation vector $\boldsymbol{r}^{(\rho)} \in \mathbb{R}^7$ of MPC path $\boldsymbol{\gamma}^{(\rho)}$ is:

$$\boldsymbol{r}^{(\rho)} = \Big[\underbrace{T_{\triangleright}^{(\rho)},\, T_{\leftrightarrow}^{(\rho)}}_{\text{Timing}},\, \underbrace{\underbrace{p_2^{(\rho)},\, p_1^{(\rho)},\, p_0^{(\rho)}}_{\boldsymbol{p}_\tau^{(\rho)}},\, \underbrace{\varrho^{(\rho)},\, \sigma^{(\rho)}}_{\text{Rice}} \Big]. \tag{7.7}$$

We denote the set of the compact descriptions of all MPC paths of an MPC path set $\mathcal{P}$ by $\mathcal{R}$.

7.2.2 Recovering MPC Path Data from Compact MPC Path Representation

The introduction of a compact representation of the MPC path evolution only makes sense when it is possible to recover[3] the data that is represented by the compact representation. This recovery process is basically the reverse of the process described in Section 7.2.1 and described in the following.

7.2.2.1 Time Vector

The first elements of $\boldsymbol{r}^{(\rho)}$ of interest for the recovering process are the elements that provide information on the timing information of the MPC path that should be generated, i.e. the start time within the global time scope $T_{\triangleright}^{(\rho)}$ and the duration $T_{\leftrightarrow}^{(\rho)}$. Since the MPC graph is discrete, these continuous values need to be converted to global block indices:

$$b_\rho^{\triangleright} = \left\lfloor \frac{T_{\triangleright}^{(\rho)}}{T_\mathrm{B}} \right\rfloor \quad \text{and} \quad b_\rho^{\square} = \left\lceil \frac{(T_{\triangleright}^{(\rho)} + T_{\leftrightarrow}^{(\rho)})}{T_\mathrm{B}} \right\rceil, \tag{7.8}$$

where T_B represents the duration of one block – which is not necessarily the same that was used in the MPC tracking stage. Thus, the length given in blocks $\beta^{(\rho)} \in \mathbb{N}$ is $\beta^{(\rho)} = b_\rho^{\square} - b_\rho^{\triangleright}$.

The time vector with respect to the *global* time scope is given by $\boldsymbol{t}_G^{(\rho)} \in \mathbb{R}^{\beta^{(\rho)}}$, where the i-th element is given by

$$\boldsymbol{t}_G^{(\rho)}[i] = \left(b_\rho^{\triangleright} + i \right) T_\mathrm{B}. \tag{7.9}$$

The polynomial fitting process in (7.3) or (7.6), respectively, was performed with respect to the *local* block indices and as such with respect to the *local* time scope of the corresponding path. Thus, the evaluation of the polynomials must be performed with respect to their respective local time scope and *not* with respect to the respective global time scope. This local time vector $\boldsymbol{t}^{(\rho)} \in \mathbb{R}^{\beta^{(\rho)}}$ can be derived from the corresponding time vector with respect to the global time scope; its i-th element is given by

$$\boldsymbol{t}^{(\rho)}[i] = \boldsymbol{t}_G^{(\rho)}[i] - b_\rho^{\triangleright} T_\mathrm{B}. \tag{7.10}$$

[3]The term "recovery" is used here, although it is of course not possible to fully recover the original MPC path evolution data due to the lossy compression process. However, it was deliberately decided not to refer to this process as "generation", as this term will be later used when synthetic MPC path data is created.

7.2.2.2 Delay and Doppler Shift Evolution

The delay and Doppler shift evolution of the MPC path is computed by evaluating the corresponding polynomial at all items of $\boldsymbol{t}^{(\rho)}$. For the delay evolution, this is achieved by inserting the third, the fourth and the fifth element of $\boldsymbol{r}^{(\rho)}$ – thus the polynomial coefficients – into (7.1):

$$\gamma_\tau^{(\rho)}[i] = p_\tau^{(\rho)}(iT_\mathrm{B}) = p_2^{(\rho)}(iT_\mathrm{B})^2 + p_1^{(\rho)} iT_\mathrm{B} + p_0^{(\rho)}, \quad \forall\, i \in \{0, 1, \ldots, \beta^{(\rho)} - 1\} \tag{7.11}$$

For the Doppler shift evolution, the polynomial coefficients are inserted into (7.2):

$$\gamma_\nu^{(\rho)}[i] = p_\nu^{(\rho)}(iT_\mathrm{B}) = -\left(2p_2^{(\rho)} iT_\mathrm{B} + p_1^{(\rho)}\right) f_\mathrm{c}, \quad \forall\, i \in \{0, 1, \ldots, \beta^{(\rho)} - 1\}. \tag{7.12}$$

7.2.2.3 Amplitude and Phase Evolution

The remaining two items of $\boldsymbol{r}^{(\rho)}$ parametrise the RICE distribution. Thus, the amplitudes of the MPC path could simply get sampled from $\mathcal{R}(\varrho^{(\rho)}, \sigma^{(\rho)})$ and the phases could get sampled from $\mathcal{U}[0, 2\pi)$, as it is suggested in [60]. However, this approach does not take the temporal evolution of the amplitude and the phase into account, since sampling from a distribution does not involve a memory.

In [68], methods for the statistically generation of the amplitude and phase of a received signal transmitted through a wireless multipath channel are discussed while taking the temporal evolution into account. The remainder of this subsection shows how this method can be integrated into the approach of generating of MPC paths presented here. The approaches presented in the following were chosen because they allow the generation of the temporal evolution of the received amplitudes fulfilling desired statistical properties with comparatively low computational effort. These approaches were also used for this purpose in some basic channel models developed by DLR as part of the development of LDACS and its predecessor projects B-VHF *(Broadband VHF)* and B-AMC *(Broadband Aeronautical Mobile Communication System)* [17].

As stated in Appendix E, it is possible to construct a RICE process from a RAYLEIGH process and – furthermore – it is possible to construct a RAYLEIGH process from two GAUSS processes, here denoted by $\mu_1^{(\rho)}(t)$ and $\mu_2^{(\rho)}(t)$. As shown in [79], such a GAUSS process can be modelled using the SoS *(Sum of Sinusoids)* method:

$$\mu_i^{(\rho)}(t) = \sum_{n=0}^{N_{\mathrm{SoS}i}-1} c_{i,n}^{(\rho)} \cos\left(2\pi f_{i,n}^{(\rho)} t + \varphi_{i,n}^{(\rho)}\right), \quad i \in \{1, 2\}, \tag{7.13}$$

where $N_{\mathrm{SoS}i} \in \mathbb{N}$ denotes the amount of sinusoids that are used for the modelling of the specific process. In theory it is $N_{\mathrm{SoS}i} \to \infty$; in practice surprisingly small values for $N_{\mathrm{SoS}i}$ are sufficient: In [68], $N_{\mathrm{SoS}i} = 100$ is already considered as "very large". To avoid undesired correlations of the real part and the imaginary part of the resulting process when inserting (7.13) into (E.14), we set $N_{\mathrm{SoS}2} = N_{\mathrm{SoS}1} + 1$.

In [68], multiple approaches to find values for the coefficients $c_{i,n}^{(\rho)}$, the discrete frequencies $f_{i,n}^{(\rho)}$, and the phases $\varphi_{i,n}^{(\rho)}$ are introduced. One of these approaches is the MEDS *(Method of Exact Doppler Spread)*, which was chosen for the implementation presented here due to its reasonable ratio of computational effort to resulting performance. The generation of some of the required parameters depends on the desired shape of the resulting spectrum type, e.g. whether a GAUSSian spectrum or JAKES spectrum is desired. As BELLO's theory on wireless channels in aeronautical communications suggests, we assume a GAUSSian spectrum [7] if not denoted otherwise.

For the chosen case, i.e. MEDS with a GAUSSian spectrum, the parameters required for (7.13) are computed as follows:

- The coefficients are determined by

$$c_{i,n}^{(\rho)} = \sigma^{(\rho)} \sqrt{\frac{2}{N_{\mathrm{SoS}i}}}. \tag{7.14}$$

- The discrete frequencies are given by

$$f_{i,n}^{(\rho)} = \nu_{\max}^{(\rho)\prime} \operatorname{erf}^{-1}\left(\frac{2n}{2N_{\mathrm{SoS}i}}\right), \tag{7.15}$$

  where $\operatorname{erf}^{-1}(\square)$ denotes the inverse of the GAUSSian error function as given in e.g. [13] and $\nu_{\max}^{(\rho)\prime}$ denotes the maximum Doppler shift expected for the MPC path indexed by ρ where the LOS Doppler from Section 4.2.2.3 compensation has been reversed.

- The phases are given by

$$\varphi_{i,n}^{(\rho)} \sim \mathcal{U}[0, 2\pi). \tag{7.16}$$

These parameters are generated accordingly and inserted into (7.13), such that two independent and uncorrelated GAUSS processes $\mu_1^{(\rho)}(t)$ and $\mu_2^{(\rho)}(t)$ are determined. These processes are then used to create a RAYLEIGH process as given in (E.14) which is eventually used to create a RICE process by using (E.13) and the shape parameter $\varrho^{(\rho)}$. This RICE process is then evaluated at all time instants given in the local time vector $\boldsymbol{t}^{(\rho)}$.

7.2.3 Kernel Fitting

The compact representation defined in Section 7.2.1 enables a straightforward analysis of the underlying statistics of the individual parameters. Based on these findings it is then possible to generate new data that match the statistics of the training data.

A tool that allows the estimation of unknown multivariate distributions is the KDE *(Kernel Density Estimation)*. In contrast to a histogram, it is not of a discrete but a continuous nature. Furthermore, a KDE supports samples of higher dimensions and is also capable of analysing correlations between these individual dimensions. A mathematical definition of the KDE as well as information on the KDE implementation used in this dissertation are given in Appendix D.

The modelling approach introduced here is based on such a KDE *(Kernel Density Estimation)*: Each item of a set of compact representation vectors $\mathcal{R}_{\text{train}}$ is pre-processed and then used to fit the KDE's kernel. The training set $\mathcal{R}_{\text{train}}$ is either a true subset of $\mathcal{R}$ or it corresponds to $\mathcal{R}$. The kernel is then used to generate new data that are used for a realisation of the channel model. In the following, the individual processing steps are explained in more detail.

Stacking Representation Vectors All vectors from $\mathcal{R}_{\text{train}}$ are stacked to a matrix $\boldsymbol{R} \in \mathbb{R}^{|\mathcal{R}_{\text{train}}| \times 7}$, such that the compact representation of an MPC path indexed by ρ corresponds to the ρ-th row of $\boldsymbol{R}$. Consequently, each column of $\boldsymbol{R}$ corresponds to $|\mathcal{R}_{\text{train}}|$ samples of the respective element of the compact representation vector, thus can be understood as a feature vector. We call $\boldsymbol{R}$ the *MPC path representation matrix*.

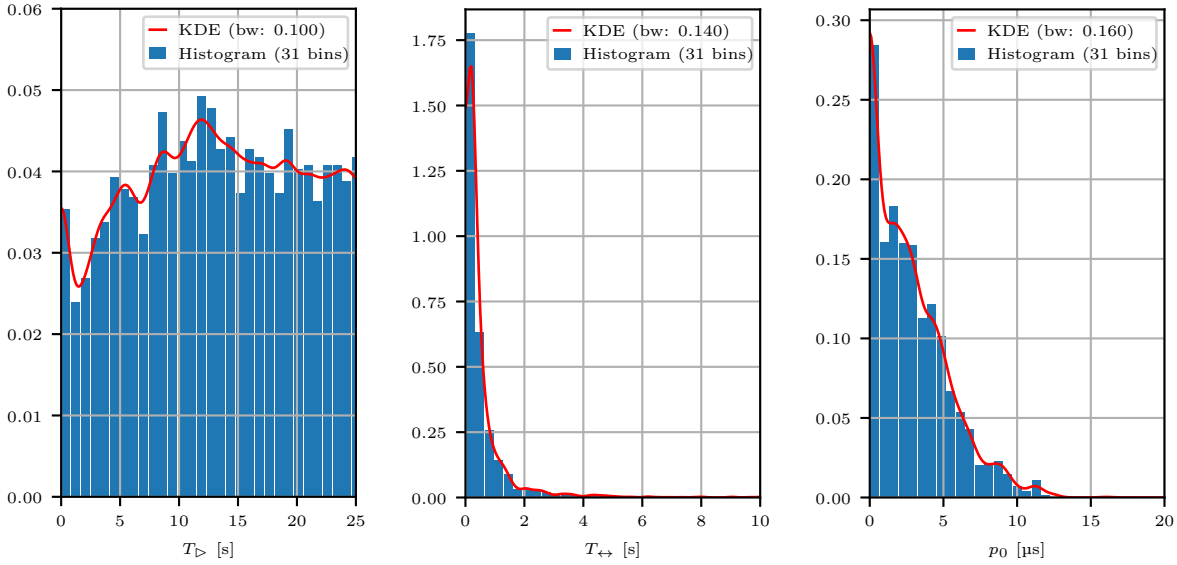
Scaling of Representation Vectors Since the individual items of the representation vectors (i.e. the columns of $\boldsymbol{R}$) are of different domains having very different scales, a feature scaling is applied to $\boldsymbol{R}$ to improve numerical stability, see Appendix D. If not denoted otherwise, the min-max scaling as given in (6.20) is applied to scale the matrix' values to the interval $[-1, +1)$. The feature scaling is applied column-wise. We denote this feature-scaled matrix by $\boldsymbol{R}'$ and call it *scaled MPC path representation matrix*. This matrix contains the actual training data.

Actual KDE Kernel Fitting A KDE with $d = 7$ dimensions is now applied to $\boldsymbol{R}'$. An important aspect of the application of a KDE is the bandwidth: A larger bandwidth increases the smoothing of the assumed statistical distribution of the corresponding dimension's data, a narrow bandwidth has the opposite effect. As the fitted KDE kernel with its "learned" distributions is later used to sample new data, the choice of the KDE bandwidth has a direct impact of the distribution of the generated data – and consequently on the resulting channel model: The narrower the bandwidth of a feature during the training phase, the closer the model's realisations are to the training data. In an extreme case, this can lead to an over-fitting. If the bandwidth of a feature is chosen (too) large, the probability of generating outliers is increased.

Although the applied feature scaling of the training data already reduces the need for significantly different bandwidths for the individual dimensions, it may still be desirable to use different bandwidths: Some features of the compact representation are more sensitive to outliers than others. One example are the items p_0 and p_2 of the representation vector: p_0 defines the constant delay offset with respect to LOS, whereas p_2 is used in a squared form and multiplied by the time vector during evaluation according to (7.11). Thus, an outlying value for p_2 has a much higher impact on the resulting MPC path data than an outlying value for p_0.

7.2.4 Analysis of the MPC Path Representation Matrix

In the following, the distribution of and the correlation between the columns of the MPC path representation matrix $\boldsymbol{R}$ as defined above is analysed. The matrix used as an example in



(A) MPC Path Start Time

(B) MPC Path Duration

(C) First Polynomial Coefficient

FIGURE 7.2: Visualisation of the distribution of selected parameters of the compact representation of a set of MPC paths.

this subsection is constructed by applying the processing steps introduced in Section 7.2.1 on the measurement data used in Figure 6.6.

7.2.4.1 Distribution of Parameters

The distribution of the values of selected columns of the matrix $\boldsymbol{R}$ – thus of selected values of the compact representation vectors of an MPC path set – is visualised in Figure 7.2. Each plot shows both a (normalised) histogram and the result of a (normalised) KDE as it is applied in Section 7.2.3. The plots demonstrate the close relationship of these two tools: Both tools provide an understanding of the distribution of the underlying data.

MPC Path Start Time Figure 7.2a sketches the distribution of the start times of the MPC paths $T_{\triangleright}$, i.e. the first parameter of the compact representation vector as defined in (7.7). One can observe that less MPC paths start during the first $\approx 5\,\mathrm{s}$ compared to the remaining observation time.

MPC Path Duration The distribution of the MPC path durations within the training set is sketched in Figure 7.2b. It can be observed that the vast majority of the MPC paths is rather short (i.e. $< 0.2\,\mathrm{s}$). Longer MPC paths become less likely with increasing duration; durations of more than 4 s appear to be rare.

First Polynomial Coefficient The plot in Figure 7.2c shows the distribution of the delay polynomial coefficient p_0, thus the coefficient that defines an MPC path's initial delay with respect to the LOS component, see (7.1). One can observe that most of the MPC paths' start delay is very close to the LOS component's delay (i.e. 0 s). Larger start delays become

TABLE 7.1: The PEARSON correlation coefficient as given in (7.17) between all permutations of the columns of $\boldsymbol{R}$ whose data were computed by applying the processing steps from Section 7.2.1 on measurement data. The notation from (7.7) is used to name the rows and columns of the table.

	$T_{\triangleright}$	$T_{\leftrightarrow}$	p_2	p_1	p_0	ϱ	σ
$T_{\triangleright}$	1.00	−0.11	−0.01	−0.00	−0.06	0.11	0.02
$T_{\leftrightarrow}$	−0.11	1.00	0.01	0.05	0.00	0.01	0.40
p_2	−0.01	0.01	1.00	−0.90	0.02	0.01	0.00
p_1	−0.00	0.05	−0.90	1.00	−0.03	0.00	0.03
p_0	−0.06	0.00	0.02	−0.03	1.00	−0.16	−0.04
ϱ	0.11	0.01	0.01	0.00	−0.16	1.00	0.25
σ	0.02	0.40	0.00	0.03	−0.04	0.25	1.00

less likely with an increasing start delay. This is explained by the fact that a larger delay of an MPC is caused by a longer travel distance of the specific MPC which decreases the component's power at the receiver due to the FSPL, see (4.13). As a matter of consequence, the corresponding MPCs are too weak to get recognised during the processing of the MPC detection, see Section 6.1.

The examples from above – especially the distribution of the path start time and the first polynomial coefficient – suggest that the distributions of the parameters are difficult to model using regular distribution functions scaled to the respective data. This assumption gets confirmed by evaluating the results of *goodness of fit* tests like the KOLMOGOROV-SMIRNOV test applied to data sampled from standard distribution functions fitted to the respective data.

7.2.4.2 Correlation between Parameters

Besides the distribution of the parameters, the correlation between the parameters is of interest. To analyze this correlation, we use the PEARSON correlation coefficient, in German-language literature also referred to as *empirical correlation coefficient* [13].

The PEARSON correlation coefficient[4] $\eta_{xy} \in \mathbb{R}_{[-1,1]}$ between two sample vectors $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{y} \in \mathbb{R}^N$ is given by

$$\eta_{xy} = \frac{\sum_{i=0}^{N-1} (x_i - \bar{\boldsymbol{x}})(y_i - \bar{\boldsymbol{y}})}{\sqrt{\sum_{i=0}^{N-1} (x_i - \bar{\boldsymbol{x}})^2} \sqrt{\sum_{i=0}^{N-1} (y_i - \bar{\boldsymbol{y}})^2}} \tag{7.17}$$

and provides a measure of the linear correlation between the elements of the two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ [13], [43].

[4]Note that this coefficient is usually referred to as r_{xy} in the literature which has been avoided here in order to avoid confusion with the MPC path representation vector $\boldsymbol{r}$.

The table given in Table 7.1 provides the PEARSON coefficient for all permutations of the columns of the MPC path representation matrix $\boldsymbol{R}$ of a set of paths. For the sale of clearness, the colours of the table cells are derived from the respective cell's value comparable to a heatmap; the text colours are adjusted accordingly to improve readability. The given table can also be seen as a symmetric matrix which is explained by the fact that the computation of (7.17) is commutative.

From (7.17), it follows that the PEARSON correlation coefficient equals 1 in case perfect correlation between the two investigated vectors is given. This is obviously the case for two identical vectors which explains the all-one diagonal of Table 7.1. A comparatively strong positive correlation can be observed between $T_{\leftrightarrow}$ and σ.

The less the two investigated vectors are correlated, the more the absolute value of the resulting PEARSON correlation coefficient tends towards zero. This low correlation can be observed between several parameters, e.g. between $T_{\leftrightarrow}$ and p_0.

A strong negative correlation can be observed between the parameters p_2 and p_1.

7.2.5 Generation of an MPC Path Set

In general, the generation[5] of a new set of MPCpaths $\tilde{\mathcal{P}}$ based on the fitted kernel is realised by performing the same steps as for the fitting in reverse.

7.2.5.1 Covered Time Frame

In general, the time frame that is covered by the MPC path set to be generated cannot be chosen arbitrarily since it is defined by the data given in $\mathcal{R}_{\text{train}}$. This is because the compact representation vectors used for training – according to their definition in (7.7) – include information on the time instance when an MPC path starts and the duration of an MPC path. Thus, this information is learned during the training process and is therefore reflected in the kernel and the data that is generated from it. In many cases, this lack of flexibility is not a real limitation in practical application:

- The training data of some scenarios, e.g. the take-off scenario, cover a representative time frame, thus a manual adaption of the time frame is not even necessary.
- If the desired time frame is *shorter* than the time frame covered by the training data, it is straightforward to generate an MPC path set covering a similar length than the training data and later just use a cut-out of the desired duration.
- If the desired time frame is *longer* than the time frame covered by the training data, it is often[6] possible to generate multiple MPC path sets from the same training data and concatenate the resulting graphs.

More flexibility regarding the time frame can be achieved by ignoring the first two elements of the generated MPC path representation vectors $\tilde{\boldsymbol{r}}^{(\rho)}$ and replacing them with manually selected values. A strategy to manually find these values in such a way that realistic channel

[5] In the following, a generated datum is denoted by a tilde over the corresponding symbol: $\tilde{\square}$.
[6] This may not be possible for all scenarios, e.g. the take-off scenario.

scenarios can still be generated is a rather complex task and will not be the subject of further discussion in this thesis.

7.2.5.2 Amount of MPC Paths

Let $\mathfrak{P}_{\text{gen}} \in \mathbb{N}$ denote the amount of MPC paths that should be generated. To achieve a result whose overall wireless channel statistics are comparable to those of the scenario that was present during the recording of the training data, it is $\mathfrak{P}_{\text{gen}} = |\mathcal{R}_{\text{train}}|$.

Increasing or decreasing the amount of MPC paths will result in a scenario that differs from the scenario used for the training of the current kernel: In the discussed modelling approach, an MPC path corresponds to a reflector (or a cluster of non-resolvable reflectors). Thus, changing the total number of MPC paths while keeping the scenario time frame fixed changes the total reflector density.

7.2.5.3 Generation of a Single MPC Path

The following steps are performed to generate a new MPC path:

1. One item $\tilde{\boldsymbol{r}}^{(\rho)\prime}$ is sampled from the KDE's kernel.
2. The inverse of the feature scaling applied in Section 7.2.3 is applied to $\tilde{\boldsymbol{r}}^{(\rho)\prime}$, resulting in $\tilde{\boldsymbol{r}}^{(\rho)}$.
3. By applying the steps given in Section 7.2.2 on $\tilde{\boldsymbol{r}}^{(\rho)}$, the data of the new MPC path $\tilde{\gamma}^{(\rho)}$ is generated. In case a modified time frame is desired as discussed in Section 7.2.5.1, the start parameters when generating the time vector in Section 7.2.2.1 need to be replaced accordingly.
4. The generated data from the previous step is subject to a plausibility check. Unless one of the following criteria is fulfilled, $\tilde{\gamma}^{(\rho)}$ is added to $\tilde{\mathcal{P}}$:
   - The generated delay evolution $\tilde{\gamma_\tau}^{(\rho)}$ becomes negative at least in one sample. Negative delays do not make sense as the LOS component has been defined to have a delay of zero – and no signal component can have a smaller value.
   - The absolute value of the generated Doppler shift evolution $\tilde{\gamma_\nu}^{(\rho)}$ exceeds the theoretical maximum Doppler shift which is physically limited by the aircraft speed (see (2.2)).
   - The estimated RICE parameters – e.g. using the approach presented in Appendix E – of the amplitude evolution $|\tilde{\gamma_\alpha}^{(\rho)}|$ show considerable deviations from the desired parameters. Since the reliability of the RICE parameter estimation can be quite low for short MPC paths, this check may result in false positives in this case.

7.2.5.4 Generation of a Whole MPC Path Set

The above steps are repeated until $|\tilde{\mathcal{P}}|$ equals $\mathfrak{P}_{\text{gen}}$. For the sake of completeness and following the notation introduced above, we define the generated MPC path representation

TABLE 7.2: The PEARSON correlation coefficient as given in (7.17) between all permutations of the columns of $\tilde{\boldsymbol{R}}$, thus applied to generated data. The reader may compare this table to Table 7.1.

	$T_{\triangleright}$	$T_{\leftrightarrow}$	p_2	p_1	p_0	ϱ	σ
$T_{\triangleright}$	1.00	−0.12	−0.03	0.01	−0.08	0.11	0.01
$T_{\leftrightarrow}$	−0.12	1.00	0.01	0.05	0.01	0.01	0.39
p_2	−0.03	0.01	1.00	−0.90	0.02	0.02	0.01
p_1	0.01	0.05	−0.90	1.00	−0.03	−0.01	0.03
p_0	−0.08	0.01	0.02	−0.03	1.00	−0.14	−0.03
ϱ	0.11	0.01	0.02	−0.01	−0.14	1.00	0.26
σ	0.01	0.39	0.01	0.03	−0.03	0.26	1.00

matrix $\tilde{\boldsymbol{R}} \in \mathbb{R}^{\mathfrak{P}_{\text{gen}} \times 7}$ that is created by stacking the representation vectors of all MPC paths of $\tilde{\mathcal{P}}$.

Analogous to Section 7.2.4.2, the PEARSON correlation coefficient has been computed for all permutations of the columns of one realisation of $\tilde{\boldsymbol{R}}$. The underlying data have been generated from a model that was trained with the data from Figure 6.6. The results are presented in Table 7.2 and show only small deviations compared to the correlation properties of $\boldsymbol{R}$ presented in Table 7.1.

7.2.6 MPC Path Set Clustering

7.2.6.1 Motivation

By intention, the parameters of the MPC paths that are generated by the approach presented above follow the distribution that the corresponding KDE has determined. However, the parameter distributions are very likely to change in different environment as the following example shows: We assume a KDE kernel has been trained using the measurement data recorded during take-off at a certain airport. It is obvious, that *this* airport's architecture (most relevant: the amount of buildings within a certain distance to the transmitter/receiver) is reflected in the measurement data – and therefore in all data generated from this KDE. Without extending the approach presented so far, it is not possible to make the modelling appraoch more generic such that it can simulate a setup on an airport with a different architecture.

An extension to achieve this desired flexibility is the clustering of the MPC paths based on certain criteria before the training process. For each of the clusters an individual KDE is performed. To generate a model instance, data from all KDE kernels is sampled and combined. By altering the amount of MPC paths that are sampled from each kernel, the user can alter the distribution of the overall generated data.

7.2.6.2 Clustering Procedure

Before the KDE kernel is fitted, the set of the representation vectors $\mathcal{R}_{\text{train}}$ is divided into $S \in \mathbb{N}$ true subsets called clusters, such that

$$\mathcal{R}_{\text{train}} = \bigcup_{\forall s \in \{0,\dots,S-1\}} \mathcal{R}_{\text{train},s}. \tag{7.18}$$

None of the subsets may be equal to the empty set and each element of $\mathcal{R}_{\text{train}}$ must be assigned to exactly one subset. It is advantageous if a considerable amount of MPC paths is assigned to each cluster, such that the analysis of its statistics is representative.

Criteria for Clustering The criterion according to which the cluster allocation is made should be chosen such that it reflects the geometry of the scene – at least in an indirect way. As shown in Section 6.3, the delay defines the size of the prolate spheroid on which the corresponding reflector is located. Thus, the delay provides some indirect information about the geometry of the setup and therefore is a reasonable choice as a criterion for the clustering. One could either use the mean of the delay evolution of the corresponding MPC path $\bar{\gamma_\tau}^{(\rho)}$ or make use of the relevant items of the path's compact representation $\boldsymbol{r}^{(\rho)}$, thus $\boldsymbol{p}_\tau^{(\rho)}$, e.g. by using the fixed offset $p_0^{(\rho)}$.

Besides this fixed offset, the sign of the Doppler can be of interest: From (2.2) it is known, that a positive sign of the Doppler shift[7] is caused when the receiver is moving towards a signal component's origin and a negative sign of the Doppler shift is caused when the receiver is moving away from a signal component's origin. In (7.2), it is shown how $p_1^{(\rho)}$ and $p_2^{(\rho)}$ define the sign of the Doppler. Evaluating those allows a rough determination whether the reflector is in front or in the back of the aircraft.

7.2.6.3 Training Phase

The training phase consists of the individual application of the procedure described in Section 7.2.3 to each subset: For each subset $\mathcal{R}_s$ an individual KDE is fitted.

7.2.6.4 MPC Path Generation

The first step of the MPC path generation is to determine how many MPC paths should be sampled from each of the S KDE kernels: Let $\mathfrak{P}_{\text{gen},s} \in \mathbb{N}$ denote the amount of MPC paths that are sampled from the kernel trained by $\mathcal{R}_{\text{train},s}$.

In case $\mathfrak{P}_{\text{gen},s} = \lceil a_s | \mathcal{R}_{\text{train},s} | \rceil, \forall s \in \{0,\dots,S-1\}$ with $a_s \in \mathbb{R}, a_s = a_{s+1} \forall a_s$, the clustering has close to no effect compared to a non-clustered approach. However, if at least one value a_s does not match the others, the overall statistics of the generated MPC paths is changed.

The actual MPC path generation is performed by the application of the steps from Section 7.2.5 on each of the S kernels individually. Eventually, the set of generated MPC paths is the joint set of all cluster-specific sets:

$$\tilde{\mathcal{P}} = \bigcup_{\forall s \in \{0,\dots,S-1\}} \tilde{\mathcal{P}}_s, \quad \text{with } |\tilde{\mathcal{P}}_s| = \mathfrak{P}_{\text{gen},s}. \tag{7.19}$$

[7] The compensation of the LOS Doppler shift needs to be considered here.

7.2.6.5 Cluster for MPC Paths Representing the LOS Component

A cluster that contains only MPC paths that are representing the evolution of the LOS component is called a LOS cluster. Due to the pre-processing described in Section 4.2.2.3, the LOS component – if present – can be found at a delay of zero and a Doppler shift of zero in each block[8]. Thus, one can define the LOS cluster by including all MPC paths whose assigned MPCs are having a zero delay and zero Doppler shift.

In aeronautical channels, where the LOS component is usually present for most of the time, this LOS cluster will often contain only very few (but long) MPC paths, since the LOS MPC paths are likely to be quite stable and since – by definition – only one single LOS component can exist at a time (i.e. within a block). As such, a cluster with that few training data is unsuitable for the training of a KDE kernel with the aim to later sample from that kernel[9]. Even though this cluster will not be used for the sampling of new MPC path data, it is still useful to create a LOS cluster, since the MPC paths of this cluster are not part of any other cluster and therefore do not influence these clusters' statistical properties. This becomes particularly relevant when MPC paths representing the LOS are created manually as described in the next paragraph.

Since sampling from the LOS cluster appears unsuitable, alternative methods for generating MPC paths representing the evolution of the LOS component must be found. A trivial approach is to manually create the compact representation vectors (7.7) of one or more (non-overlapping) LOS MPC paths. These vectors' elements are set as follows:

- Due to the compensation during pre-processing, the delay and Doppler shift of the LOS component are constantly zero; thus, all polynomial coefficients are set to zero (compare to (7.1) and (7.2)).
- The parameters defining the amplitude and phase evolution (ϱ, σ) can be set according to the corresponding values that have been found for MPC paths assigned to the LOS cluster. Varying these values has a direct influence on the K-factor of the resulting model realisation. Thus, in case non-stationary LOS conditions are desired, multiple LOS MPC paths with varying amplitude parameters can be defined to reflect these changing conditions.
- The start time and duration of each MPC path can be set freely, as long as none of the defined LOS MPC paths overlap.

Following the nomenclature from above, we denote the set of these manually created LOS MPC paths by $\tilde{\mathcal{P}}_{\mathrm{LOS}}$. The elements of this MPC path set are then added to $\tilde{\mathcal{P}}$ as described in (7.19).

[8] Due to imperfect compensation caused by imperfect GT data, minor deviations are likely when dealing with actual measurement data.

[9] The small amount of training data and the fact that at least parts of the compact representation vectors of the LOS MPC paths are usually very similar, lead to numerical instability.

7.3 Evaluation

In the following section, we evaluate several realisations of the modelling approach presented so far by comparing its general channel characteristics to the respective characteristics of the measured channel.

7.3.1 Evaluation Metrics

In [60], the statistical channel measures introduced in Section 2.2.2 are adapted to be applicable to the channel representation through MPC paths. Based on this work, we define the following two measures to evaluate the characteristics of the channels generated by the modelling approach introduced above.

7.3.1.1 Delay Spread

Combining the definition of the discrete RMS-DS as given in (2.18) and the notation from the above sections, the RMS-DS of a block b is given by

$$\sigma_\tau^{(b)} = \sqrt{\frac{\sum_{l=0}^{L^{(b)}-1} |\alpha_l|^{(b)2} \tau_l^{(b)2}}{\sum_{l=0}^{L^{(b)}-1} |\alpha_l|^{(b)2}} - \mu_\tau^{(b)2}}, \tag{7.20}$$

where the mean energy delay of the block $\mu_\tau^{(b)}$ is given by

$$\mu_\tau^{(b)} = \frac{\sum_{l=0}^{L^{(b)}-1} |\alpha_l|^{(b)2} \tau_l^{(b)}}{\sum_{l=0}^{L^{(b)}-1} |\alpha_l|^{(b)2}}. \tag{7.21}$$

7.3.1.2 K-factor

Analogous to the above definition of the blockwise RMS-DS, we define the blockwise K-factor using the definition from Section 2.2.2.2 by

$$K^{(b)} = \frac{|\alpha_{\mathrm{LOS}}^{(b)}|^2}{\sum\limits_{\forall \zeta_l^{(b)} \in \mathcal{M}^{(b)} \setminus \{\zeta_{\mathrm{LOS}}^{(b)}\}} |\alpha_l^{(b)}|^2}, \tag{7.22}$$

where b – as usual – describes the block index.

7.3.2 Evaluation of Channel Properties

In this section, the evaluation metrics from above are applied to measurement data processed according to the steps introduced in Chapter 6 and Section 7.2.1, respectively, and data generated using the approach introduced in Section 7.2.5. In all cases, the data of three arbitrarily chosen realisations of the model are used as examples. The underlying data are the data used to generate the plot in Figure 6.6. For both metrics, both the evolution over time as well as the overall distribution are analysed.

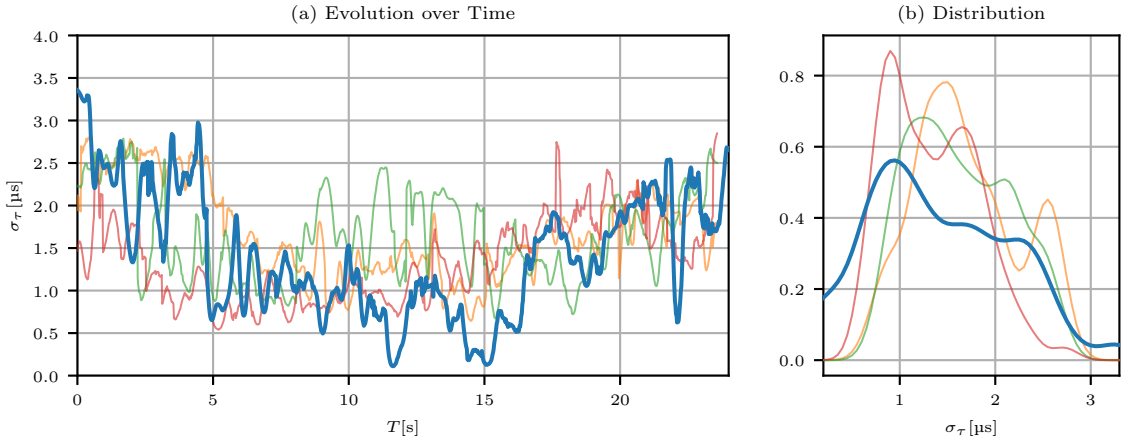


FIGURE 7.3: Comparing the RMS-DS of measurement data (bold blue line) and data from three realisations of the proposed model (other): (a) Evolution over time (b) Approximation of the distribution using a KDE.

7.3.2.1 Delay Spread

The temporal evolution of the RMS-DS as defined in Section 7.3.1.1 for a time period of 16 s is shown in Figure 7.3(a). In case of the model realisations, the range of the data points is 0.5 µs to 2.8 µs, which is a little less compared to the range of the actual measurement data (0.1 µs to 3.35 µs). Apparently, the model is unlikely to generate outliers for the RMS-DS. The general nature of the curves representing the model realisations and the curve representing the measurement data looks quite similar: In all cases, no rapid oscillations can be observed. The curves show some smooth parts that are interrupted by "jumps". These jumps happen when comparatively strong MPC paths either start or end and illustrate the influence of these strong MPC paths on the power distribution along the delay axis.

The plots in Figure 7.3(b) show an estimate of the distribution of the RMS-DS. All plots suggest that the underlying distribution is multivariate and it is impossible to provide a closed form solution for the distributions. Due to the wider range of the values – see above – the distribution of the RMS-DS of the measurement data is more stretched along the x-axis compared to the distribution of the RMS-DS of the model realisations. All distributions have in common that smaller values for σ_τ appear to be more likely than larger values for σ_τ.

7.3.2.2 *K*-factor

Figure 7.4(a) shows the evolution over time of the logarithmic *K*-factor as defined in Section 7.3.1.2. The data the *K*-factors have been determined for are the same that were used to determine the delay spread in the previous subsection. One can observe that the range covered by the *K*-factor is more or less the same for all curves, regardless of whether the underlying data are measurement data or generated data. All curves also show sections where the a *K*-factor is more or less stable for a certain time period, e.g. the blue curve from 22 s to 24 s and the red curve from 1 s to 3 s.

In contrast to the RMS-DS, a comparison of the distribution of the *K*-factor between the measured data and the generated data from the model, see Figure 7.4(b), shows only small

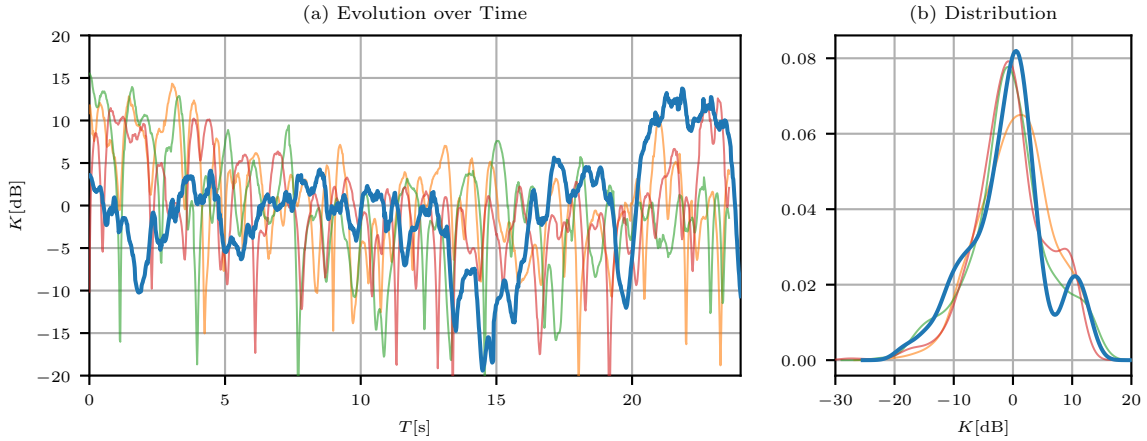


FIGURE 7.4: Comparing the logarithmic K-factor of measurement data (bold blue line) and data from three realisations of the proposed model (other): (a) Evolution over time (b) Approximation of the distribution using a KDE.

deviations. The shape of the estimated distributions suggests a multivariate nature of the underlying distribution. This is particularly evident for the measurement data where a peak around 0.3 dB and another peak around 10 dB can be identified. We explain this by varying LOS conditions within the observed time period: While the K-factor is around 0.3 dB for most of the time between the periods from 3 s to 5 s and from 7 s to 2 s, the K-factor rapidly increases up to 10 dB (and even more) starting from 21 s. According to the distribution, a K-factor around 6 dB to 8 dB is less likely than a slightly lower or slightly higher K-factor in the given scenario. This assumption gets also supported by analysing the temporal evolution in Figure 7.4(a).

7.4 Generation and Use of Impulse Responses

As explained in the the introductory part of this dissertation in Section 2.2.1, the impulse response is *the* central tool to describe a channel. The *practical* use of a channel model is to generate impulse responses that can be used for simulations to support the design of new communication or navigation systems. In the following, it is described how impulse responses are generated from the presented channel modelling approach and how these impulse responses can be used for the evaluation of communication systems.

7.4.1 Generation of Impulse Responses

Based on the graph structure as introduced in Section 6.2.1 – regardless of whether it is based on measurement data or generated data – the CIR can be determined by evaluating the MPCs assigned to a certain block.

We denote the discrete CIR of block b by $\boldsymbol{h}^{(b)} = [h_0^{(b)}, h_1^{(b)}, \ldots]$, where the k-th element is given by

$$h_k^{(b)} = \sum_{l=0}^{L^{(b)}-1} \alpha_l^{(b)} \ \mathrm{sinc}\left(\frac{k T_\mathrm{B} - \tau_l^{(b)}}{T_\mathrm{B}} \right), \tag{7.23}$$

where sinc(□) denotes the sinc-function as introduced in Section 3.2.2.3.

Setting the symbol duration along the delay axis in (7.23) equal to the block duration T_B simplifies the later processing. This does not limit the field of application, since it is possible to evaluate MPC paths of an MPC graph at arbitrary time instants once the MPC paths are given in the compact representation (see Section 7.2.2). It is therefore possible to set arbitrary block durations.

The length of the discrete block CIR $\boldsymbol{h}^{(b)}$ is given by $h_{\text{len}} \in \mathbb{N}$ and determined by the MPC with the longest delay with respect to the LOS component. In the following, we assume that the CIRs of all β blocks of an MPC graph are of common length

$$h_{\text{len}} = \max_{\forall b \in \{0,1,\ldots,\beta-1\}} \left\{ \max_{\forall l \in \{0,1,\ldots,L^{(b)}-1\}} \left\{ \left\lceil \frac{\tau_l^{(b)}}{T_B} \right\rceil \right\} \right\}. \tag{7.24}$$

7.4.2 Use of Generated Impulse Responses

In the following, we want to transmit a discrete baseband signal represented by a complex vector $\boldsymbol{s}_{\text{Tx}} = [s_{\text{Tx},0}, s_{\text{Tx},1}, \ldots] \in \mathbb{C}^{N_s}$ through a channel given by an MPC graph of sufficient length, thus given by at least N_s consecutive blocks. We assume the time interval between adjacent samples of $\boldsymbol{s}_{\text{Tx}}$ to be T_B. To allow a time varying CIR, the following processing is applied:

1. A transmission matrix $\boldsymbol{S}_{\text{Tx}} \in \mathbb{C}^{h_{\text{len}} \times N_s + h_{\text{len}} - 1}$ is created, where the n-th row corresponds to $\boldsymbol{s}_{\text{Tx}}$ with n prefixed zeros and $h_{\text{len}} - n - 1$ appended zeros – thus $\boldsymbol{S}_{\text{Tx}}$ is a TOEPLITZ matrix. Figure 7.5a provides an example transmission matrix with $N_s = 8$ and $h_{\text{len}} = 4$.

2. The CIRs that were generated according to (7.23) are concatenated to a matrix $\boldsymbol{W} = (W_{kb}) \in \mathbb{C}^{h_{\text{len}} \times N_s + h_{\text{len}} - 1}$ where the b-th column corresponds to the CIR of block b, thus $W_{kb} = h_k^{(b)}$. We use the symbol $\boldsymbol{W}$ to emphasise that this matrix is a discrete representation of a weight function as explained in Section 2.2.1.1, where the columns correspond to the delay axis and the rows correspond to the time axis. An example with $N_s = 8$ and $h_{\text{len}} = 4$ is given in Figure 7.5b.

3. We define the matrix $\boldsymbol{S}_{\text{Rx}} = (S_{\text{Rx},kb}) \in \mathbb{C}^{h_{\text{len}} \times N_s + h_{\text{len}} - 1}$ as the element-wise multiplication of the previously defined matrices: $\boldsymbol{S}_{\text{Rx}} = \boldsymbol{S}_{\text{Tx}} \cdot \boldsymbol{W}$.

4. The signal after the channel $\boldsymbol{s}_{\text{Rx}} \in \mathbb{C}^{N_s + h_{\text{len}} - 1}$ is now given by taking the sum along all columns of $\boldsymbol{S}_{\text{Rx}}$; thus, the b-th element of $\boldsymbol{s}_{\text{Rx}}$ is given by $s_{\text{Rx},b} = \sum_{k=0}^{h_{\text{len}}-1} S_{\text{Rx},kb}$.

The reader may note that the above processing corresponds to a convolution of $\boldsymbol{s}_{\text{Tx}}$ with $\boldsymbol{h}^{(b)}$ if *all* columns of $\boldsymbol{W}$ were set to a *fixed* $\boldsymbol{h}^{(b)}$.

$s_{\mathrm{Tx},0}$	$s_{\mathrm{Tx},1}$	$s_{\mathrm{Tx},2}$	$s_{\mathrm{Tx},3}$	$s_{\mathrm{Tx},4}$	$s_{\mathrm{Tx},5}$	$s_{\mathrm{Tx},6}$	$s_{\mathrm{Tx},7}$	0	0	0
0	$s_{\mathrm{Tx},0}$	$s_{\mathrm{Tx},1}$	$s_{\mathrm{Tx},2}$	$s_{\mathrm{Tx},3}$	$s_{\mathrm{Tx},4}$	$s_{\mathrm{Tx},5}$	$s_{\mathrm{Tx},6}$	$s_{\mathrm{Tx},7}$	0	0
0	0	$s_{\mathrm{Tx},0}$	$s_{\mathrm{Tx},1}$	$s_{\mathrm{Tx},2}$	$s_{\mathrm{Tx},3}$	$s_{\mathrm{Tx},4}$	$s_{\mathrm{Tx},5}$	$s_{\mathrm{Tx},6}$	$s_{\mathrm{Tx},7}$	0
0	0	0	$s_{\mathrm{Tx},0}$	$s_{\mathrm{Tx},1}$	$s_{\mathrm{Tx},2}$	$s_{\mathrm{Tx},3}$	$s_{\mathrm{Tx},4}$	$s_{\mathrm{Tx},5}$	$s_{\mathrm{Tx},6}$	$s_{\mathrm{Tx},7}$

(A) Example for a transmission signal matrix

$h_0^{(0)}$	$h_0^{(1)}$	$h_0^{(2)}$	$h_0^{(3)}$	$h_0^{(4)}$	$h_0^{(5)}$	$h_0^{(6)}$	$h_0^{(7)}$	$h_0^{(8)}$	$h_0^{(9)}$	$h_0^{(10)}$
$h_1^{(0)}$	$h_1^{(1)}$	$h_1^{(2)}$	$h_1^{(3)}$	$h_1^{(4)}$	$h_1^{(5)}$	$h_1^{(6)}$	$h_1^{(7)}$	$h_1^{(8)}$	$h_1^{(9)}$	$h_1^{(10)}$
$h_2^{(0)}$	$h_2^{(1)}$	$h_2^{(2)}$	$h_2^{(3)}$	$h_2^{(4)}$	$h_2^{(5)}$	$h_2^{(6)}$	$h_2^{(7)}$	$h_2^{(8)}$	$h_2^{(9)}$	$h_2^{(10)}$
$h_3^{(0)}$	$h_3^{(1)}$	$h_3^{(2)}$	$h_3^{(3)}$	$h_3^{(4)}$	$h_3^{(5)}$	$h_3^{(6)}$	$h_3^{(7)}$	$h_3^{(8)}$	$h_3^{(9)}$	$h_3^{(10)}$

(B) Example for a weight matrix

Chapter 8

Channel Model Application

In this chapter, the channel modelling approach presented in the previous chapter is used to simulate the transmission of the waveform of an example communication system. The communication system investigated here is based on the "popular" [31] OFDM *(Orthogonal Frequency-Division Multiplex)* modulation technique as it is used in established end-user communication standards like IEEE 802.11a/g/n (known as "Wi-Fi"), LTE, DVB-T *(Terrestrial Digital Video Broadcasting)*, or DAB *(Digital Audio Broadcasting)* as well as in aeronautical systems like LDACS or AeroMACS *(Aeronautical Mobile Airport Communication System)*.

8.1 Orthogonal Frequency-Division Multiplex

OFDM is a digital multi-carrier modulation scheme, thus the symbols that should be modulated into the waveform are distributed to individual sub-carriers. The core idea of OFDM is to "transform" a frequency selective channel into multiple parallel non-dispersive channels [31], where each sub-carrier is intended to be transmitted through its individual non-dispersive channel. In general, the resulting baseband waveform is complex[1]. This baseband waveform is than modulated onto a carrier signal such that it can be transmitted through the designated channel.

Besides the fact that OFDM can be implemented very efficiently in digital signal processors [31], the central advantages of OFDM are its flexibility and its robustness to frequency-selective fading [26].

One significant drawback of multi-carrier signals in general and OFDM systems in particular is the high PAPR, that increases with the number of used sub-carriers. The effects of a high PAPR have already been explained in Section 3.2. However, the straightforward solution applied in the referenced section – iterative clipping and filtering – is not applicable to a signal where actual data symbols are modulated on. An overview of several classes of PAPR reduction techniques for OFDM signals can be found in [16], [109] and is not discussed here any further.

This section focuses on the aspects of OFDM that are relevant for the understanding of the simulation results presented later in this chapter. A detailed OFDM introduction can be found in literature, e.g. [22], [26], [31], [87].

[1] Due to the properties of the FOURIER transform, it is possible to modify the OFDM principle such that the resulting baseband waveform is real; this approach is called DMT *(Discrete Multitone Transmission)*.

8.1.1 Processing Structure and Implementation

In the following, we briefly discuss the architecture of a discrete OFDM baseband system. We start and finish our explanation of an OFDM system with information bits in the transmitter and their estimates in the receiver.

8.1.1.1 Transmitter

In a real-world application like LDACS, an FEC and an interleaver are usually applied to the information bits; we refer to the result of this processing as code bits. The code bits are then mapped to (usually) complex symbols from a constellation like N-PSK *(Phase Shift Keying)* or N-QAM *(Quadtrature Amplitude Modulation)*.

These symbols are split up into multiple sets, where each set is later modulated onto an individual sub-carrier. Figure 8.1 visualises the schematic architecture of an OFDM frame of an example OFDM system: The n-th row of the matrix contains the symbols that are later modulated onto the n-th sub-carrier, while the m-th column of the matrix is later used to compute the m-th OFDM symbol of the OFDM frame. The figure also shows how symbols like pilot symbols – used for channel estimation in the receiver – or synchronisation symbols – used for synchronisation in the receiver – are distributed within the frame. It can also be seen, that not all sub-carriers of the frame are actually used for data transmission, but are set to zero: This is not only the case for the DC *(Direct Current)* carrier, but also for the guard carriers that are added here to reduce the actual OFDM signal bandwidth.

In contrast to a generic multi-carrier system, the sub-carriers of an OFDM system are – as the name suggests – *orthogonal* after demodulation. In [107] it is shown that this orthogonal sub-carrier modulation can be realised using an IDFT whose length corresponds to the amount of sub-carriers. In practice, the IDFT is almost always realised as an IFFT *(Inverse Fast Fourier Transform)* of length $N_{\text{FFT}} \in \mathbb{N}$, which allows an efficient implementation of an OFDM transmitter (and receiver, where the FFT *(Fast Fourier Transform)* is used accordingly), see [14]. Following this widely used approach, the m-th OFDM symbol of the frame is computed by applying an IFFT to the m-th column of the matrix given in Figure 8.1.

The time-discrete time-domain OFDM baseband signal is now created by concatenating the generated OFDM symbols. It is common practice to prepend each OFDM symbol with a copy of its last $N_{\text{CP}} \in \mathbb{N}, N_{\text{CP}} < N_{\text{FFT}}$ samples[2], called the *cyclic prefix*.

8.1.1.2 Receiver

For the description of the receiver, we assume the reception of the time-discrete time-domain OFDM baseband signal as it was generated in the above subsection. We also assume a perfect synchronisation in both time and frequency[3].

The received time-discrete signal is realigned to a matrix with $N_{\text{CP}} + N_{\text{FFT}}$ rows, where the first N_{CP} rows correspond to the cyclic prefixes of each OFDM symbol that were added in the transmitter. These first N_{CP} rows are removed and an DFT (in practice: FFT) is applied

[2] So far, we assume a setup without any additional upsampling.

[3] In practice, the SCHMIDL-COX synchronisation approach [81] is used in many systems.

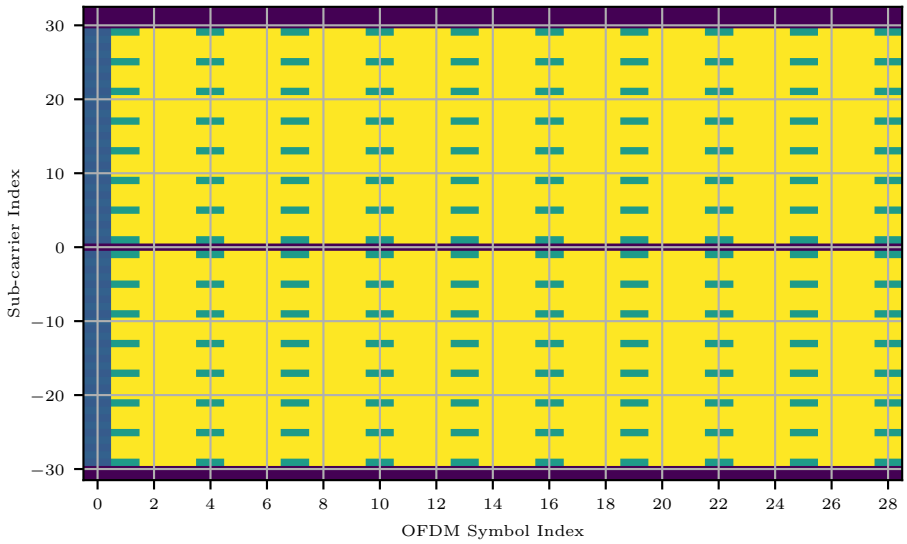


FIGURE 8.1: Visualisation of an example frame of an OFDM system with $N_{\text{FFT}} = 64$ and an OFDM frame length of $N_{\text{OS}} = 29$ in frequency-domain. Unset symbols – here: the guard carriers and the DC carrier – are given in purple, pilot symbols are given in turquoise, the synchronisation symbol – here: the OFDM symbol at index zero – is given in blue, and the remaining symbols used as data symbols are given in yellow.

to the remaining N_{FFT} rows of each column to transform the received OFDM symbol into frequency-domain.

The next step is the channel estimation: First, each of the *received* pilot symbols is divided by its corresponding *transmitted* pilot symbol at the same position within the OFDM frame. The result is a set of estimates of the complex frequency-domain channel coefficients, where $\hat{h}_{\text{CC}nm} \in \mathbb{C}$ denotes the estimate of the frequency-domain channel coefficient of the n-th sub-carrier and the m-th OFDM symbol. The remaining estimates of the channel coefficients can be computed by an interpolation using the already estimated coefficients as the basis: In [18], it is suggested to apply a linear interpolation along the OFDM symbol axis and another linear interpolation along the sub-carrier axis. The result is a matrix containing the estimates of the frequency-domain channel coefficients for every symbol in the received frame. We denote this matrix by $\hat{\boldsymbol{H}}_{\text{CC}} \in \mathbb{C}^{N_{\text{SC}} \times N_{\text{OS}}}$, where $N_{\text{SC}} \in \mathbb{N}, N_{\text{SC}} \leq N_{\text{FFT}}$ denotes the amount of sub-carriers that contain actual data, thus all carriers but the guard carriers and the DC carrier. The matrix containing the *actual* channel coefficients is denoted by $\boldsymbol{H}_{\text{CC}} \in \mathbb{C}^{N_{\text{SC}} \times N_{\text{OS}}}$ and is not known to the receiver in a real-world system.

The channel equalisation is now performed using $\hat{\boldsymbol{H}}_{\text{CC}}$: The simplest linear equalisation approach is to divide the received data symbol at position (n, m) within the OFDM frame by $\hat{h}_{\text{CC}nm}$. This equalisation approach fulfils the ZF *(Zero-Forcing)* criterion and is not taking noise into account. A more practice-relevant equalisation approach it the equaliser fulfilling the MMSE *(Minimum Mean Squared Error)* criterion, as it tries to minimise the MSE *(Mean Squared Error)* between the transmitted and the received symbols and therefore takes the noise power into account [31]. A deeper discussion of channel equalisation in OFDM systems goes beyond the scope of this chapter and can be found in literate, e.g. [76].

The equalised data symbols can now be demodulated using the appropriate demodulator (e.g. N-PSK or N-QAM) and eventually the interleaver and FEC decoder can be applied to

the demodulator's output. The result are the estimates of the information bits.

8.1.2 Key OFDM Parameters

The above introduction has already mentioned some key parameters, which are explained in detail in this section. During the design process of an OFDM system, these parameters must be adapted to the properties of the channel the designed waveform is intended to be sent through.

8.1.2.1 Sub-carrier Spacing

As described above, OFDM is a multi-carrier system and, as such, the frequency gap between the sub-carriers – called sub-carrier spacing denoted by Δf – is of central relevance.

In general it is advised to set the sub-carrier spacing according to certain properties of the channel: the coherence bandwidth $W_\mathrm{c} \approx \sigma_\tau^{-1}$ [31] and the Doppler bandwidth W_ν. However, there are no more than a few rules of thumb in the literature to define the actual sub-carrier spacing: While [26] suggest to set $\Delta f \ll W_\mathrm{c}$, [83] is more precise and suggests to set $\Delta f < 0.1 W_\mathrm{c}$. A lower limit is proposed in [83] as $\Delta f > 10 W_\nu$.

Other OFDM system parameters are derived from this sub-carrier spacing, e.g.:

- The duration of one single OFDM symbol is given by $T_\mathrm{OS} = (\Delta f)^{-1}$.
- Since one OFDM symbol comprises N_FFT samples, the duration of one single sample within an OFDM system is given by $T_\mathrm{Smpl} = T_\mathrm{OS}/N_\mathrm{FFT} = (\Delta f N_\mathrm{FFT})^{-1}$.

8.1.2.2 Amount of Sub-carriers

Both the OFDM symbol duration T_OS and the overall OFDM bandwidth $W_\mathrm{OFDM} = N_\mathrm{FFT}\Delta f$ generally depend on the amount of sub-carriers. The *occupied* OFDM bandwidth, however, depends on the actually used carriers: In practice, guard carriers (i.e. sub-carriers without any power) are often added on the lower and the upper part of the spectrum. This allows a flexible setting of the occupied signal bandwidth while still having the option to set N_FFT to a power of two – which is beneficial for the computation of the (I)FFT [14] – while keeping the sub-carrier spacing fixed. These guard carriers can of course also be used to generate an inherently oversampled OFDM signal.

8.1.2.3 Cyclic Prefix Length

The transmission of a signal through a channel can be described as a convolution of the signal with the CIR *(Channel Impulse Response)* (see Section 2.2.1.1 for definition and assumptions). In the discrete case, this operation generally corresponds to a *linear* convolution. However, it is desirable to "transform" this *linear* convolution to a *circular* convolution as this allows the application of the DFT's convolutional theorem [26]. This greatly simplifies the channel equalisation because it can now be implemented as a one-tap equaliser in frequency-domain [31]. In practice, one tries to achieve this "transformation" from a linear convolution to a

circular convolution by prepending a cyclic prefix to each OFDM symbol. This cyclic prefix of length

$$T_{\mathrm{CP}} = N_{\mathrm{CP}} T_{\mathrm{Smpl}} \tag{8.1}$$

should exceed the CIR[4], thus $T_{\mathrm{CP}} \geq \tau_{\max}$. If this condition is fulfilled, ISI *(Inter-Symbol Interference)* is eliminated. If this condition is not fulfilled, the sub-carriers' orthogonality is corrupted and the one-tap frequency-domain channel equalisation is not able to neutralise all channel impairments.

8.1.2.4 Distribution of Pilot Symbols

The estimation of the channel coefficients is the fundament of the channel equalisation. The channel can get estimated by the application of pilot symbols, i.e. symbols known to the receiver that are distributed over an OFDM frame. Since pilot symbols cannot be used to transmit actual information, they have a negative effect on the bandwidth efficiency η_W of an OFDM system. As such, the amount of pilot symbols should be chosen wisely, i.e. by taking the relevant channel properties into account; this is the change rate of the channel coefficients along the time axis and along the frequency axis. Thus, the pilot distance shall fulfil the sampling theorem along both the time and the frequency axis to avoid aliasing in the channel estimation. According to [83], this is the case if the pilot distance along the *time* axis given in samples is set to

$$N_{\mathrm{P}t} \leq \left\lfloor \left(W_\nu \left(T_{\mathrm{OS}} + T_{\mathrm{CP}} \right) \right)^{-1} \right\rfloor \tag{8.2}$$

and if the pilot distance along the *frequency* axis given in samples is set to

$$N_{\mathrm{P}f} \leq \left\lfloor \left(\tau_{\max} \, \Delta f \right)^{-1} \right\rfloor . \tag{8.3}$$

In the example in Figure 8.1, it is $N_{\mathrm{P}t} = 3$ and $N_{\mathrm{P}f} = 4$.

The resulting distance between pilot symbols along the time axis is given by

$$\Delta T_{\mathrm{P}} = N_{\mathrm{P}t} (T_{\mathrm{OS}} + T_{\mathrm{CP}}), \tag{8.4}$$

the resulting distance between pilot symbols along the frequency axis is given by

$$\Delta f_{\mathrm{P}} = N_{\mathrm{P}f} \, \Delta f. \tag{8.5}$$

8.2 Simulations

To demonstrate the application of the channel modelling approach proposed in Chapter 7, OFDM signals of different configurations are generated and sent through the model. Each of these configurations is a set of the parameters listed in Section 8.1.2.

[4] To be more precise: Exceed the parts of the CIR that contribute significant power.

8.2.1 Implementation

In the following, the steps for the simulation of the transmission of an OFDM signal through the channel model are explained. These steps are repeated for each OFDM configuration individually until a certain amount of information bits has been transmitted.

A block diagram of the processing is part of the thesis overview given in Figure 1.1.

8.2.1.1 Deriving a Model Instance

For the simulations described here, it is assumed that a model has already been created according to the steps introduced in Section 7.2. The derivation of a model *instance* as described in Section 7.2.5, however, *is* part of each simulation.

The derived instance is used for the simulation of the transmission of $N_{\mathrm{Frm}} \in \mathbb{N}$ OFDM frames before a new instance is derived. By using the *same* model instance – hence the *same* MPC graph – as a base for the simulation of the transmission of N_{Frm} OFDM frames, it is possible to make use of the central property of the proposed channel modelling approach, i.e. the evolution of MPCs over time.

8.2.1.2 Generating the Weight Matrix

The MPC graph of the derived model instance is now created and evaluated at each relevant time instance. These time instances correspond to the timestamps of the samples of the signal that is later transmitted through the channel. As we know from Section 7.4.1, the block duration T_{B} is the central value for the evaluation of the MPC graph. We set $T_{\mathrm{B}} = T_{\mathrm{Smpl}}/f_{\mathrm{up}}$, where f_{up} denotes the oversampling factor of the OFDM signal used in the simulation.

As mentioned above, the transmission of N_{Frm} frames instead of just one single frame is simulated. Thus, the length of the signal that is sent through the channel is given by

$$N_s = \underbrace{\overbrace{(N_{\mathrm{FFT}} + N_{\mathrm{CP}})}^{\substack{\text{One OFDM symbol}\\ \text{with cyclic prefix}}} N_{\mathrm{OS}}}_{\text{One OFDM frame}} N_{\mathrm{Frm}} f_{\mathrm{up}}. \tag{8.6}$$

The time instances, the MPC graph is evaluated at, are given by the vector

$$\boldsymbol{t} = \left[\, 0,\ T_{\mathrm{B}},\ 2T_{\mathrm{B}},\ 3T_{\mathrm{B}},\ \ldots,\ (N_s + h_{\mathrm{len}} - 2)T_{\mathrm{B}} \,\right], \tag{8.7}$$

where h_{len} denotes the length of the discrete CIR of a block. We understand that the b-th entry of $\boldsymbol{t}$ corresponds to the time instance of the b-th block of the MPC graph. The CIR of the MPC graph's b-th block given by $\boldsymbol{h}^{(b)}$ is computed according to (7.23).

8.2.1.3 Transmitting a Signal Through the Channel

The OFDM signal is generated according to the steps in Section 8.1.1.1 using the given set of OFDM parameters. The signal is then oversampled by f_{up} and transformed to the TOEPLITZ structure shown in Figure 7.5a.

The transmission through the channel is performed by executing the remaining steps given in Section 7.4.2, i.e. the TOEPLITZ matrix and the weight matrix are multiplied element-wise and the sum along the resulting matrix' columns is computed. Complex white GAUSSian noise is then added to the signal according to a given SNR (here: 16 dB).

The noised signal after the channel is now downsampled by f_{up} using a polyphase filtering/resampling approach, see e.g. [90]. The downsampled signal is then split into its individual OFDM frames. Each frame is reshaped such that it can be processed by the OFDM receiver as described in Section 8.1.1.2.

8.2.1.4 Evaluating the Performance of the Channel Estimation

For each frame, the MSE between the elements of the matrix of the estimated channel coefficients $\hat{\boldsymbol{H}}_{\mathrm{CC}}$ and the elements of the matrix of the actual channel coefficients $\boldsymbol{H}_{\mathrm{CC}}$ is determined. The averaged MSE over all transmitted frames of a certain OFDM configuration denoted by e_{CC} is later used to compare the performance of the channel estimation of the different OFDM configurations for the given channel. It is computed according to

$$e_{\mathrm{CC}} = \frac{1}{N_{\mathrm{Frm}}} \sum_{i=0}^{N_{\mathrm{Frm}}-1} \underbrace{\frac{1}{N_{\mathrm{SC}} N_{\mathrm{OS}}} \sum_{n=0}^{N_{\mathrm{SC}}-1} \sum_{m=0}^{N_{\mathrm{OS}}-1} \left| \hat{h}_{\mathrm{CC}nm}^{(i)} - h_{\mathrm{CC}nm}^{(i)} \right|^2}_{\text{MSE of channel estimation for frame } i}. \tag{8.8}$$

The values of $\boldsymbol{H}_{\mathrm{CC}}$ are determined as follows: Analogous to the steps in Section 8.2.1.3, a noise-free copy of the OFDM signal behind the channel is processed by an independent OFDM receiver. As shown in Section 8.1.1.2, the processing in the OFDM receiver includes the reshaping and the transformation into frequency-domain. Each of these frequency-domain symbols of the *received* OFDM frame is divided by the corresponding frequency-domain symbol of the corresponding *transmitted* OFDM frame. The result is the matrix of channel coefficients that *perfectly* represents the impairments of the channel on the transmitted signal (except for the noise). It can therefore be used as a benchmark for the performance of the channel estimation.

The reader may note that the described determination of $\boldsymbol{H}_{\mathrm{CC}}$ is also independent of any ISI, i.e. even in case of an inappropriately short cyclic prefix, $\boldsymbol{H}_{\mathrm{CC}}$ can always be used to perfectly equalise the OFDM frame. Thus, e_{CC} is a *joined* measure for the effect of the OFDM parameters ΔT_{P}, Δf_{P}, and T_{CP} as they are defined in (8.1), (8.4) and (8.5), respectively, on the channel estimation performance.

8.2.2 OFDM Configurations

For the following simulations, a fixed FFT length of $N_{\mathrm{FFT}} = 64$ is used. The determination of the remaining OFDM parameters as given in Section 8.1.2 is explained in the following.

First the sub-carrier spacing Δf is set – here, we use an integer multiple of 12.207 kHz.

Then, the cyclic prefix length in samples N_{CP} is determined based on a desired cyclic prefix length in seconds T'_{CP}. As shown in Section 8.1.2, this derivation depends on the sample duration and as such depends on Δf. Due to the fixed Δf and the required quantisation in a discrete environment, it is not possible to set arbitrary values for T_{CP}, thus we set

$N_{CP} = \lceil T'_{CP}/T_{Smpl} \rceil$, such that $T_{CP} \geq T'_{CP}$ is ensured. For the OFDM configurations simulated in the following, $T'_{CP} \in \{1\,\mu s, 10\,\mu s, 20\,\mu s\}$ has been used.

The pilot distances along the time and frequency axis are determined using a similar approach: A desired maximum distance along time and frequency, respectively, is given and the closest possible (lower) values using the given Δf are used for the actual pilot distances for the given configuration.

The guard carriers are determined such that the pilot symbols form a symmetric grid as given in the example in Figure 8.1, while using as many sub-carriers as possible.

The reader may note that certain combinations of desired OFDM parameters may be impossible to achieve, e.g. if the sub-carrier spacing is larger than the desired maximum pilot distance along the frequency axis.

8.2.3 Results

In this section, the results of the simulation of several OFDM waveform configurations are presented and discussed. These results are presented in figures showing scatter plots that all share the same structure: The XY-position of a data point within the scatter plot shows the pilot distance along the time axis ΔT_P and frequency axis Δf_P, respectively. The marker type of a data point indicates the corresponding sub-carrier spacing Δf according to the legend of each figure. The value of the data point – thus the actual measured variable – is colour-coded using a heatmap which scale is shown on the right-hand side of each figure.

The individual sub-figures show the results for OFDM configurations with different desired cyclic prefix lengths T'_{CP}. As explained above, this desired cyclic prefix length is not always exactly achievable and thus slightly varies between the configurations shown in a sub-figure. Nevertheless, it was chosen to group the results of OFDM configurations with similar T_{CP} in one sub-figure. In the presented results, the deviation never exceeds 12 % of the respective T'_{CP}.

8.2.3.1 Observations

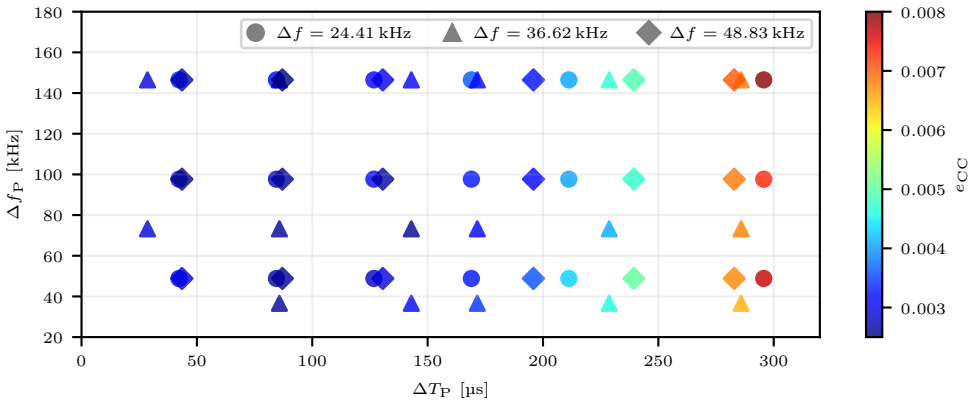
Figure 8.2 shows the resulting e_{CC} as introduced in Section 8.2.1.4 for different OFDM configurations. We can make the following observations:

1. In all cases e_{CC} increases with an increasing ΔT_P. This is especially the case for $\Delta T_P > 180\,\mu s$.

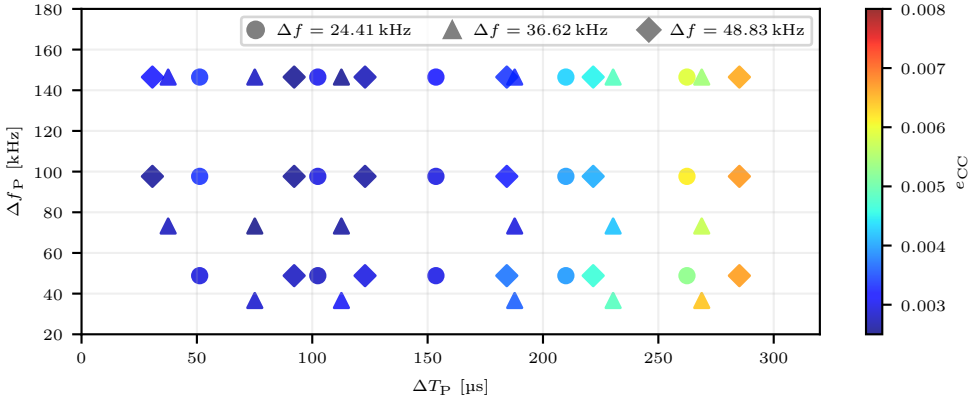
2. Although the dependency is not as distinct as for ΔT_P, e_{CC} also slightly increases with increasing values of Δf_P.

3. In most cases, a larger sub-carrier spacing leads to lower values of e_{CC}.

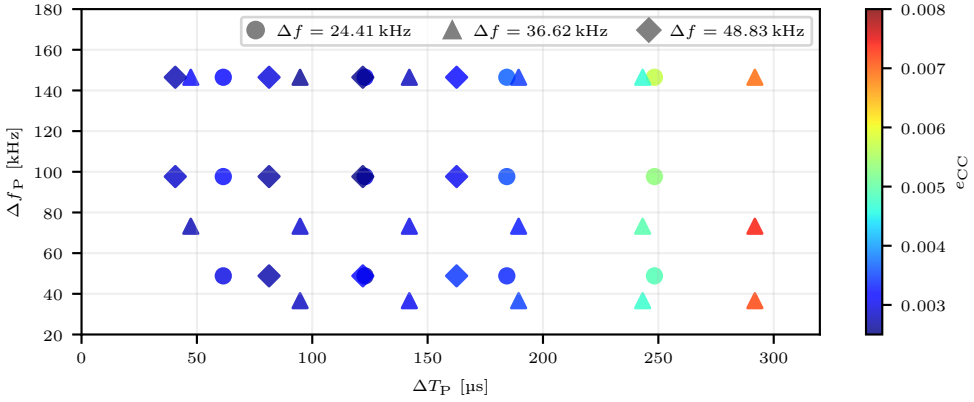
4. Comparing the results for $\Delta T_P > 250\,\mu s$ between $T'_{CP} = 1\,\mu s$ and $T'_{CP} = 10\,\mu s$ shows a clear decrease in the channel estimation error. However, this decrease is larger than the decrease comparing the results for $T'_{CP} = 10\,\mu s$ and $T'_{CP} = 20\,\mu s$.



(A) $T'_{CP} = 1\,\mu s$



(B) $T'_{CP} = 10\,\mu s$



(C) $T'_{CP} = 20\,\mu s$

FIGURE 8.2: Resulting channel estimation error e_{CC} for several OFDM configurations.

8.2.3.2 Interpretation and Discussion

We interpret the above observations as follows:

- Observations 1. and 2. can be explained by the hint to the sampling theorem mentioned in Section 8.1.2.4: In case of too distant pilot symbols, the channel changes quicker than the pilot symbols can detect; thus, the interpolation is unable to sufficiently reconstruct the missing channel coefficients. Apparently, this effect is more relevant for the investigated ΔT_P than for the investigated Δf_P.

- If N_FFT is kept constant, a decreasing Δf leads to longer OFDM symbols. From observation 3. we therefore conclude that shorter OFDM symbols are beneficial for the given channel. As we know from Section 8.1.2.1, this can be explained by the Doppler bandwidth; however, literature lacks a strict definition for an optimal relation.

- The purpose of the cyclic prefix has been explained in Section 8.1.2.3 and helps to understand observation 4.: Apparently, the simulated channel's CIRs are (usually) longer than the cyclic prefix of the OFDM configurations used in Figure 8.2a. Thus, the estimation of the channel becomes more difficult due to the ISI. If the cyclic prefix' length gets increased, the ISI is reduced and the channel estimation error is reduced. However, a further increase appears to have no significant positive effect on the system performance for the given channel. Thus, we assume that the simulated channel's CIRs do not have much power for delays larger than 10 µs. The RMS-DSs shown in Figure 7.3 – which were determined based on the same channel data that were used for the OFDM simulations – support this assumption.

8.2.3.3 Concluding Remarks

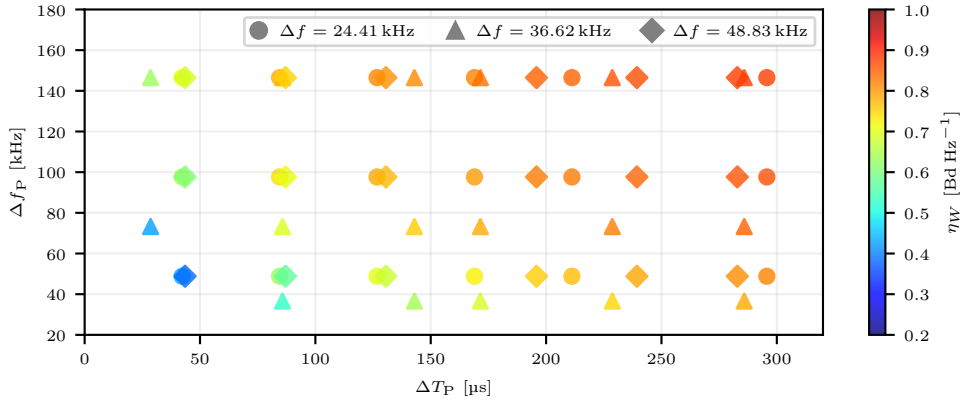
The results clearly showed that the channel estimation error increases in case the parameter settings do not match the actual channel parameters, e.g. in case the pilot symbols are too distant and/or the cyclic prefix is too short. It is intuitive to understand that a higher density of pilot symbols or a longer cyclic prefix has a negative effect on the *bandwidth efficiency* η_W (also known as spectral efficiency), since no additional information is transmitted although the transmission of the signal itself consumes more resources.

The bandwidth efficiency quantifies the rate of information per bandwidth that is transmitted in a system. It is usually defined with respect to bits, thus its unit is $[\mathrm{bit\,s^{-1}\,Hz^{-1}}]$. However, the above discussion of the OFDM system used for the simulations did not involve any definition of a symbol alphabet that would allow a quantification of the transmitted information given in [bit]. We therefore define the bandwidth efficiency with respect to symbols instead to bits; its unit is therefore $[\mathrm{Bd\,Hz^{-1}}]$, where [Bd] is the unit of the symbol rate[5].

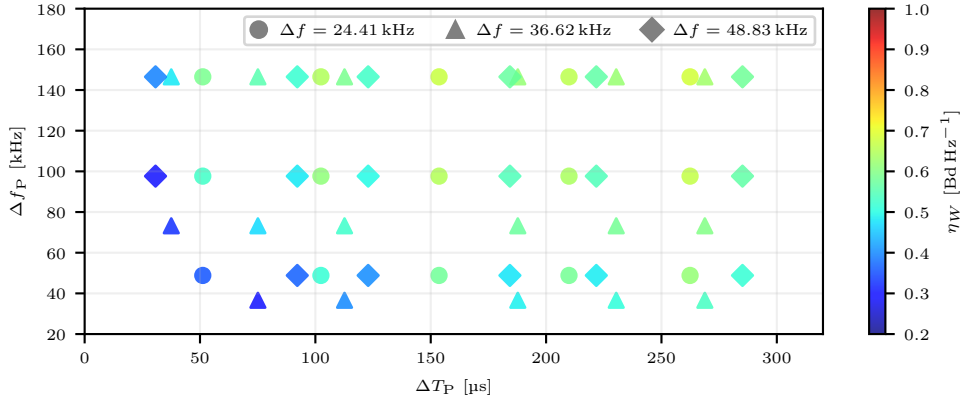
The bandwidth efficiency according to this definition is given in Figure 8.3 for all simulated OFDM configurations. The decrease of the bandwidth efficiency for higher pilot distances and longer cyclic prefixes described above is clearly visible. Comparing the data points from

[5]The symbol rate is also called baud rate, named after BAUDOT.

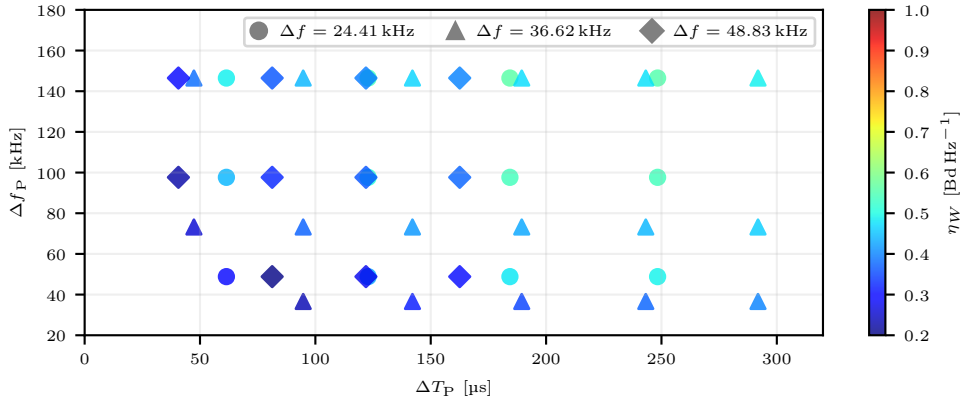
Figure 8.2 with the corresponding data point's value from Figure 8.3 shows that the lower channel estimation error comes at the cost of a lower bandwidth efficiency.



(A) $T'_{\mathrm{CP}} = 1\,\mu\mathrm{s}$



(B) $T'_{\mathrm{CP}} = 10\,\mu\mathrm{s}$



(C) $T'_{\mathrm{CP}} = 20\,\mu\mathrm{s}$

FIGURE 8.3: Resulting bandwidth efficiency η_W for several OFDM configurations. The reader may note that η_W is given with respect to symbols and not bits since the cardinality of the symbol alphabet is not defined.

Chapter 9

Conclusion

This dissertation covered important steps in the development of a wireless communication system for civil aviation. A flight measurement campaign has been executed and much channel sounding data of many different flight scenarios have been collected, see Chapter 3. Although the data collected during this comprehensive campaign are the basis for the remaining dissertation, they are by far not evaluated completely. They are expected to be valuable to the scientific community for further evaluation since they contain much more information that can be used for the development of aeronautical communication and navigation systems than can be discussed in a single dissertation.

Chapter 4 is not only intended as a detailed documentation of the pre-processing of the measurement data and the computation and efficient organisation of the GT data, but is also expected to be helpful for the processing and organisation of future measurement campaign data.

The different processing steps of the collected measurement data have been discussed in the following chapters: While the analysis of the dominant component as explained in Chapter 5 has shown that comparatively simple models like two-ray channel models can be used to model the evolution of the dominant component in en-route flight scenarios, novel approaches in dealing with MPCs have been discussed in Chapter 6.

The modular nature of the separated detection and tracking of MPCs enables an efficient processing: The laborious processing of the huge amounts of collected IQ data to extract the truly relevant information (in terms of the MPCs) is parallelised where possible due to the reduction of data dependencies. Where those dependencies cannot be reduced – e.g. when tracking MPCs – the processing is not performed in the domain of IQ data, but in the domain of the extracted information from the previous processing step.

So far, the MPC extraction has been performed by cutting the IQ data into consecutive data blocks without any overlapping; however, it would also be interesting to compare the results of this approach to an alternative approach where the processed blocks overlap, i.e. by using a sliding window where the window length corresponds to the block length and the step size is considerable smaller than the block size. The introduced MPC tracking approach – and further processing steps based on the results of this tracking – can then be applied to the detected MPC without any adaptions.

Chapter 6 also proposed an approach to locate the reflectors that cause the MPC environment by combining GT data from Chapter 4 with results from the MPC tracking. The

successful localisation in the given example is understood as a validation of the whole processing chain (pre-processing of IQ data, computation of GT data, MPC detection, MPC tracking, and the reflector localisation itself) that lead to these results: Due to the constantly changing environmental conditions in the given example, it is close to impossible that the localisation would produce any useful outcome if just one step in the described processing chain would be erroneous.

Some properties of the MPC paths resulting from the MPC tracking motivated their further investigation in Chapter 7. The outcome is the compact representation, where the delay evolution of an MPC is described by a polynomial, whose coefficients are determined using an optimisation algorithm, and the amplitude of an MPC is described by statistical properties. Finding alternative approaches for this compact representation is an interesting research topic for the future.

The channel model itself is essentially a set of KDE kernels, where each kernel has been computed over the compact representation vectors of a cluster of MPC paths. An instance of the channel model is generated by sampling (and evaluating) a certain amount of compact representation vectors from each kernel. It has been shown, that the overall channel statistics of these generated channels are comparable to the corresponding properties of the measured channel. For the future it would be interesting to run the described model fitting approach over more data, recorded during different phases of flight.

Since the evolution of the MPC parameters is central for both the creation of the model, as well as the generation of a chanel model instance, it is well reflected in the generated channel and allows, in contrast to other channel modelling approaches, a realistic simulation of the evolution of the MPC parameters over time.

The channel modelling approach is used for the simulation of multiple OFDM waveforms – as they could be used for a C2 link for unmanned aviation – in Chapter 8. This chapter not only explains the actual implementation of the simulations, but also shows and discusses simulation results. These results do not only show that the proposed channel modelling approach is actually useable in practice, but also show that some of the OFDM parameter sets used in the simulations are more suitable for the given example channel than others.

Appendix A

Notation

This dissertation aims to consequently follow common notation standards that are well established in the community of information theory. However, since not all notation is entirely agreed on within the community, this appendix shall provide some clear definitions to avoid misunderstandings.

A.1 Vectors and Matrices

A vector is denoted by a small letter in bold type. A vector with N elements from a field $\mathbb{A}$ is defined as $\boldsymbol{a} \in \mathbb{A}^N$ and given by

$$\boldsymbol{a} = [a_0, a_1, \ldots, a_{N-1}], \text{ with } a_i \in \mathbb{A} \forall i \in \{0, 1, \ldots, N-1\}. \tag{A.1}$$

Commonly used fields are integers $\mathbb{N}$, real numbers $\mathbb{R}$, and complex numbers $\mathbb{C}$.

The elements of a vector are indexed, the first element is indexed by 0. The i-th element of a vector $\boldsymbol{a}$ is given by a_i. Alternatively, the i-th element of a vector $\boldsymbol{a}$ may be given by $\boldsymbol{a}[i]$ to improve readability. These two notations are used interchangeably.

A matrix is denoted by a capital letter in bold type. A matrix with M rows and N columns and elements from a field $\mathbb{A}$ is defined as $\boldsymbol{A} \in \mathbb{A}^{M \times N}$ and given by

$$\boldsymbol{A} = (a_{mn}) = (a_{m,n}) = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}$$
$$\text{with } a_{mn} \in \mathbb{A} \forall m \in \{0, 1, \ldots, M-1\} \forall n \in \{0, 1, \ldots, N-1\}. \tag{A.2}$$

Analogous to the vector definition, a matrix' elements are indexed starting from 0. The element of the m-th row and the n-th column of a matrix $\boldsymbol{A}$ is given by $a_{mn} = a_{m,n} = \boldsymbol{A}[m, n]$.

A.2 Sequences

Finite sequences are commonly used throughout this dissertation. A sequence is denoted by a bold symbol and is defined by

$$\boldsymbol{s} = (a_A, a_{A+1}, \ldots, a_B) = (\, a_i \,)_{i=A}^{B}, \tag{A.3}$$

where the index of the first element of the sequence is given by $A \in \mathbb{N}$ and the index of the last element of the sequence is given by $B \in \mathbb{N}$. In some cases, a superscript notation is used instead of the subscript notation from above to improve readability; the definition is given by

$$\boldsymbol{s} = \left(a^{(A)}, a^{(A+1)}, \ldots, a^{(B)}\right) = \left(\, a^{(i)} \,\right)_{i=A}^{B}. \tag{A.4}$$

To access a specific element of a sequence, the vector notation from above is applied; thus the i-th element of the sequence $\boldsymbol{s}$ from above is given by a_i or $\boldsymbol{s}[i]$, respectively, where the latter notation is preferred for the sake of readability.

Appendix B

FOURIER Analysis

B.1 Continuous Signals

B.1.1 From Time-Domain to Frequency-Domain

The FOURIER transform of a time-continuous signal $s(t) \in \mathbb{C}$ is given by

$$\begin{aligned} S(f) &= \int_{-\infty}^{\infty} s(t)\, \mathrm{e}^{-\mathrm{j}2\pi f t} \mathrm{d}t \\ &= \mathcal{F}\{s(t)\}_{\mathrm{d}t}. \end{aligned} \tag{B.1}$$

B.1.2 From Frequency-Domain to Time-Domain

Using the above nomenclature, the inverse FOURIER transform is hence given by

$$\begin{aligned} s(t) &= \int_{-\infty}^{\infty} S(f)\, \mathrm{e}^{\mathrm{j}2\pi f t} \mathrm{d}f \\ &= \mathcal{F}^{-1}\{S(f)\}_{\mathrm{d}f}. \end{aligned} \tag{B.2}$$

German-language literature in particular often uses the nomenclature

$$s(t) \circ\!\!-\!\!\bullet\, S(f) \quad \text{and} \quad S(f) \bullet\!\!-\!\!\circ\, s(t), \tag{B.3}$$

respectively, to show the FOURIER relation between two signals.

B.1.3 Selected Properties

B.1.3.1 Time- and Frequency-Shift

A shift in time-domain by a time period $t_{\mathrm{shft}} \in \mathbb{R}$ corresponds to a multiplication with an exponential term in frequency-domain:

$$s(t - t_{\mathrm{shft}}) \circ\!\!-\!\!\bullet\, S(f)\, \mathrm{e}^{-\mathrm{j}2\pi f t_{\mathrm{shft}}}. \tag{B.4}$$

Vice versa, a shift in frequency-domain by a frequency offset of $f_{\text{shft}} \in \mathbb{R}$ corresponds to a multiplication with an exponential term in time-domain:

$$s(t)\ \mathrm{e}^{\mathrm{j}2\pi f_{\text{shft}} t} \circ\!\!-\!\!\bullet\ S(f - f_{\text{shft}}). \tag{B.5}$$

B.1.3.2 Convolution and Multiplication

According to the convolution theorem, the time-domain convolution of two signals $a(t)$ and $b(t)$ corresponds to a multiplication in frequency-domain:

$$\begin{aligned} a(t) * b(t) &= \mathcal{F}^{-1}\{\ \mathcal{F}\{a(t)\}_{\mathrm{d}t} \mathcal{F}\{b(t)\}_{\mathrm{d}t}\ \}_{\mathrm{d}t} \\ \text{thus: } a(t) * b(t) &\circ\!\!-\!\!\bullet\ A(f)\ B(f). \end{aligned} \tag{B.6}$$

A multiplication of two time-domain signals $a(t)$ and $b(t)$, however, corresponds to a convolution in frequency domain:

$$a(t)\ b(t) \circ\!\!-\!\!\bullet\ A(f) * B(f). \tag{B.7}$$

B.2 Discrete Signals

B.2.1 From Time-Domain to Frequency-Domain

While the FOURIER transform as explained above is applied to time-continuous signals, the DFT *(Discrete Fourier Transform)* is applied to time discrete signals. The DFT of a time-discrete signal represented by a vector $\boldsymbol{x} = [x_0, x_1, \ldots, x_{N-1}] \in \mathbb{C}^N$ is given by

$$\begin{aligned} X_k &= \sum_{n=0}^{N-1} x_n\ \mathrm{e}^{-\mathrm{j}2\pi \frac{nk}{N}},\ \ \forall\, k \in \{0, 1, \ldots, N-1\} \\ \text{or as shorthand: } \boldsymbol{X} &= \mathrm{DFT}\{\boldsymbol{x}\},\ \ \text{with } \boldsymbol{X} \in \mathbb{C}^N \end{aligned} \tag{B.8}$$

According to the above definition

- X_0 corresponds to the zero-frequency term (often called DC carrier in multi-tone modulation schemes),
- $[X_1, X_2, \ldots, X_{N/2}]$ correspond to the positive-frequency terms in order of increasingly positive frequency, and
- $[X_{N/2+1}, X_{N/2+2}, \ldots, X_{N-1}]$ correspond to the negative-frequency terms in order of decreasingly negative frequency.

If we assume a vector $\boldsymbol{X}' \in \mathbb{C}^N$ with elements $X'_k = X_{\underset{N}{\mathrm{mod}}\{k+\lceil N/2 \rceil\}}$, the zero-frequency term is in its middle and the elements of this vector are representing strictly increasing frequencies. Transforming $\boldsymbol{X}$ to $\boldsymbol{X}'$ according to this definition is called a DFT-shift and denoted by the operator $\boldsymbol{X}' = \mathrm{DSHFT}\{\boldsymbol{X}\}$. The inverse of this index transformation is given by $X_k = X'_{\underset{N}{\mathrm{mod}}\{k+\lfloor N/2 \rfloor\}}$ and called inverse DFT-shift and denoted by the operator $\boldsymbol{X} = \mathrm{DSHFT}^{-1}\{\boldsymbol{X}'\}$.

B.2.2 From Frequency-Domain to Time-Domain

Analogous to the time-continuous case, we define the IDFT *(Inverse Discrete Fourier Transform)*:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k \, \mathrm{e}^{\mathrm{j}2\pi \frac{nk}{N}}, \quad \forall \, n \in \{0, 1, \ldots, N-1\}$$
$$\text{or as shorthand: } \boldsymbol{x} = \mathrm{DFT}^{-1}\{\boldsymbol{X}\}. \tag{B.9}$$

B.2.3 Selected Properties

B.2.3.1 Time- and Frequency-Shift

According to the shift theorem of the DFT, shifts can be handled similarly to the continuous case, however, they usually correspond to a circular shift. A shift according to $x_{n-n_{\mathrm{shft}}}, n_{\mathrm{shft}} \in \mathbb{N}$ in time-domain corresponds to a multiplication with an exponential term according to $X_k \, \mathrm{e}^{-\mathrm{j}2\pi n_{\mathrm{shft}} k/N}$ in frequency-domain.

Correspondingly, a shift in frequency-domain $X_{k-k_{\mathrm{shft}}}, k_{\mathrm{shft}} \in \mathbb{N}$ corresponds to a multiplication by an exponential term according to $x_n \, \mathrm{e}^{\mathrm{j}2\pi k_{\mathrm{shft}} n/N}$ in time-domain.

B.2.3.2 Convolution and Multiplication

The convolution theorem is also defined for the discrete case: The convolution of two discrete time-domain signals $\boldsymbol{a} \in \mathbb{C}^N$ and $\boldsymbol{b} \in \mathbb{C}^N$ as defined in e.g. [29], [31] is given by

$$(\boldsymbol{a} * \boldsymbol{b})_n = \sum_{n'=0}^{N-1} a_{n-n'} \, b_{n'} = \sum_{n'=0}^{N-1} a_{n'} \, b_{n-n'} \tag{B.10}$$

and corresponds to the (element-wise) multiplication of their frequency-domain counterparts $\boldsymbol{A} \in \mathbb{C}^N$ and $\boldsymbol{B} \in \mathbb{C}^N$, respectively. Applying the nomenclature introduced above to the discrete case as suggested in [31], we get

$$\boldsymbol{a} * \boldsymbol{b} \circ\!\!-\!\!\bullet \, \boldsymbol{A} \cdot \boldsymbol{B}. \tag{B.11}$$

Correspondingly, the element-wise multiplication in time-domain corresponds to a convolution in frequency-domain:

$$\boldsymbol{a} \cdot \boldsymbol{b} \circ\!\!-\!\!\bullet \, \boldsymbol{A} * \boldsymbol{B}. \tag{B.12}$$

B.2.4 Additional Remarks

In case the input to the DFT is a matrix instead of a vector, the DFT operator is expanded by an index that indicates whether the DFT is performed either column-wise $\mathrm{DFT}_{\mathrm{cols}}\{\square\}$ or row-wise $\mathrm{DFT}_{\mathrm{rows}}\{\square\}$. The same nomenclature applies to all other operators defined in this section.

In the actual implementation, the (inverse) FFT *(Fast Fourier Transform)* according to COOLEY and TUKEY [14] is used instead of the (inverse) DFT if not denoted otherwise.

Appendix C

Computation of CIR and Other CIR-based Data

This appendix explains the actual implementations used in this dissertation. It therefore focuses only on time-discrete signals that are represented by vectors.

C.1 Calculation of the Channel Impulse Response

From literature [70] it is known that the impulse response of an LTI *(Linear Time Invariant)* system can be estimated by cross-correlating the input with the system output. It is also known that the cross-correlation is closely related to the convolution, which – as described in Appendix B – is straightforward to compute in frequency-domain, as it just corresponds to a simple element-wise multiplication. This section explains how these two properties are exploited for the implementation of the estimation of the CIRs.

The transmitted signal corresponds to the reference signal $\boldsymbol{x}_{\text{ref}} \in \mathbb{C}^N$ as described in Section 4.2.2.2; by definition, it contains exactly one complete instance of the channel sounding sequence having the length of N samples. The received signal assumed here is denoted by $\boldsymbol{x} \in \mathbb{C}^{BN}$; thus, its length corresponds exactly to the length of B consecutive instances of the channel sounding sequence.

The CIRs of a block of received data $\boldsymbol{x}$ are computed as follows:

1. In a first step, $\boldsymbol{x}$ is converted to a matrix $\boldsymbol{Y} \in \mathbb{C}^{N \times B}$ where the item in the n-th row and the m-th column is set according to $y_{n,m} = x_{mB+n}$.

2. Based on the reference signal, a matrix $\boldsymbol{X} \in \mathbb{C}^{N \times B}$ is created. Each column of this matrix corresponds to a copy of the complex conjugate of the reverted reference signal $\boldsymbol{x}_{\text{ref}}$, thus the matrix elements are set according to $x_{n,m} = x^*_{\text{ref},N-1-n}$ for all columns $m \in \{0, 1, \ldots, B-1\}$.

3. Both $\boldsymbol{Y}$ and $\boldsymbol{X}$ are column-wise converted to frequency-domain by appling the DFT as described in Appendix B.2: $\boldsymbol{Y}' = \text{DFT}_{\text{cols}}\{\boldsymbol{Y}\}$ and $\boldsymbol{X}' = \text{DFT}_{\text{cols}}\{\boldsymbol{X}\}$, respectively.

4. An element-wise multiplication $\boldsymbol{W}' = \boldsymbol{Y}' \cdot \boldsymbol{X}'$ is performed.

5. Eventually, an IDFT along the columns of $\boldsymbol{W}'$ is performed: $\boldsymbol{W} = \text{DFT}^{-1}{}_{\text{cols}}\{\boldsymbol{W}'\}$.

The n-th column of $\boldsymbol{W} \in \mathbb{C}^{N\times B}$ now corresponds to the CIR of the n-th instance of the channel sounding sequence in the input data. We therefore call $\boldsymbol{W}$ the CIR matrix. The mathematical symbol $\boldsymbol{W}$ was chosen to emphasise the close relationship to the weight function $w(\tau, t)$ as discussed in Section 2.2.1.1.

Assuming that $\boldsymbol{x}_{\text{ref}}$ and $\boldsymbol{x}$ are sampled at a rate of f_{sr} and that the time axis of $\boldsymbol{W}$ got upsampled by a factor of $f_{\text{up}} \in \mathbb{N}$, the delay resolution of the CIRs is given by

$$\Delta_\tau = (f_{\text{sr}} f_{\text{up}})^{-1}. \tag{C.1}$$

We define a function $\mathcal{I} : \mathbb{C}^{BN} \mapsto \mathbb{C}^{N\times B}$, that combines the steps from above:

$$\mathcal{I}\{\boldsymbol{x}\} = \text{DFT}^{-1}{}_{\text{cols}} \left\{ \text{DFT}_{\text{cols}}\{\boldsymbol{Y}\} \cdot \text{DFT}_{\text{cols}}\{\boldsymbol{X}\} \right\}, \tag{C.2}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are defined as in step 1 and 2, respectively.

C.2 Calculation of the Averaged CIR

We further define the *averaged CIR* denoted by $\bar{\boldsymbol{h}} \in \mathbb{C}^N$, whose n-th element $\bar{h}_n$ is computed according to

$$\bar{h}_n = \frac{1}{B} \sum_{m=0}^{B-1} w_{n,m}, \tag{C.3}$$

where $w_{n,m}$ denote the elements of the CIR matrix $\boldsymbol{W}$. We understand that the averaging over a whole processing block, as it is done above, reduces the time resolution (not to be confused with the delay resolution!).

C.3 Calculation of the Power Delay Profile

The PDP *(Power Delay Profile)* $\boldsymbol{s} \in \mathbb{R}^N$ is of a similar nature to the CIR; however, it is real-valued and therefore cannot contain any phase information. Since the computation of both the coherent and the incoherent PDP involves averaging over more than one CIR, the time resolution is reduced in the same way as for the averaged CIR. The delay resolution of the PDP is always the same as the delay resolution of the CIRs it was computed from, thus it is given by (C.1).

C.3.1 Coherent Power Delay Profile

The coherent PDP is computed by taking the absolute square of the mean of the rows of $\boldsymbol{W} = (w_{n,m})$. The n-the element of the coherent PDP $\boldsymbol{s}_{\text{coh}}$ is given by

$$s_{\text{coh},n} = \left| \frac{1}{B} \sum_{m=0}^{B-1} w_{n,m} \right|^2 \overset{(C.3)}{=} \left| \bar{h}_n \right|^2. \tag{C.4}$$

We also define a function $\mathrm{PDP}_{\mathrm{coh}} : \mathbb{C}^{BN} \mapsto \mathbb{R}^{N}$ that performs all the steps introduced in this appendix to compute the coherent PDP of a vector $\boldsymbol{x}$ containing received data:

$$\boldsymbol{s}_{\mathrm{coh}} = \mathrm{PDP}_{\mathrm{coh}}\{\boldsymbol{x}\} = |\mathrm{mean}_{\mathrm{rows}}\{\mathcal{I}\{\boldsymbol{x}\}\}|^2. \tag{C.5}$$

The benefit of the coherent integration of the CIRs is, that in case the detected peaks of the individual CIRs remain in phase over time (i.e. along the rows of $\boldsymbol{W}$), the peaks remain constant while the noise floor, expected to have random phases, is averaged out and drastically reduced. Eventually, this results in an increased SNR and allows the detection of weak (but stable) MPCs.

However, in case the phase of a peak is drifting over time, the coherent processing leads to a reduction of this peak's amplitude and the peak gets washed out. Keeping the peaks of the CIRs in phase over time is challenging and requires a precise pre-processing of measurement data.

C.3.2 Incoherent Power Delay Profile

The incoherent PDP is computed similarly to the coherent PDP, however, the order of the calculation of the mean and of the absolute value is reversed. The n-the element of the incoherent PDP $\boldsymbol{s}_{\mathrm{inc}}$ is given by

$$s_{\mathrm{inc},n} = \left[\frac{1}{B}\sum_{m=0}^{B-1}|w_{n,m}|\right]^2. \tag{C.6}$$

Analogous to the coherent PDP, we define $\mathrm{PDP}_{\mathrm{inc}} : \mathbb{C}^{BN} \mapsto \mathbb{R}^{Nf_{\mathrm{up}}}$ to compute the incoherent PDP of a data vector $\boldsymbol{x}$:

$$\boldsymbol{s}_{\mathrm{inc}} = \mathrm{PDP}_{\mathrm{inc}}\{\boldsymbol{x}\} = \mathrm{mean}_{\mathrm{rows}}\{|\mathcal{I}\{\boldsymbol{x}\}|\}^2. \tag{C.7}$$

The incoherent processing usually results in a higher noise floor than the coherent processing. However, in case the phases of the peaks are drifting over time, e.g. due to insufficient pre-processing, the peaks are not washed out and remain detectable.

C.4 Calculation of the Delay-Doppler Spread Function

The results of the processing described so far provide only information on the *delays* of the processed data $\boldsymbol{x}$. To achieve information on the data's *Doppler shifts*, we introduce the DDSF *(Delay-Doppler Spread Function)*, denoted by $\mathrm{DD} : \mathbb{C}^{BN} \mapsto \mathbb{C}^{N\times B}$. It is computed based on the results from Appendix C.1 according to

$$\boldsymbol{S} = \mathrm{DD}\{\boldsymbol{x}\} = \mathrm{DSHFT}_{\mathrm{rows}}\left\{\mathrm{DFT}_{\mathrm{rows}}\left\{\mathcal{I}\{\boldsymbol{x}\}\right\}\right\} \tag{C.8}$$

and can be understood as a discrete representation of (2.9). The reader may note the DFT-shift as defined in Appendix B.2 applied to the result of the DFT, such that the zero-frequency is shifted from column 0 to column $B/2$. The columns of the resulting matrix $\boldsymbol{S} \in \mathbb{C}^{N\times B}$

correspond to the delay axes, the rows correspond to the Doppler axes. While the resolution along the delay axis is the same as for the CIRs as given in (C.1), the resolution along the Doppler axis is given by

$$\Delta_\nu := \frac{f_{\mathrm{sr}}}{NB}. \tag{C.9}$$

C.5 Addressing Spectral Leakage

In theory, the FOURIER transform, as well as its discrete counterpart, the DFT – and consequently the FFT, too – are designed to be applied to *infinite* signals, which is impossible in practice for obvious reasons. Thus, this transform is applied to finite (and in general non-periodic) signals; this is also the case for the computations presented in this appendix.

Applying the DFT on finite, non-periodical signals results in the so called spectral leakage effect: The detected frequency components of the analysed signal are periodically repeated in the spectrum. These repeated peaks are called side lobes [73]. In case a peak detection is applied to the result of the DFT, the algorithm will detect many peaks, that are in fact just processing artifacts. If the analysed signal consists of just a few strong components and many weak components, the algorithm is also likely to miss weak peaks that are actually real. Furthermore, the peaks' height is reduced, since its power is distributed over the different lobes.

The problem of spectral leakage is well-known in the field of digital signal processing and can be addressed by the application of window functions to reduce the amplitudes of the side lobes, see e.g. [73]. Instead of using a BOXCAR window having the length of the signal, which corresponds to the case of not using a window, the signal is multiplied by a more sophisticated window function before the DFT is applied. Common window functions are BLACKMAN, BLACKMAN-HARRIS, KAISER, KAISER-BESSEL, and HAMMING. If not denoted otherwise, a KAISER-BESSEL window using the BESSEL function of 0-th order is used when one of the functions listed in this appendix is applied.

Appendix D

Kernel Density Estimation

A KDE *(Kernel Density Estimation)* is used to investigate a data set's distribution – more precise: its PDF *(Probability Density Function)* [47] – , similar to what a histogram can be used for. However, in contrast to a histogram, a KDE provides not discrete, but continuous information on the underlying data's distribution. This property is enabled by applying a certain kernel function on the data.

In the general case, a KDE can also handle multidimensional data and is able to represent potential correlations within the dimensions of the data set it is applied to. In case such multidimensional data is used, it must be assured that all dimensions of the data share a similar scale. If this is not the case, the elements of the KDE's covariance matrix will spread which is likely to result in numerical instabilities. In practice, it is straightforward to avoid this problem by applying feature scaling to the data in advance of the KDE training.

The kernel that is created during the training process of the KDE can also be used to sample data having the same statistical properties than the training data [39].

D.1 Mathematical Description

D.1.1 Estimating a Kernel

Let $\boldsymbol{x}_i \in \mathbb{R}^d$ be a d-dimensional data sample from a set of N samples (i.e. the training set). The corresponding kernel density estimate at a position $\boldsymbol{x} \in \mathbb{R}^d$ is then given by

$$\hat{f_{\boldsymbol{H}}}(\boldsymbol{x}) = \sum_{i=0}^{N-1} w_i K_{\boldsymbol{H}}(\boldsymbol{x} - \boldsymbol{x}_i), \tag{D.1}$$

where $w_i \in \mathbb{R},\ \forall\, i \in \{0, 1, \ldots, N-1\}$ denote weights with

$$\sum_{i=0}^{N-1} w_i = 1. \tag{D.2}$$

If not denoted otherwise, we set $w_i = 1/N,\ \forall\, i \in \{0, 1, \ldots, N-1\}$.

$K_{\boldsymbol{H}} : \mathbb{R}^d \mapsto \mathbb{R}$ denotes the *scaled* kernel function given by

$$K_{\boldsymbol{H}}(\boldsymbol{x}) = ||\boldsymbol{H}||^{-\frac{1}{2}} K(\boldsymbol{H}^{-\frac{1}{2}}\boldsymbol{x}), \tag{D.3}$$

that is based on the kernel function $K : \mathbb{R}^d \mapsto \mathbb{R}$ scaled by $\boldsymbol{H} \in \mathbb{R}^{d\times d}$, a symmetric and positive definite matrix [47] defining the KDE bandwidth. In practice, multiple kernel functions are commonly used (see [106] for some examples); the most common one is the GAUSSian Kernel [39], given by

$$K(\boldsymbol{x}) = (2\pi)^{-\frac{d}{2}} \exp\left\{-\frac{||\boldsymbol{x}||^2}{2}\right\}. \tag{D.4}$$

In this work, only the GAUSSian kernel as defined above is used.

D.1.2 Generating Data Using a KDE

Let $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ denote the covariance matrix [29] of the d-dimensional data used to train the GAUSSian kernel (D.4) of a KDE. A CHOLESKY decomposition is used to find a lower-triangular matrix $\boldsymbol{L} \in \mathbb{R}^{d\times d}$, such that

$$\Sigma \cdot \boldsymbol{H} \cdot \boldsymbol{H} = \boldsymbol{L}\boldsymbol{L}^T \tag{D.5}$$

is fulfilled.

Let $\boldsymbol{n} \in \mathbb{R}^d$ be taken from a d-dimensional (multivariate) normal distribution $\mathcal{N}(\boldsymbol{0}_d, \boldsymbol{I}_d)$, where $\boldsymbol{0}_d$ denotes a d-dimensional all-zero vector and $\boldsymbol{I}_d$ denotes a $d \times d$-dimensional identity-matrix (also known as *eye* matrix).

We create a new sample $\tilde{\boldsymbol{x}}$ by an element-wise addition of a "scaled" version of $\boldsymbol{n}$ with a randomly chosen sample from the training set:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x}_i + \boldsymbol{L}\boldsymbol{n}, \text{ with } i \sim \left\lfloor \mathcal{U}_{[0,N)} \right\rfloor. \tag{D.6}$$

We now analyse the covariance of $\tilde{\boldsymbol{x}}$ to verify it actually corresponds to the desired covariance (D.5). First, we realise that $\tilde{\boldsymbol{x}}$ is a vector of d random variables. According to the definition in [13], its covariance is hence given by

$$\text{cov}\{\tilde{\boldsymbol{x}}\} = \mathbb{E}\{(\tilde{\boldsymbol{x}} - \mathbb{E}\{\tilde{\boldsymbol{x}}\})\,(\tilde{\boldsymbol{x}} - \mathbb{E}\{\tilde{\boldsymbol{x}}\})^T\}. \tag{D.7}$$

Inserting (D.6) leads to

$$\begin{aligned}
\text{cov}\{\tilde{\boldsymbol{x}}\} &= \mathbb{E}\{(\boldsymbol{x}_i + \boldsymbol{L}\boldsymbol{n} - \underbrace{\mathbb{E}\{\boldsymbol{x}_i + \boldsymbol{L}\boldsymbol{n}\}}_{=\boldsymbol{x}_i})(\boldsymbol{x}_i + \boldsymbol{L}\boldsymbol{n} - \underbrace{\mathbb{E}\{\boldsymbol{x}_i + \boldsymbol{L}\boldsymbol{n}\}}_{=\boldsymbol{x}_i})^T\} \\
&= \mathbb{E}\{(\boldsymbol{L}\boldsymbol{n})(\boldsymbol{L}\boldsymbol{n})^T\} = \boldsymbol{L}\underbrace{\mathbb{E}\{\boldsymbol{n}\boldsymbol{n}^T\}}_{=\text{cov}\{\boldsymbol{n}\}}\boldsymbol{L}^T \\
&= \boldsymbol{L}\boldsymbol{I}_d\boldsymbol{L}^T = \boldsymbol{L}\boldsymbol{L}^T \overset{\text{(D.5)}}{=} \Sigma \cdot \boldsymbol{H} \cdot \boldsymbol{H}.
\end{aligned} \tag{D.8}$$

D.2 Software Implementation

If not denoted otherwise, the software implementation used to compute the KDE throughout this work is the PYTHON implementation described in [39].

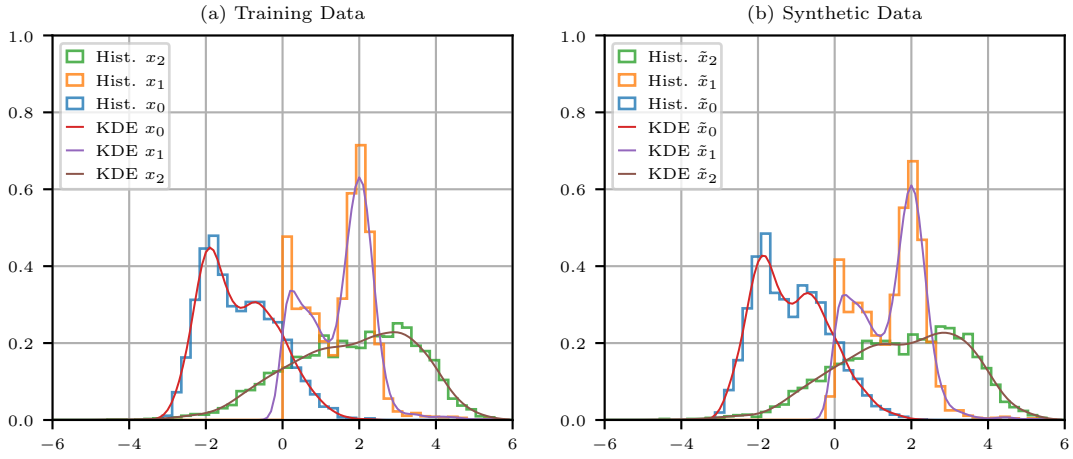


FIGURE D.1: Estimated distributions of the random variables of the (a) training data and the (b) synthetic data. In each case, both a step-histogram and a KDE are used to represent the respective distribution.

D.3 Practical Example

To demonstrate the properties and possible applications of the KDE, an example with $d = 3$ is provided, where each dimension represents a random variable that may or may not be independent of the others. The training data with $N = 3{,}000$ are sampled from multiple superimposed distributions, including the GAUSSian, the WEIBULL, and the RAYLEIGH distribution.

The distributions of the three random variables x_0, x_1, and x_2 are given in Figure D.1a; both the normalised step-histogram and an estimation of the PDF using a KDE are given. The bandwidths of the KDE have been estimated using the SILVERMAN rule as given in [47]. It can be seen that none of the underlying distributions of the three random variables can be represented by a simple distribution like a single GAUSSian or a single WEIBULL distribution.

In the left part of Table D.1, the PEARSON correlation coefficient as introduced in (7.17) is given for the training data. A strong negative correlation between x_0 and x_2 can be observed, whereas the (absolute) correlation between the other permutations is below 0.5.

Using the procedure explained in Appendix D.1.2, a set of synthetic data has been generated and analysed using the same tools as above. We denote the vector containing the three random variables representing this generated data by $\tilde{\boldsymbol{x}} = [\tilde{x}_0, \tilde{x}_1, \tilde{x}_2]$.

The distributions of the generated random variables are given in Figure D.1b. Both the step-histogram and the KDE-based probability density estimation show only small deviations from the corresponding data from the training set presented in Figure D.1a. Also, the correlation between the random variables is very similar to the corresponding values of the training data as the right part of Table D.1 shows.

TABLE D.1: The PEARSON correlation coefficient as given in (7.17) between all permutations of the random variables of the training data $\boldsymbol{x}$ (left) and the generated (i.e. synthetic) data $\tilde{\boldsymbol{x}}$ (right).

	x_0	x_1	x_2
x_0	1.00	−0.48	−0.93
x_1	−0.48	1.00	0.45
x_2	−0.93	0.45	1.00

	$\tilde{x}_0$	$\tilde{x}_1$	$\tilde{x}_2$
$\tilde{x}_0$	1.00	−0.49	−0.92
$\tilde{x}_1$	−0.49	1.00	0.45
$\tilde{x}_2$	−0.92	0.45	1.00

Appendix E

RICE Distribution

E.1 Probability Density Function

The PDF *(Probability Density Function)* of the RICE distribution is given by

$$f(x) = \frac{x}{\sigma^2} \exp\left\{-\frac{x^2+\varrho^2}{2\sigma^2}\right\} \mathrm{I}_0\left(\frac{x\varrho}{\sigma^2}\right), \text{ with } \varrho \geq 0 \text{ and } \sigma \geq 0, \tag{E.1}$$

where $\mathrm{I}_0(.)$ is the modified BESSEL function of 0-th order[1]. Using the definition given in [13], this function is given by:

$$\mathrm{I}_0(x) = \sum_{k=0}^{\infty} \frac{\left(\frac{x^2}{4}\right)^k}{(k!)^2}. \tag{E.2}$$

The K-factor – that represents the ratio of the power of the LOS component and the power of all scattering components [80] in a fading process – is given in [68] by

$$K = \frac{\varrho^2}{2\sigma^2}. \tag{E.3}$$

An alternative definition of the RICE distribution that is used in [101] is given by

$$f'(x) = x \exp\left\{-\frac{x^2+b^2}{2}\right\} \mathrm{I}_0(xb), \text{ with } b \geq 0. \tag{E.4}$$

This definition only holds in case the scaling is set to $\sigma = 1$. To allow generic scaling, the standard scale family equation is applied:

$$\begin{aligned} f(x) = \frac{1}{\sigma} f'\left(\frac{x}{\sigma}\right) &= \frac{1}{\sigma} \left[\frac{x}{\sigma} \exp\left\{-\frac{\left(\frac{x}{\sigma}\right)^2+b^2}{2}\right\} \mathrm{I}_0\left(\frac{x}{\sigma}b\right)\right] \\ &= \frac{x}{\sigma^2} \exp\left\{-\frac{x^2+(b\sigma)^2}{2\sigma^2}\right\} \mathrm{I}_0\left(\frac{xb}{\sigma}\right). \end{aligned} \tag{E.5}$$

Comparing this equation's coefficients with the coefficients of (E.1) shows the relationship $b = \frac{\varrho}{\sigma}$.

[1] The shape parameter ϱ is often denoted by ν in literature. An alternative symbol is used here to avoid confusion with the Doppler shift.

E.2 Moment-based Parameter Estimation

According to [89], the first two even moments of the RICE distribution are given by

$$\mu_2' = 2\sigma^2 + \varrho^2 \quad \text{and} \tag{E.6}$$

$$\mu_4' = 8\sigma^4 + 8\sigma^2\varrho^2 + \varrho^4, \tag{E.7}$$

respectively. Solving (E.6) for ϱ^2 and inserting in (E.7) leads to:

$$\begin{aligned}\mu_4' &= 8\sigma^4 + 8\sigma^2(\overbrace{\mu_2' - 2\sigma^2}^{\varrho^2}) + (\overbrace{\mu_2' - 2\sigma^2}^{\varrho^2})^2 \\ &= -4\sigma^4 + 4\mu_2'\sigma^2 + {\mu_2'}^2.\end{aligned} \tag{E.8}$$

Using the substitution $A = \sigma^2$, this can be transformed to a reduced quadratic function with respect to A:

$$A^2 + pA + q = 0, \text{ with } p := -\mu_2' \text{ and } q := \frac{\mu_4' - {\mu_2'}^2}{4}. \tag{E.9}$$

The application of standard solving methods leads to the two solutions

$$\begin{aligned}\sigma_{1,2}^2 = A_{1,2} &= \frac{\mu_2'}{2} \pm \sqrt{\frac{2{\mu_2'}^2 - \mu_4'}{4}}, \\ &= \frac{\mu_2' \pm \sqrt{R}}{2}, \text{ with } R := 2{\mu_2'}^2 - \mu_4'.\end{aligned} \tag{E.10}$$

The remaining ambiguity about the solution will be resolved later. Since only a solution where $\sigma^2 \in \mathbb{R}$ is applicable here, the following condition applies for the radicand R:

$$R \geq 0. \tag{E.11}$$

Inserting (E.10) into (E.6) allows a trivial solution for ϱ:

$$\begin{aligned}\varrho^2 = \mu_2' - 2\sigma^2 = \mu_2' - 2\left(\frac{\mu_2' \pm \sqrt{R}}{2}\right) = \mp\sqrt{R} \\ \Leftrightarrow \varrho = \sqrt{\mp\sqrt{R}} \overset{(E.11)}{=} \mp\sqrt[4]{R} \overset{(E.1)}{=} \sqrt[4]{R}\end{aligned} \tag{E.12}$$

Now that ϱ is known, the ambiguity on σ^2 can be resolved by inserting into (E.6).

E.3 Relation to Other Distributions

E.3.1 RAYLEIGH Distribution

The well-known RAYLEIGH distribution is a special case of the RICE distribution where $\varrho = 0$ (and therefore $K = 0$). If $\mu(t)$ is a RAYLEIGH process, then, according to [68], a RICE process

$\mu_\varrho(t)$ is given by

$$\mu_\varrho(t) = \mu(t) + m(t), \tag{E.13}$$

where $m(t)$ is a process modelling the LOS component's amplitude.

In wireless radio communication, it is common to model the received amplitude by either a RICE or a RAYLEIGH distribution, depending on whether a LOS component is available or not [31].

E.3.2 GAUSSian Distribution

The absolute value of a complex process $\mu(t)$ follows a RAYLEIGH distribution with the variance $2\sigma^2$ if its real part and its imaginary part both follow a GAUSSian distribution with a variance of σ^2, see [68]:

$$\mu(t) = \mu_1(t) + \mathrm{j}\mu_2(t), \text{ with } \mu_{\{1,2\}}(t) \sim \mathcal{N}(0, \sigma^2). \tag{E.14}$$

List of Abbreviations

ACF	Autocorrelation Function
ADC	Analog-to-Digital Converter
AeroMACS	Aeronautical Mobile Airport Communication System
APNT	Alternative Positioning, Navigation, and Timing
APU	Auxiliary Power Unit
AG	Air-Ground
AS	Air Station
ATC	Air Traffic Control
ATM	Air Traffic Management
AWG	Arbitrary Waveform Generator
AWGN	Additive White Gaussian Noise
B-AMC	Broadband Aeronautical Mobile Communication System
BOBYQA	Bound Optimization by Quadratic Approximation
BP	Band-Pass
B-VHF	Broadband VHF
BVLOS	Beyond Visual Line of Sight
C2	Command and Control
CE2R	Curved-Earth Two-Ray
CIR	Channel Impulse Response
CNS	Communication, Navigation, Surveillance
CONOPS	Concept of Operation
CPDLC	Controller-Pilot Data Link Communications
CSS	Channel Sounding Sequence
CSV	Comma Separated Values
DAB	Digital Audio Broadcasting
DAC	Digital-to-Analog Converter
DC	Direct Current
DDSF	Delay-Doppler Spread Function
DFT	Discrete Fourier Transform
DLR	German Aerospace Center
DMT	Discrete Multitone Transmission
DS	Delay Spread
DVB-T	Terrestrial Digital Video Broadcasting

ECEF	Earth Centered, Earth-Fixed Cartesian Coordinate System
ENR	En-route
ENU	East-North-Up
FCI	Future Communications Infrastructure
FEC	Forward Error Correction
FANS	Future Air Navigation System
FFT	Fast Fourier Transform
FSPL	Free Space Path Loss
GAN	Generative Adversarial Network
GCP	Gain Compression Point
GLONASS	Global'naya Navigatsionnaya Sputnikovaya Sistema
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GS	Ground Station
GT	Ground Truth
GTD	Geometrical Theory of Diffraction
HF	High Frequency
HPA	High Power Amplifier
ICAO	International Civil Aviation Organization
ICF	Iterative Clipping and Filtering
IDFT	Inverse Discrete Fourier Transform
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
IKN	Institut für Kommunikation und Navigation
IMU	Inertial Measurement Unit
IR	Impulse Response
ISI	Inter-Symbol Interference
ITU	International Telecommunication Union
KDE	Kernel Density Estimation
LB	Lower Bound
LDACS	L-Band Digital Aeronautical Communication System
LNA	Low Noise Amplifier
LOS	Line-of-Sight
LP	Low-Pass
LS	Least Squares
LTE	Long Term Evolution
LTI	Linear Time Invariant
MEDS	Method of Exact Doppler Spread
ML	Maximum Likelihood

MMSE	Minimum Mean Squared Error
MPC	Multipath Component
MSE	Mean Squared Error
NATO	North Atlantic Treaty Organization
NED	North-East-Down
NextGen	Next Generation Air Transportation System
OFDM	Orthogonal Frequency-Division Multiplex
PAPR	Peak-to-Average Power Ratio
PDP	Power Delay Profile
PDF	Probability Density Function
PKI	Public Key Infrastructure
PPP	Precise Point Positioning
PSK	Phase Shift Keying
QAM	Quadtrature Amplitude Modulation
RAID	Redundant Array of Independent Disks
Rb	Rubidium
RF	Radio Frequency
RMS	Root Mean Square
RNN	Recurrent Neural Network
RINEX	Receiver Independent Exchange Format
RPA	Remotely Piloted Aircraft
RPAS	Remotely Piloted Aircraft System
RxCCCS	Receiver Centred Cartesian Coordinate System
RxCPCS	Receiver Centred Polar Coordinate System
SESAR	Single European Sky ATM Research
SMS	Short Message Service
SNR	Signal-to-Noise Ratio
SoS	Sum of Sinusoids
SQL	Structured Query Language
TDL	Tapped Delay Line
TxCCCS	Transmitter Centered Cartesian Coordinate System
TxCPCS	Transmitter Centered Polar Coordinate System
UA	Unmanned Aircraft
UB	Upper Bound
UPS	Uninterruptible Power Supply
VDL	VHF Data Link
VHF	Very High Frequency
WGS84	World Geodetic System 1984
WSSUS	Wide-Sense Stationary Uncorrelated Scattering

w. r. t.	with respect to
ZF	Zero-Forcing

List of Symbols

α	amplitude	unit-less, often: dB
W	bandwidth	Hz
η_W	bandwidth efficiency	$\mathrm{bit\,s^{-1}\,Hz^{-1}}$ or $\mathrm{Bd\,Hz^{-1}}$
d	distance	unit-less or m
τ	delay	s
ν	Doppler shift	Hz
f	frequency	Hz
G	gain	unit-less, often: dB
N	integer, e.g. used to describe length of a discrete signal	unit-less
$\mathbb{C}$	set of complex numbers	-
$\mathbb{N}$	set of natural numbers	-
$\mathbb{R}$	set of real numbers	-
P	power	W ($\mathrm{J\,s^{-1}}$)
ϵ	residual	usually unit-less
v	velocity	$\mathrm{m\,s^{-1}}$
f_{sr}	sample rate	Hz
Δf	sub-carrier spacing (e.g. in a multi-carrier system)	Hz
T	time period (e.g. a symbol duration)	s

Bibliography

[1] F. A. Administration, *Airplane flying handbook FAA-H-8083-3B*, United States Department of Transportation, P.O. Box 25082, Oklahoma City, OK 73125, 2016.

[2] *Aeronautical telecommunications annex 10, vol. 3*, ICAO, Jul. 2007.

[3] P. Almers, E. Bonek, A. Burr, N. Czink, M. Debbah, V. Degli-Esposti, H. Hofstetter, P. Kyösti, D. Laurenson, G. Matz, *et al.*, "Survey of channel and radio propagation models for wireless MIMO systems", *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, pp. 1–19, 2007.

[4] M. Armitage, *Unmanned Aircraft* (Air Power Series). Brassey's Defence Publishers, 1988, ISBN: 9780080347448.

[5] G. Battista, R. Kumar, E. Nossek, and O. Osechas, "Placing LDACS-based ranging sources for robust RNP 1.0 accuracy en-route", in *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, 2017, pp. 1–9. DOI: 10.1109/DASC.2017.8102135.

[6] D. Becker, U.-C. Fiebig, and L. Schalk, "Wideband channel measurements and first findings for low altitude drone-to-drone links in an urban scenario", in *2020 14th European Conference on Antennas and Propagation (EuCAP)*, 2020, pp. 1–5. DOI: 10.23919/EuCAP48036.2020.9135494.

[7] P. Bello, "Aeronautical channel characterization", *IEEE Transactions on Communications*, vol. 21, no. 5, pp. 548–563, 1973. DOI: 10.1109/TCOM.1973.1091707.

[8] P. Bello, "Characterization of randomly time-variant linear channels", *IEEE Transactions on Communications Systems*, vol. 11, no. 4, pp. 360–393, 1963. DOI: 10.1109/TCOM.1963.1088793.

[9] E. A. Bender and S. G. Williamson, *Lists, Decisions and Graphs*. S. Gill Williamson, 2010. [Online]. Available: https://books.google.de/books?id=vaXv_yhefG8C.

[10] T. Boegl, M. Rautenberg, B. Haindl, C. Rihacek, J. Meser, P. Fantappie, N. Pringvanich, J. Micallef, K. Hauf, J. MacBride, P. Sacre, B. van den Einden, T. Gräupl, and M. Schnell, *LDACS white paper – a roll-out scenario*, ICAO-DCIWG, Oct. 2019. [Online]. Available: https://www.ldacs.com/wp-content/uploads/2013/12/ACP-DCIWG-IP01-LDACS-White-Paper.pdf.

[11] S. Boyd, "Multitone signals with low crest factor", *IEEE Transactions on Circuits and Systems*, vol. 33, no. 10, pp. 1018–1022, Oct. 1986. DOI: 10.1109/TCS.1986.1085837.

[12] C. Bresteau, S. Guigui, P. Berthier, and J. M. Fernandez, "On the security of aeronautical datalink communications: Problems and solutions", in *2018 Integrated Communications, Navigation, Surveillance Conference (ICNS)*, 2018, 1A4-1-1A4–13. DOI: 10.1109/ICNSURV.2018.8384830.

[13] I. N. Bronstein, K. A. Semendjajew, G. Musiol, and H. Mühlig, *Taschenbuch der Mathematik*, 7th ed. Frankfurt am Main: Verlag Harri Deutsch, 2008, ISBN: 978-3-8171-2017-8.

[14] J. Cooley and J. Tukey, "An algorithm for the machine calculation of complex fourier series", *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.

[15] G. Del Galdo, "Geometry-based channel modeling for multi-user MIMO systems and applications", Ph.D. dissertation, TU Ilmenau, Ilmenau, Jan. 2007.

[16] L. Dewangan, M. Singh, and N. Dewangan, "A survey of PAPR reduction techniques in LTE-OFDM system", in *International Journal of Recent Technology and Engineering (IJRTE)*, 5, vol. 1, IJRTE, 2012.

[17] DLR, "Report D5: Expected B-AMC system performance", EUROCONTROL, Tech. Rep., 2007. [Online]. Available: https://www.eurocontrol.int/sites/default/files/2019-05/24092007-b-amc-project-deliverable-d5-v11.pdf.

[18] U. Epple and M. Schnell, "Channel estimation in OFDM systems with strong interference", Sep. 2010.

[19] FAA Safety Team, *Fly the aircraft first*, Federal Aviation Administration, Feb. 2018. [Online]. Available: https://www.faa.gov/news/safety_briefing/2018/media/SE_Topic_18-07.pdf.

[20] "FAA unmanned aircraft systems integration pilot program", Federal Aviation Administration, Tech. Rep., Jul. 2021. [Online]. Available: https://www.faa.gov/sites/faa.gov/files/uas/programs_partnerships/completed/integration_pilot_program/IPP_Final_Report_20210712.pdf.

[21] A. Filip and D. Shutin, "Ambiguity function analysis for OFDM-based LDACS passive multistatic radar", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 3, pp. 1323–1340, 2018. DOI: 10.1109/TAES.2017.2780678.

[22] R. F. H. Fischer and J. B. Huber, *Digital Communications: A Foundational Approach*. Cambridge University Press, 2024, ISBN: 9781009429665. DOI: https://doi.org/10.1017/9781009429658.

[23] Fortune Business Insights, *Medical Drone Market Size, Share & COVID-19 Impact Analysis*, Aug. 2021. [Online]. Available: https://www.fortunebusinessinsights.com/medical-drone-market-105805.

[24] Fortune Business Insights, *Unmanned Aerial Vehicle (UAV) Market Size, Share & COVID-19 Impact Analysis*, Jul. 2020. [Online]. Available: https://www.fortunebusinessinsights.com/industry-reports/unmanned-aerial-vehicle-uav-market-101603.

[25] GlobalData, *Drones in power – thematic intelligence*, Oct. 2022. [Online]. Available: https://www.globaldata.com/store/report/drones-in-power-theme-analysis/?scalar=true&pid=198453&sid=21.

[26] A. Goldsmith, *Wireless Communications*. Stanford University, 2004.

[27] F. van Graas, S. Craig, W. Pelgrum, and S. Ugazio, "Laboratory and flight test analysis of rubidium frequency reference performance", *NAVIGATION*, vol. 60, no. 2, pp. 123–131, 2013. DOI: https://doi.org/10.1002/navi.34. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/navi.34.

[28] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, and H. Nakayama, "GAN-based synthetic brain MR image generation", in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 2018, pp. 734–738. DOI: 10.1109/ISBI.2018.8363678.

[29] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy", *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. DOI: 10.1038/s41586-020-2649-2. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2.

[30] P. Hoeher and E. Haas, "Aeronautical channel modeling at VHF-band", in *Gateway to 21st Century Communications Village. VTC 1999-Fall. IEEE VTS 50th Vehicular Technology Conference (Cat. No.99CH36324)*, vol. 4, 1999, 1961–1966 vol.4. DOI: 10.1109/VETECF.1999.797280.

[31] P. Höher, *Grundlagen der digitalen Informationsübertragung: Von der Theorie zu Mobilfunkanwendungen*. Springer Fachmedien Wiesbaden, 2013, ISBN: 9783834822147. [Online]. Available: https://books.google.de/books?id=rzggBAAAQBAJ.

[32] J. Hoydis, S. Cammerer, F. Ait Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An open-source library for next-generation physical layer research", *arXiv preprint*, Mar. 2022.

[33] Z. Huang, J. Rodríguez-Piñeiro, T. Domínguez-Bolaño, X. Cai, and X. Yin, "Empirical dynamic modeling for low-altitude UAV propagation channels", *IEEE Transactions on Wireless Communications*, vol. 20, no. 8, pp. 5171–5185, 2021. DOI: 10.1109/TWC.2021.3065959.

[34] P. Hullah, P. Bosman, and M. Liossone, "Unmanned aircraft systems (UAS) ATM integration operational concept", Eurocontrol, Tech. Rep., Nov. 2018. [Online]. Available: https://www.eurocontrol.int/sites/default/files/publication/files/uas-atm-integration-operational-concept-v1.0-release%2020181128.pdf.

[35] I-CNSS, "PJ14-W2-60 TRL6 CBAT – LDACS A/G", SESAR, Tech. Rep., Jan. 2023.

[36] ICAO RPAS Panel, *Remotely piloted aircraft systems concept of operations for international IFR operations*, ICAO, Aug. 2016. [Online]. Available: https://www.icao.int/safety/UA/Documents/ICAO%20RPAS%20CONOPS.pdf.

[37] "IEEE standard letter designations for radar-frequency bands", *IEEE Std 521-2002 (Revision of IEEE Std 521-1984)*, pp. 1–10, 2003. DOI: 10.1109/IEEESTD.2003.94224.

[38] S. G. Johnson, *The NLopt nonlinear-optimization package*, https://github.com/stevengj/nlopt, 2007.

[39] L. Z. Kelley, "Kalepy: A python package for kernel density estimation, sampling and plotting", *Journal of Open Source Software*, vol. 6, no. 57, p. 2784, 2021. DOI: 10.21105/joss.02784. [Online]. Available: https://doi.org/10.21105/joss.02784.

[40] R. Kerczewski, J. Wilson, and W. Bishop, "Frequency spectrum for integration of unmanned aircraft", Oct. 2013, pp. 6D5–1, ISBN: 978-1-4799-1536-1. DOI: 10.1109/DASC.2013.6712625.

[41] H. Kesteloo, *Formula one FPV drone captures max verstappen's win in spain*, May 2022. [Online]. Available: https://dronexl.co/2022/05/22/formula-one-fpv-drone/.

[42] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications", *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009. DOI: 10.1137/07070111X. [Online]. Available: https://doi.org/10.1137/07070111X.

[43] J. Kundrata, D. Fujimoto, Y. Hayashi, and A. Barić, "Comparison of pearson correlation coefficient and distance correlation in correlation power analysis on digital multiplier", in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 146–151. DOI: 10.23919/MIPRO48935.2020.9245325.

[44] D. Kunertova, "The war in ukraine shows the game-changing effect of drones depends on the game", *Bulletin of the Atomic Scientists*, vol. 79, no. 2, pp. 95–102, 2023. DOI: 10.1080/00963402.2023.2178180.

[45] P. Laly, D. P. Gaillot, M. Lienard, P. Degauque, E. Tanghe, W. Joseph, and L. Martens, "Flexible real-time MIMO channel sounder for multidimensional polarimetric parameter estimation", in *2015 IEEE Conference on Antenna Measurements & Applications (CAMA)*, 2015, pp. 1–3. DOI: 10.1109/CAMA.2015.7428142.

[46] D. Larm, "Expendable remotely piloted vehicles for strategic offensive airpower roles", M.S. thesis, School of Advanced Airpower Studies, Maxwell Air Force Base, Jun. 1996.

[47] T. Ledl, "Kernel density estimation: Theory and application indiscriminant analysis", *Austrian Journal of Statistics*, vol. 33, no. 3, pp. 267–279, 2004.

[48] C. Lesjak, D. Hein, and J. Winter, "Hardware-security technologies for industrial IoT: Trustzone and security controller", in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, 2015, pp. 002 589–002 595. DOI: 10.1109/IECON.2015.7392493.

[49] *Manual on remotely piloted aircraft systems*, ICAO, 2015. [Online]. Available: https://skybrary.aero/sites/default/files/bookshelf/4053.pdf.

[50] C. Marischka, "Der "erste echte Drohnenkrieg", Europas Anteil und deutsche Aufrüstung", *Telepolis*, Apr. 2021. [Online]. Available: https://www.telepolis.de/features/Der-erste-echte-Drohnenkrieg-Europas-Anteil-und-deutsche-Aufruestung-6014986.html?seite=all.

[51] D. W. Matolak and R. Sun, "Air-ground channel characterization for unmanned aircraft systems - part i: Methods, measurements, and models for over-water settings", *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 26–44, Jan. 2017, ISSN: 0018-9545. DOI: 10.1109/TVT.2016.2530306.

[52] D. W. Matolak and R. Sun, "Air-ground channel measurements & modeling for UAS", in *2013 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, 2013, pp. 1–9. DOI: 10.1109/ICNSurv.2013.6548539.

[53] D. W. Matolak and R. Sun, "Air–ground channel characterization for unmanned aircraft systems—part iii: The suburban and near-urban environments", *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 6607–6618, 2017. DOI: 10.1109/TVT.2017.2659651.

[54] N. Mäurer, T. Gräupl, C. Schmitt, G. D. Rodosek, and H. Reiser, "Advancing the security of LDACS", *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 5237–5251, 2022. DOI: 10.1109/TNSM.2022.3189736.

[55] F. Maussion, TimoRoth, J. Landmann, M. Dusch, R. Bell, and tbridel, *Fmaussion/salem: V0.3.4*, version v0.3.4, Mar. 2021. DOI: 10.5281/zenodo.4635291. [Online]. Available: https://doi.org/10.5281/zenodo.4635291.

[56] K. Mehlhorn and P. Sanders, *Algorithms and Data Structures: The Basic Toolbox*. Springer, Berlin, 2008, ISBN: 3540779779.

[57] D. M. Mielke, "C-band digital aeronautical communication for unmanned aircraft systems", in *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, Sep. 2017, pp. 1–7. DOI: 10.1109/DASC.2017.8102111.

[58] D. M. Mielke and N. Schneckenburger, "Towards a data link for unmanned aviation: DLR flight measurement campaign for c2 data link development", in *2019 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, Apr. 2019, pp. 1–8. DOI: 10.1109/ICNSURV.2019.8735378.

[59] D. Mielke, N. Mäurer, T. Gräupl, and M. Bellido-Manganell, "Getting civil aviation ready for the post quantum age with LDACS", *Digitale Welt*, vol. 5, pp. 28–33, Oct. 2021. DOI: 10.1007/s42354-021-0401-1.

[60] D. M. Mielke, D. Becker, and M. Walter, "Path-based statistical modeling of multipath components in propagation channels for wireless communications in unmanned aviation", in *2023 Integrated Communication, Navigation and Surveillance Conference (ICNS)*, 2023, pp. 1–8. DOI: 10.1109/ICNS58246.2023.10124256.

[61] D. M. Mielke, N. Schneckenburger, U.-C. Fiebig, M. Walter, and M. A. Bellido-Manganell, "Analysis of the dominant signal component of the air-ground channel based on measurement data at c-band", *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 2955–2968, 2021. DOI: 10.1109/TVT.2021.3065143.

[62] D. M. Mielke, M. Walter, D. Becker, M. A. Bellido-Manganell, and U.-C. Fiebig, "Application of path-based multipath component tracking on air-ground channel measurement data", *IEEE Transactions on Vehicular Technology*, vol. 72, no. 5, pp. 5546–5559, 2023. DOI: 10.1109/TVT.2022.3230357.

[63] J. Milanovic, S. Rimac-Drlje, and I. Majerski, "Radio wave propagation mechanisms and empirical models for fixed wireless access systems", *Technical Gazette*, vol. 17, no. 1, pp. 43–52, 2010.

[64] B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster", *Computer Communications*, vol. 156, pp. 1–10, 2020, ISSN: 0140-3664. DOI: https://doi.org/10.1016/j.comcom.2020.03.012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366419318602.

[65] J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization", *The Computer Journal*, vol. 7, no. 4, pp. 308–313, Jan. 1965, ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308.

[66] J. Painter, S. Gupta, and L. Wilson, "Multipath modeling for aeronautical communications", *IEEE Transactions on Communications*, vol. 21, no. 5, pp. 658–662, 1973. DOI: 10.1109/TCOM.1973.1091704.

[67] J. Parsons, *The Mobile Radio Propagation Channel*. Wiley, 1992, ISBN: 9780470218242. [Online]. Available: https://books.google.de/books?id=xxFTAAAAMAAJ.

[68] M. Pätzold, *Mobile Fading Channels*. Wiley, 2002, ISBN: 9780471495499. [Online]. Available: https://books.google.de/books?id=ffIyrlCyf5cC.

[69] H. Pei, K. Ren, Y. Yang, C. Liu, T. Qin, and D. Li, "Towards generating real-world time series data", *CoRR*, vol. abs/2111.08386, 2021. arXiv: 2111.08386. [Online]. Available: https://arxiv.org/abs/2111.08386.

[70] R. J. Polge and E. M. Mitchell, "Impulse response determination by cross correlation", *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, no. 1, pp. 91–97, 1970. DOI: 10.1109/TAES.1970.310015.

[71] B. Pong, "The art of drone warfare", *Journal of War & Culture Studies*, vol. 15, no. 4, pp. 377–387, 2022. DOI: 10.1080/17526272.2022.2121257.

[72] M. J. D. Powell, "The BOBYQA algorithm for bound constrainedoptimization without derivatives", Department of Applied Mathematics and Theoretical Physics, Cambridge University, Tech. Rep. 2009/NA06, 2009.

[73] K. Prabhu, *Window Functions and Their Applications in Signal Processing*. Oct. 2013, ISBN: ISBN 9781466515833. DOI: 10.1201/9781315216386.

[74] *R&S SMBV100A Vector Signal Generator Operating Manual*, Rohde & Schwarz, Munich, 2018.

[75] S. A. Ramakrishna, "Physics of negative refractive index materials", *Reports on Progress in Physics*, vol. 68, no. 2, p. 449, Jan. 2005. DOI: 10.1088/0034-4885/68/2/R06. [Online]. Available: https://dx.doi.org/10.1088/0034-4885/68/2/R06.

[76] N. S. Randhawa, S. Sharma, and R. K. Dubey, "A survey of equalization techniques for an effective equalizer design in MIMO-OFDM systems", in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, 2015, pp. 1–5. DOI: 10.1109/ICCPCT.2015.7159297.

[77] *Recommendation ITU-R p.527-3*, ITU, Nov. 1992. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.527-3-199203-S!!PDF-E.pdf.

[78] *Report itu-r m.2171*, ITU, Dec. 2009. [Online]. Available: https://www.itu.int/en/ITU-R/space/snl/Documents/R-REP-M.2171-2009-PDF-E.pdf.

[79] S. O. Rice, "Mathematical analysis of random noise", *The Bell System Technical Journal*, vol. 23, no. 3, pp. 282–332, 1944. DOI: 10.1002/j.1538-7305.1944.tb00874.x.

[80] S. O. Rice, "Statistical properties of a sine wave plus random noise", *The Bell System Technical Journal*, vol. 27, no. 1, pp. 109–157, 1948. DOI: 10.1002/j.1538-7305.1948.tb01334.x.

[81] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM", *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1613–1621, 1997. DOI: 10.1109/26.650240.

[82] N. Schneckenburger, "A wide-band air-ground channel model", Ph.D. dissertation, University of Ilmenau, 2017.

[83] M. Schnell, *OFDM-basierte Übertragungstechniken*, Lecture Notes University of Karlsruhe, 2013.

[84] M. Schnell, U. Epple, M. Mostafa, T. Gräupl, and T. Bögl, "The future terrestrial data link LDACS – current status and way ahead", in *ENRI International Workshop on ATM/CNS (EIWAC 2015)*, Nov. 2015. [Online]. Available: https://elib.dlr.de/102367/.

[85] W. Schulz, G. Kurz, E. Hering, R. Martin, and M. Stohrer, *Physik für Ingenieure* (Springer-Lehrbuch). Springer Berlin Heidelberg, 2007, ISBN: 9783540718567.

[86] *Sesar2020 –pj14-02-01 -LDACS A/G specification*, SESAR, Aug. 2019. [Online]. Available: https://www.ldacs.com/wp-content/uploads/2013/12/SESAR2020_PJ14_D3_3_030_LDACS_AG_Specification_00_02_02-1_0.pdf.

[87] C. Shahriar, M. La Pan, M. Lichtman, T. C. Clancy, R. McGwier, R. Tandon, S. Sodagari, and J. H. Reed, "PHY-layer resiliency in OFDM communications: A tutorial", *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 292–314, 2015. DOI: 10.1109/COMST.2014.2349883.

[88] D. Shutin and G. Kubin, "Tracking and prediction of multipath components in wireless MIMO channels", in *Proceedings of the 65th IEEE Vehicular Technology Conference VTC2007*, 2007, pp. 229–333. [Online]. Available: papers/vtc2007.pdf.

[89] J. Sijbers, A. den Dekker, P. Scheunders, and D. Van Dyck, "Maximum-likelihood estimation of Rician distribution parameters", *IEEE Transactions on Medical Imaging*, vol. 17, no. 3, pp. 357–361, 1998. DOI: 10.1109/42.712125.

[90] J. O. Smith, S. Levine, and H. Thornburg, *Multirate, Polyphase, and Wavelet Filter Banks*, Stanford University, Stanford, California 94305, Jun. 2020.

[91] J. Sparks, *Aircraft communications*, May 2007. [Online]. Available: https://www.aviationpros.com/aircraft/article/10381753/aircraft-communications.

[92] K. Stepanenko, R. Bailey, G. Mappes, and F. W. Kagan, *Russian offensive campaign assessment*, Apr. 2023. [Online]. Available: https://www.understandingwar.org/sites/default/files/Russian%20Offensive%20Campaign%20Assessment%20April%2021,%202023.pdf.

[93] R. Sun and D. W. Matolak, "Air-ground channel characterization for unmanned aircraft systems - part ii: Hilly and mountainous settings", *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 1913–1925, Mar. 2017, ISSN: 0018-9545. DOI: 10.1109/TVT.2016.2585504.

[94] R. Sun, D. W. Matolak, and W. Rayess, "Air-ground channel characterization for unmanned aircraft systems—part IV: Airframe shadowing", *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 7643–7652, 2017.

[95] F. Talebi and T. Pratt, "Channel sounding and parameter estimation for a wideband correlation-based MIMO model", *IEEE Transactions on Vehicular Technology*, vol. 65, no. 2, pp. 499–508, 2016. DOI: 10.1109/TVT.2015.2404571.

[96] J. Taylor and K. Munson, *Jane's Pocket Book of Remotely Piloted Vehicles: Robot Aircraft Today*. Collier Books, 1977, ISBN: 9780020806400. [Online]. Available: https://books.google.de/books?id=8o9TAAAAMAAJ.

[97] A. Temme and S. Helm, "Unmanned freight operations", in *Deutscher Luft- und Raumfahrtkongress 2016*, German Aerospace Center (DLR), 2016. [Online]. Available: https://www.dglr.de/publikationen/2016/420292.pdf.

[98] P. Tétreault, J. Kouba, P. Héroux, and P. Legree, "CSRS-PPP: An internet service for GPS user access to the canadian spatial reference frame", *Geomatica*, vol. 59, pp. 17–28, Jan. 2005.

[99] The European Aviation System Planning Group, "Report of the first meeting", ICAO, Paris, Report, Dec. 2019. [Online]. Available: https://www.icao.int/EURNAT/EUR%20and%20NAT%20Documents/EUR%20Documents/EASPG%20Reports/EASPG1/_EASPG01%20Final%20Report.pdf.

[100] United Nations Development Programme, *In moldova, agricultural drones are used more and more by farmers*, Sep. 2022. [Online]. Available: https://www.undp.org/european-union/stories/moldova-agricultural-drones-are-used-more-and-more-farmers.

[101] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python", *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: 10.1038/s41592-019-0686-2.

[102] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "Scikit-image: Image processing in python.", eng, *PeerJ*, vol. 2, e453, 2014.

[103] M. Walter, D. Shutin, and U.-C. Fiebig, "Prolate spheroidal coordinates for modeling mobile-to-mobile channels", *IEEE Antennas and Wireless Propagation Letters*, vol. 14, pp. 155–158, 2015. DOI: 10.1109/LAWP.2014.2358265.

[104] M. Walter, D. Shutin, M. Schmidhammer, D. W. Matolak, and A. Zajic, "Geometric analysis of the doppler frequency for general non-stationary 3d mobile-to-mobile channels based on prolate spheroidal coordinates", *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 10 419–10 434, 2020. DOI: 10.1109/TVT.2020.3011408.

[105] Y.-C. Wang and Z.-Q. Luo, "Optimized iterative clipping and filtering for PAPR reduction of OFDM signals", *IEEE Transactions on Communications*, vol. 59, no. 1, pp. 33–37, 2011. DOI: 10.1109/TCOMM.2010.102910.090040.

[106] Weglarczyk, Stanislaw, "Kernel density estimation and its application", *ITM Web Conf.*, vol. 23, p. 00 037, 2018. DOI: 10.1051/itmconf/20182300037. [Online]. Available: https://doi.org/10.1051/itmconf/20182300037.

[107] S. Weinstein and P. Ebert, "Data transmission by frequency-division multiplexing using the discrete fourier transform", *IEEE Transactions on Communication Technology*, vol. 19, no. 5, pp. 628–634, 1971. DOI: 10.1109/TCOM.1971.1090705.

[108] Wissenschaftliche Dienste des Deutschen Bundestags, *Zum Drohneneinsatz im Krieg um Bergkarabach im Jahre 2020*, Jan. 2021. [Online]. Available: https://www.bundestag.de/resource/blob/825428/5b868defc837911f17628d716e7e1e1d/WD-2-113-20-pdf-data.pdf.

[109] G. Wunder, R. F. Fischer, H. Boche, S. Litsyn, and J.-S. No, "The PAPR problem in OFDM transmission: New directions for a long-lasting problem", *IEEE Signal Processing Magazine*, vol. 30, no. 6, pp. 130–144, 2013. DOI: 10.1109/MSP.2012.2218138.

[110] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications", *IEEE Access*, vol. 3, pp. 1089–1100, 2015, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2015.2453991.

[111] Q. Zhu, F. Bai, M. Pang, J. Li, W. Zhong, X. Chen, and K. Mao, "Geometry-based stochastic line-of-sight probability model for A2G channels under urban scenarios", *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 7, pp. 5784–5794, 2022. DOI: 10.1109/TAP.2022.3161277.

[112] X. Zhu, W. Pan, H. Li, and Y. Tang, "Simplified approach to optimized iterative clipping and filtering for PAPR reduction of OFDM signals", *IEEE Transactions on Communications*, vol. 61, no. 5, pp. 1891–1901, 2013. DOI: 10.1109/TCOMM.2013.021913.110867.