NTNU – Trondheim
Norwegian University of
Science and Technology

Digital Signal Processing for UAV Wireless Communication System

Eirik Heggedal

Master of Science in Cybernetics and Robotics
Submission date: June 2015
Supervisor: Amund Skavhaug, ITK

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Abstract

This thesis was motivated by a student project with the goal of building a prototype for a vertical take-off and landing (VTOL) unmanned aerial vehicle (UAV). The goal was to determine what performance could be expected from the wireless communication system of this UAV by employing advanced digital signal processing techniques. This study was important to determine which applications this UAV is suitable for, and hence its commercial potential. A wast theoretic study was conducted to identify what factors prevents reliable communication with high data transfer rate. These factors was found to be limitations on signal bandwidth and emitted power, besides the high mobility of the UAV antenna. A fundamental challenge was identified: Increasing data transfer rate results in increased signal distortion. To facilitate a study on how to remedy this problem, an ergodic model of the communication system was derived, based on the filtered Gaussian noise method. Great effort was invested in finding a digital signal processing technique that could optimize for data transfer rate and reliability simultaneously. Various filtering techniques was studied. On-line identification of the communication channel was found to be a prerequisite for achieving an effective filter. The system identification problem was approached as a black box parameter estimation problem. Simulations revealed that the problem was non-convex. The problem was solved in a simulation with a channel model that exhibited perhaps unrealistically mild time-variant behaviour, using a global optimization method. However, the solution was not found within a satisfactory time period. It was therefore concluded that the non-convex problem is not suited to be solved on-line by an embedded computer within reasonable price range. Reliable communication must therefore be achieved by trading off data transfer rate. The three primary trade off mechanisms are the modulation order, the period between transmitted symbols, and data overhead containing error correcting code. Increasing the period between transmitted symbols is expected to yield the best results. Consequently, the expectation of limited data transfer rate is taken into consideration by the UAV development team.

Sammendrag

Denne avhandlingen var motivert av et studentprosjekt med mål om å bygge en prototype for et ubemannede fly som kan lande og ta av vertikalt. Målet var å finne hvilke ytelsen som kan forventes fra det trådløse kommunikasjonssystemet til dette flyet ved bruk av avansert digitale signalbehandling. Denne studien var viktig å finne ut hvilke bruksområder dette flyet er egnet for, og dermed dets kommersielle potensial. En grundig teoretisk studie ble utført for å identifisere hvilke faktorer som hindrer pålitelig kommunikasjon med høy dataoverføringshastighet. Disse faktorene ble funnet å være begrensninger på signalbåndbredden og sendeeffekten, i tillegg til den høye mobilitet til antennen på flyet. En grunnleggende utfordring ble identifisert: Økende dataoverføringshastighet resulterer i økt signalforvrengning. For å muliggjøre en studie på hvordan man skal bøte på dette problemet, ble en ergodisk modell av kommunikasjonssystemet utledet. Stor innsats ble investert i å finne en digital signalbehandlingsteknikk som kan optimalisere for dataoverføringshastighet og pålitelighet samtidig. Ulike filtreringsteknikker ble studert. Kontinuerlig identifisering av kommunikasjonskanalen ble funnet å være en forutsetning for å oppnå et effektivt filter. Systemidentifikasjonsproblemet ble ble tilnærmet som et svart boks parameterestimeringsproblem. Simuleringer viste at problemet var ikke-konveks. Problemet ble løst i en simulering, med en kanalmodell som oppviste kanskje urealistisk mild tidsvarierende oppførsel, ved hjelp av en global optimaliseringsmetode. Løsningen ble ikke funnet innenfor en tilfredsstillende tidsperiode. Det ble derfor konkludert med at det ikkekonvekse problemet ikke er egnet til å bli løst av et innvevd datasystem innen rimelig prisklasse. Pålitelig kommunikasjon må derfor oppnås på bekostning av dataoverføringshastigheten. Avveining mellom disse kan realiseres ved å endre på modulasjongraden, perioden mellom sendte symboler, og tilleggsdata som inneholder feilkorrigerende kode. Økning av perioden mellom sendte symboler forventes å gi beste resultaten. Det ble konkludert med at begrenset dataoverføringshastighet må forventes.

Acknowledgements

I would like to thank my supervisor Amund Skavhaug for excellent advice throughout this project.

Contents

List of Figures

Abbreviations

ACF	**A**uto **C**orrelation **F**unction
ADC	**A**nalogue to **D**igital **C**onverter
ASK	**A**mplitude **S**hift **K**eying
BER	**B**it **E**rror **R**ate
AWGN	**A**dditative **W**hite **G**aussian **N**oise
DAC	**D**igital to **A**nalogue **C**onverter
DFE	**D**ecision **F**eedback **E**qualizer
DSP	**D**igital **S**ignal **P**rocessor
EKF	**E**xtended **K**alman **F**ilter
FGN	**F**iltered **G**aussian **N**oise
FIR	**F**inite **I**mpulse **R**esponse
FLOPS	**FL**oating point **O**perations **P**er **S**econd
FPU	**F**loating **P**oint **U**nit
FSK	**F**requency **S**hift **K**eying
GCS	**G**round **C**ontrol **S**tation
IIR	**I**nfinite **I**mpulse **R**esponse
ISI	**I**nter **S**ymbol **I**nterference
LMS	**L**east **M**ean **S**quare
LOS	**L**ine **O**f **S**ight
LTI	**L**inear and **T**ime-**I**nvariant
MLSE	**M**aximum **L**ikelihood **S**equence **E**stimator
MMSE	**M**inimum **M**ean **S**quare **E**rror
MSE	**M**ean **S**quare **E**rror
PSK	**P**hase **S**hift **K**eying

RLS	**R**ecursive **L**east **S**quares
RNN	**R**ecurrent **N**eural **Network**
SER	**S**ymbol **E**rror **R**ate
SNR	**S**ignal to **N**oise **R**atio
UAS	**U**nmanned **A**erial **S**ystem
UAV	**U**nmanned **A**erial **V**ehicle
UKF	**U**nscented **K**alman **F**ilter
VTOL	**V**ertical **T**ake-**O**ff and **L**anding
WSSUS	**W**ide **S**ense **S**tationary **U**ncorrelated **S**cattering
QAM	**Q**uadrature **A**mplitude **M**odulation

Chapter 1

Introduction

1.1 Background and motivation

[2]

This thesis was motivated by a student project with the goal of building a prototype for a small VTOL UAV, dubbed "X-Drone". A group of students, mainly from the engineering cybernetics and mechanical engineering programs at NTNU, participated with contribution to different aspect of the development. This master project contributes with theoretic study aiming to identify possible challenges that could arise in the radio communication system for this UAV. This information is vital to determine potential commercial applications for the UAV.

Existing literature in the field are based on other applications, e.g. mobile networks such as 4G, and might not suitable for UAS. UAS differs from these application on key parameters such as higher mobility in six spatial dimensions, and over large geographical areas in short time span. Also, literature that is only few years old might be outdated due to the ever increasing capabilities of modern embedded computers. Embedded computers enables the use advanced algorithms, that are able to dynamically adapt communication scheme based on real-time computation.

The use of Unmanned aerial systems (UAS) has seen a rise in popularity the recent past. This can be contributed to the increasing availability of the technology, UAS have become better and cheaper. Although the majority of UAS are used for recreational purposes, the technology has proven useful in commercial and academic applications. These applications include, but are not limited to, remote sensing, logistics and communication relay. Remote sensing is the acquisition of information about an object or phenomenon without making physical contact with the object and thus in contrast to on site observation. For UAS, the most viable commercial applications involve some form of imaging. Examples of such applications are:

- Hyperspectral imaging applications, including mineralogy, agriculture and surveillance.
- Steriographic pairs of areal photographs to make topographic maps.
- Infra-red imaging for object detection and tracking, such as search and rescue.

Common for these applications are the need for processing large amounts of data. For some applications, this may be done by collecting data during flight, and processing the data after the flight. Other applications calls for real-time processing of data. This can be achieved by on-board processing or by transferring the data to the GCS. These two approaches have distinct key advantages. Although microprocessors are becoming ever more powerful, they cannot match the processing power of high-end desktop computers. On the other hand, real-time GCS data processing requires reliable and high data-rate communication between the UAV and GCS.

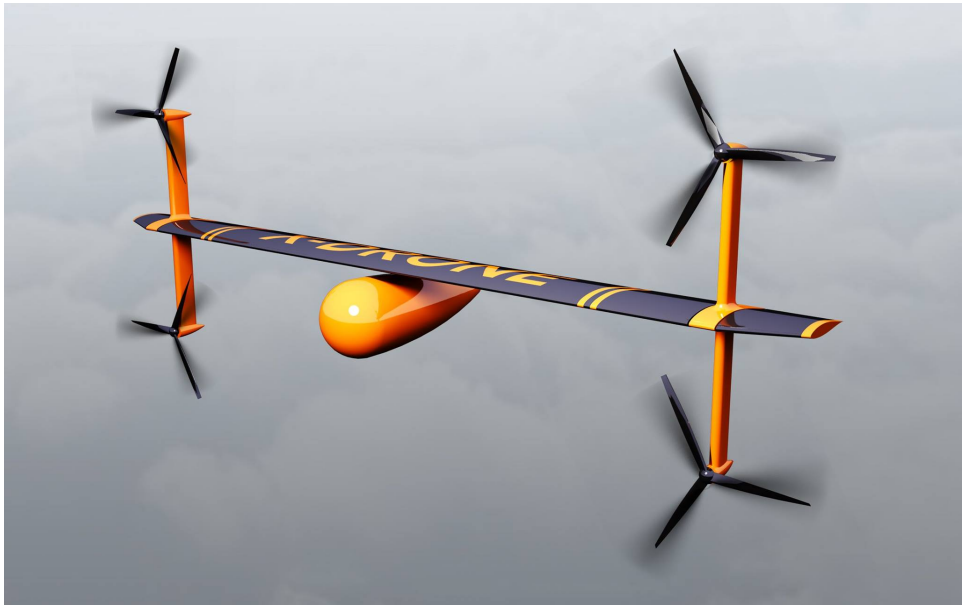

FIGURE 1.1: Illustration of an UAV.

1.2 Goals

The goal of this project is to identifying both underlying physical restriction for the communication system, and other technical challenges that prevents us from achieving reliable communication with high data transfer rate. Focus will then be shifted towards techniques for combating these technical challenges, with the physical restrictions in mind. Effort will be put towards finding application specific solutions. Due to spatial limitations in the UAV under consideration here, the techniques will be limited to signal processing, as opposed to antenna and front end design. The approach taken here is inspired by techniques from cybernetics and applied mathematics, such as optimization and estimation theory, rather than the traditional approach from information theory. This goal of this approach is to provide original insight to the problems regarding high mobility wireless communication, and determine whether this approach is worth perusing any further.

1.3 Thesis outline

Chapter 2 will provide an introduction to the challenges related to high UAS wireless communication. In chapter 3, a mathematical model of an wireless communication system will be derived. This model aims to accurately capture the characteristics of typical scenarios that occurs during UAS operation. Chapter 4 will summarize a study on digital signal processing techniques, and give recommendation for techniques to combat the technical challenges presented in chapter 2 and modelled in chapter 3. Chapter 5 describes experiments conducted to determine the performance of the digital signal processing recommended in chapter 4. Chapter 6 concludes the thesis with discussion and recommendations for further work.

Chapter 2

Theory

This chapter will present the underlying physical restriction for wireless communication as well as technical challenges specificity for UAS radio communication. Most notably, these challenges are results of multipath propagation, bandwidth and energy limitations along with mobility of the UAV. The goal is to provide the reader with intuition of these challenges.

FIGURE 2.1: Illustration of a radio communication system. [**?**]

2.1 Radio wave propagation

In digital radio communication, data is encoded into electromagnetic waves, with radio frequency, which are propagated between antennas. In Telecom literature, the data-carrying electromagnetic waves are referred to as signals, and the medium in which these waves are propagated are referred to as the channel. These terms, where "channel" refer to the analogue section of the communication system, are

adopted in this thesis. This section will briefly describe the main phenomena which affects radio wave propagation.

2.1.1 Propagation mechanisms

Reflection occurs when a propagating electromagnetic wave impinges upon an object which has very large dimensions when compared to the wavelength of the propagating wave. E.g. the earth or buildings. If the surface of the object is smooth, the reflection is specular (mirror-like). If the surface is rough, the reflection is diffuse. When reflection causes the wave not to arrive to receiver, it is often referred to as shadowing in Telecom literature. **Refraction** is the change in direction of propagation of a wave due to a change in its transmission medium. **Diffraction** occurs when the radio path between the transmitter and receiver is obstructed by a surface that has sharp irregularities (edges). This gives rise to a bending of waves around the obstacle. **Scattering** is the process where electromagnetic waves are forced to deviate from a straight trajectory by one or more paths due to localized non-uniformities in the channel. Scattering occurs when there are objects with dimensions that are small compared to the wavelength λ in the channel, and where the number of obstacles per unit volume is large. Scattered waves are produced by rough surfaces, small objects, or by other irregularities in the channel. In practice, droplets, foliage etc. **Absorption** refer to the phenomena where the energy of an electromagnetic wave is absorbed by matter in the channel. Most notably water vapour. This typically occurs when the energy of the photons are absorbed by electrons of an atom. The reduction in energy of a propagation wave due to absorption is often referred to as attenuation.

2.1.2 Multipath Propagation

Due to these propagation mechanisms, transmitted waves can reach the receiver via different paths. Due to different path lengths, these waves are time delayed and arrives from a different direction compared to the direct path. As a result,

the received signal may consist of a superposition of several adjacent signals with varying time delay.

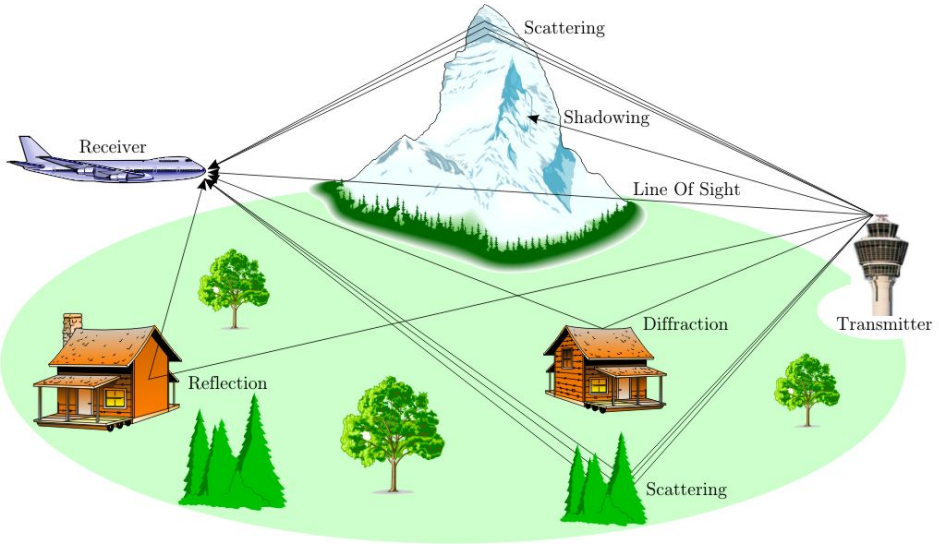

FIGURE 2.2: Illustration of multipath propagation.

Multipath propagation leads to interference, since the time delayed signals are relatively phase shifted. There exists countermeasures for this problem, which will be discussed in chapter 4, but for now lets consider this interference as random noise. Although this assumption is clearly wrong, it will simplify the introduction of other topics in this chapter.

2.2 Bandwidth limitations

This section will introduce sources of bandwidth limitations, such as regulations governing the use of the electromagnetic spectrum for wireless communication. Technical difficulties resulting from bandwidth limitation will also be presented.

2.2.1 Regulations

The assignment of frequencies for wireless communication is regulated by international agreements, and governed internationally by the International Telecommunication Union, which is a sub-organization of the United Nations. ITU established guidelines for the usage of spectrum in different regions and countries. Further regulations are issued by frequency regulators of individual countries [3]. For example the Federal Communications Commission in the USA, and Nkom in Norway. Regulated frequency bands may be divided into two groups. There are bands which are regulated with regards to whom may use them. These are typically assigned to users by national authorities. The other group, such as the 2.45 Ghz band, may be used by anyone, but are regulated with regards to emission power and bandwidth. There exists clever work-around techniques to limit the effect of these restrictions. Most notably cognitive radio techniques, and orthogonal frequency-division multiplexing. In this thesis the assumption of a coherent and limited frequency band will be used, as this most likely will be the case for the UAS motivating this thesis. Either way, considering the worst case scenario is good engineering practise.

2.2.2 Shannon's theorem

Shannon's theorem provides a good intuition to how bandwidth limitation affects the performance of the communication system. It states the tight upper bound of the data transfer rate for a communication system in the presence of noise:

$$C = B \log_2 \left(1 + \frac{S}{N}\right) \tag{2.1}$$

where:

C is the theoretical maximum data transfer rate, in bits per second.

B is the width of the frequency band, in Hertz.

S is the average received signal power over the bandwidth, in Watts.

N is the average power of the AWGN, in Watts.

This theorem assumes constant spectral density of both signal and noise power, and the noise to be purely additive white Gaussian. Hence, this upper bound cannot realistically be achieved. In practice, distortion and noise from radio wave propagation, antenna and front-end electronics determines the the actual transfer rate limit.

2.2.3 Modulation

Digital modulation is the process of mapping digital data to signal waveforms that can be transmitted trough a wireless channel, which is inherently analogue. This encoding can therefore be viewed as a form of analogue to digital conversion. The terms symbol will be used for the pair of discrete-time and continuous-time signal. The discrete-time signal, i.e. the digital data, is an integer number usually represented in either decimal or binary. The continuous-time signal is a waveform which may be represented as a complex vector. The relationship between the discrete- and continuous-time signal is illustrated in figure 2.4. The demodulator in the receiver aims to perform the inverse transform of the modulator. Hence the entire continuous-time system between the modulator and demodulator, referred to as the channel, may be represented as a mapping of the complex vector representing of the symbol.

A set of M symbols that may be transferred is referred to as an M-ary alphabet. Each symbol in an M-ary alphabet can be related to a unique sequence of k bits, expressed as $k = log_2 M$, where M is the size of the alphabet. Hence $M = 2^k$. Since one of M symbols is transmitted during each symbol duration T_s, the data rate R in bits per second can be expressed as

$$R = \frac{k}{T_s} = \frac{\log_2 M}{T_s} \tag{2.2}$$

The most fundamental digital modulation schemes are:

PSK (phase-shift keying): A finite number of phases are used to map symbols.

FSK (frequency-shift keying): A finite number of frequencies are used to map symbols.

ASK (amplitude-shift keying): A finite number of amplitudes are used to map symbols.

QAM (quadrature amplitude modulation): A finite number of at least two phases and at least two amplitudes are used to map symbols.

QAM will be used in all examples and simulations in this thesis, since it's a standard used extensively in the industry. QAM subsumes ASK, FSK and PSK, as the final waveform of QAM is a combination of ASK and PSK, and FSK is a special case of PSK. Since QAM in theory supports any alphabet size, it can achieve arbitrarily high spectral efficiency. In a transmitter using QAM, the real and complex (quadratur and in-phase) components of the symbol mapping are multiplied to two sinusoids in quadrature ($\frac{\pi}{2}$ out of phase). These sinusoids have the same frequency f_c, referred to as the carrier frequency. The final signal is an additive combination of the two sinusoids, as illustrated below.

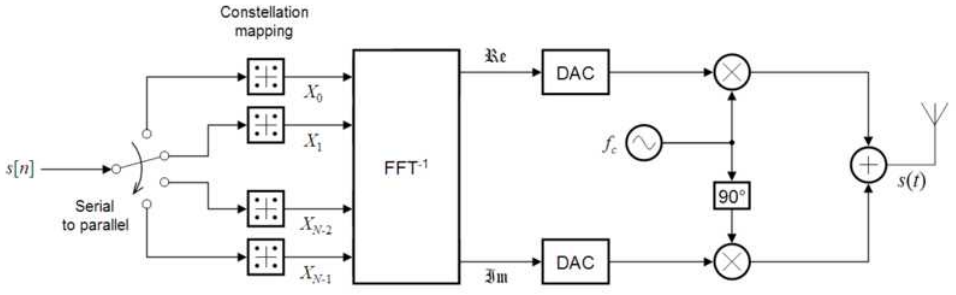

FIGURE 2.3: Illustration of a QAM transmission system.

The encoding procedure endeavours to make each waveform as unalike as possible; the goal is to render the cross correlation coefficient c_f as small as possible. For example, if there are only two symbols in the alphabet, the smallest possible c_f can

be achieved by making the signals antipodal, which makes them anti-correlated. This may be more easily visualized as increasing the distance between the complex vectors. A constellation diagram provides this representation, as exemplified by figure 2.4.

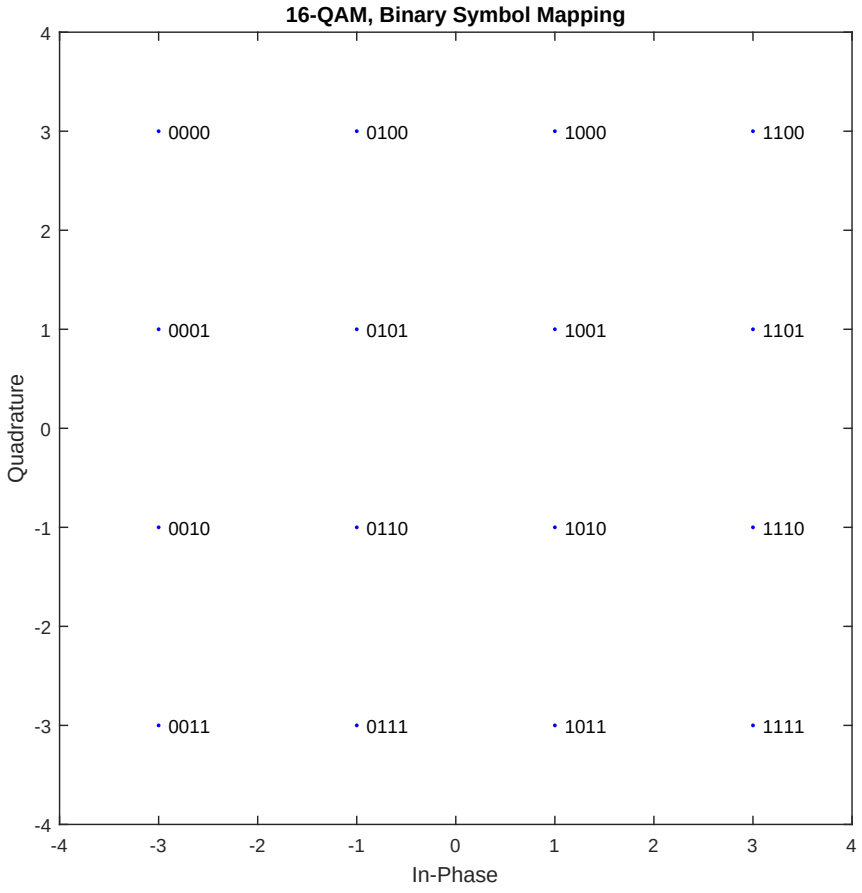

FIGURE 2.4: Illustration of QAM symbol mapping to the complex plane, with alphabet size $M = 16$.

Since the signal power restrictions, and received noise power, of course are unrelated to the choice of alphabet size, the relative noise power will increase with the modulation order. Hence, the choice of alphabet size can be used to trade off between reliability and data transfer rate.

2.2.4 Pulse shaping

The signal must be filtered to conform with the bandwidth restrictions. This is typically done by low-pass filtering the baseband signal before it's multiplied with the carrier frequency. Ideally this would be done with a sinc filter. Typically, a raised cosine filter is used. The figures below illustrates how low-pass filtering with a raised cosine filter affects the signal in the time domain, i.e. the Fourier transform.

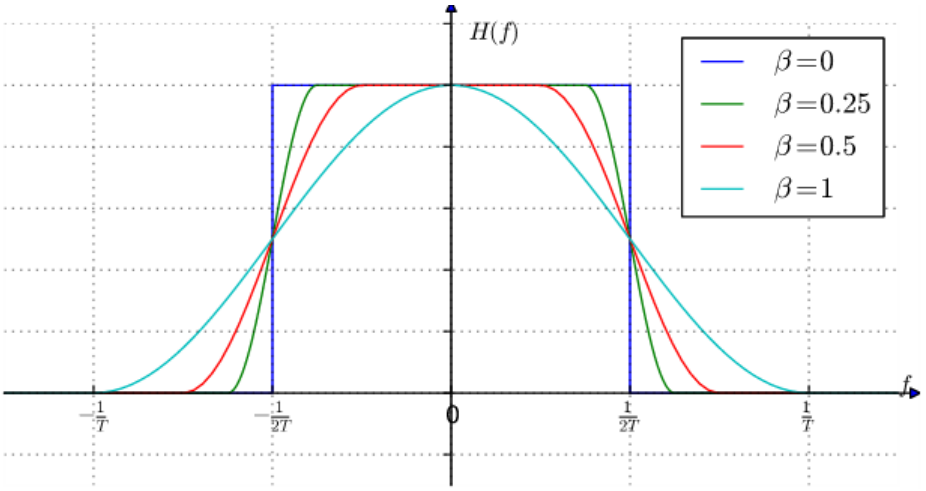

FIGURE 2.5: Illustration of the frequency response of raised-cosine filter with various roll-off factors.

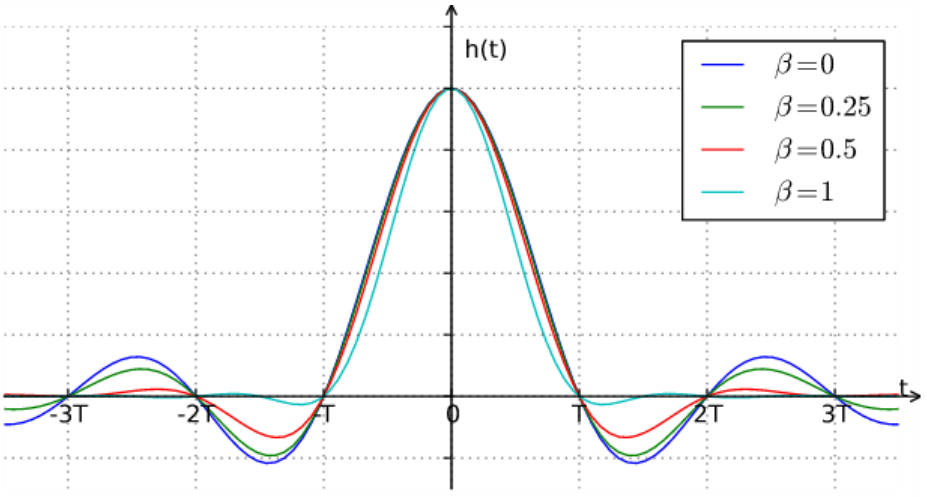

FIGURE 2.6: Illustration of the impulse response of raised-cosine filter with various roll-off factors.

In the figures above, T is the symbol period and β is the roll-off factor of the raised cosine filter. Hence, adjacent symbols overlap at all times except the points that

is an integer multiplicand of the symbol period. This as illustrated in the figure below.

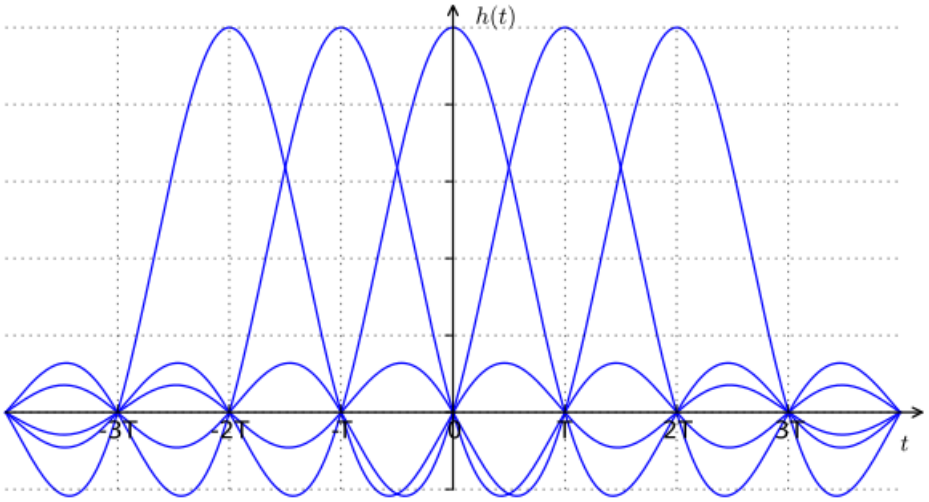

FIGURE 2.7: Illustration of overlapping of consecutive raised cosine impulses.

Recall that the continuous-time system between the ADC in the modulator and the DAC in the demodulator at the receiver, is referred to as the channel. If this system is linear and time invariant (LTI) the received signal would also have points in time where only one symbol was non-zero. That is, the signal would remain within the same bandwidth. Theoretically the signal could be sampled at these points. This would result in a linear mapping of the complex vectors representing the symbol in the constellation diagram. However, a real channel is never LTI. The channel is inherently non-linear and, for a UAS communication system, also time-variant. This will become more obvious in the next two sections. For a real channel there would be no points in time where only one symbol was non-zero. They would always overlap. For any sampling technique the signal would be distorted. This distortion is referred to as intersymbol interference (ISI). A typical manifestation in the continuous-time domain would be spreading of the symbols. Equally, the complex vectors representing the symbols would undergo a non-linear mapping. An obvious solution to this problem would be to increase the time between symbols. That is increasing T_s while retaining B. Since this would directly impact the data transfer rate, alternative techniques will be presented in chapter 4.

2.2.5 Antenna

The bandwidth limitations discussed thus far in this section, originates from political regulations. These regulations does not always apply for UAS which, for example, may be used in military applications. The communication system will nonetheless be band limited due to properties of the antenna.

The purpose of an antenna is to transform electrical power into electromagnetic waves and vica versa. Transmission is done by applying an alternating current to the antenna terminals, the antenna radiates the energy from the current as electromagnetic waves, with the same frequency. The opposite happens in reception, the antenna intercepts some of the energy from an electromagnetic wave and produces a voltage at its terminals. To achieve good performance from an antenna, its characteristics must be chosen for a particular application. The main characteristic to consider is the antenna gain, which is the product of the directivity and the electrical efficiency. The antenna gain describes how well the antenna converts electrical power to electromagnetic waves transmitted in a specified direction. The opposite applies for a receiving antenna. Another important characteristic is the physical dimensions of the antenna. The antenna directivity will be discussed in section 2.4.

The term "antenna efficiency" is a measure of the efficiency with which a radio antenna converts the radio-frequency power accepted at its terminals into radiated power. To achieve high efficiency, the antenna must be impedance matched with the transmission line and the transceiver. For a resonating antenna, the input impedance depends on its resonant frequency. The input impedance is defined as "the impedance presented by an antenna at its terminals or the ratio of the voltage to current at a pair of terminals or the ratio of the appropriate components of the electric to magnetic fields at a point" [4]. The resonance frequency f_r of a resonator is given by: $f_r = \frac{Nv}{2d}$, where N is an integer, v is the velocity of the wave and d is the distance between the sides of the resonator, in this case the length of the antenna. At the resonance frequency, the standing wave has a current peak and voltage node at the input terminals. At this frequency the current and voltage is

in phase, i.e. the input impedance of the antenna is minimal and purely resistive. Meaning that the antenna efficiency is proportional to the factor of the radiation resistance to the loss resistance (heat). When the applied frequency is moved away from the resonance frequency, the input impedance will be mismatched and hence get a reactive component. In simple terms, this means that the antenna performs band-pass filtering with center frequency at the resonance frequency.

2.3 Energy limitations

The technical challenges related to the SNR of the received signal, introduced in the previous section, could obviously be solved by increasing the power of transmitted signal. In practice the signal power is limited, primarily by regulations on transmitted energy. But also implicitly by the power amplifiers at the transmitter, since they will introduce distortion to the signal when operated in the high region. This section explains why energy limitations is a problem.

2.3.1 Free Space Loss

For wireless communication systems where the transmitter and receiver has a clear, unobstructed line of sight path between them, the Friis transmission equation (2.3) can be used to predict received signal strength [4]. The assumption of the entire signal being transmitted via a single unobstructed LOS path is obviously not valid for the system under consideration here, but will nevertheless provide some good intuition to some of the factors that affects the received signal strength.

$$\frac{P_r}{P_t} = e_{cdt} e_{cdr} \left(1 - |\Gamma_t|^2\right) \left(1 - |\Gamma_r|^2\right) \left(\frac{\lambda^2}{4\pi R}\right)^2 D_t(\theta_t, \phi_t) D_r(\theta_r, \phi_r) |\hat{\rho}_t \cdot \hat{\rho}_r|^2 \quad (2.3)$$

where:

P_r = Power delivered to the receiver load.

P_t = Input power of the transmitting antenna.

$e_r = (1 - |\Gamma|^2)$ = Reflection (mismatch) efficiency.

e_{cdr} = Antenna radiation efficiency of the receiving antenna.

e_{cdt} = Antenna radiation efficiency of the transmitting antenna.

$\left(\frac{\lambda^2}{4\pi R}\right)^2$ = The *free-space loss* which takes into account the spherical spreading of the energy by the antenna.

R = The distance between the antennas.

$D_t(\theta_t, \phi_t)$ = Directivity of the transmitting antenna.

$D_r(\theta_r, \phi_r)$ = Directivity of the receiving antenna.

$|\hat{\rho}_t \cdot \hat{\rho}_r|^2$ = PLF (polarization loss factor).

Although the equation gives a statistically good estimate of the received power under idealized conditions, it is unsuited for predicting instantaneous P_r. This can be contributed to it's neglect of important factors such as water absorption, interference, temperature noise and Doppler spread.

2.4 Mobility

Mobility of the antennas in a wireless communication system is a problem for two distinct reasons, which will be discussed in this section. One is related to the directional mobility, i.e. the heading of the antenna, the other is related to the translational mobility of the antenna. For an UAS the antenna mounted on the UAV is mobile under flight. The other antenna may also be mobile, e.g. in the case of satellite communication. In this thesis the assumption of a stationary earth mounted antenna will be used.

2.4.1 Antenna Directivety

The term antenna directivity is as "the ratio of the radiation intensity in a given direction from the antenna to the radiation intensity averaged over all directions. The average radiation intensity is equal to the total power radiated by the antenna devided by 4π. If the direction is not specified, the direction of maximum radiation intensity is implied" [4]. The antenna directivity measures, in transmit, the power density the antenna radiates in the direction of its strongest emission, versus the power density radiated by an ideal isotropic radiator, in dBi. Directivity is an important measure since the total transmitted power is limited. Hence, higher directivity enables more signal power to be transferred in the direction of the receiver. This will in turn contribute to a higher SNR. However high directivity antennas are not suitable for all applications, since they introduce new challenges. Using a high directivity antenna in a point to point communication link presupposes that the location of the receiver, in this case the UAV, is known. Since an UAV is non-stationary, the heading of the high directivity antenna must have the ability to be dynamically actuated, preferably autonomously. This is not trivial. For the UAV, a high directivity antenna would be significantly more difficult to implement. Not only would it require control algorithm of the antenna heading. It could also be a major impairment of the aerodynamic profile of the UAV, since the antenna would, e.g. have to be gimbal mounted. Although the antenna heading does not have to stabilized towards the GCS for the signal to be transferred, due to multipath propagation, a high directivity antenna would worsen the time-variant behaviour of the channel. A low directivity antenna would be more suited. In reality an omnidirectional antenna, with non-directional radiation in azimuth, would be a better choice. This would also cause time-variant behaviour, e.g. related to banking manoeuvres of the UAV.

2.4.2 Doppler Shift

The relative translational velocity of the antennas cause a Doppler shift in the signal. The change in frequency f_D due to the Doppler shift of a signal component arriving from azimuth angle α, not accounting for the effects of special relativity, is related to the translational velocity of the receiver relative to the source Δv, the speed of light c, and the frequency of the signal f as follows:

$$f_D = \frac{\Delta v}{c} f cos(\alpha) \tag{2.4}$$

For a UAS communication system, the Doppler shift will vary due to the UAV translational and rotational mobility. The variable α will depend on how the signal is multipath propagated. As seen from the equation, the Doppler shift is also a function of the signal frequency, thus the Doppler shift will not be constant over the signal bandwidth. Doppler shifts is a significant source of intersymbol interference, which may be more easily understood from it's Fourier transform in the time domain. How a signal is affected by Doppler shift hard to predict due to it's rapid time-variant behaviour. This is further discussed in chapter 3.

Chapter 3

Channel modelling

In this chapter a mathematical model of an wireless communication channel will be derived. This model aims to accurately capture the characteristics of typical scenarios that occurs during UAS operation. Ideally, this model should incorporate the system characteristics discussed in chapter 2, while having feasible computational efficiency. More specifically, the model will generate a dataset transmitted signals and their respective distorted received signals, i.e. the input and output of the channel.

3.1 Assumptions

A proper deterministic model of the channel would consist of the transfer functions of the electronics of both the transmitter and receiver, including accurate models of the antennas. Initial conditions calculated from these transfer functions could be used to solve a version of Maxwell's equation taking account for surrounding topography and electrodynamic properties of the transmission medium. Since geometrical optics and unified theory of diffraction require accurate knowledge of material and geometry of all objects in the transmission medium, deterministic methods require high computational effort, and are sensitive to change in the

initial conditions [5]. Hence, a deterministic model is not feasible. Neither is it needed to model typical system dynamics.

A stochastic model can be used to satisfactory capture the dynamics of a specific scenario. For a model to be stochastic, at least one of it's parameter must be a random variable. The scenario to be modelled here is described by the assumptions below:

1. The UAV has a single dipole antenna.
2. The UAV antenna polarization is always polarized in azimuth (horizontal polarization).
3. The GCS has a single stationary antenna headed towards the UAV.
4. The communication system operates with a single coherent frequency band.
5. The UAV is airborne, a few kilometres away from the GCS.
6. There is a line of sight (LOS) between the GCS and the UAV.
7. The resolved signal is propagated over multiple paths, including the LOS.
8. The signal is perturbed by AWGN from background radiation and front-end electronics.
9. There is a LNA in the receiver with expected output power equal to the signal power at the input of the transmit amplifier. That is $E[\sum_{n=1}^{N} C_n^2 = 1]$ where C_n is the n'th signal component.

3.2 Stochastic model

The impulse response of the channel, i.e. the complex vector representing the received signal, is assumed to be an additive combination of the complex vectors representing:

- The resolved signal, which is the combination of components from one or more transmitted signals. Due to multipath propagation, these signal components may originate from different, typically adjacent, transmitted symbols. The number of different symbols contributing to the received signal, and their magnitude, depends on their spreading in the time domain due to Doppler shift f_D and their time delay τ.

- Additive white Gaussian noise (AWGN) introduced by the front-end electronics and antennas. This noise primarily arise from background noise (temperature noise) and unresolved signals (interference) observed by the antenna, in addition to noise introduced by the front-end electronics in both the receiver and transmitter. The quantization error of the ADC is also assumed to be AWGN, although this is not strictly true.

Ignoring the AWGN, the linear time-variant system interpretation may be described by a simple deterministic model, where the received lowpass complex signal $y(t)$ is related to the transmitted complex signal $x(t, \tau)$ by:

$$y(t) = \int_{-\infty}^{\infty} x(t, \tau) h(t, \tau) d\tau \tag{3.1}$$

Equation 3.1 assumes sinc (raised cosine with $\beta = 0$) low-pass filter. The spectrum of the received signal is given by:

$$Y(\tilde{f}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(f) H(t, f) e^{j2\pi f t} e^{j2\pi \tilde{f} t} df dt \tag{3.2}$$

Which does not reduce to Y(f) = X(f)H(f) [6]. Fourier transforming the transfer function with respect to t, yields the Doppler spread f_D and time delay τ parametrized function, often referred to as the spreading function:

$$s(f_D, \tau) = \int_{-\infty}^{\infty} h(t, \tau) e^{j2\pi f_D t} dt \tag{3.3}$$

These interpretations are Fourier transforms of each other. As mention a deterministic model is not feasible, instead a stochastic model limited to second order

statistics, may be derived using the autocorrelation function (ACF). The ACF of the raceived signal $R_{yy}(t,t')$ is given by:

$$R_{yy}(t,t') = E\left[\int_{-\infty}^{\infty} x^*(t-\tau)h^*(t-\tau)d\tau \int_{-\infty}^{\infty} x(t'-\tau')h(t'-\tau')d\tau'\right] \tag{3.4}$$

The transmitted signal can be interpreted as a stochastic process that is independent of the transfer function representing the channel. Under this assumption, the expectation over the transmitted signal and the transfer function may be performed separately. The ACF of the received signal may then be written as:

$$R_{yy}(t,t') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} E\left[x^*(t-\tau)x(t'-\tau')\right]E\left[h^*(t-\tau)h(t'-\tau')\right]d\tau d\tau' \tag{3.5}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} R_{xx}(t-\tau,t'-\tau')R_h(t,t',\tau,\tau')d\tau d\tau' \tag{3.6}$$

R_h depends on four variables since the underlying stochastic process is two dimensional. To further assumptions about the underlying physical system may lead to simplification of R_h. The most popular are called the "wide sense stationary" and "uncorrelated scatterers" assumptions [7]. Models using both assumptions are referred to as WSSUS models. The wide sense stationary assumptions is that the ACF does not depend on t and t' seperately, but rather their difference $\Delta t = t - t'$. That is, the statistic properties of the transfer function representing the channel does not change with time:

$$R_h(t,t',\tau,\tau') = R_h(\Delta t,\tau,\tau') \tag{3.7}$$

The "uncorrelated scatterers" assumption is that signal components with different delay are uncorrelated. That is:

$$R_h(t,t',\tau,\tau') = P_h(t,t',\tau)\delta(\tau-\tau') \tag{3.8}$$

Where $P_h(\cdot)$ is the power spectral density function. Combining the two assumptions yield the following relationships:

$$R_h(t, t+\Delta t, \tau, \tau') = P_h(\Delta t, \tau)\delta(\tau - \tau') \tag{3.9}$$

$$R_H(t, t+\Delta t, f, f+\Delta f) = R_H(\Delta t, \Delta f) \tag{3.10}$$

$$R_s(f_D, f'_D, \tau, \tau') = P_s(f_D, \tau)\delta(f_D - f'_D)\delta(\tau - \tau') \tag{3.11}$$

$$R_B(f_D, f'_D, f, f+\Delta f) = P_B(f_D, \Delta f)\delta(f_D - f'_D) \tag{3.12}$$

Under these assumption a discrete time model representing the complex signal $r(k)$ sampled by the receiver at time k may formulated as:

$$r(k) = \sum_{n=-N_1}^{N_2} x(k-n)g_n(k) \tag{3.13}$$

Where k is spaced by the sample period $T_s = 1/R_s$ (due to optimal low pass filtering), and $g_n(k)$ is the gain process. Proof is derived in [8].

$$h(n, \tau) = \sum_i C_i(n)\delta(\tau - \tau_i) \tag{3.14}$$

In [9] it has been shown that the power spectrum $P_s(\tau, f_D)$ is proportional to the probability density function $p(\tau, f_D)$. Under these requisites it's sufficient to specify $p(\tau)$ and $p(f_D)$ along with the AWGN $v(t)$ to construct a stochastic model of the system. $P_s(\tau)$ and $P_s(f_D)$ for a variety of scenarios occurring in aeronautical channels are described in [5]. There are two main methods for implementing a simulator described by $P_s(\tau, f_D)$; sum of sinusoid (SoS) methods, and Filtered Gaussian noise (FGN) methods. A thorough comparison of these methods are provided in [8].

3.3 MATLAB implementation

The FGN channel model described above can be easily implemented in MATLAB, using the `ricianchan` channel object. All parameters used in creating the channel object, is based on this study on aeronautical channels [5], and modified to better suit the application under consideration here. The source code for the channel model is appended. The channel had the following attributes:

```
                ChannelType: 'Rician'
          InputSamplePeriod: 6.3272e-06
            DopplerSpectrum: [1x1 doppler.ajakes]
            MaxDopplerShift: 416.6667
                 PathDelays: [1x5 double]
              AvgPathGaindB: [0 0 0 0 0]
                    KFactor: 5.6234
  DirectPathDopplerShift: 0
     DirectPathInitPhase: 0
      NormalizePathGains: 1
```

To get a proper understanding the behaviour of this channel model, the source code and MATLABs "`Channel Visualization Tool`" in the communication systems toolbox should be advised, as it does not support figure exportation. Nevertheless, two snapshots of typical impulse responses is provided below.

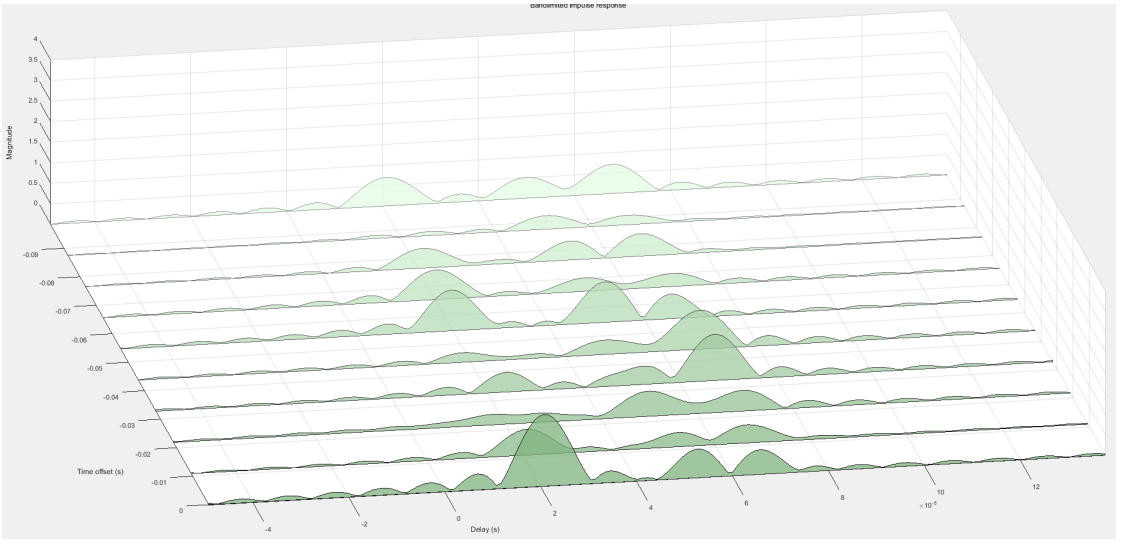

FIGURE 3.1: Illustration of typical impulse responses

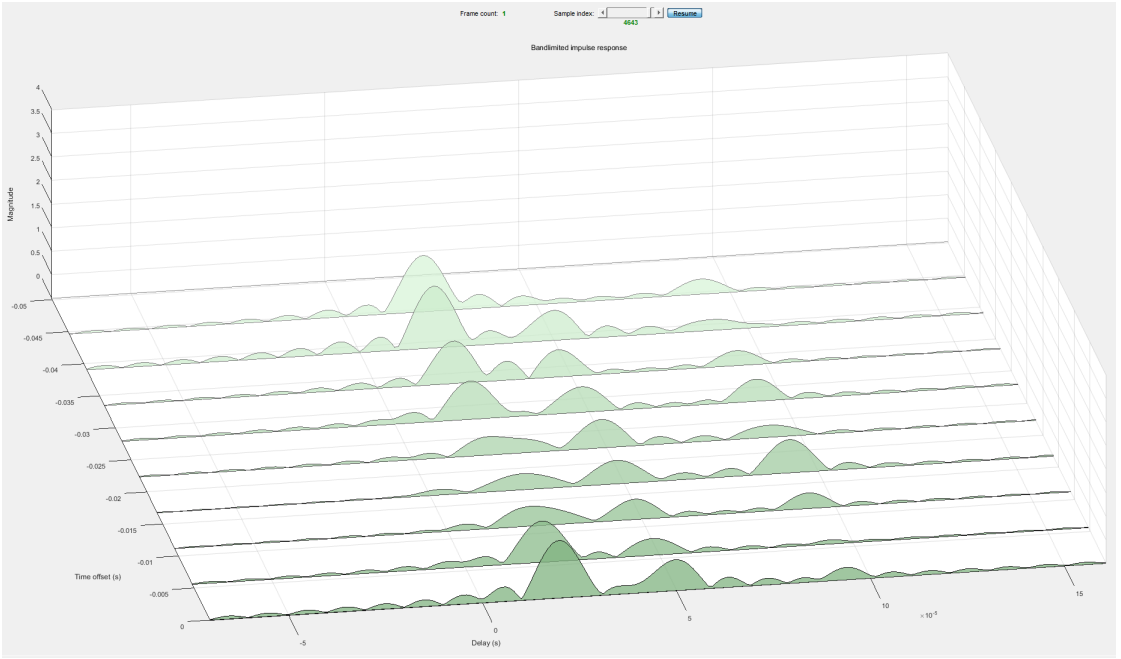

FIGURE 3.2: Illustration of typical impulse responses

In both figure 3.1 and 3.2, the first peaks are located at two sample periods. The impulse responses are roughly 8 sample period long in figure 3.1, and 10 in figure 3.2.

Chapter 4

Signal Processing

This chapter will present techniques to perform the functionality of the demodulator. Hence, a digital signal processing scheme to combat the technical challenges presented in chapter 2 and modelled in chapter 3. First, lets define:

- $k \in \mathbb{N}, k \leq n$, where $n \in \mathbb{N}$ is the current time.
- $s[k] \in \mathbb{C}$ is the sequence of transmitted symbols.
- $z[k] \in \mathbb{C} = G(h[k] * r[k]) - s[k]$ is the additive noise process from signal distortion.
- $v[k] \in \mathbb{C}$ is the AWGN process.
- $e[k] = v[k] + z[k]$ is the error process.
- $r[k + \Delta t] = s[k + \Delta t] + e[k + \Delta t]$ is the received symbols.
- $\Delta t \in \mathbb{R}$ is an arbitrary transmission delay. For simplicity, $\Delta t = 0$ will be used.
- $\mathbf{x}_r[k] = [r[k], ..., r[k - m + 1]]^T$ is the channel response vector.

The signal processing scheme at the receiver will attempt to recover the transmitted symbol $s[k]$ from $r[k]$. A system which performs this functionality is called an

equalizer in telecom literature and filter in estimation theory. To avoid confusion the term filter will be used here. Thus, $r[k]$ is the *observation* to the filter, which aims to estimate $\hat{s}[k] = s[k]$. The performance of the filter is measured in its ability to estimate the transmitted symbol from the received signal, as the fraction of errors; symbol error rate (SER) or equally bit error rate (BER). Hence, the ideal filter $H_o(\cdot)$ always satisfies:

$$y[k] = \hat{s}[k] = H_o(r[k]) \tag{4.1}$$

Where $y[k]$ is the symbol estimate, i.e. output from the filter, at time k. Due to the time-variant non-linear behaviour of the channel, $H_o(\cdot)$ has to be non-linear and time-variant. A non-linear filter is one whose estimate is not a linear function of the observation. The simplest way to achieve this is to use a channel estimator in combination with a sequence estimator. The channel estimator is a filter that attempts to estimate the transmitted symbol by predicting the distortion introduced by the channel. The sequence estimator attempts to estimate the transmitted symbol by finding the most likely sequence of symbols. In the trivial case where the sequence has length one, the sequence estimator is a simple symbol classifier.

4.1 Data Estimator

Recall that demodulation is the process of transforms the received symbol to its respective set of data bits, and that the symbol may be perceived as a complex vector. This functionality can be achieved by applying a limiter, also known as a classifier or a decision device. A limiter is a filter that maps an observation to an estimate which is member of a predetermined set. If the members of the set are the symbols of the symbol alphabet the functionality of the demodulator is achieved. In the following, the limits are constant and equidistant from adjacent symbols, i.e. the input is mapped to the closest valid symbol. The output from the limiter

$\hat{s}[k]$ is referred to as the symbol decision or simply decision. If the limiter $H_L(\cdot)$ is combined with a filter $H_1(r[k])$ as illustrated below, $H_1(r[k])$ merely needs to produce estimates within the limits for the correct symbol for the concatenation of these filters to be optimal.

Alternatively, the data may be estimated from an observation of a sequence of symbols, using a maximum likelihood sequence estimator (MLSE). Typically implemented with the Viterbi algorithm. The general idea of the MLSE is to determine the sequence of symbols under the assumption of a stationary channel with perturbation by AWGN. The first assumption as obviously not valid for the system under consideration here. It would still be possible to implement this type of data estimator by using an adaptive filter, as described below, compensate for the time-variant behaviour of the channel, and ideally produce a stationary estimate of the received sequence.

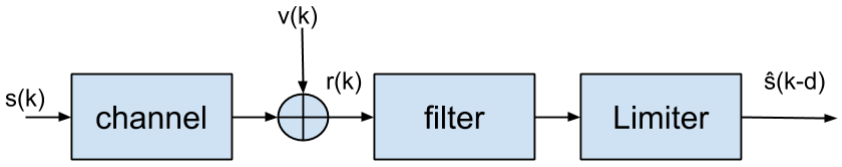

FIGURE 4.1: Illustration of a limiter.

The distance between the limits are directly proportional to the inverse square of the modulation order in the case of QAM. This means the sufficient performance of $H_1(r[k])$, in terms of worst case $e[k]$, depends on the modulation order.

4.2 Linear filters

The optimal filter may consist of a linear filter $h[k]$ concatenated to a limiter. That is:

$$H_o(r[k]) = H_L(r[k] * h[k]) \tag{4.2}$$

Intuitively, one may think that $h[k]$ should have the inverse frequency response of the transfer function of the channel, such that their concatenation has a flat

frequency response. Such a filter would be optimal for eliminating distortion from a noiseless linear channel. In reality this filter is very noise sensitive, since it must have strong amplification at frequencies where the transfer function of the channel attains small values. Assuming the noise to be flat over the bandwidth of the signal, the SNR at the frequencies of the signal would determine the amplification of noise relative to the signal. As a consequence, the the noise power at the output of the filter is greater then at the input.

A better contestant for $h[k]$ is a filter that attempts to minimize the mean square error of $e[k]$. Hence, a filter that minimizes:

$$MSE = E[|e[k]|^2] = E[\mathfrak{Re}\{|e[k]|\}^2 + \mathfrak{Im}\{|e[k]|\}^2] \tag{4.3}$$

These filters are called MMSE filters, or filters with a quadratic cost functions. Strictly speaking, an arbitrary filter does not have a quadratic cost function to be optimal, but these filters are thoroughly studied and well understood. The Wiener filter [Wiener] is an optimal MMSE filter under the conditions that its physically realizable, and $r[k]$ is a stationary stochastic process with known first and second moment statistics. Thus, the Wiener filter is in fact any filter that fulfils these conditions. However, the Wiener filter is inadequate if the channel is time-variant, which would render $r[k]$ nonstationary. In this situation there does not exist a single optimal filter. The optimal filter is one that tracks the statistical variations caused by the time-variant behaviour of the channel.

4.3 Feedback filter

Inserting a feedback filter as depicted below is results in a filter commonly referred to as a decision feedback equalizer (DFE) in Telecom literature. The abriviation DFE will be adopted here to refer to the combination of a feed-forward filter, feedback filter and limiter. The goal of the feedback filter is to remove distortion from a symbol to subsequent symbols. Hence, this filter relies on the correctness of previous symbol estimates. Consider a sequence of exclusively correct estimates.

The job of the feedback filter is then to determine how this symbols interferes with subsequent symbols, and remove this this part of $z[k]$, from subsequent sequence of symbols. For a stationary input process, this can be achieved with a FIR filter, where the coefficients determines how a symbol interferes in subsequent symbols. Now, consider an erroneous estimation occurs. This error will now be fed back to subsequent symbols, which in turn decreases the probability of an correct estimate. This is a manifestation of the stability problem which is inherent to introducing feedback to a system. A possible solution to this problem is to decode the data sequence from the limiter, perform error check, and re-encode them before they are used in the feedback filter. This will, however, introduce a time delay.

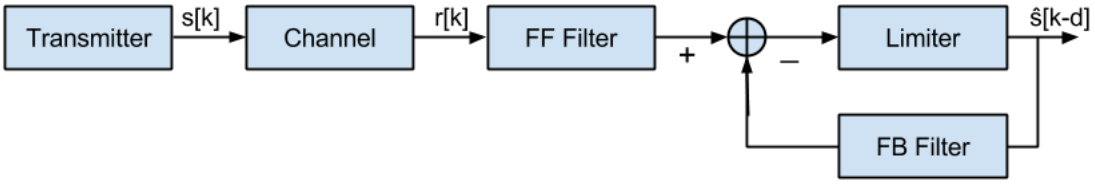

FIGURE 4.2: Illustration of a DFE

A DFE is characterized by the order of the feed-forward filter, m, the order of the feedback filter, n, and the "decision delay", d, which will be discussed below.

4.4 Adaptive filters

Consider the following example. A Wiener filter is implemented as a FIR filter for a stationary process with known statistics. The filter could have IIR structure, but FIR is assumed here to ignore stability issues. This filter is optimal only when the statistics of the process match the prior information the filter was designed for. If the statistic properties of the process diverge from the original properties, the filter will no longer be optimal. The solution to this problem is to adjust the coefficients of the filter in a way that minimizes the cost function of the filter. The adjustment of the coefficients are done with an adaptation algorithm. A filter with this property is said to be adaptive. The adaptation algorithm must be provided

with a desired filter response, which corresponds to the transmitted symbol, with a decision delay $s[k-d]$. The decision delay should correspond to the sample for which the received value attains the largest value. In other words, the largest expected value of the discrete impulse response. This is typically one of the first samples, which corresponds to a path close to LOS. The simulations described in chapter 3, suggests that the second or third should yield the best results. In the following $d = 2$ was used.

Adaptive filtering can of course be stated more generally in terms of parametric regression: The goal is to determine the parameters $\mathbf{w}$ of a model, described by a set of input-output relations $\{\mathbf{x}[k], \mathbf{y}[k]\}$, and the model structure. In the case of a FIR filter, the input vector $\{\mathbf{x}[k]$ contains a sequence $r[k], ..., r[k+m]$ where m is the length of the filter. Here the model structure is described by the filter structure, and the model parameters $\mathbf{w}$ are the filter coefficients. The vector $\mathbf{w}$ will be referred to as the parameter vector. In all practical applications, the adaptation algorithm is recursive. There is no universal optimal recursive algorithm for the adaptive filtering problem. Rather, there is an abundance of algorithms which can be used for this purpose. Their performance may be described by the following criteria [2]:

1. **Convergence rate:** This is defined as the time required for the algorithm, in response to stationary inputs, to converge "close enough" to the optimum. In this case the this would be a filter which minimizes the MSE to a satisfactory degree, which is typically determined by the modulation order as discussed above. For iterative algorithms the convergence rate can conveniently be measured in number of iterations if the algorithms under consideration receive the same inputs at the same rate. The importance of quick convergence depends on the time-variant behaviour of the channel; a slow algorithm may not be able to keep up with the time evolution of the channel.

2. **Misadjustment:** The deviation of the final converged state to the optimal solution. This criteria is typically measured by an average over an ensemble of adaptive filters.

3. **Tracking:** When an adaptive filtering algorithm operates in a nonstationary environment, the algorithm is required to track the statistical variations in the environment. The tracking performance of the algorithm, however, is influenced by two contradictory features: **(1)** rate of convergence and **(2)** steady-state fluctuation due to algorithm noise.

4. **Robustness:** For an adaptive algorithm to be robust, small disturbances (i.e. disturbances with small energy) can only result in small estimation errors. The disturbances may arise from a variety of factors, internal or external to the filter.

5. **Computational effort:** Here, the issue of concern include **(a)** the number of operations required by the processor to perform one iteration of the algorithm. This is not analogue to operation per time unit, as the algorithm may be executed in parallel. **(b)** The size of memory locations required to store the data and program, and **(c)** the investment required to implement the algorithm on a computer, including engineering and hardware expenses. The upper limit to the computational effort of an algorithm is also dependant on available technology. Today this limit is rarely an issue, due to TeraFLOPS DSPs and incredibly dense and cheap computer memory.

6. **Structure:** This refers to the structure of information flow in the algorithm, determining the manner in which it is implemented in hardware. For example, an algorithm whose structure exhibits high modularity, parallelism or concurrency is well suited for implementation using very large-scale integration (VLSI).

7. **Numerical properties:** When an algorithm is implemented numerically, inacuracies are produced due to quantization errors, which are dependent on ADC resolution and the digital representation of internal calculations.

Ordinarily, it is the latter source of quantization errors that poses a serious design problem. In particular there are two basic issues of concern: numerical stability and numerical accuracy. **Numerical stability** is an inherent characteristic of an adaptive filter algorithm. It essentially refers to the ability of the algorithm to damp errors introduced by numerical inaccuracy. This is obviously related to criteria 1, 2 and 4. **Numerical accuracy** is determined by the number of bits used in the numerical representation of data samples and filter coefficients. This number determines the magnitude of errors mainly caused by round off and truncating infinite sums. An adaptiv filter algorithm is said to be numerically robust when its insensitive to variations in wordlength used in its digital implementation.

8. **Global minimum:** Also known as absolute minimum of the error surface, refers to the optimum solution of the parameter estimation problem. Although it is impossible to construct an algorithm that will find a global minimum for an arbitrary function in deterministic time, some adaptation algorithms are better than others in finding the global minimum of the error surface, in terms of escaping local minima. The ability to find the global minim is dependent on the filter structure and number of parameters. Filters with fewer parameters are generally easier to adapt to the optimum solution, but this solution may not be able to provide good estimates although its optimal for that particular filter.

There are two main approaches to derive a recursive adaptive filtering algorithm: the method of stochastic gradient descent, and the method of least squares.

4.4.1 Stochastic Gradient Descent

Gradient decent algorithms are first order optimization algorithms that attempts to minimize the cost function $J(\mathbf{w})$ by changing its parameters, denoted by the

vector $\mathbf{w}$, with respect to the gradient vector of the cost function:

$$\nabla J(\mathbf{w}) = \left[\frac{\delta J}{\delta w_1}, \frac{\delta J}{\delta w_2}, ..., \frac{\delta J}{\delta w_L}\right]^T \tag{4.4}$$

For example, consider the method of steepest descent, where the successive adjustments applied to the parameter vector $\mathbf{w}$ are in the direction of the steepest descent of the error plane. Hence, in a direction opposite to the gradient vector $\nabla J(\mathbf{w})$. Now, reconsider the example of a FIR Wiener filter and limiter, as illustrated in figure 4.3. If the inputs to the filter are nonstationary or their statistic properties are otherwise not completely determined, a stochastic cost function can be employed. One could for example apply the method of steepest descent to the cost function of the instantaneous squared value of the error signal: $J(\mathbf{w}) = \frac{1}{2}e^2[k]$. Doing this recursively is called the least mean square (LMS) algorithm. There exist a platter of algorithms based on these method, with the general update rule:

$$\hat{\mathbf{w}}[k+1] = \hat{\mathbf{w}}[k] + \mu[k]\theta[k] \tag{4.5}$$

Where the vector $\mu[k]$ is a scaling parameter determining the iterative step size in a direction, with respect to the gradient vector, determined by θ_k. The stochastic gradient algorithms differs in their derivation of $\mu[k]$ and $\theta[k]$.

4.4.2 Least Squares

The method of least squares, developed by Gauss, minimizes a cost function of the sum of weighted error squares:

$$J(\mathbf{w}) = \sum_k \lambda[k]|\mathbf{s}[k] - \phi(\hat{\mathbf{w}}[k], \mathbf{r}[k])|^2 = \sum_k \lambda[k]|\mathbf{e}[k]|^2 \tag{4.6}$$

Where $\phi(\hat{\mathbf{w}}[k], \mathbf{r}[k])$ is the filter response at time k. Lets once again consider the case of an adaptive FIR filter approximation to the Wiener filter, under the same assumptions as above. The method of recursive least squares (RLS) can be used

to update the filter coefficient in a way that minimizes the sum of squared differences between the inputs and their corresponding modelled values. The update algorithm is on the form:

$$\hat{\mathbf{w}}[k+1] = \hat{\mathbf{w}}[k] + \mathbf{g}[k]\alpha[k] \tag{4.7}$$

Where $\mathbf{g}[k]$ is the "gain vector", which determines how the current parameters should be changed with respect to the "new" information $\alpha[k]$ that entered the system at time-step k. There exists a large family of algorithms based on the RLS, with different approach to deriving $\mathbf{g}[k]$. To the best of the authors knowledge $\mathbf{e}[k] = \mathbf{s}[k] - \phi(\hat{\mathbf{w}}[k], \mathbf{r}[k])$ is always used as α_k. A common trait of RLS based algorithms is their use of second moment statistics of the observation process $\mathbf{r}[k]$ in their derivation of the gain vector. More specifically the cross-correlation of the correlation matrix of the input vector $\mathbf{x}_k$, and the variance of the desired filter response $\mathbf{s}[k]$. When choosing a RLS based algorithm, some key aspects should be taken under consideration:

- Prior information about the statistic properties of the input process.
- Characteristics of the filter structure.
- Computational effort with respect to number of parameters. This is generally quadratic, with some popular algorithms having cubic complexity.

For example: a linear filter with statistically independent Gaussian distributed inputs calls for the Kalman filter, its equivalent RLS or the information filter. The use of second moment statistics facilitates rapid convergence rate at the expense of increased computational complexity, when compared to stochastic gradient descent based algorithms. Conversely, these does not make any statistical assumptions about $r[k]$.

4.5 Choice of filter structure

In this sections the choice of filter structure will be justified. The goal is to find a structure that minimizes SER while attaining feasibility in terms of computational complexity. When the impulse response is long, the symbol decision of the MLSE has too long delay to be efficient in a DFE structure [Bayesian DFE mobile]. Hence, the two general structures under consideration is the DFE, and the adaptive MLSE algorithm. These structures have different key characteristics:

- The MLSE structure is an optimal estimator for stationary gaussian channels. Thus, the adaptive MLSE performance relies on the adaptive forward filters ability to render the MLSE input stationary. The performance is therefore highly dependent on the performance of the adaptive receive filter when the channel has rapid time-variant behaviour [10].
- The MLSE, void the adaptive receive filter, has exponential complexity on the form $2(M^N)$, where M is the modulation order, e.g. alpabet size, and N is the discrete time length of the impulse response of the channel, in number of sample periods. For the channel modelled in chapter 3, $N \approx 10$. On the other hand, the DFE type filter typically has more parameters than the MLSE adaptive receive filter. The complexity of the adaptive filters, both the DFE and the MLSE receive filter, depends on the number of parameters and the adaptation algorithm. RLS based algorithms typically has a complexity on the form L^2, where L is the number of parameters.
- The DFE structure is more robust against such rapid time-variant behaviour [11].

The DFE structure was chosen due to its superior scalability in terms of modulation order, as well as the reported superior performance for channels with rapid time-variant behaviour. A simple DFE with FIR structure in both feed-forward filter and feedback filter will be used in the experiments in chapter 5.

4.6 Artificial Neural Network

Instead of conventional FIR or IIR structures, the DFE structure can be implemented as a nonlinear filter modelled as an artificial neural network [12]. This is an obvious advantage, since such a filter can capture the characteristics of an actual transmission system, which is inherently nonlinear. The filter structure should ideally be robust enough to capture the characteristics of the channel all scenarios which may occur during UAS communication. Including the general model derived in chapter 3. The number of adjustable parameters in the structure should be minimized since the computational complexity of the adaptation algorithm increase with number of parameters.

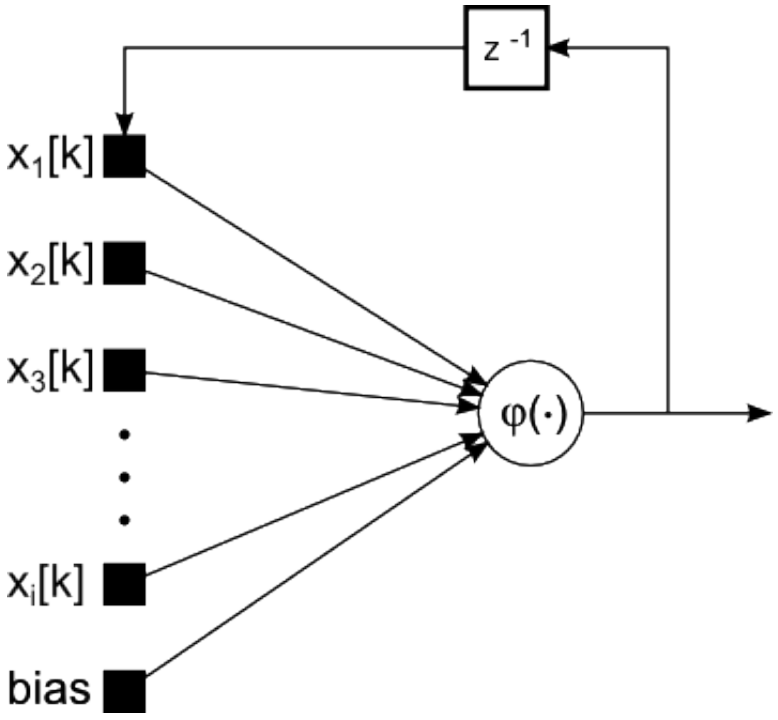

FIGURE 4.3: Illustration of a recurrent neuron

An artificial neural network is made up of interconnected "neurons", loosely attempting to model the way a brain attempts to perform a particular task. The output of these neurons are typically either a weighted sum, or a sigmoid function of the weighted sum, of the neuron inputs. In the following the hyperbolic tangent function is used as the sigmoid function. A recurrent neuron is one with a feedback of the output to the input, as illustrated in figure 4.3. Artificial neural networks

with recurrent neurons are referred to as recurrent neural networks (RNN). The use of feedback enables information of previous states to remain within the network, similar to IIR filters, this reduces the necessary number of adjustable parameters [2]. A recurrent neural networks can in theory be applied as a black box model to any system, with arbitrary accuracy [13]. A black box model is a system that can be characterized by the input-output relationship without any information about the internals of the system. The internals of a neural network is obviously not unknown, but not on the same form as any reasonable channel model. Hence, it is difficult to study the actual channel by studying the neural network, even though the input-output characteristics are similar. In figure 4.3 there is an input denoted "bias". This input has a fixed value of +1 and its magnitude is therefore determined by its corresponding weight. The bias has the affect of applying an affine transformation to the other weighted inputs. Mathematically a recurrent neuron can be described as:

$$y = \phi\left(\sum \mathbf{wx}\right) \tag{4.8}$$

Where y is the output, $\mathbf{x}$ is a vector containing the inputs, $\mathbf{w}$ is a vector containing the corresponding weights (gains) and $\phi(\cdot)$ is, in this case, the hyperbolic tangent function.

An Elman type fully recurrent neural network with three hidden nodes was chosen as the general structure due to its successive application as black box model for similar systems [14] [1] [15]. This is a very compact RNN consisting of a hidden layer (internal layer) of sigmoid neurons [16]. The output layer of the network is the sum of the outputs from the hidden layer and a bias. An illustration of an Elman type RNN is provided in figure 4.4. Since $r[n] \in \mathbb{C}$ (the received signal is complex), the real and imaginary part will be handled separately. This can be justified by the fact that complex numbers are treated as a two dimensional vector by the computer, and will therefore inherently require twice as many computations as a real number would. In practise the calculation of the RNN response of the real and imaginary part may be done in parallel. However, proper implementation of the algorithm, in e.g. C/POSIX, is beyond the scope of this thesis. All simulations

are performed in the MATLAB environment. Source code for the implemented algorithm is appended.

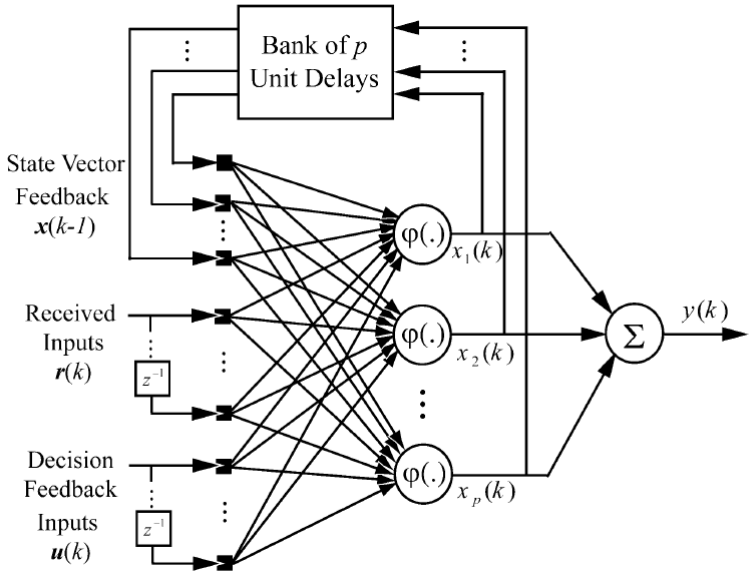

FIGURE 4.4: Illustration of a Elman RNN implementation of a DFE [1], where the bias nodes are omitted for simplicity.

Mathematically the Elman RNN can be described on discrete state-space form:

$$x[k] = \phi(\mathbf{Z}[k], \mathbf{W}_p) \tag{4.9}$$

$$y[k] = f\left(\{[1, \mathbf{x}^T[k]]^T \mathbf{w}_o[k]\right) \tag{4.10}$$

Where:

- $\phi(\cdot)$ is the activation function for the neurons in the hidden layer. $tanh(\cdot)$ is used as activation function here.
- $f(\cdot)$ is the activation function for the output layer. A linear combiner (summation) is used here.
- $\mathbf{Z}[k] = [\mathbf{z}_1[k], ..., \mathbf{z}_p[k]]$ is the matrix containing p equal input vectors for the neurons.
- $\mathbf{z}[k] = [1,\ \mathbf{r}^T[k]],\ \mathbf{u}^T[k]],\ \mathbf{x}^T[k-1]]^T$ is the neuron input vector, containing:
  - The node bias input.

  - The vector of received inputs $\mathbf{r}[k] = [r[k], ..., r[k-m+1]]^T$
  - The vector of previous symbol decisions $\mathbf{u}[k] = [u[k], ..., u[k-n]]^T$
  - The vector of hidden layer feedback vector $\mathbf{x}[k-1]$

- $\mathbf{W}_p[k] = [\mathbf{w}_1[k], ..., \mathbf{w}_p[k]]$ is the matrix containing the weights for the p neurons.

- $\mathbf{w}_o[k]$ is the $1+p$ long output summation node weight vector, including the bias weight.

The parameter vector which contains all the parameters of the RNN is defined as:

- $\mathbf{w}[k] = [\mathbf{w}_1^T[k], ..., \mathbf{w}_p^T[k], \mathbf{w}_o^T[k]]^T$

A DFE is characterized by the integers m, n, p described above, and implicitly the decision delay d. As discussed above, $p = 3$ and $d = 2$ was chosen. The minimal sufficient value of m and n can be calculated by the method described in [11]. That is: $m = d + 1 = 3$ and $n = N + m - d - 2 = 9$. Hence, $L = 2(p(m+n+p+1)+p+1) = 104$. Where the scalar coefficient 2 is due to $r[k]$ being complex.

4.7 Parameter estimation

Adapting the weights of the RNN to minimize MSE can be regarded parameter estimation of a nonlinear system. It is also known as a neural net training. As described above, the concept is to apply a set of inputs with corresponding desired outputs $\mathcal{T}\{\mathbf{r}_d, \mathbf{y}_d\}$, and calculating the error as the difference (MSE) between the actual filter response and the desired response. In practice $\mathbf{r}_d$ corresponds to a sequence of transmitted signal with values and time of transmission that is known to the receiver. This can be implemented by transmitting a predetermined data sequence at predetermined points in time. The necessary length of the sequence

is determined by the convergence time of the adaptation algorithm. The necessary frequency of sequences is determined by how rapid the characteristics of the channel is changing over time. What "necessary" actually constitutes is typically determined by a BER requirement. How the training sequence is implemented to the communication system is beyond the scope of this thesis, but its assumed that the sequence is not part of the payload data. Hence, faster convergence time yields higher data transfer rate.

Modern UAVs typically have an embedded computer with a floating point unit (FPU), such as the pixhawk, to run the autopilot with motion control algorithm, sensor fusion, path planning and so on. It is assumed here that the UAV has a modern microcontroller, for example based on the ARM-M4, with a FPU. Hence, real-time implementing a RLS adaptation algorithm is not a problem in terms of reaching deadlines. In other words the available floating point operations per second (FLOPS) is sufficient if the parameter estimation task is given highest priority by the scheduler and may pre-empt other tasks. This assumption means that it is possible to implement RLS based parameter estimation methods, but the higher computational complexity may impair other tasks that the microcontroller has to perform. This must be taken under consideration when implementing the algorithm on an embedded system, but is not considered any further here. Under these assumptions, RLS based methods are superior to stochastic gradient based methods, such as backpropagation through time or real-time recurrent learning. Both due to the higher convergence rate and the vanishing gradient problem [1]. It was therefore decided to use RLS methods for both the FIR based DFE filter, and the RNN.

Among the RLS methods considered for the RNN, are the nonlinear kalman filter algorithms. Mainly because of their successful application to similar problems [1], but also due to the extensive literature on using these filters for neural net parameter estimation [17]. Lets first describe the RNN behaviour in a way that is suitable for Kalman filter formulation, using the notation of [15]

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \omega[k] \tag{4.11}$$

$$\mathbf{y}_d = h(\mathbf{w}[k], \mathbf{z}[k]) + \upsilon[k] \tag{4.12}$$

Where $\mathbf{w}[k]$ is the L-by-1 weight vector describing the state of the system (RNN), and $\mathbf{z}[k]$ is a 90-by-1 input vector, aggregated from the three real and three imaginary hidden node input vectors described above. The process equation (4.11) describes the state of the system when characterized as a stationary process corrupted by process noise $\omega[k]$. The measurement equation (4.12) describes the desired system output $\mathbf{y}_d$ as a nonlinear function of the weight vector $\mathbf{w}$ and the input vector $\mathbf{z}[k]$; this equation is augmented by random measurement noise $\upsilon[k]$. $\mathbf{y}_d$ corresponds to $\mathbf{s}[k-d]$ when the filter parameters are being adapted (in training mode), and $\hat{\mathbf{s}}[k-d]$ otherwise.

The general concept of RLS methods was discussed above. The Kalman filter is a RLS method that uses an Bayesian approach to determined the optimal recursive estimation, in the MMSE sense, of the state (in this case $\mathbf{w}$) given the observations up to the current time. Mathematically, this can be stated as [17]

$$\hat{\mathbf{w}}[k] = \mathbb{E}[\mathbf{w}[k]|Y_0[k]] \tag{4.13}$$

Where $Y_0[k]$ is the sequence of observations up to time k. To determine this expectation, or any "best estimator" under non-MMSE criteria, requires knowledge of the *a posteriori* density $p(w[k]|Y_0[k])$. This may be determined recursively using the Bayesian approach:

$$p(\mathbf{w}[k]|\mathbf{Y}_0[k]) = \frac{p(\mathbf{w}[k]|\mathbf{Y}_0[k-1])p(\mathbf{y}[k]|\mathbf{w}[k])}{p(\mathbf{y}[k-1]|\mathbf{Y}_0[k-1])} \tag{4.14}$$

Where

$$\mathbf{w}[k]|\mathbf{Y}_0[k-1]) = \int p(\mathbf{w}[k]|\mathbf{w}[k-1])p(\mathbf{w}[k-1]|\mathbf{Y}_0[k-1])d\mathbf{w}[k-1] \tag{4.15}$$

and the normalizing constant $(\mathbf{y}[k]|\mathbf{Y}_0[k-1])$ is given by:

$$\mathbf{w}[k]|\mathbf{Y}_0[k-1]) = \int p(\mathbf{w}[k-1]|\mathbf{Y}_0[k-1])p(\mathbf{y}[k]|\mathbf{w}[k-1])d\mathbf{w}[k] \tag{4.16}$$

This recursion specifies the current state density as a function of the previous density and the most recent measurement data. The state space model into play by specifying the state transition probability $p(\mathbf{y}[k]|\mathbf{w}[k-1])$ and measurement probability $p(\mathbf{y}[k]|\mathbf{w})$. Specifically, $p(\mathbf{y}[k]|\mathbf{w}[k-1])$ is determined by the process noise density $p(\omega[k])$, with the state update equation

$$\mathbf{w}[k-1]) = \phi(\mathbf{w}[k], \mathbf{u}[k], \omega[k]) \tag{4.17}$$

This reduces to (4.11) when there is no exogenous process input $\mathbf{u}[k]$, and the noise is additive. Similarly, $p(\mathbf{y}[k]|\mathbf{x}[k])$ is determined by the observation noise density $p(\upsilon[k])$ and the measurement equation (4.12). Thus, knowledge of these densities and the initial condition $p(\mathbf{w}_0|\mathbf{y}_0) = p(\mathbf{y}_0|\mathbf{w}_0)p(\mathbf{w}_0])/p(\mathbf{y}_0)$ determines $p(\mathbf{x}[k]|\mathbf{Y}_0[k]) \; \forall \; k$.

For most systems it is intractable to derive a closed-form solution for the multi-dimensional integration indicated by (4.14) - (4.16). The only general approach is to apply Monte Carlo sampling techniques that essentially convert integrals to finite sums, which converge to the true solution in the limit. The particle filter [18] and the unscented particle filter [19] are examples of this approach. These estimators have very high computational complexity [20], and will not be discussed further here. The Bayesian recursion can be greatly simplified under the assumption of Gaussian distributed densities. In this case, only the conditional mean $\hat{\mathbf{w}}[\mathbf{k}] = \mathbb{E}[\mathbf{x}[k]|\mathbf{Y}_0[k]]$ and covariance $P_{\mathbf{w}}[k]$ needs to be evaluated. This leads to the recursive estimation, equal to that in (4.7).

$$\hat{\mathbf{w}}[k] = (\textit{prediction of } \hat{\mathbf{w}}[k]) + \mathcal{K}[k](\mathbf{y}_d[k] - (\textit{prediction of } \mathbf{y}[k])) \tag{4.18}$$

$$\mathbf{P}_{\mathbf{w}}[k] = \mathbf{P}_{\mathbf{w}}^{-}[k] - \mathcal{K}[k]\mathbf{P}_{\tilde{\mathbf{y}}}[k]\mathcal{K}^{T}[k] \tag{4.19}$$

The optimal terms in this recursion are given by

$$\hat{\mathbf{w}}^{-}[k] = \mathbb{E}[\mathbf{w}[k] + \omega[k]] \tag{4.20}$$

$$\mathcal{K}[k] = \mathbf{P}_{\mathbf{x}[k]\mathbf{y}_d[k]}\mathbf{P}^{-1}_{\mathbf{e}[k]\mathbf{e}[k]} \tag{4.21}$$

$$\hat{\mathbf{y}}^{-}[k] = \mathbb{E}[\phi(\mathbf{w}[k], \mathbf{z}[k]) + \upsilon[k]] \tag{4.22}$$

Where the optimal prediction (prior mean) of $\mathbf{w}[k]$ is written $\hat{\mathbf{w}}^{-}[k]$ and corresponds to the expectation of $\mathbf{w}[k-1]$, with similar interpretation for $\hat{\mathbf{y}}^{-}[k]$. The optimal gain term is a function of the posterior covariance matrices with $\mathbf{e}[k] = \mathbf{y}_d - \hat{\mathbf{y}}[k]$.

As mentioned above, the Kalman filter [21] calculates the optimal terms of (4.20) - (4.22) in the case of a linear model. In the case of a nonlinear model, the Kalman filter is no longer optimal, and a *nonlinear* Kalman filter should be used instead. There exists several nonlinear versions of the Kalman filter. The simplest being the extended Kalman filter (EKF), and the most complex are the cubature Kalman filters. EKF is the Kalman filter applied to a local linearization of the descrete state space model under consideration. In this case (4.9) and (4.10). The linearisation is performed with a first order Taylor approximation. Higher order Taylor approximation may be used, but the increased accuracy rarely justifies the increased computational complexity, since there exists nonlinear Kalman filters at this complexity class that typically yields better performance. The approximation of the optimal terms may lead to large errors in the true posterior mean of the transformed Gaussian variable, which may lead to suboptimal performance and sometimes divergence of the filter.

Another nonlinear interpretation of the Kalman filter, which attempts to address the flaws of the EKF, is the unscented Kalman filter (UKF) [22], which instead of linearising the state space model, samples the state distribution in a deterministic way (as opposed to the particle filter). The sample points are propagated through the *true* nonlinear model and used to calculate the posterior mean and covariance accurately to the second order Taylor approximation of any nonlinear system. The UKF uses the unscented transform (UT) to perform the transformation of the random variable, the algorithm [22] is presented below for convenience. The general idea is to find a minimal number of samples that accurately captures the statistics of the random variable. The typical method is to us the mean vector and two conjugated vectors for each dimension of the random variable. Consider

propagating a random variable x with dimension L through a nonlinear function $y = f(x)$. Assume x has a mean $\bar{\mathbf{x}}$ and covariance $\mathbf{P_x}$. To calculate the statistics of y, a vector $\mathcal{X}$ of $2L+1$ *sigma vectors* $\mathcal{X}_i$ are formed in the following way:

$$, \mathcal{X}_0 = \bar{\mathbf{x}} \tag{4.23}$$

$$\mathcal{X}_i = \bar{\mathbf{x}} + (\sqrt{(L+\lambda)P_{\mathbf{x}}})_i, \qquad i = 1, ..., L \tag{4.24}$$

$$\mathcal{X}_i = \bar{\mathbf{x}} - (\sqrt{(L+\lambda)P_{\mathbf{x}}})_i, \qquad i = L+1, ..., 2L \tag{4.25}$$

Where $\lambda = \alpha^2(L+\kappa) - L$ is a scaling parameter. The constant α determines the spreading of the sigma points around the mean, and is usually set to a small positive number (e.g. $10^{-4} \leq \alpha \leq 1$). The constant κ is a secondary scaling parameter that may also be used to determine the sigma point spreading, whileβ is used for taking advantage of prior knowledge of the distribution of $\mathbf{x}$. It has been shown that $\beta = 2$ is optimal for Gaussian distribution. The sigma vectors are propagated through the nonlinear function

$$\mathcal{Y}_i = f(\mathcal{X}_i), \qquad i = 0, ..., 2L \tag{4.26}$$

and the mean and covariance of $\mathbf{y}$ are approximated using a weighted sample mean and covariance of the posterior sigma points

$$\bar{\mathbf{y}} \approx \sum_{i=0}^{2L} W_i^m \mathcal{Y}_i \tag{4.27}$$

$$\mathbf{P_y} \approx \sum_{i=0}^{2L} W_i^c (\mathcal{Y}_i - \hat{\mathbf{y}})(\mathcal{Y}_i - \hat{\mathbf{y}})^T \tag{4.28}$$

with the weights W_i given by

$$W_0^m = \frac{\lambda}{L+\lambda} \tag{4.29}$$

$$W_0^c = \frac{\lambda}{L+\lambda} + 1 - \lambda^2 + \beta \tag{4.30}$$

$$W_0^m = W_0^c = \frac{\lambda}{2(L+\lambda)}, \qquad i = 1, ..., 2L \tag{4.31}$$

A comparison of the UKF and the EKF for a similar parameter estimation problem was conducted in [Haykin, 2004]. It concluded that the UKF typically has slightly better performance than the first order EKF in terms of tracking and misadjustment, and similar performance in terms of convergence rate and robustness. In that experiment the matrix square root in (4.24) and (4.25) was done with lower-triangular Cholesky factorization (`chol`() in MATLAB). This method, as well as other matrix inversion methods such as QR, yields cubic complexity ($\mathcal{O}(L^3)$), which is significantly higher than the quadratic complexity of the EKF. However, the square root implementation of the UKF [23] (`cholupdate`() in MATLAB), claims to yield similar performance with quadratic complexity ($\mathcal{O}(L^2)$).

Kalman filters experiences numerical stability issues known as the *divergence phenomenon* [24]. This manifests by the covariance matrix of the system state $\mathbf{P_w}$ being rendered *not* non-negative definite, which obviously is unacceptable. This is a result of the fact that $\mathbf{P_w}$ is defined as the difference between two matrices, and the calculation is performed with finite length arithmetic. Techniques to remedy this problem includes

- square root filtering.
- eigen decomposition of $\mathbf{P_w}$ to ensure positive eigenvalues.
- adding artificial noise to $\mathbf{P_w}$.

The latter technique is especially useful for nonlinear Kalman filters, since they tend to underestimate the true covariance matrix. For the case of neural net parameter estimation it has been shown that artificial noise also accelerates the convergence rate, both with EKF and UKF [1]. The numerical stability issues tends to increase with the dimension of the state vector, and for large dimensions the artificial noise technique may not be sufficient to retain non-negative definite $\mathbf{P_w}$. In that case, it may be used in combination with one of the other techniques.

To summarise with respect to the list in section 4.4, the EKF and UKF have similar performance for convergence rate, robustness, structure and numerical properties.

They differ in terms of computational complexity, where the EKF is slightly better, and tracking, ability to find global minimum [25] and misadjustment where the UKF is slightly better. The UKF was chosen as the adaptation algorithm in the following simulations.

Chapter 5

Experiments

In this chapter the performance of the filters presented in chapter 4 will be tested on the model presented in chapter 3. The goal of these experiments is to determine whether system identification with a black block approach is suitable for joint optimization of the data transfer rate and the reliability of an UAS communication system. As discussed in the previous chapter, literature in the field suggests that the DFE structure is most suitable for channels with rapid time-variant behaviour, and the filters under consideration are two DFE filters realized with FIR structure, and recurrent artificial neural network. The FIR version uses the RLS algorithm for parameter estimation, with forgeting factor $\lambda = 0.99$ (4.6). The neural net uses the UKF. In theory the FIR DFE has less expressive power, in the sense that the accuracy in which it can identify the optimal filter is worse than the neural net. On the other hand, it has only 12 parameters $(m + n)$ to be estimated, which is far less then the 104 parameters in the neural net. The source code used in these experiments are appended.

All the experiments in this section was conducted in the same general manner: A set of inputs, generated from the channel model described in chapter 3, with the corresponding ideal responses, was a applied to the parameter estimator. The ability to converge to the optimal solution for different QAM modulation orders was studied. Recall the optimal solution is any filter that result in no erroneous symbol estimates.

The parameters was initialized to random values between -1 and 1, using the `rand()` function in MATLAB, which has flat distribution. This makes sense since the expected average value of the filter outputs is zero. The transmitted data sequence, i.e. the input to the channel model, was also generated with a flat distributed random integer generator (`randi()` in MATLAB).

5.1 RNN with UKF

The first experiment conducted on the neural net endeavoured to determine the required length of training set to converge to the optimum solution when 16-, 4- and 2-QAM modulation was used. 2-QAM is equivalent to BPSK (2-PSK). A training set of ten thousand inputs and desired outputs was used in twenty independent experiments. The convergence characteristics is depicted below for the different modulation orders. These experiments are performed without adding random noise to the channel output. That is, $v[k] = 0,\ \forall k$. The y-axis is the sum of absolute error of the real and imaginary component of the error, $Re\{e[k]\} + Im\{e[k]\}$.

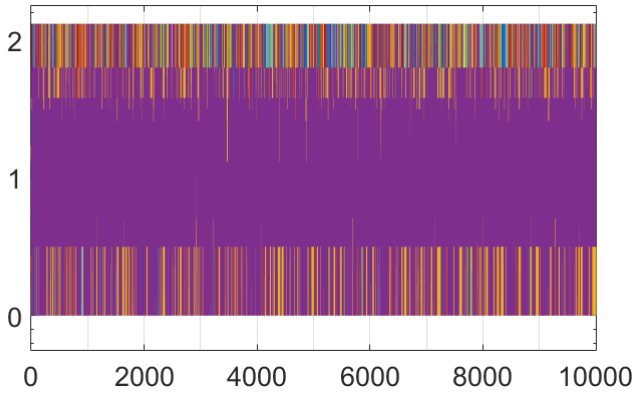

FIGURE 5.1: Convergence of the RNN with UKF for 16-QAM modulation.

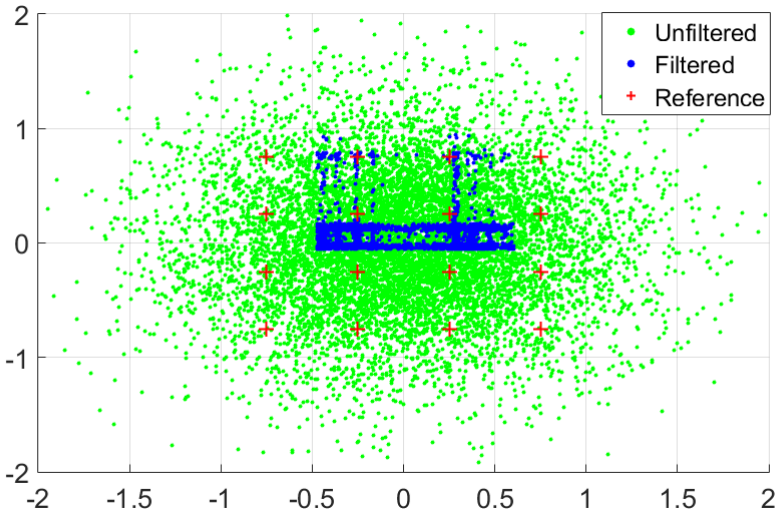

FIGURE 5.2: Constellation of the RNN with UKF for 16-QAM modulation.

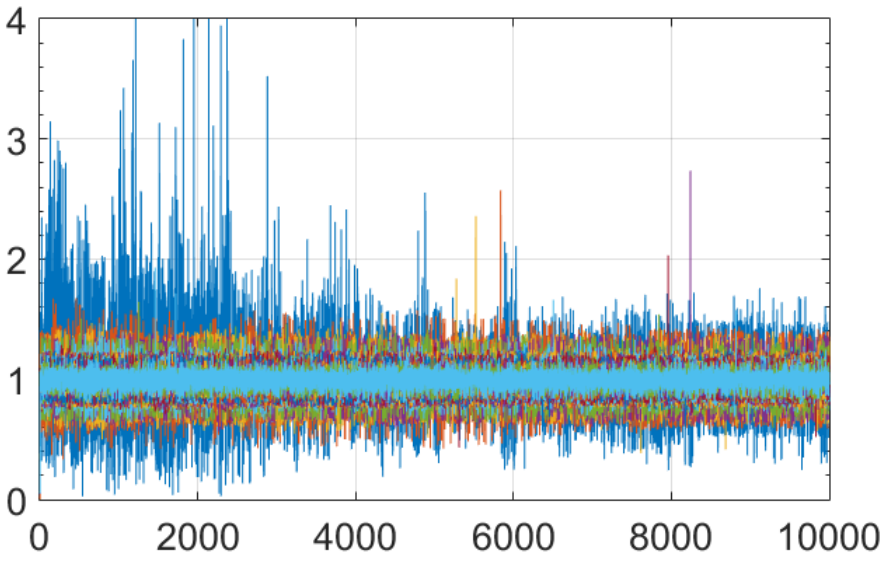

FIGURE 5.3: Convergence of the RNN with UKF for 4-QAM modulation.

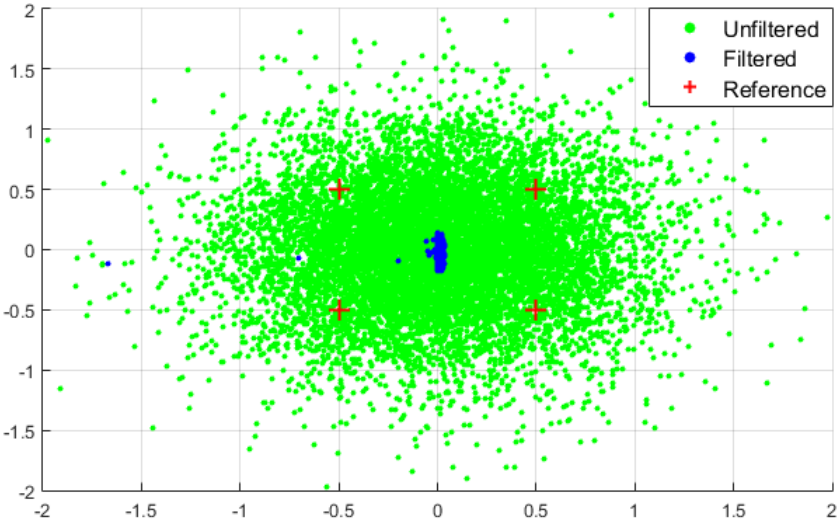

FIGURE 5.4: Constellation of the RNN with UKF for 4-QAM modulation.

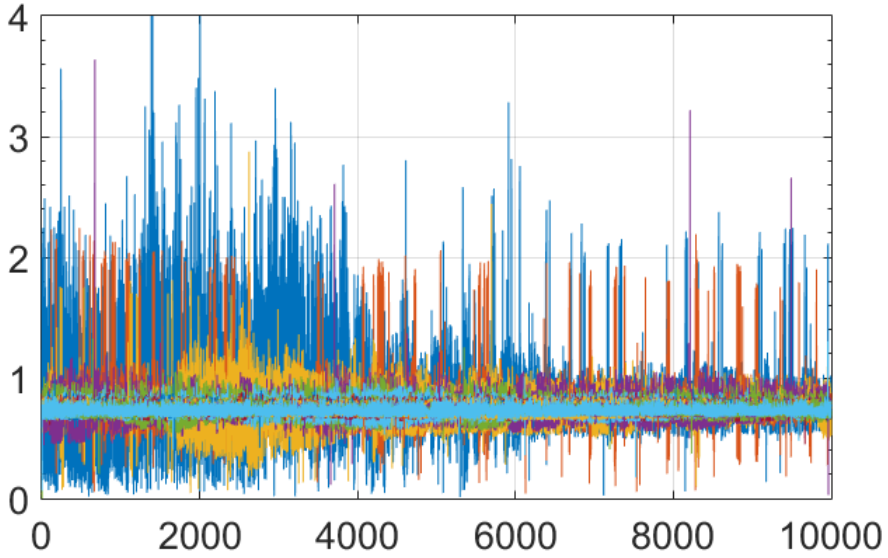

FIGURE 5.5: Convergence of the RNN with UKF for 2-QAM modulation.

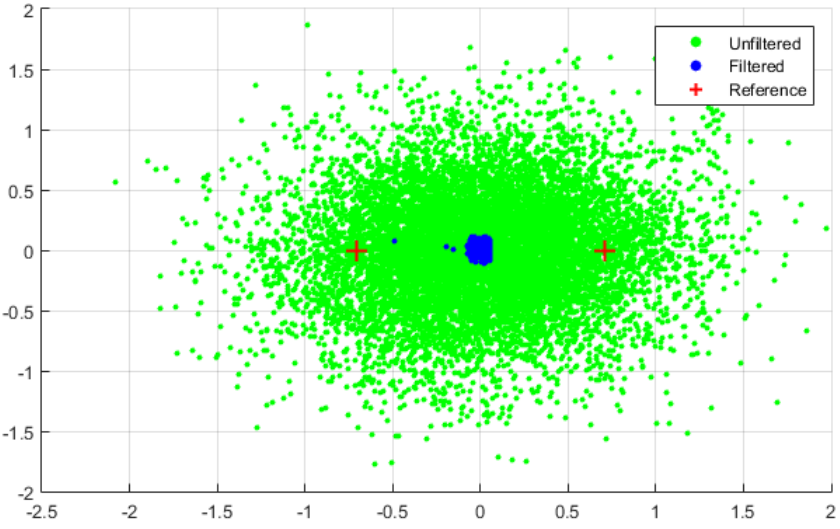

FIGURE 5.6: Constellation of the RNN with UKF for 2-QAM modulation.

As seen above this RNN with UKF converged to the same local minimum for all experiments. In this local minimum, the filter produces estimates along the axes. This result simply means that the filter converged to an estimate of the expectation of the average value of the set. Intuitively, this is an easy to find local minima for the algorithm which is trying to find the least sum of squared errors.

5.2 FIR DFE with RLS

The same experiments that was done on the neural net was conducted on the FIR DFE filter: QAM modulation with modulation order 16, 4 and 2 was used in fifty independent experiments with set length of ten thousand.

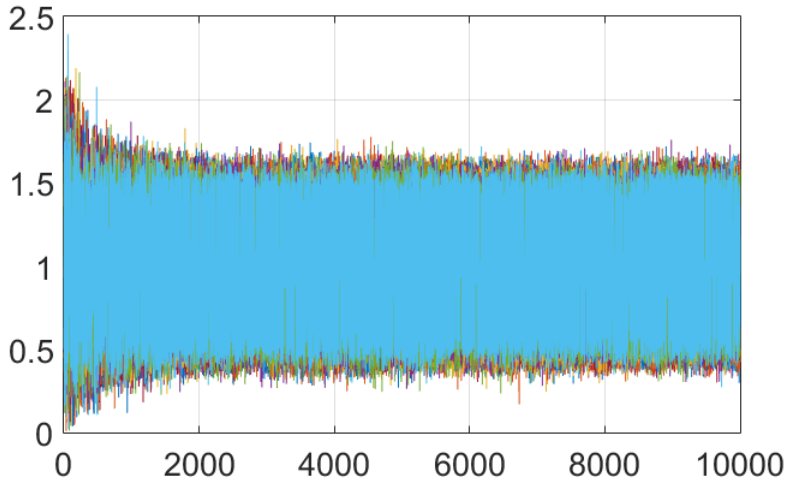

FIGURE 5.7: Convergence of the FIR DFE with RLS for 16-QAM modulation.

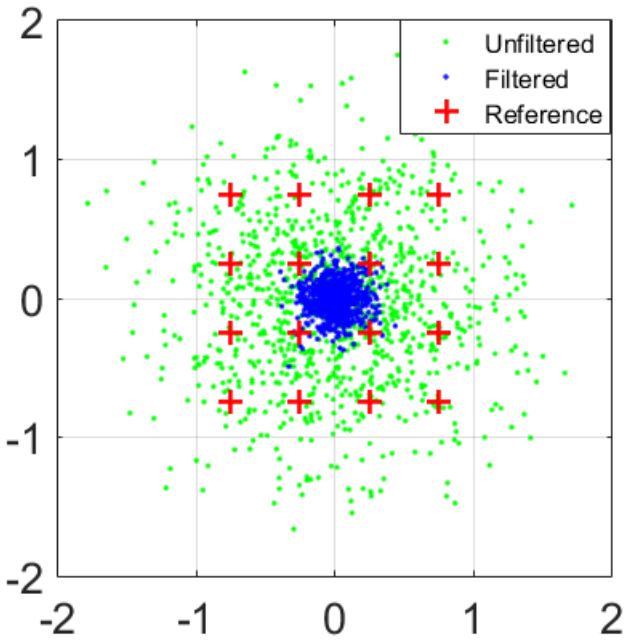

FIGURE 5.8: Constellation of the FIR DFE with RLS for 16-QAM modulation.

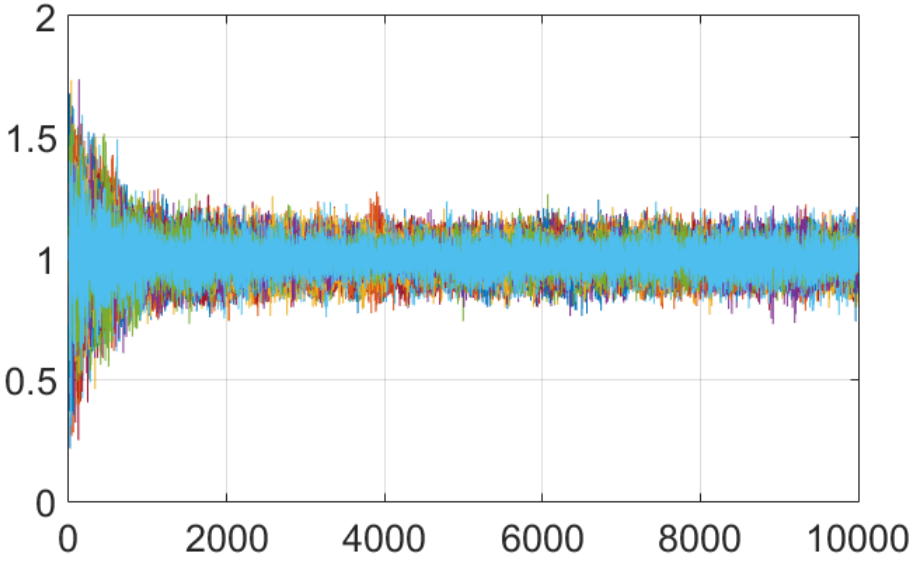

FIGURE 5.9: Convergence of the FIR DFE with RLS for 4-QAM modulation.

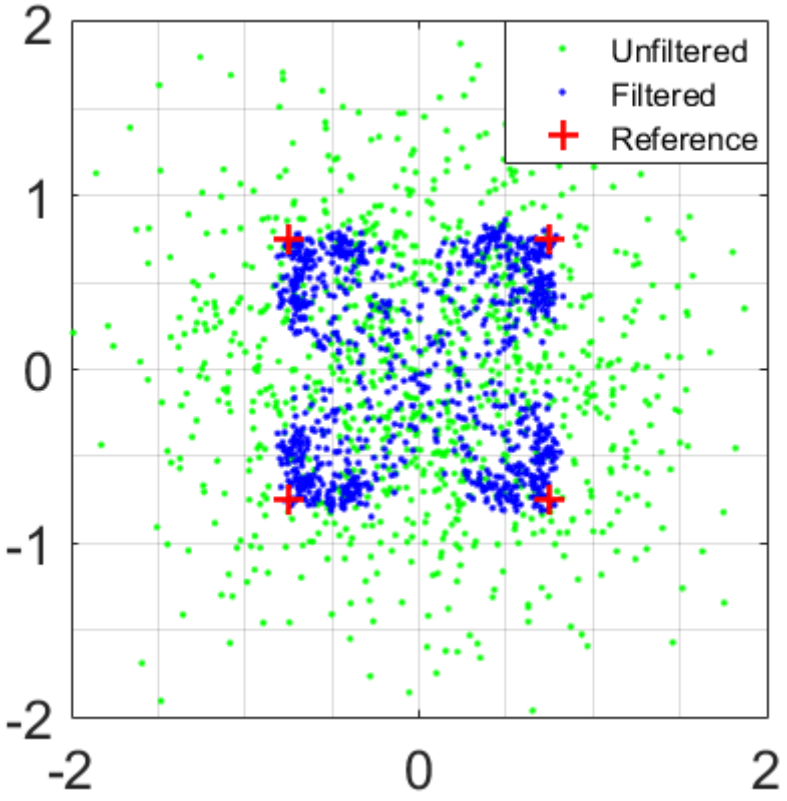

FIGURE 5.10: Constellation of the FIR DFE with RLS for 16-QAM modulation.

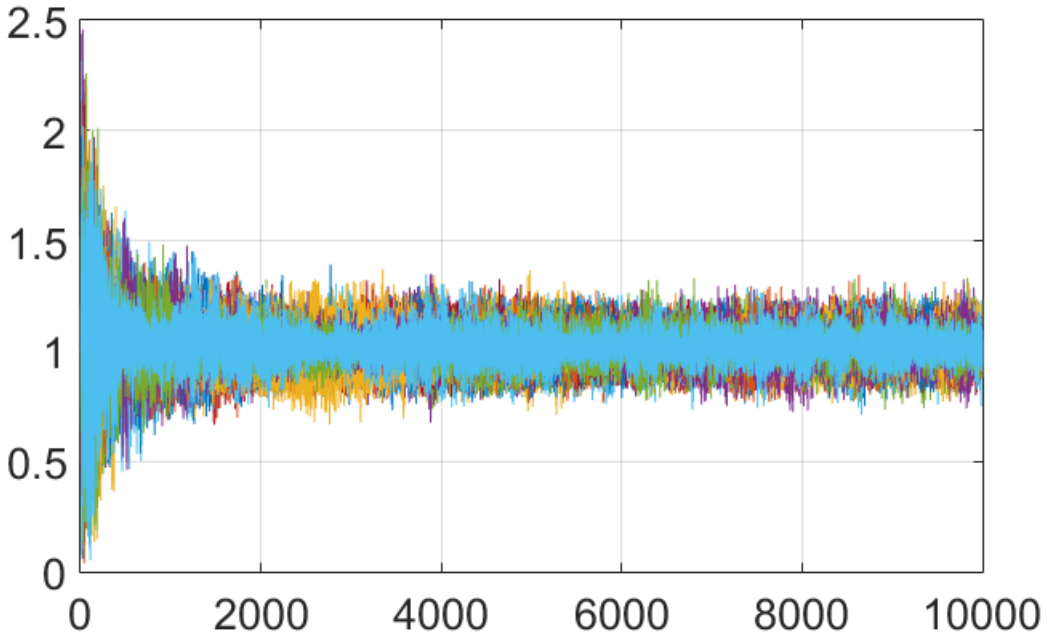

FIGURE 5.11: Convergence of the FIR DFE with RLS for 2-QAM modulation.

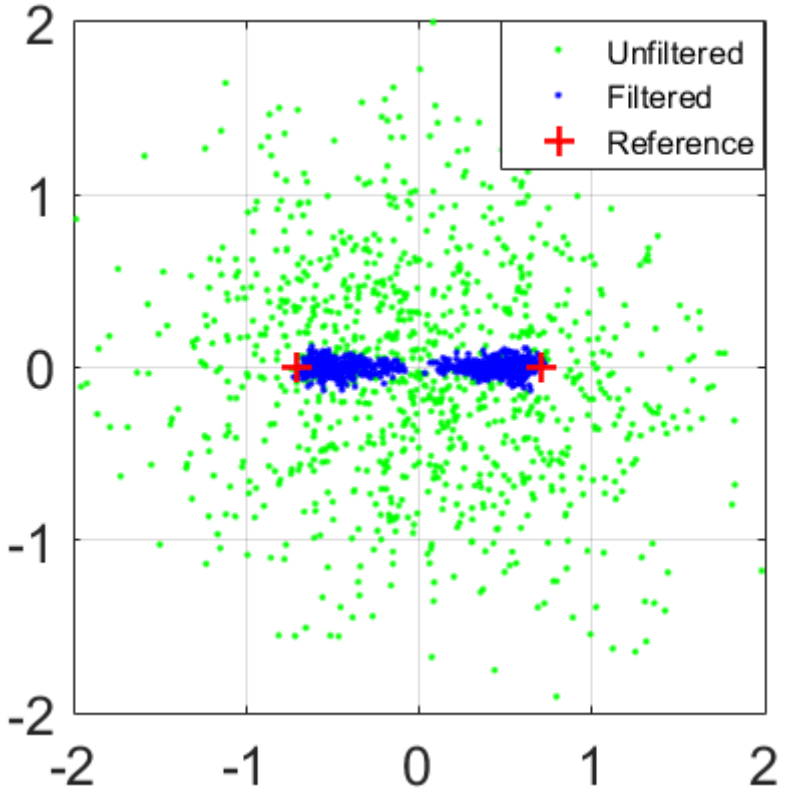

FIGURE 5.12: Constellation of the FIR DFE with RLS for 2-QAM modulation.

Some key observations can be made from these experiments:

- None of the experiments was successful in converging to the optimal solution.
- A training set of 2000 symbols seems to be sufficient.
- The FIR DFE with 2-QAM and 4-QAM modulation seems to successfully filter the signal.
- The neural net does not seem to be able to filter the signal successfully for any of the experiments.

5.3 SER improvement

The experiments described above indicates that the FIR DFE filter with low modulation order are most suited for the channel model under consideration here. In the following experiments, the symbol error rate improvement will be quantified. The filter will first use a 2000 samples long training set to converge, and the parameters will then be held constant. A set of ten thousand symbols will then be applied to both the filter, and directly to the data estimator/limiter. The errors of the estimates from the filtered and unfiltered sets will then be compared, for different SNR. The SER is averaged over 20 independent experiments for each SNR value.

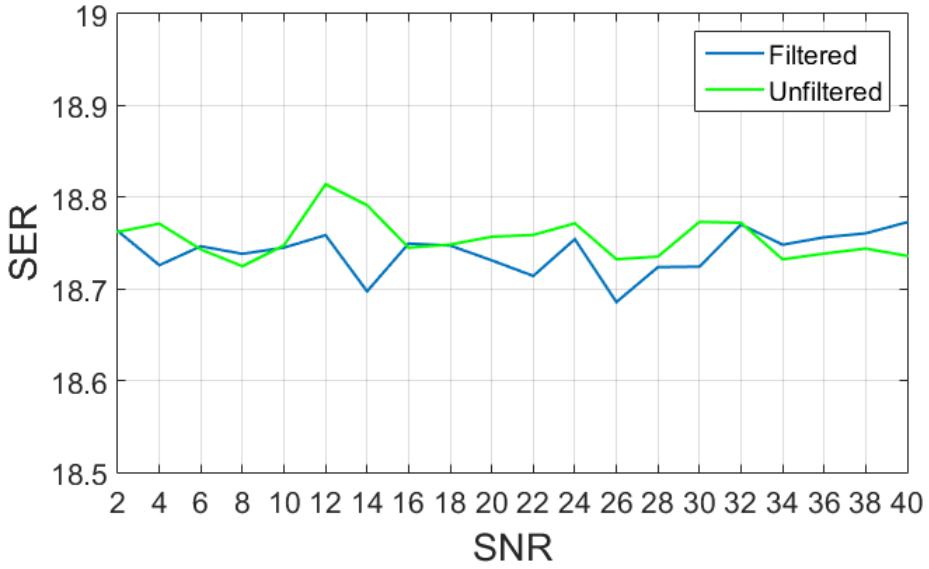

FIGURE 5.13: SER performance, in percent, for different SNR of the FIR DFE with RLS for 4-QAM modulation.

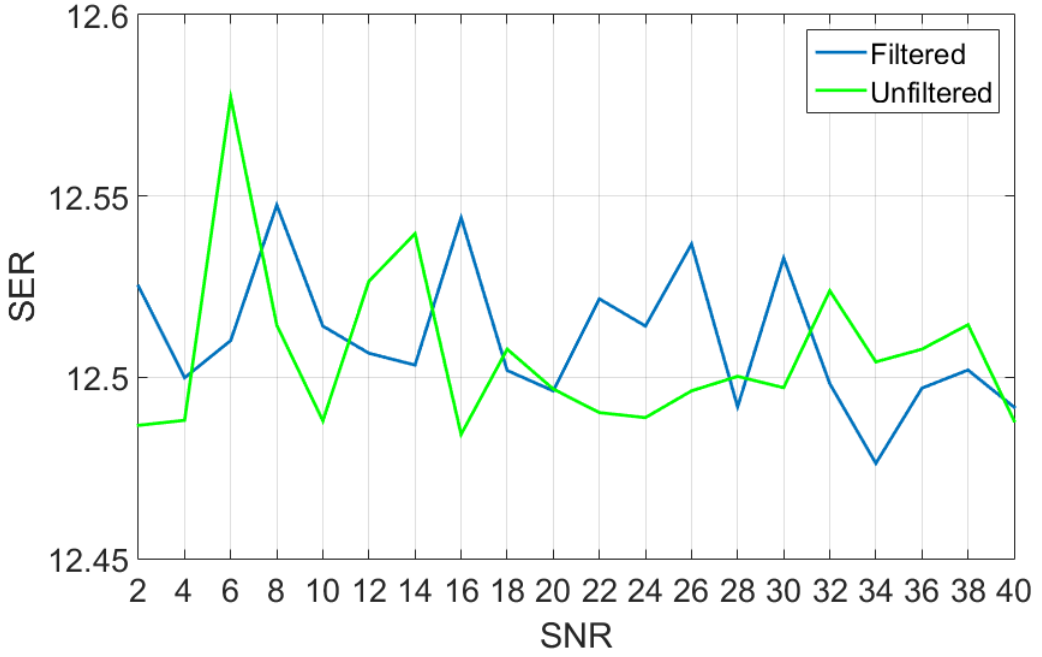

FIGURE 5.14: SER performance, in percent, for different SNR of the FIR DFE with RLS for 2-QAM modulation.

Two key observations can be made from these experiments:

- The filter did not improve the SER, even though it was able to produce estimate in groups close to the reference values. Hence, these estimates was

mostly wrong.

- The signal distortion introduced by AWGN does not affect the SER. This implies that the distortion is dominated by superposition of adjacent symbols (ISI).

5.4 Global optimization

Since the result in the experiments using the neural net was negative, an experiment using a simple global optimization technique was conducted to ensure that the problem is in fact due to nonconvex optimization. This technique is to converge to a local minimum and save the corresponding parameters and error. If the local minimum is not the global minimum, the procedure was repeated with new initial parameters and the error at this minimum was compared with the previous. This type of brute force optimization was not expected to converge to the global optimal solution within reasonable time. Meaning that the convergence time is far longer than what would be tractable for being used for on-line parameter estimation on an UAV. Therefore the experiment was conducted with a channel model that generates a training set which should present a much easier parameter estimation challenge. The channel model has been used to produce positive results in literature [1] [15]. The transfer function of the channel model is s a third order FIR filter on the form:

$$C(z) = (c_0 + a_0[k]) + (c_1 + a_1[k])z^{-1} + (c_2 + a_2[k])z^{-2} \tag{5.1}$$

Where $\mathbf{c} = [c_0 \ c_1 \ c_2]^T = [0.3482 \ 0.8704 \ 0.3482]^T$ and the coefficients $a_i[k](i = 0, 1, 2)$ are generated by a second order Markov model. Where an AWGN source drives a secound-order Butterworth low-pass filter with cut-off frequency 0.1. This random-walk process is meant to simulate random time-variant behaviour of the channel. The neural net with the UKF was used in this experiment. The parameters of the neural net was chosen in the same manner as described in chapter 4. Hence, $p = 3$, $d = 1$ $m = 2$ and $n = 2$. This results in a naural net with 50 parameters. Training sets with length of ten thousand symbols where used. Figure 5.15 shows the result after 800 training sets, where the a close approximation to the global optimal solution is achieved.

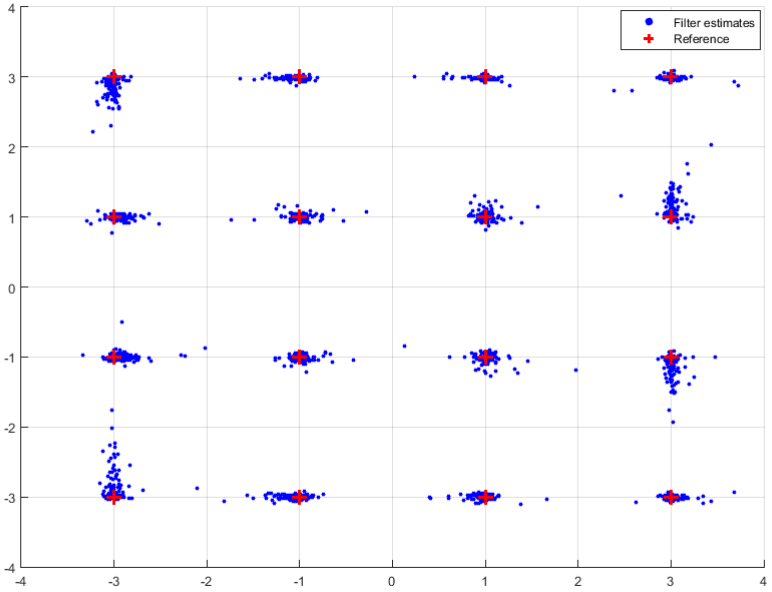

FIGURE 5.15: Illustration of a Elman RNN implementation of a DFE [Haykin paper], where the bias nodes are omitted for simplicity.

This simulation was performed with an Intel Core i7 Ivy Bridge-E processor. The simulation time was more than five hours. This is obviously utterly unacceptable for on-line applications. Reason implies that the results on the channel model derived in chapter 3 would be even worse, since this channel model introduces far more distortion to the signal. Although better techniques for initializing the parameters between training sets might yield better performance, global optimization techniques was not studied any further.

Chapter 6

Discussion And Recommendations For Further Work

The bulk of the work described in this thesis, focused on finding a filtering techniques to enable reliable communication with minor impairs to the data transfer rate for the UAV under consideration here. This is a very ambitious endeavour, which unfortunately yielded negative results despite great effort. The degree of non-convexity of the filtering problem perplexed the author, since high profile scientists in the field had reported overwhelmingly positive results for the filtering techniques in similar problems [26] [1] [15] [14] [11] [27] [28]. Consequently, great effort was put towards verifying the results, which in the end are trusted to be accurate. The reason for the negative results is contributed to the use of minimal time period between transmitted symbols. This is a fundamental characteristic in the channel model described in chapter 3. Theory implies that the signal distortion from ISI could be reduced by adjustable time spacing between symbols, called guard periods. By seting the guard period equal to the length of the impulse response of the channel, ISI would be completely eliminated. Experiments on the effect of varying the length the guard period would have been very interesting and probably quite useful since the length of the channel is time-varying. However, the

author was unable to incorporate adjustable guard periods to the channel model while retaining its accuracy in regards to the other characteristics it must capture. A representative dataset of transmitted and received signals, collected from an actual UAV communication system could have been used to remedy this problem. Either to fit a channel model to the dataset, or simply to employ it directly to the simulations. Nevertheless, the expectation of limited data transfer rate is taken into consideration by the UAV development team.

6.1 Recommendations For Further Work

1. As mentioned, the channel model should be augmented to incorporate guard period as an adjustable parameter,
2. and the model should be validated with a dataset.
3. Under the assumption that the guard periods amends the non-convexity problem to a satisfactory degree, the filter algorithms should be tested on a real radio system. Obviously not in any critical applications, but rather in a representative test scenarios using software defined radio.
4. The effect of error correcting codes should be evaluated with respect to other trade off mechanisms, e.g. modulation order and guard periods. Interleaving should among the techniques to be evaluated.
5. The next step should be implementation to the UAV communication system.
6. An algorithm that adapts the parameters m, n and N, based on estimates derived from the system states of the autopilot should be implemented to minimize the number of filter parameters to be estimated.

Chapter 7

Conclusion

This thesis was motivated by a student project with the goal of building a prototype for a VTOL UAV. The goal was to determine what performance could be expected from the wireless communication system of this UAV by employing advanced digital signal processing techniques. The ideal technique would optimize for both reliability and data transfer rate. A literature survey on filtering techniques concluded that an adaptive filter with the general structure of an DFE was best suited for the task. This techniques relies on parameter estimation to perform system identification of the channel. The system identification problem proved to be non-convex in simulations conducted on a FGN WSSUS aeronautic channel model. An experiment using a global optimization technique implied that the filtering problem was ill-suited to be solved on-line by an embedded computer within reasonable price range. Hence, the filtering technique deemed inadequate to achieve high data transfer rate while retaining the reliability for the UAV communication system under consideration here. Reliable communication must therefore be achieved by trading off data transfer rate. The main three trade off mechanisms are the modulation order, the period between transmitted symbols, and data overhead containing error correcting code. Increasing the period between transmitted symbols is expected to yield the best results, since simulations suggested that the

signal distortion was completely dominated by interference. Consequently, the expectation of limited data transfer rate was taken into consideration by the UAV development team.

Bibliography

[1] Haykin et al. Kalman filter-trained recurrent neural equalizers for time-varying channels. *IEEE Transactions on communications*, 53(3), March 2005.

[2] Simon Haykin. *Adaptive Filter Theory*. Pearson, 5th edition, 2014.

[3] Andreas F. Molisch. *Wireless Communications*. John Wiley and Sons, 2nd edition, 2011.

[4] Constantine A. Balanis. *Modern Antenna Handbook*. John Wiley and Sons, 1st edition, 2008.

[5] Erik Haas. Aeronautic channel modeling. *IEEE Transactions on Vehicular Technology*, 51(2), March 2002.

[6] Matz and Hlawatsch. Generalized evolutionary spectral analysis and the weyl spectrum of nonstationary random process. *IEEE Transaction on Signal Processing*, 45(6), June 1997.

[7] Pilip A. Bello. Characterization of randomly time-variant linear channels. *IEEE Transactions on Communications Systems*, CS-11(4):360 – 393, December 1963.

[8] Cyril-Daniel Iskander. A matlab-based object-oriented approach to multipath fading channel simulation. Technical report, Hi-Tek Multisystems (Formerly with MathWorks), 7945 Avenue de Cornouailles, Quebec, QC, Canada, 2008.

[9] P. Hoeher. A statistical discrete-time model for the wssus multipath channel. *IEEE Transactions on Vehicular Technology*, 41:461 – 468, 1992.

[10] Savo Glisic. *Advanced Wireless Communications and Internet*. John Wiley and Sons, 3rd edition, 2011.

[11] Shen Chen et al. Adaptive bayesian equalizer with decision feedback. *IEEE Transactions on signal processing*, 41(9), September 1993.

[12] Simon Haykin. *Neural Networks and Learning Machines*. Pearson, 3rd edition, 2009.

[13] K. Funahashi. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6:801 – 806, 1993.

[14] S. C. Shrivastava Kavita Burse, R. N. Yadav. Channel equalization using neural networks: A review. *IEEE Transactions on Systems and Cybernetics*, 40(3), 2010.

[15] Simon Haykin et al. Unscented kalman filter-trained recurrent neural equalizer for time-varying channels. In *IEEE International Conference on Communications*, volume 5, pages 3241 – 3245. IEEE.

[16] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14:179 – 211, 1990.

[17] Simon Haykin. *Kalman Filtering and Neural Networks*. John Wiley and Sons, 1st edition, 2001.

[18] P. Fearnhead J. Carpenter, P. Clifford. Improved particle filter for nonlinear problems. *IEEE Proceedings on Radar, Sonar and Navigation*, 146:2 – 7, 1999.

[19] N. Freitas E. A. Wan, A. Doucet and R. van der Merwe. The unscented particle filter. Technical report, Oregon Graduate Institute, Cambridge University, UC Berkley, 2000.

[20] F. Gustafsson R. Karlsson, T. Schon. Complexity analysis of the marginalized particle filter. Technical report, Linkjoping University, 2004.

[21] RE Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82:35 – 45, 1960.

[22] J.K. Uhlmann S.J. Julier. A new extension of the kalman filter to nonlinear systems. *Proceedings of AeroScience*, 1997.

[23] Rudolph van der Merwe and Eric A. Wan. The unscented kalman filter for state and parameter-estimation. Technical report, Oregon Graduate Institute of Science and Technology, 2001.

[24] P. Kaminski. Discrete square root filtering: A survey of current techniques. *IEEE Transactions on Automatic Control*, 16:727 – 736, 1971.

[25] C. Moraga B. Todorovic, M. Stankovic. On-line learning in recurrent neural networks using nonlinear kalman filters. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology*, pages 802 – 805. IEEE, 2003.

[26] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *SCIENCE*, 304:78–80, April 2004.

[27] H. Duarte-Ramos P. Gil, J. Henriques and A. Dourado. State-space neural networks and the unscented kalman filter in on-line system identification. In *International Conference on Inteligent Systems and Control*, 2001.

[28] Mauri Aparecido de Oliveira. An application of neural networks trained with kalman filter variants (ekf and ukf) to heteroscedastic time series forecasting. *Applied Mathematical Sciences*, 6(74):3675 – 3686, 2012.