

# GNSS Chirp Interference Estimation and Mitigation

## Master's Thesis

by

Daniël Kappelle

to obtain the degree of Master of Science  
at the Delft University of Technology,  
to be defended publicly on May 26, at 10:00.  
Conducted at the Royal Netherlands Aerospace Centre.

Student number: 4299108  
Project duration: September 1, 2020 – May 26, 2021  
Thesis committee: prof. dr. ir. A. J. van der Veen, TU Delft, supervisor  
dr. ir. J. J. P. van Es, Royal Netherlands Aerospace Centre  
dr. F. Fioranelli, TU Delft

An electronic version of this thesis is available at <https://repository.tudelft.nl/>.





# Abstract

GNSS receivers can suffer severely from *radio frequency interference* (RFI). RFI can introduce errors in the position and time calculations or if the interference is very severe, can lead to a total loss of GNSS reception. This vulnerability of GNSS can have large implications on critical infrastructure such as power plants, telephony, aviation or search and rescue operations. RFI is a real threat to GNSS as many interfering incidents are reported every day.

A common type of RFI is chirp interference, which is a sweep over a wide range of frequencies that overlap with the frequencies used by GNSS. This is often emitted by cheap *Personal Privacy Devices* that can be bought online. The question in this thesis was how well such interference can be modelled and if modelling could help mitigation against it.

This thesis consists of two main parts. In the first part a novel estimator is proposed that assumes a mathematical model of a chirp and estimates its parameters from recordings of chirps. The estimator has shown to work well in simulations for chirps with an SNR of  $-9$  dB or more. On real recordings the estimates were accurate for 66.7 % of the signals.

In the second part the estimator was used to derive a filter. The filter is based on the subtraction of a replica of the chirp interference from the received signal. It uses the proposed estimator to create the replica. In simulations, the filter is able to improve correlation strength by up to 7 dB. On real recordings the performance was worse as for only 46 % of the recordings the GNSS correlation was increased.

Both the estimator and filter have many ways in which they could be improved. The estimator can be improved to allow for more complex chirps, which would in turn improve the filter. Both can also be made more computationally efficient.

Furthermore, in order to get a better understanding of Personal Privacy Devices, one such device has been tested. It was found that the signal from the device was very unstable and changed much over time, it was also highly dependent on ambient temperature.



# Preface

This thesis is the product of almost nine months of reading papers, writing Matlab code, verifying, thinking, discussing and much more. Looking back it was a long period which had its highs and its lows. Doing a full-time individual project for such a period of time in the middle of a pandemic can be difficult at times. Working from home you miss the the quick chats, questions, criticism and motivation from people surrounding you working on similar matter. Fortunately at times it was possible to work from either the TU Delft or the NLR and these days were by all means the best days, both in terms of productivity as in terms of joy and motivation.

If I look back at my work from the beginning of this research it becomes clear to me how much I have learned over this period. There are also many things that, looking back at it now, I would have done differently. It turns out that you easily get caught up in the work and it is difficult to keep a good overview of what it is that you're doing. When I was working on the final thesis the past weeks I did get a good overview. This led me to finding many things I would have liked to research further, but didn't have the time to. These things can always still be researched, but for now there is this thesis and what I have achieved the previous months. Enjoy the read.

## Acknowledgements

Doing this research and writing this thesis would not have been possible were it not for a lot of people. Let me start by expressing my sincerest thanks to my daily supervisor Jan-Joris, accompanied by Nikki and Yuri. Every week in the past nine months you reserved time for me to help me going forward. You have been very supportive and motivating, but you have also been able to ask pressing questions that usually led me to a better understanding and a better end result. I don't know exactly how many versions of this thesis you have seen and commented on, but it must be quite a few. My thanks also go out to other colleagues from the NLR who have helped me, either by physically carrying out measurements, or by listening to my presentations and commenting with great suggestions!

Secondly I'd like to thank my professor Alle-Jan. It was a real pleasure to have you as my supervising professor. Our biweekly meetings were very important for me to gauge my progress. You were very clear about deadlines and expectations and also helped a lot in finding the exact direction to go in with this thesis. Your feedback on my earlier iterations of the thesis were very precise and thought provoking, adding greatly to the final result.

Of course none of this would have been possible if it weren't for my parents. You have made the path to this point possible and I am very thankful for that. You have been supportive all those seven (almost eight) years and you have been always interested in trying to understand what I was doing.

The Electrotechnische Vereeniging has also played an important role in my years at the TU Delft. I thank everyone for all the fun experiences, but also the valuable lessons that I have learned from the numerous activities. Especially the year as a Board member with the best Board members I could ask for is year I will never forget.

Thanks also to my roommates from the EduP for all the unforgettable years. I have enormously enjoyed my time at the EduP, but let's also not forget the intense library study sessions that have contributed to this point!

I am also very thankful for my lovely girlfriend Marije for motivating me whenever I needed it and also for pushing me to get things done when they had to get done.

Finally I'd like to thank all my friends throughout the years for all the fun experiences and for making studying such a pleasant time!

*Daniël Kappelle  
Rotterdam, May 2021*



# Contents

<b>List of abbreviations</b>	<b>ix</b>
<b>Nomenclature</b>	<b>xi</b>
<b>List of Figures</b>	<b>xiii</b>
<b>List of Tables</b>	<b>xv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 GNSS	1
1.1.1 Interference	1
1.1.2 Chirps	2
1.2 Research question	3
1.2.1 Scope	3
1.3 Organisation of this thesis	4
<b>2 GNSS Interference</b>	<b>5</b>
2.1 Types of interference	5
2.1.1 Continuous wave	5
2.1.2 Multi tone	7
2.1.3 Pulsed	7
2.1.4 Chirp	8
2.1.5 Multi-chirp	9
2.1.6 Wide-band noise	9
2.1.7 GNSS Modulated	9
2.1.8 Other modulation	10
2.2 Detecting interference	10
2.2.1 GNSS receiver structure	11
2.2.2 Antenna array	11
2.2.3 AGC monitoring	12
2.2.4 Spectral estimation	12
2.2.5 C/N0 monitoring	12
2.3 GNSS interference monitoring	13
2.3.1 STRIKE3	13
2.3.2 SENTINEL	13
2.3.3 J911	13
2.3.4 NLR	13
<b>3 Chirps estimation</b>	<b>15</b>
3.1 Signal model	15
3.1.1 Mathematical description of chirp	16
3.1.2 Band-Limited chirps	17
3.1.3 Window choice	18
3.2 Theoretical bound of estimator	18
3.3 Estimation algorithms from literature	19
3.3.1 Cubic Phase function	19
3.3.2 (modified) Discrete Chirp Fourier Transform	20
3.3.3 Phase unwrapping	21
3.3.4 Hankel Rank Reduction	21
3.3.5 Radon transform of Ambiguity Function	22
3.3.6 Fractional Fourier Transform	22
3.3.7 Conclusion on estimators	22

3.4	Proposed estimation algorithm. . . . .	23
3.4.1	Measurement setup . . . . .	30
3.4.2	Results . . . . .	31
3.5	Conclusion . . . . .	33
<b>4</b>	<b>Filtering chirp interference</b>	<b>35</b>
4.1	Filters from literature . . . . .	35
4.1.1	Filter bank pulse blanking . . . . .	35
4.1.2	Transform domain processing . . . . .	35
4.1.3	Adaptive Notch Filter . . . . .	36
4.2	Proposed filter . . . . .	36
4.3	Simulations . . . . .	38
4.3.1	Measuring performance . . . . .	38
4.3.2	Results . . . . .	39
4.4	Conclusion . . . . .	41
<b>5</b>	<b>Application</b>	<b>43</b>
5.1	Types of chirps . . . . .	43
5.1.1	From the NLR monitoring . . . . .	43
5.2	Parameter estimation on real data. . . . .	44
5.3	Filtering on real data . . . . .	44
5.3.1	Qualitative results . . . . .	48
5.3.2	Quantitative results. . . . .	52
<b>6</b>	<b>Conclusion</b>	<b>55</b>
6.1	Estimation. . . . .	55
6.2	Filtering . . . . .	55
6.2.1	Filter limitations . . . . .	56
6.2.2	Comparison. . . . .	56
6.3	Future work . . . . .	56
6.3.1	Non-linear chirps . . . . .	56
6.3.2	Out-of-band chirps . . . . .	57
6.3.3	Computational complexity . . . . .	57
<b>A</b>	<b>Jammer measurements</b>	<b>59</b>
A.1	First glance . . . . .	59
A.2	EM Leakage . . . . .	60
A.3	Stability . . . . .	61
A.3.1	Long term . . . . .	61
A.3.2	Temperature dependence . . . . .	61
A.3.3	Supply voltage dependence . . . . .	64
A.3.4	Conclusion . . . . .	64
	<b>Bibliography</b>	<b>67</b>

# List of abbreviations

- ADC** Analog Digital Converter.
- AF** Ambiguity Function.
- AGC** Automatic Gain Control.
- AM** Amplitude Modulation.
- BPSK** Binary Phase Shift Keying.
- CAF** Cross Ambiguity Function.
- CDMA** Code-Division Multiple Access.
- CRLB** Cramèr-Rao Lower Bound.
- CW** Continuous Wave.
- DCFT** Discrete Chirp Fourier Transform.
- DME** Distance Measuring Equipment.
- DoS** Denial of Service.
- FFT** Fast Fourier Transform.
- FM** Frequency Modulation.
- FrFT** Fractional Fourier Transform.
- GNSS** Global Navigation Satellite System.
- GPS** Global Positioning System.
- GSA** European Global Navigation Satellite Systems Agency.
- iid** Independent and identically distributed.
- IQ** In-phase, Quadrature-phase.
- JNR** Jammer-to-Noise Ratio.
- L1 C/A** L1 band Coarse Acquisition (most used GPS signal).
- MUSIC** Multiple Signal Classification.
- n-PSK** n-ary Phase Shift Keying.
- NLR** Koninklijk Nederlands Lucht- en Ruimtevaartcentrum (Royal Netherlands Aerospace Centre).
- PM** Phase Modulation.
- PPD** Personal Privacy Device.
- PRN** Pseudorandom Noise.
- PSD** Power Spectral Density.
- RF** Radio Frequency.

**RFI** Radio Frequency Interference.

**SDR** Software Defined Radio.

**SNR** Signal-to-Noise Ratio.

**STFT** Short Time Fourier Transform.

**STRIKE3** Standardisation of GNSS Threat Reporting and Receiver Testing through International Knowledge Exchange, Experimentation and Exploitation.

**SV** Space Vehicle.

**SVD** Singular Value Decomposition.

**TACAN** Tactical Air Navigation.

**TDoA** Time Difference of Arrival.

**VCO** Voltage Controlled Oscillator.

**VOR** VHF Omnidirectional Range.

**WB** Wide-band.

**WSS** Wide-Sense Stationary.

# Nomenclature

In general bold face lower case letters denote vectors, bold face upper case letters denote matrices. Otherwise it is a scalar.

$\hat{\cdot}$	Denotes an estimator
$*$	Denotes the complex conjugate
$[\mathbf{A}]_{ij}$	$ij$ 'th element of matrix $\mathbf{A}$
$\mathcal{F}\{\cdot\}$	Fourier transform operator
$\mathcal{F}^{-1}\{\cdot\}$	Inverse Fourier transform operator
$E[\cdot]$	Energy of a signal
$\mathbb{R}$	Set of real numbers
$\mathcal{CN}(\boldsymbol{\mu}, \mathbf{C})$	Complex normal distribution around $\boldsymbol{\mu}$ with covariance matrix $\mathbf{C}$
$\mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$	Normal distribution around $\boldsymbol{\mu}$ with covariance matrix $\mathbf{C}$
$\alpha$	Chirp rate
$\bar{x}$	Sample mean of signal $\mathbf{x}$
$\boldsymbol{\theta}$	Vector of parameters
$\mathbf{0}$	All zero vector
$\mathbf{C}$	Covariance matrix
$\mathbf{I}$	Identity matrix
$\mathbf{I}(\boldsymbol{\theta})$	Fisher information matrix
$\mathbf{r}_{xx}$	Autocorrelation sequence of $\mathbf{x}$
$\mathbf{s}$	Desired (satellite) signal
$\mathbf{u}$	Undesired (jamming) signal
$\mathbf{w}$	Noise signal
$\mathbf{x}$	Received signal
$\mathbf{x}_i$	Window $i$ of $\mathbf{x}$
$\omega$	Angular frequency in rad/s
$\phi$	Phase offset
$\phi_0$	Initial phase
$A$	Amplitude
$B$	Bandwidth
$C/N_0$	Carrier-to-noise-density-ratio
$f$	Frequency, either normalised or assuming some sampling frequency

---

$f_0$	Initial frequency
$f_c$	Centre frequency
$h$	Channel impulse response (time-domain)
$i$	Window index
$L$	Number of samples in a window
$M$	Number of windows
$N$	Number of samples
$n$	Discrete time index
$N_{mc}$	Number of Monte Carlo runs
$P_x$	Power in signal $x$
$T$	Chirp period
$X$	Stochastic variable
$x(t)$	Continuous-time signal $x$ at time $t$
$x[n]$	Sample $n$ from discrete-time signal $x$

# List of Figures

1.1	Frequency bands of Galileo and GPS. Image taken from [1]	2
1.2	Types of interference	2
1.3	An example of a chirp signal in the time-frequency domain	2
2.1	Continuous wave interference	5
2.2	Two tone interference	7
2.3	Pulsed interference	8
2.4	Chirp interference	8
2.5	Multi-chirp interference	9
2.6	Wide-band noise interference	10
2.7	GPS L1 C/A modulated interference	10
2.8	Schematic overview of GNSS receiver	11
2.9	Beamforming using multiple antennas	11
2.10	The GNSS detector from NSL (GSS200D)	14
3.1	The parameters of a chirp, shown in the time-frequency domain	17
3.2	The parameters that define the band-limited chirp	18
3.3	O'Shea algorithm figures	20
3.4	Estimation algorithms compared	24
3.5	Proposed algorithm, schematically	25
3.6	Autocorrelation of $x$	26
3.7	Window drift because of inaccurate window length.	27
3.8	Offset as function of window index, showing window drift.	28
3.9	The shaded part is only noise. This area is what's left after removing the maximum of the column and 4 pixels up and down from the maximum.	29
3.10	Chirp spectrogram and estimated instantaneous frequency for a chirp with an SNR of 0 dB	31
3.11	Effect of very strong chirp on surrounding noise. In both cases the noise has variance 1.	32
3.12	Accuracy of proposed estimator	32
4.1	Overview of the filtering algorithm	36
4.2	Illustration of how subtracting one unit amplitude sinusoid from another with frequencies 10 kHz apart leads to a beating pattern.	37
4.3	Correlation peak for a specific satellite using one PRN period and 30 periods. In this simulation $C/N_0 = 40$ dB – Hz	39
4.4	Gain in correlation peak for SNR = –20 dB	40
4.5	Gain in correlation peak for SNR = 0 dB	40
4.6	Phase shift required for proper filtering, using 20 sample windows	41
4.7	The effect of the window size on the filter	41
5.1	Received chirp-like interference signals.	45
5.2	Relative occurrence of the different chirp types.	46
5.3	Estimator results on real data. The $\sigma$ by hand values indicate the standard deviation resulting from estimating parameters by hand.	46
5.4	This figure shows some examples of the errors that the estimator might make and how that compares to a visual estimate.	47
5.5	Filtering results for fast chirp	48
5.6	Filtering results for medium chirp	49
5.7	Filtering results for slow chirp	50
5.8	Filtering results for slow chirp	51

---

5.9	Filtering results for slow chirp . . . . .	53
A.1	The jammer . . . . .	60
A.2	The circuit board of the jammer. On the right is the SMA connector that connects to the antenna. . . . .	60
A.3	The main lobe of the jammer, centered around 1.575GHz . . . . .	61
A.4	The main lobe is at $f_c = 1.575\text{GHz}$ and there are clear harmonics at $2f_c$ , $3f_c$ and $4f_c$ . . . . .	61
A.5	The spectrogram of the output of the jammer, it is a chirp. . . . .	61
A.6	The full spectrum from 25 MHz to 8 GHz . . . . .	62
A.7	Estimated parameters of the chirp over 1 hour. . . . .	62
A.8	The signal of the jammer at 5 °C . . . . .	63
A.9	Estimated parameters of the chirp for different temperatures. . . . .	63
A.10	Estimated parameters of the chirp for different supply voltages. . . . .	64
A.11	The signal from the jammer when only 5 V is applied. . . . .	65

# List of Tables

2.1	Overview of different interference types . . . . .	6
3.1	Overview of different estimation algorithms. . . . .	23
3.2	The allowed bounds for the different parameters, assuming a sampling frequency of $f_s = 52\text{MS/s}$ . . . . .	31
5.1	This table lists for how many signals the correlation performance was increased by applying the filter. . . . .	52



# Introduction

## 1.1. GNSS

A Global Navigation Satellite System, GNSS for short, is the general term for a satellite navigation system. Well known systems such as GPS (US) and Galileo (European Union) are examples of GNSS. These systems have become ubiquitous in our modern society. Most people will know GNSS from their smart phones and turn-by-turn navigation for accurately pinpointing their location. The system is however used in many more applications such as navigation in aviation or the maritime sector, military applications or geodesy. Perhaps lesser known are its applications as a very precise source of time. Due to the way GNSS operates, it is able to tell time with an accuracy of  $\leq 40$  ns [2]. This is used for example for synchronising power plants, for the operation of mobile telephony and for time stamping financial transactions.

Most users will not question the presence of GNSS and rely on it assuming it always works. For everyday applications such as navigation this is reasonable, especially since it indeed provides accurate position and time info at least 95 % of the time [2]. For the more critical applications however, it is very important not to take GNSS for granted, but to be prepared in case the system malfunctions.

GNSS systems can suffer multiple problems. These can be urban environments which lead to multipath and fading effects, solar storms or just satellite maintenance. Another threat to GNSS is intentional or unintentional radio frequency interference, which is the scope of this thesis. In the next section this problem will be elaborated on.

### 1.1.1. Interference

GNSS signals are very weak when they reach the surface of the earth, they can have powers as low as  $-158$  dBW ( $\approx -1.6 \times 10^{-16}$  W) [3]. This makes them very vulnerable to *Radio Frequency Interference* (RFI). Only very limited transmit power is required to emit a much stronger signal than the GNSS signals and render them useless. The GNSS frequency bands, see figure 1.1, are restricted virtually anywhere on the planet, making this illegal and therefore also unexpected for most users. Unfortunately, interfering signals are not uncommon and their impact can be big. Please note that this is not only a hypothetical problem, but many incidents have already been reported. There are a few different types of interfering signals as will now be discussed.

There are some important distinctions to be made in the context of interference with respect to GNSS. First of all interference can either be manmade or the result of a natural process. Natural processes could for example be heavy solar activity or ionospheric effects and are clearly something nobody has control over. Manmade interference can be subdivided further.

Manmade interference could be unintentional or intentional. Unintentional interference is in many cases the result of out-of-band emissions (e.g. higher harmonics) from malfunctioning electronics, such as telecommunication broadcasting towers. Then finally there is intentional interference, which can be very problematic since it can be targeted to a specific victim. The intentional interference could be either jamming or spoofing. In the case of jamming, a strong interfering signal is emitted within the band which causes a reduced performance of the GNSS receiver and can in severe cases can even cause the receiver to fail entirely. This is also known as a *denial-of-service* (DoS) attack. These

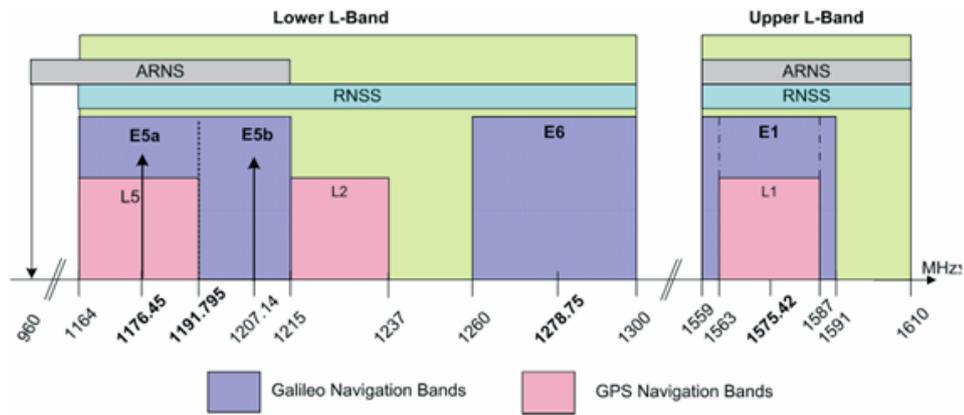


Figure 1.1: Frequency bands of Galileo and GPS. Image taken from [1]

attacks are illegal, but easy to perform for anyone as simple GNSS jammers can be bought online. One common type of jammer plugs into the 12V socket in your car and that is all that is needed. One such jammer has been examined in further detail, the observations can be found in appendix A. Commercial jammers are also referred to as *Personal Privacy Devices* or PPDs. The signals that they emit vary, as will be the focus of section 2.1.

Spoofing is a bit more sophisticated as it requires well engineered signals and specific hardware and software to work properly. The idea here is to mimic the GNSS satellites and trick the receiver into thinking it is in a different location and possibly at a different time.

The distinction made in this section is shown schematically in figure 1.2.

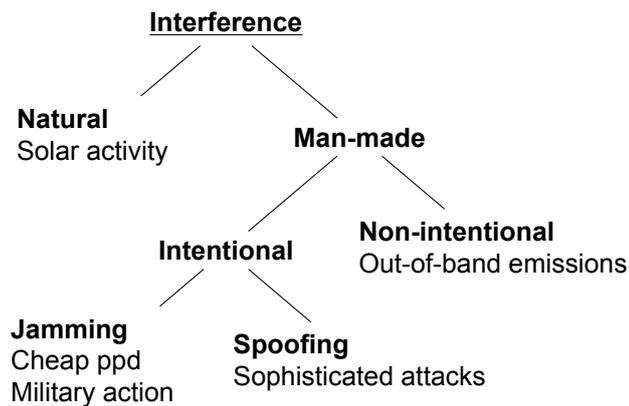


Figure 1.2: Types of interference

### 1.1.2. Chirps

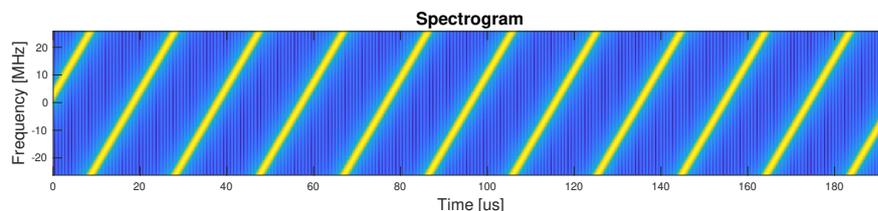


Figure 1.3: An example of a chirp signal in the time-frequency domain

Interference signals come in many different forms. One particularly interesting type of interference is the so called chirp interference. This is named after the sound birds make, which is a sweep over a

certain frequency band. This type of signal is also called (linear) frequency modulated and it is used often in radar, but also by some species of bats that use echolocation to find their way in the dark [4]. In the time-frequency domain the signal looks like a sawtooth as is shown in figure 1.3.

There are three main reasons chirp signals are of interest. The first reason is that it is not a signal that would accidentally appear somewhere. It has to be deliberately crafted and therefore signifies with great probability that someone is actively trying to jam the GNSS signals. Secondly, GNSS receivers have great difficulty operating when chirp-like interference is present. This is because the signal energy is in a wide band and varies with time, making it difficult to filter out. Lastly, chirp-like interference is seen regularly on monitoring stations: more than once every day at the NLR monitoring station!

## 1.2. Research question

The research question of this thesis is formulated as: *how to effectively estimate and filter out chirp interference signals in the GNSS frequency bands to improve GNSS signal reception?*

This automatically leads to some subquestions such as

- How can the signal parameters of the chirp be estimated? And what is limiting the performance of the estimation?
- How well can the chirp signal be predicted, i.e. how stable is the interference source and what is the impact of multipath/fading effects.
- What methods can be used to filter out chirp interference from the received GNSS signal?
- How does filtering improve GNSS receiver performance?

### 1.2.1. Scope

In order to constrain the research question a number of boundary conditions are imposed on it:

**Single antenna** The thesis assumes a system that uses only one receiving antenna. This rules out any spatial techniques to filter out interference from the GNSS signals as described in section 2.2.2. Assuming a single antenna makes that the results of this research will be applicable to virtually all current-day GNSS receivers and interference monitoring systems as these devices rarely have more than 1 receiving antenna.

**IQ samples** To be able to apply digital signal processing techniques, it is assumed that the received signal is digitised and that the raw IQ samples can be accessed. This means at discrete time intervals (reciprocal of the sampling frequency) there will be a quantised in-phase and quadrature amplitude, which can be combined to form a complex valued sample.

Some further conditions can be assumed. Note that values may vary from device to device, but below is a list of lower bounds on the hardware.

- **Bit depth** At least 2 bits per phase, so 4 bits per sample
- **Sample rate** At least 50 MS/s, so there is a wide enough receiver bandwidth and many chirps will fall entirely within the band.
- **Recording time** At least one L1 C/A period, so  $>1$  ms.

**Civilian only** In this thesis GNSS interference in a civil environment is studied, i.e. the interferers and receivers are assumed non-military. Military or state-organised jamming as part of electronic warfare, is beyond the scope of this work.

**Low power jammers** Only jammers with relatively low power are studied. Low power in this sense means that the radio front-end does not suffer clipping from the interferer and renders the samples useless.

### 1.3. Organisation of this thesis

This thesis is further organised as follows. Chapter 2 gives more insights in GNSS interference as a whole. The types of interference that may be encountered as well as ways to detect it are discussed.

Chapter 3 then zooms in on the chirp type interference. Mathematical models are derived and different techniques are discussed to estimate the parameters that constitute the chirp. Simulation results of the estimator will also be studied in chapter 3.

In chapter 4 filtering techniques are discussed and one particular technique is proposed and implemented. The estimator from chapter 3 plays an important role in the filter of chapter 4. At the end of chapter 4 the performance of the filter will be analysed based on simulations.

In chapter 5 actual recorded signals are presented and used to test the performance of the derived algorithms for estimating and filtering chirp interference signals. The performance of the methods used are presented and discussed. The results are also compared to the simulation results from earlier chapters.

In chapter 6 conclusions are drawn about the proposed estimation and filtering method. The shortcomings of these methods are exposed and recommendations are made for future work to improve the proposed methods.

During this thesis an actual GPS jammer has been analysed to get more insights in their stability as well as how they work. A report of this analysis is added at the end of this thesis as appendix A.

# 2

## GNSS Interference

This chapter gives more insights into radio frequency interference that may occur on GNSS frequency bands. First eight different types of interference are presented. The parameters of interest of these signals will be listed and the signals visualised. Then four techniques are discussed that can be used to detect the presence of interfering signals.

### 2.1. Types of interference

Radio Frequency Interference comes in many different forms. The interfering signals vary greatly in their waveform/spectrum, but also come from very different sources. Some might be intentional, while others are probably not. In this section, an overview is given of some common signals. This list has been compiled based on measurements from the monitoring station at the NLR as well as from literature such as [5–8].

The signals are also presented visually by their *Power Spectral Density* (PSD) and Spectrogram representations. The former shows how the total energy in the signal is distributed over the frequency spectrum. The latter shows how the PSD changes with time, so that the intensity of the plot shows how much energy is present, the vertical axis shows the frequency spectrum and the horizontal axis shows time (in most cases discrete time, i.e. samples).

Table 2.1 summarises the list in a comprehensible manner.

#### 2.1.1. Continuous wave

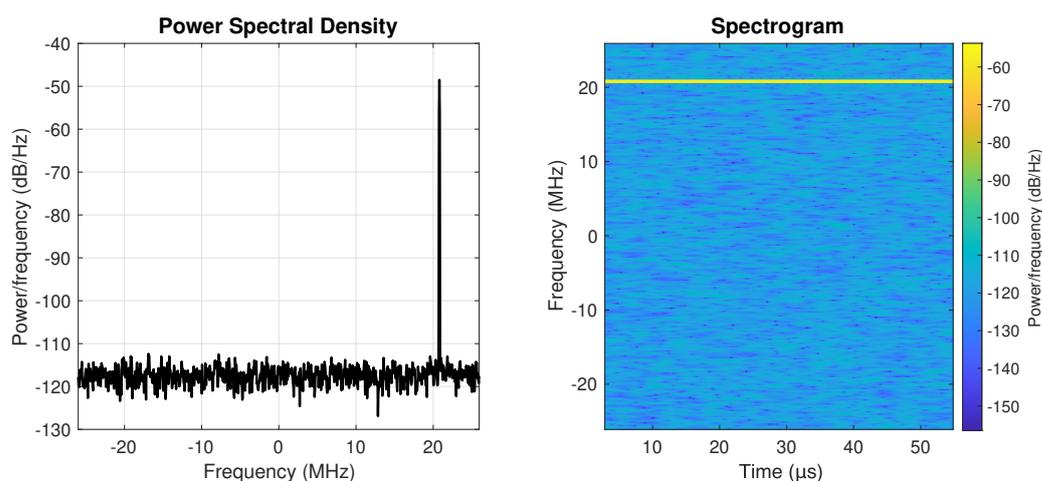


Figure 2.1: Continuous wave interference

The first type of signal is the *continuous wave* or CW signal. This is another name for a single tone or a single sinusoid, see figure 2.1. Sinusoids are used as carriers for modulated signals but

Type	Properties of interest	Possible sources	Impact	Filtering
Continuous wave	<ul style="list-style-type: none"> <li>• frequency</li> <li>• power</li> </ul>	<ul style="list-style-type: none"> <li>• harmonics in electronic device</li> <li>• clock frequency</li> <li>• PPD</li> </ul>	Low	Easy
Multi tone	<ul style="list-style-type: none"> <li>• frequencies</li> <li>• power of different components</li> <li>• number of different tones</li> </ul>	<ul style="list-style-type: none"> <li>• multiple harmonics of electronic device</li> <li>• multiple CW sources</li> <li>• input clipping, nonlinearity</li> <li>• PPD(s)</li> </ul>	Low	Easy
Pulsed	<ul style="list-style-type: none"> <li>• Underlying signal</li> <li>• Pulse period</li> <li>• Duty cycle</li> </ul>	<ul style="list-style-type: none"> <li>• DME/TACAN</li> <li>• Radar</li> </ul>	Low to medium	Medium
Chirp	<ul style="list-style-type: none"> <li>• Up/down (direction)</li> <li>• Bandwidth</li> <li>• Chirp rate</li> <li>• Centre frequency</li> <li>• Linear or curved frequency modulation</li> </ul>	<ul style="list-style-type: none"> <li>• PPD</li> <li>• Radar</li> </ul>	High	Difficult
Multi-chirp	<ul style="list-style-type: none"> <li>• Up/down (directions)</li> <li>• Bandwidths</li> <li>• Chirp rates</li> <li>• Centre frequencies</li> <li>• Linear or curved modulations</li> <li>• Number of chirps</li> </ul>	<ul style="list-style-type: none"> <li>• Advanced PPD</li> </ul>	High	Very difficult
Triangular	<ul style="list-style-type: none"> <li>• Bandwidth</li> <li>• Centre frequency</li> <li>• Frequency rate</li> </ul>	<ul style="list-style-type: none"> <li>• PPD</li> </ul>	High	Difficult
Wide-band noise	<ul style="list-style-type: none"> <li>• bandwidth</li> <li>• centre frequency</li> </ul>	<ul style="list-style-type: none"> <li>• unknown</li> </ul>	Very high	Very difficult
Other modulation	<ul style="list-style-type: none"> <li>• modulation type</li> </ul>	<ul style="list-style-type: none"> <li>• Out-of-band emission</li> </ul>	N/A	N/A

Table 2.1: Overview of different interference types

can also arise from higher order harmonics. Possible sources could in fact be any type of electronics. These could be RF appliances with leaking clock frequencies or harmonics of a clock for example. It is however also possible that a piece of electronics which is not even intended to emit any RF still does so by poor design and traces or wires that behave like antennas.

The fact that these type of signals can so easily manifest themselves makes that they are seen very often, see for example [8, 9]. They are also very often reported by the monitoring station at the NLR: around 95% of the events are classified as CW signals. They are all also classified as very low impact.

There are some commercial jammers available that use CW to jam GNSS, see for example [6], but the majority use chirp-like signals. This, in combination with the fact that CW signals can easily arise from malfunctioning electronics makes it reasonable to assume that these type of interferences are in most cases unintentional.

With respect to GNSS reception, the severity of CW interferers is minimal. First of all, since they are stationary, i.e. they do not change with time, and they are also very narrow band they can easily be filtered out with a notch filter. Removing a tiny portion of the spectrum will not have a severe impact on GNSS reception due to its wide band nature. Furthermore, even if the CW is not filtered out, its impact on reception is acceptable. During the demodulation of GNSS signals, the energy of the CW signal is spread out over the receiver frequency bandwidth while the satellite energy is despread and confined to one very narrow band. Because of this operation, the effect on the correlation process is minimal.

### 2.1.2. Multi tone

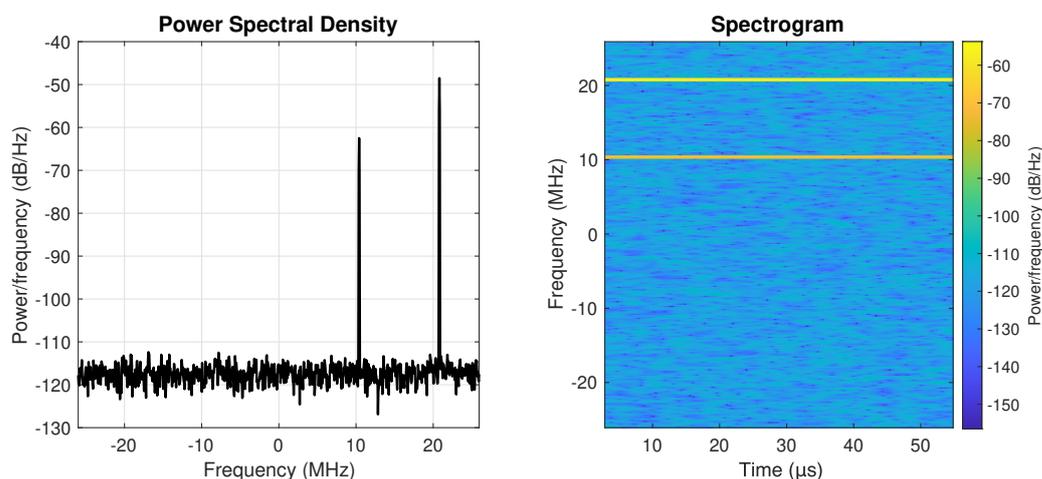


Figure 2.2: Two tone interference

Multi tone interference is nothing but multiple CW signals superimposed, see figure 2.2. There could be an arbitrary number of sinusoids with arbitrary frequency and amplitude. Obviously, they could come from multiple CW sources. Another possibility is that a single piece of electronics emits multiple harmonics at the same time. What is however also possible is that the RF input (or any of the analog stages) of the GNSS receiver is saturated due to a single CW source for example. This saturation, or clipping, will distort the signal leading to multiple harmonics.

PPDs using multiple CW signals as a jamming signal do not seem to exist, so that is a very unlikely source. It is of course possible that there are multiple CW PPDs present, but that seems quite unlikely as well. It is therefore assumed that in virtually all cases, this type of interference is unintentional.

As long as the number of tones and their amplitudes are somewhat limited, the effect on GNSS reception will also be limited. Since the signals are also stationary, multiple notch filters could easily filter all of them out and the GNSS signal will suffer minimally.

### 2.1.3. Pulsed

Any type of signal could be pulsed to get a more complex kind of signal. Figure 2.3 shows a pulsed CW signal. Such a signal could be a harmonic as described earlier, but then in a piece of electronics that is switched on and off. Another possible source of pulsed signals is radar, where pulsed signals

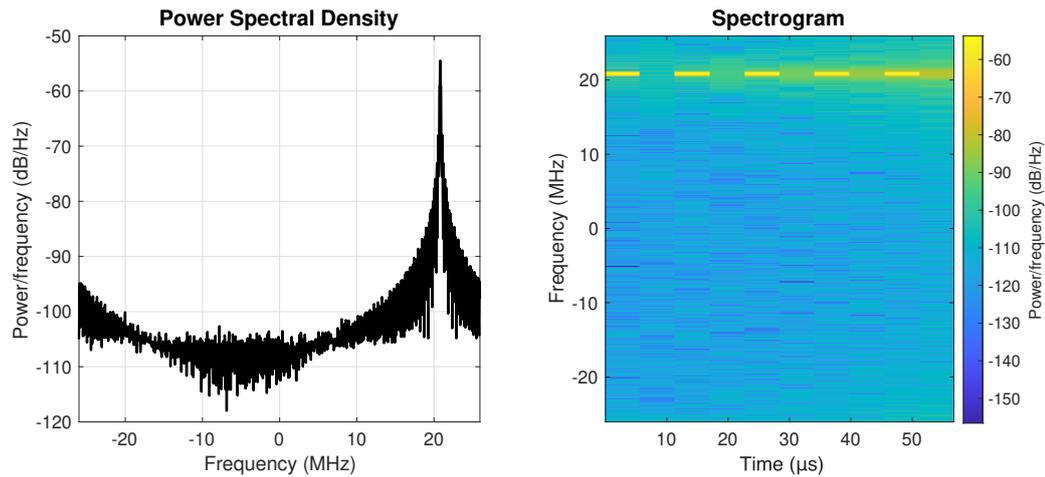


Figure 2.3: Pulsed interference

are sometimes used [10].

A third source of pulsed interference is *Distance Measuring Equipment* (DME) or *Tactical Air Navigation* TACAN systems. These systems provide distance information to airplanes and are often found at airports and VOR (VHF Omnidirectional Range) radio beacons. They use pulse trains and operate in a frequency band that overlaps with the L5 and E5 bands [11, 12].

Since there is only interference energy present part of the time, the effect is minimal. Apart from filtering the underlying signal, which would be easy in case of CW, pulse blanking can be used to filter out the pulse signal. These reasons make it unlikely that anyone who wanted to deliberately jam GNSS would use such a signal.

#### 2.1.4. Chirp

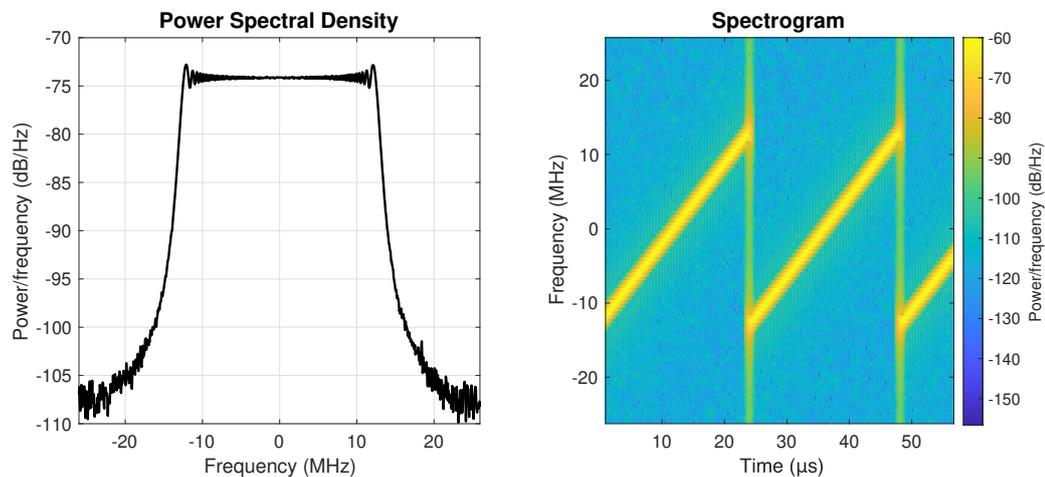


Figure 2.4: Chirp interference

Chirps have already been briefly described in the introduction of this thesis (sec. 1.1.2). They are slightly more complex than the signals previously mentioned. An example of the spectrum of a chirp signal is shown in figure 2.4. Unlike a CW signal, the frequency of the chirp signal varies with time. The chirp is a so called frequency modulated signal and sweeps a certain frequency band in a certain time. In some cases this frequency sweep is linear in time, making it a *Linear Frequency Modulated* (LFM) signal. In this thesis we will primarily look at the linear type.

Chirp like signals are encountered in for example radar and sonar (also by certain bats). Chirps are however also used by the majority of the GNSS jammers [5, 6]. Since the frequency bands of GNSS

are restricted and the chirp type would not accidentally arise from any piece of electronics, this is a clear sign that the chirp interference is deliberately used to jam the GNSS signals.

Both at the NLR as at other monitoring stations it turns out that chirp interference is also very common [8, 9]. The NLR monitoring station reported more than one chirp interferer every day (see sec. 2.3.4).

The varying nature of the chirp makes it difficult to filter it out. One cannot simply apply a notch filter to get rid of the signal. Moreover, the impact of the chirp signal on GNSS reception is more severe than that of CW interference.

The fact that chirps are very common, they are almost certainly deliberately used for jamming and they are very harmful for GNSS reception makes them the focus of this thesis.

### 2.1.5. Multi-chirp

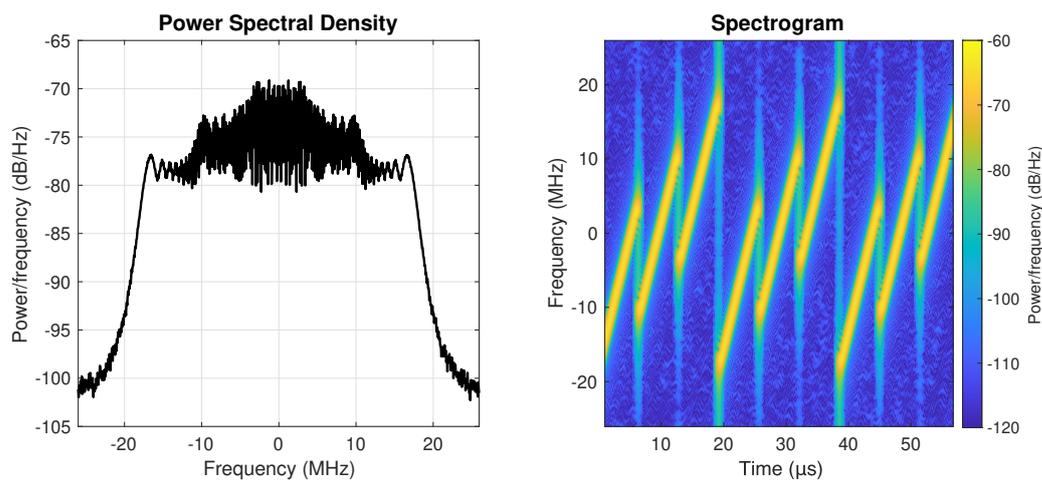


Figure 2.5: Multi-chirp interference

An extension of the chirp signal is a multi-chirp signal, see figure 2.5. This is where multiple chirp signals are modulated on top of one another, creating an even more complex signal. These signals have no known usage other than deliberate jamming.

The extra chirps makes tracking the instantaneous frequency of the signal much more complex, making it nearly impossible to design a proper filter for this type of signal.

Fortunately these signals are rarely seen being used by jammers, although they do exist [6].

### 2.1.6. Wide-band noise

Wide band noise signals are not seen often in practice and receive little attention in literature. The spectra of the signals would look like the one in figure 2.6, which is basically a massive amount of energy in a relatively wide band.

Since this signal is Gaussian it does not have a predictable structure, making it nearly impossible (if not impossible) to filter it out and leave the GNSS signal intact. The GNSS reception would also be severely degraded.

### 2.1.7. GNSS Modulated

Although this might not technically be considered jamming, but would be referred to as spoofing (refer back to figure 1.2), it is possible to see a stronger GNSS modulated signal like the GPS L1 C/A signal in figure 2.7.

This could indicate a spoofing attack, i.e. an attacker tries to craft a GNSS signal that tricks GNSS receivers into thinking it is somewhere else (or some time else) than it really is.

Another possible source is a GNSS repeater. These devices enhance GNSS reception in indoor environments where reception would otherwise be bad or impossible. If, however, these repeaters are not handled with care it could be possible to leak the signals to other areas where it could disrupt the normal GNSS operation.

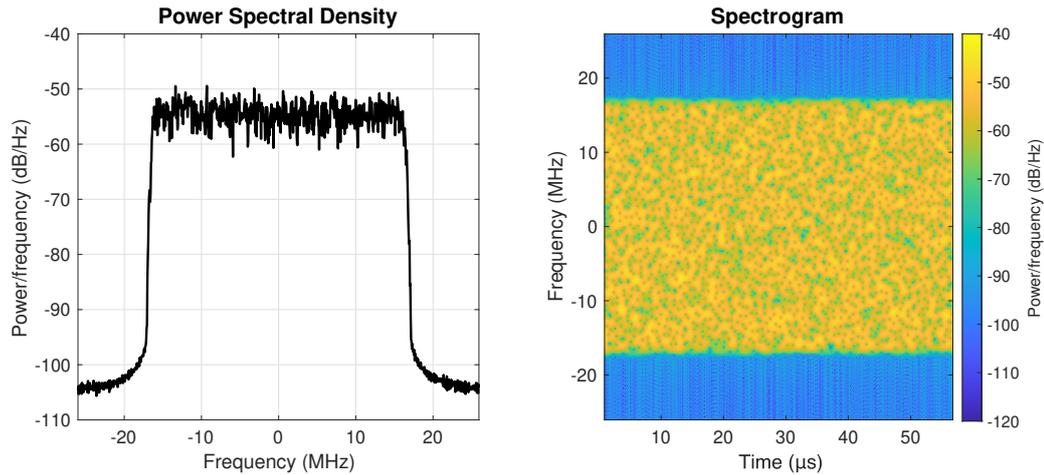


Figure 2.6: Wide-band noise interference

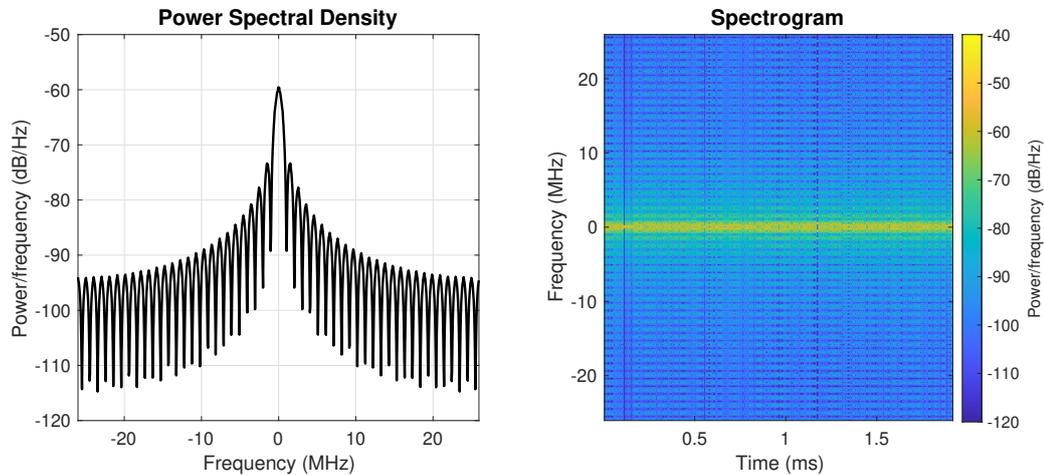


Figure 2.7: GPS L1 C/A modulated interference

If no spatial techniques can be leveraged, it is very difficult to distinguish between the real GNSS signal and the rogue one. If, on the other hand, multiple antennas are used and spatial techniques can be used, it is usually possible to filter out the rogue GNSS signal as it often comes from a single point source, usually at low elevation. This is different from the real GNSS signals which come from different satellites scattered around the sky.

### 2.1.8. Other modulation

In addition to all the previously mentioned signal types, any type of modulated signal could also be seen interfering. This can be coming from radio or tv transmitters for example. If the electronics are faulty and the transmitters emit outside their assigned bands, this type of signal can be seen.

## 2.2. Detecting interference

Different techniques have been presented in literature - and are used in practice - to detect GNSS interference. They differ in what data they use for detection, which is closely related to during what stage of the GNSS receiver they are implemented. The next section gives an overview of the GNSS receiver structure and in subsequent sections the different techniques for detection will be briefly introduced.

### 2.2.1. GNSS receiver structure

Figure 2.8 shows a schematic overview of how a typical GNSS receiver looks. The electromagnetic waves are picked up by an antenna and fed through an analog network. This can contain for example filters, amplifiers or matching networks. The signal is then passed on into an *Automatic Gain Control* (AGC) block.

The AGC consists of a variable amplifier, *analog digital converter* (ADC) and digital gain control. As the GNSS signal is typically very weak, it is important to use it as effectively as possible. The AGC makes sure that the signal is amplified just the right amount by a variable gain amplifier so that it matches the input range of the ADC. To do this, a feedback loop from the ADC controls the gain of the variable gain amplifier [13, pp. 108].

The digital signal coming from the ADC is then passed onto the signal processing block where the demodulation of the GNSS signals takes place and pseudo range measurements are computed.

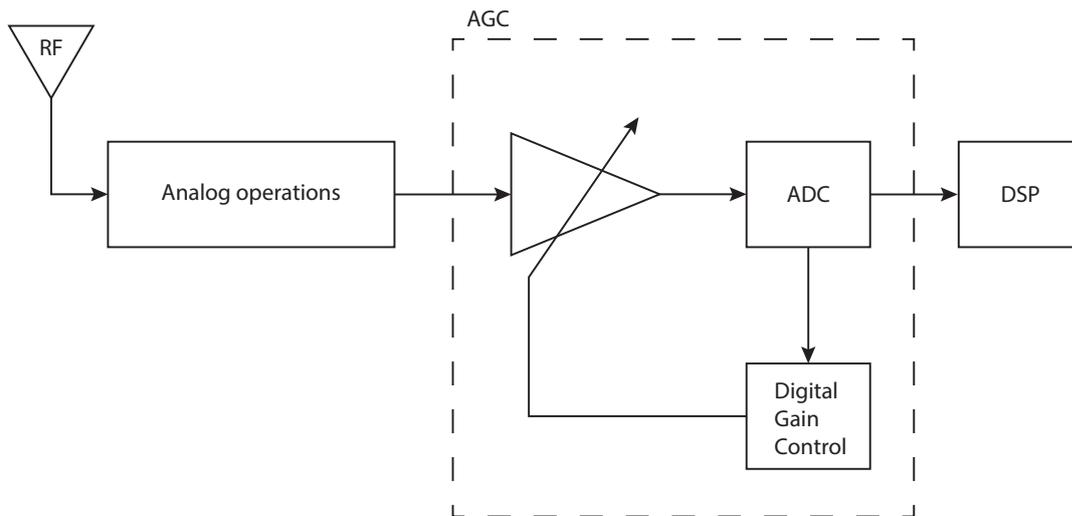


Figure 2.8: Schematic overview of GNSS receiver

### 2.2.2. Antenna array

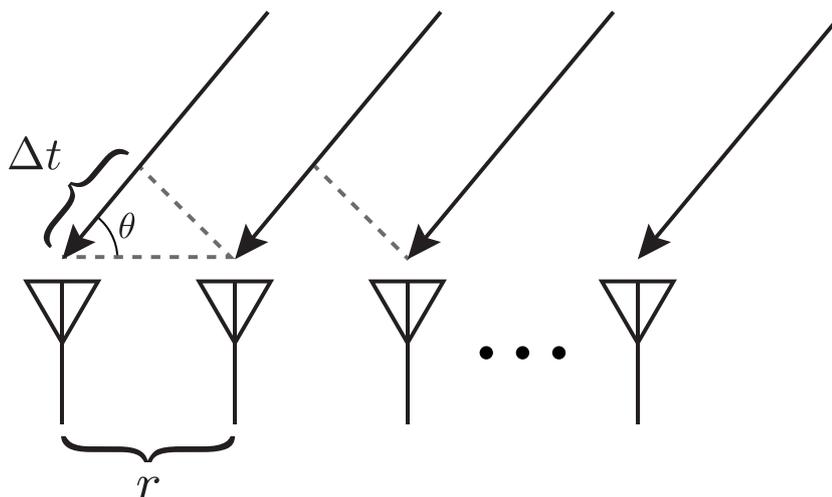


Figure 2.9: Beamforming using multiple antennas

Antenna arrays can be used for a technique known as beam forming. This is where the *time difference of arrival* (TDoA) of a signal at the different antennas is used to determine the location of the source. This setup is shown schematically in figure 2.9. Consider a number of antennas with known

separation  $r$ , then the TDoA is proportional to the incident angle  $\theta$ . By for example delaying and summing the outputs of the antennas one can *steer* the beamformer to a certain direction. Note that this is just the general idea and much more effective beamforming algorithms exist, but they rely on the same principle.

Since jammer and GNSS satellites are typically very well separated spatially, this technique can be very successful. GNSS satellites are often at high elevation, while jammers will mostly be somewhere on the surface of the earth. Some beam forming techniques allow steering *zeros*, i.e. places in space where the signal is completely attenuated, so that such a zero can be steered in the location of the jammer.

Using multiple antennas and processing the output of them requires complex hardware. This makes the technique very costly to implement. Also, because special hardware is required, such a system cannot be implemented using typical off-the-shelf GNSS receivers.

### 2.2.3. AGC monitoring

Virtually all GNSS receivers are equipped with an Automatic Gain Controller (AGC) as described earlier in section 2.2.1. Interfering signals are typically orders of magnitude more powerful than the GNSS signals and noise floor. This means that if an interferer is present, the total input energy picked up by the antenna and passed on to the AGC is much higher than in the nominal case. The AGC will try to compensate for the higher input energy by lowering the gain on the variable amplifier.

If the control signal that controls the variable amplifier is available, this signal can be used to measure an increase in received energy. This is then an indication that an interfering signal may be present [14]. Fortunately in many GNSS receivers the AGC control signal is in fact available, so it can actually be leveraged for this detection method.

A disadvantage of using the AGC control signal for detection is that it is quantised in a limited number of levels, giving it a poor resolution. Other parameters such as integration time or response time are usually not alterable, so there is limited control over the detector [15, p. 17].

Another drawback is that the AGC only measures the total input energy and it gives no insights into the spectral shape of the interfering signal. It can therefore not be used to model the interferer or estimate its parameters.

### 2.2.4. Spectral estimation

If the output of the ADC, which consists of raw IQ samples, are available, more complex techniques are available. These samples, that have not yet been processed, contain all available information about the interfering signal. This makes it possible to analyse the signal in different domains, most probably the time-frequency domain. In this case the time domain signal is windowed and the *Short Time Fourier Transform* (STFT) of each window is taken, resulting in a spectrogram. This spectrogram shows how the spectral shape of the signal changes with time. From this many conclusions can be drawn and the information can also be used to derive an accurate model of the interfering signal, which in turn can help filter it.

Off-the-shelf GNSS receivers will typically not provide these samples, so special hardware is required to acquire them. Simple *Software Defined Radios* (SDRs) or more precise, expensive radio front-ends can be used for example to acquire these samples. These samples can be recorded for later inspection and processing.

### 2.2.5. C/N0 monitoring

In order to give users a metric of how well the satellites are received, most commercial GNSS receivers give an estimate of the carrier-to-noise-density ratio,  $C/N_0$ . This metric is calculated based on the signal demodulation and pseudo range measurements, so this is at the last stage of the receiver. Although a jammer signal is not exactly noise, it will still cause a serious drop in the  $C/N_0$  because the receiver has much more trouble finding the carrier. Because of this, the metric can be used to determine whether an interferer is present [16].

This technique has the advantage that virtually all GNSS receivers give a  $C/N_0$  estimate, so off-the-shelf receivers can be used. If the  $C/N_0$  drops for another reason than an interferer being present, this cannot be seen. This is because there is no further information on the received signal. For example, imagine driving past large buildings or into a tunnel. This will greatly drop the carrier-to-noise-density-ratio but it has nothing to do with an interferer. For this reason, only if the receiver is stationary and the

surroundings are free of obstructions, this method will yield accurate results [15, p. 18].

Another disadvantage is that there is no way of controlling how the receiver does the calculation of the metric. Also, determining a proper threshold might be challenging.

## 2.3. GNSS interference monitoring

The increasing awareness of the threat of GNSS interference has fueled various interference monitoring initiatives. Generally the goal of these initiatives is to gain more insight in the actual occurrence of interference and the nature of the interfering waveforms. Improved understanding of the interference threat will be helpful making GNSS receivers more robust against this threat.

### 2.3.1. STRIKE3

The STRIKE3 project was an effort to standardise GNSS threat reporting internationally, (co-)funded by the *European Global Navigation Satellite Systems Agency* (GSA) and conducted by a consortium consisting of a number of institutions. The project was active from 2016 to 2019 and its main focus was to develop a threat monitoring and reporting standard and to get an understanding of how widespread and severe GNSS interference is on a global level. As part of the STRIKE3 project dozens of interference sensors were deployed in 23 countries around the world, collecting more than 450,000 interference events in the L1/E1 band [17].

### 2.3.2. SENTINEL

SENTINEL was an earlier project originating in the UK, which was concluded in 2014. This project mainly aimed at deploying sensors around the globe to get more information on GNSS threats and also to create more awareness for GNSS vulnerabilities. It states that many high-profile services depend on GNSS without acknowledging the risks of this. As this report is written in 2014, an important remark is also made that jamming not only affects GPS but also other GNSS services such as Glonass and the - at the time very new - Galileo [18].

### 2.3.3. J911

Although this idea has not been implemented (yet), it is worth mentioning. In 2010 Scott [19] describes a distributed GNSS monitoring network using mobile phones. Although GNSS receivers in phones might not be of the highest quality, they could provide an estimate of the *jammer-to-noise-ratio* ( $J/N$ ) ratio based on available metrics, such as the AGC control signal. With so many sensors working together, it should be possible to deliver accurate detections. Also, since GNSS equipped smart phones are so ubiquitous, the monitoring network would cover a great portion of the globe. The idea of using smart-phones to form a large interference detecting network is still being pursued. In recent years Google has actually made changes to the Android operating system (Android 7 and later) of smartphones to allow access to GNSS raw measurement data to support this effort [20, 21].

### 2.3.4. NLR

An interference monitoring system from Nottingham Scientific Limited (NSL) can be (and is) used to detect interference events, which the NLR has deployed in Amsterdam. The system - GSS200D, see figure 2.10 - works autonomously and monitors the GNSS frequency bands continuously. As soon as an increase in received RF signal power is detected an interference event is registered, together with the date and time of the event. The system also registers and stores the duration of the event and the variation in signal power strength during the event. For signal characterisation and further analysis, a 100ms long snapshot is stored at the moment the maximum power is reached. These snapshots consist of IQ samples with 2-bit resolution per phase, so a total of 4 bits per sample. The sampling rate is 52MS/s.<sup>1</sup>

Signals that have been recorded by this system will be used in chapter 5 to verify the proposed detection and filtering algorithms on real signals.

---

<sup>1</sup>More information:

<https://www.gpsworld.com/spirent-gss200d-automates-monitoring-and-analysis-of-rf-interference/>  
and <https://gmvnsl.com/gnss-environment/gnss-interference-detection>



Figure 2.10: The GNSS detector from NSL (GSS200D)

# 3

## Chirps estimation

This chapter will give a detailed overview of estimation techniques from literature that may be applied to chirp interference in GNSS applications. Before the estimators are presented, a mathematical description of the signal model is derived. At the end of the chapter a chirp estimator is proposed, which is composed of some of the elements of the discussed estimators from literature. This proposed estimator is then validated by testing it in simulations.

The estimation accuracy is very important in this thesis, because of the filtering technique that was chosen. As will be clarified in the next chapter, the interferer will be removed using a time-domain subtraction. This requires accurate knowledge of the interfering signal, because a replica needs to be constructed. To get this accurate replica, this chapter attempts to create an estimator that is as accurate as possible.

### 3.1. Signal model

In this section a general signal model is described that is used throughout this thesis. This is the received signal after the ADC in the receiver model. This means it has passed the radio front end and any analog steps and has been downconverted, discretised and quantised. Any effects of the front-end, such as nonlinearities or filtering are not explicitly treated. Any noise, such as thermal noise or quantisation noise is combined together in one additive complex Gaussian noise term.

The received signal consists of the superposition of the signals from a number of satellites, noise and possible interferers. Only one interferer is assumed to be present at the receiver. The model then looks as follows

$$x[n] = \sum_i (s_i * h_{s_i})[n] + (u_s * h_u)[n] + w[n] \quad \mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}). \quad (3.1)$$

In this equation,  $s_i[n]$  are the signals from the different satellites. Their signals do not appear right at the receiver, but travel all the way from space to the antenna of the receiver. Along its way the signal will be attenuated and possibly filtered or reflected for example. These effects are captured in this equation by the channel impulse response  $h_{s_i}[n]$ . Mathematically this results in the convolution between the signal at the source (the satellite in this case) and the channel impulse response.

The unwanted interferer is denoted by  $u_s[n]$ . This signal also first passes through a certain channel and its impulse response is denoted by  $h_u[n]$ . This channel again accounts for effects such as reflections or filters.

Finally, the noise  $w[n]$  is assumed to be complex additive Gaussian noise with some covariance matrix  $\mathbf{C}$ . As described earlier this consists of all the noise in the system, such as RF noise, thermal noise and quantisation noise.

Some assumptions can be made to simplify the equation. First of all, note that in this thesis the primary interest is not in the signals coming from the satellites, but in the interfering signals. Since the signals from space are of extremely low power and are also of relatively wide band, their contribution may be regarded as noise as well and the term can be discarded.

The interferer term may also be simplified. The eventual goal is to remove the interfering signal from the received signal. Since what is received is the interfering signal passed through its channel and not the interfering signal at the jammer (source), it is more interesting to find a model for  $(u_s * h_u)[n]$  than for  $u_s[n]$ . As long as the effect of the channel is minimal in the sense that the received signal is still a chirp and there are no prominent reflections, then this is a reasonable idea. The received interferer as it has passed through the channel will be referred to as  $u[n]$ .

If the aforementioned assumptions are applied to (3.1) then the signal model to work with is

$$x[n] = u[n] + w'[n] \quad \mathbf{w}' \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}'). \quad (3.2)$$

For clarity the noise term has been denoted as  $w'$  to emphasise that the contribution of the satellites has been captured in it as well. From here on the primes will be dropped.

### 3.1.1. Mathematical description of chirp

As mentioned before, chirps are a form of frequency modulated signals. This means they can be described as a sinusoid where the frequency is a function of time. Starting with a complex sinusoid, i.e.

$$u(t) = \exp(j2\pi\phi(t)), \quad (3.3)$$

let us define the instantaneous frequency as the derivative of the phase, i.e.

$$f(t) = \frac{d}{dt}\phi(t). \quad (3.4)$$

*For reference, for a continuous wave signal, i.e. a normal (complex) sinusoid a function for the phase would be  $\phi(t) = f_0t + \phi_0$ , resulting in a constant frequency of  $f_0$ .*

For a linear chirp, a function for the phase is

$$\phi(t) = f_0t + \frac{\alpha}{2}t^2 + \phi_0. \quad (3.5)$$

This gives an instantaneous frequency of

$$f(t) = \frac{d}{dt}\phi(t) = f_0 + \alpha t. \quad (3.6)$$

This is indeed a linear function of  $t$  with a certain initial frequency  $f_0$ . The parameter  $\alpha$  is called the chirp rate. The chirp model in the continuous time domain then is

$$u(t) = \exp j[2\pi(f_0t + \frac{1}{2}\alpha t^2) + \phi_0], \quad t \in \mathbb{R}, \quad (3.7)$$

where the initial phase  $\phi_0$  has been brought outside so it is not normalised like the frequency, but is a phase in radians. Similarly, the discrete time domain signal is

$$u[n] = \exp j[2\pi(f_0n + \frac{1}{2}\alpha n^2) + \phi_0], \quad n = 0, 1, \dots, N-1. \quad (3.8)$$

This means that a chirp is described by three parameters, namely the initial phase  $\phi$ , initial frequency  $f_0$  and chirp rate  $\alpha$ . Figure 3.1 shows the time-frequency domain representation of such a chirp along with the mentioned parameters. Note that since this is in discrete time, the normalised frequency ( $f = \omega/(2\pi)$ , where  $\omega$  is angular velocity) is bounded to the domain  $[-1/2, 1/2)$  and therefore the instantaneous frequency keeps wrapping around. This gives the chirp its distinct sawtooth-like shape in the time-frequency domain. In the next section the case will be addressed where the instantaneous frequency is bounded to a smaller bandwidth.

After incorporating the noise term from (3.2), the model is

$$x[n] = \exp j[2\pi(f_0n + \frac{1}{2}\alpha n^2) + \phi_0] + w[n], \quad n = 0, 1, \dots, N-1 \quad w[n] \sim \mathcal{N}(0, \sigma^2). \quad (3.9)$$

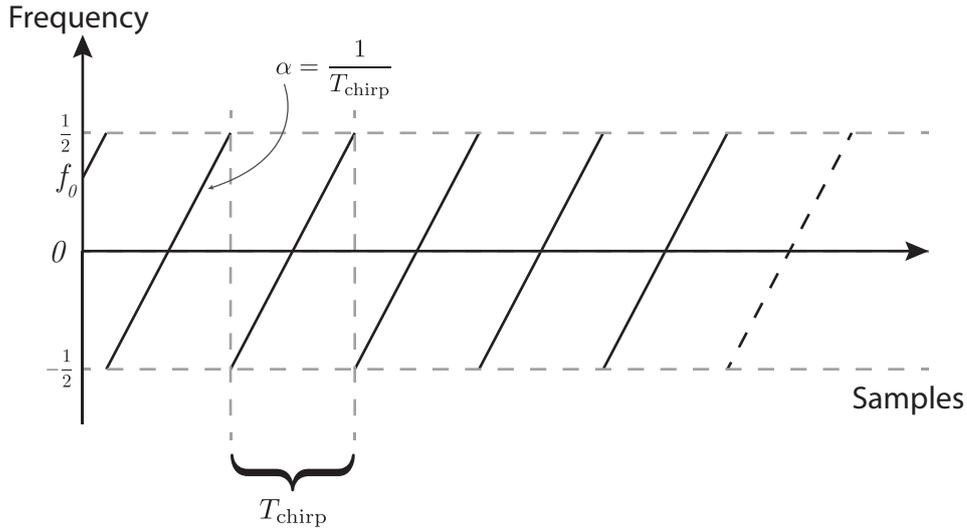


Figure 3.1: The parameters of a chirp, shown in the time-frequency domain

### 3.1.2. Band-Limited chirps

In practical situations where measurements of an unknown chirp signal are performed, it is very unlikely that the instantaneous bandwidth of the measurement equipment matches the bandwidth of the chirp. Usually the chirp will either have a smaller bandwidth or a larger bandwidth. If the chirp has a larger bandwidth, it will be outside the measured frequency range for at least part of each period. If the equipment has proper out-of-band rejection, that part of the chirp is thus suppressed. Also a chirp with a small bandwidth can be outside the measured frequency range, because the central frequency of the chirp might have an offset with respect to the measured band. The current paragraph explains how these practical issues can be taken into account in the signal model. Only the case where the chirp has a smaller bandwidth than the measurement bandwidth and the chirp falls entirely within this measurement bandwidth is discussed.

Mathematically, the previous section assumed a chirp with an ever increasing instantaneous frequency. Due to the discrete time, this instantaneous frequency keeps wrapping around the  $[-1/2, 1/2)$  interval (for normalised frequencies). In this section the interval around which it keeps wrapping around is smaller, but still within  $[-1/2, 1/2)$ .

In order to describe the band-limited chirp, some new parameters are required. These could either be the centre frequency  $f_c$  and the bandwidth  $B$  or, likewise, the lower bound  $f_{LO}$  and upper bound  $f_{HI}$ . These both contain the same information. Additionally, the chirp period depends on both the bandwidth and the chirp rate in this scenario as  $T_{\text{chirp}} = B/\alpha$ . This new case is shown in figure 3.2.

Let the instantaneous frequency of the band-limited chirp be defined as

$$f_i[n] = (\alpha n - f'_0 \bmod B) - \frac{B}{2} + f_c. \quad (3.10)$$

Note first that the periodicity arises from the modulo operation <sup>1</sup>. This operation ensures that the instantaneous frequency stays within the bandwidth  $B$  and wraps around when it reaches the top. The subtraction by  $\frac{B}{2}$  translates the frequency from the  $[0, B)$  to  $[-B/2, B/2)$  range and finally the centre frequency  $f_c$  is added so that the frequency is centered around it, i.e. the range is now  $[f_c - B/2, f_c + B/2)$ .

Another thing to take note of is that  $f'_0$  is used, this is the offset from the lower bound, i.e. for  $f'_0 = 0$  the chirp starts out at  $f_{LO}$ . The relation to  $f_0$  is then  $f'_0 = f_0 - f_{LO} = f_0 - f_c + B/2$ .

Now to be able to construct a chirp, the phase needs to be found as the integral of this initial frequency. Being careful with the modulo operator, the following expression is found

<sup>1</sup>This is the floored division modulo, defined as  $a \bmod b := a - b \left\lfloor \frac{a}{b} \right\rfloor$

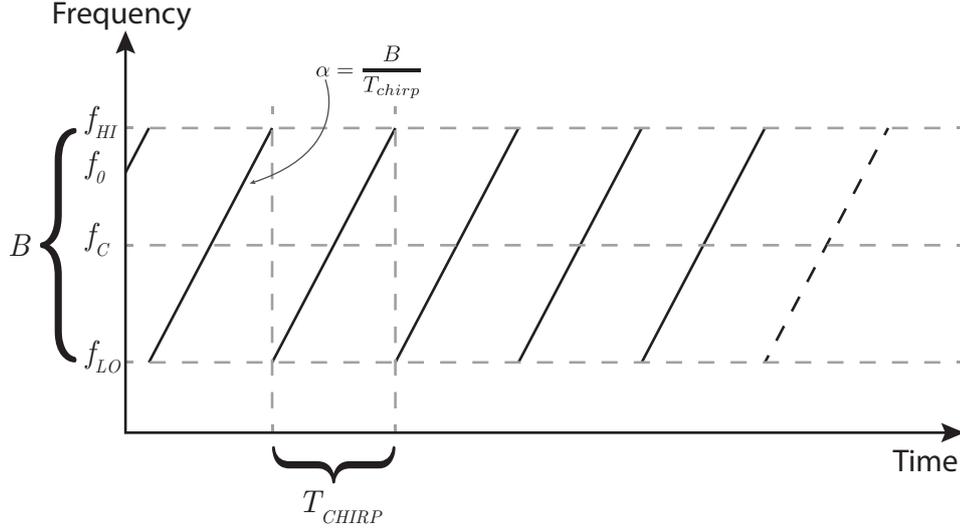


Figure 3.2: The parameters that define the band-limited chirp

$$\phi[n] = \frac{1}{2\alpha}(\alpha n + f'_0 \bmod B)^2 + \frac{B^2}{2\alpha} \left\lfloor \frac{\alpha n}{B} \right\rfloor + f_c n + \phi_0. \quad (3.11)$$

There is still a modulo operator in the expression accounting for the periodicity and the middle term of the equation makes sure that the phase has no discontinuities. Finally there is an initial phase  $\phi_0$ . Assuming the frequencies are normalised, the full chirp along with noise is then again defined as

$$u[n] = \exp(j2\pi\phi[n]) + w[n]. \quad (3.12)$$

### 3.1.3. Window choice

For estimation of the chirp rate, it might be more useful to work with a shorter window of the signal than working with the entire signal. In fact, the difficulty of dealing with the wrapping that arises with the band-limited chirp can be mitigated by choosing the window in such a way that the signal does not wrap around within the window. This requires finding the chirp period and also the point in time when the wrapping occurs beforehand. If this is achieved, chirp rate estimators that work better on full-band chirps can be leveraged.

## 3.2. Theoretical bound of estimator

An estimate of some parameter of a stochastic variable is itself a stochastic variable. In other words, when trying to estimate parameters from noisy observations, the estimate will inherently be noisy as well. It is often possible, however, to reduce the variance of the estimate by using more realisations of the stochastic variable for example. Intuitively this can be understood as follows. Imagine a stochastic variable  $X$ , which is normally distributed as  $X \sim \mathcal{N}(0, \sigma_x^2)$ . An estimator could be  $\hat{x} = x$ , where  $x$  is one realisation of  $X$ . The estimator has the same variance as  $X$ , i.e.  $\hat{x} \sim \mathcal{N}(0, \sigma_x^2)$ . One could alternatively use multiple realisations  $x_i$  of  $X$  and use the estimator  $\hat{x} = \frac{1}{N} \sum_i x_i$ . Intuitively, this reduces the variance of the estimator, so that  $\hat{x} \sim \mathcal{N}(0, \sigma_{\hat{x}}^2)$  where  $\sigma_{\hat{x}}^2 < \sigma_x^2$ . If the realisations of  $X$  are *independent and identically distributed* (iid) then in fact  $\sigma_{\hat{x}}^2 = \frac{1}{N} \sigma_x^2$ .

It turns out that there is a lower bound on the variance of any unbiased estimator, called the *Cramér-Rao Lower Bound* (CRLB) [22, p. 40]. This lower bound is studied in this section for the estimation problem at hand. This metric can then be used later on to compare estimators to.

This derivation assumes a chirp that does not wrap around, which can be achieved by appropriate windowing as stated in the previous section. The interferer can stochastically be described as a deterministic signal with unknown parameters buried in random noise. The vector  $\mathbf{u}$ , which is the vector of samples of  $u$  is then distributed as

$$\mathbf{u} \sim \mathcal{CN}(\boldsymbol{\mu}(\boldsymbol{\theta}), \sigma^2 \mathbf{I}). \quad (3.13)$$

Here the noise is assumed to be white. The vector  $\boldsymbol{\theta}$  contains the deterministic, but unknown parameters, i.e.  $\boldsymbol{\theta} = [A \ f_0 \ \alpha \ \phi_0]^T$ . The mean of the distribution in (3.13) is the deterministic chirp as described by the first term in (3.12). The variance stems purely from the noise term  $w[n]$  in (3.12). The mean can be written in vector form as

$$\boldsymbol{\mu}(\boldsymbol{\theta}) = \begin{bmatrix} A \exp j[2\pi(f_0 \cdot 0 + \frac{1}{2}\alpha \cdot 0^2) + \phi_0] \\ A \exp j[2\pi(f_0 \cdot 1 + \frac{1}{2}\alpha \cdot 1^2) + \phi_0] \\ A \exp j[2\pi(f_0 \cdot 2 + \frac{1}{2}\alpha \cdot 2^2) + \phi_0] \\ \vdots \\ A \exp j[2\pi(f_0 \cdot (N-1) + \frac{1}{2}\alpha \cdot (N-1)^2) + \phi_0] \end{bmatrix}. \quad (3.14)$$

To find lower bounds for the variance of said parameters first the Fisher information matrix, commonly denoted by  $\mathbf{I}(\boldsymbol{\theta})$ , must be found. The variance of an estimator  $\hat{\theta}_i$  of parameter  $\theta$  is then bounded by

$$\text{var}(\hat{\theta}_i) \geq [\mathbf{I}^{-1}(\boldsymbol{\theta})]_{ii} \quad (3.15)$$

where  $[\cdot]_{ii}$  denotes the  $ii$ 'th element of the matrix and the  $\mathbf{I}^{-1}(\boldsymbol{\theta})$  is the inverse of that Fisher information matrix.

For the case of a complex signal in complex Gaussian noise a useful representation for  $\mathbf{I}(\boldsymbol{\theta})$  is as follows [22, pp. 525]:

$$[\mathbf{I}(\boldsymbol{\theta})]_{ij} = \text{tr} \left[ \mathbf{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \mathbf{C}(\boldsymbol{\theta})}{\partial \theta_i} \mathbf{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \mathbf{C}(\boldsymbol{\theta})}{\partial \theta_j} \right] + 2\Re \left[ \frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\theta})}{\partial \theta_i} \mathbf{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_j} \right]. \quad (3.16)$$

where  $\mathbf{C}(\boldsymbol{\theta})$  is the covariance matrix of  $\mathbf{u}$  as a function of  $\boldsymbol{\theta}$ . Note that in our case the covariance matrix is equal to  $\sigma^2 \mathbf{I}$  (i.e. white noise) and does not depend on  $\boldsymbol{\theta}$ . This means the first term of (3.16) vanishes.

Working out the partial derivatives in (3.16) using (3.14) leads to the following expression for  $\mathbf{I}(\boldsymbol{\theta})$

$$\mathbf{I}(\boldsymbol{\theta}) = \frac{2}{\sigma^2} \begin{bmatrix} N & 0 & 0 & 0 \\ 0 & \frac{2}{3}\pi^2 A^2 N(N+1)(2N+1) & \frac{1}{2}\pi^2 A^2 N^2(N+1)^2 & \pi A^2 N(N+1) \\ 0 & \frac{1}{2}\pi^2 A^2 N^2(N+1)^2 & \frac{1}{30}\pi^2 A^2 N(N+1)(2N+1)(3N^2+3N-1) & \frac{1}{6}\pi A^2 N(N+1)(2N+1) \\ 0 & \pi A^2 N(N+1) & \frac{1}{6}\pi A^2 N(N+1)(2N+1) & N A^2 \end{bmatrix}. \quad (3.17)$$

This matrix is hard to invert by hand so for the calculation of the lower bound the inverse is calculated using Matlab. The result of this can be seen for  $\alpha$  as the CRLB line in figure 3.4a.

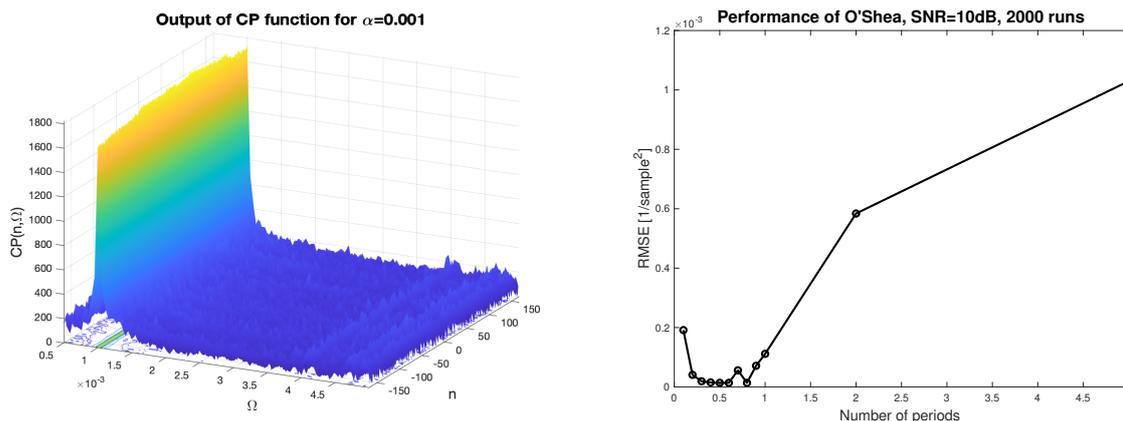
Since the CRLB gives the lowest possible variation on any estimator for the given parameter, it is a solid metric to validate estimators against.

### 3.3. Estimation algorithms from literature

With the ultimate goal in mind of being able to filter out chirp signals, the idea is to first estimate the signal parameters as well as possible. The more accurately the interfering signal is known, the better it can be filtered out. A number of techniques from literature have been analysed. All the relevant techniques will be discussed below. All of those techniques stem from signal processing papers, mostly only loosely tied to specific applications. The most prevalent applications mentioned are radar and sonar.

#### 3.3.1. Cubic Phase function

In 2002 O'Shea proposed to use the cubic phase function to find the chirp rate [23]. It assumes a signal model very similar to (3.9), but the indices  $n$  are chosen differently. In this paper  $n$  is chosen such that  $n = 0$  is at the centre of the data, i.e.  $n = -\frac{N-1}{2}, \dots, -1, 0, 1, \dots, \frac{N-1}{2}$ . The discrete cubic phase function is then defined as



(a) Surface and contour plot of cubic phase function

(b) Performance of O'Shea for different sample lengths

Figure 3.3: O'Shea algorithm figures

$$CP(n, \Omega) = \sum_{m=0}^{(N-1)/2} x[n+m]x[n-m]e^{-j\Omega m^2} \quad (3.18)$$

and it is maximal for  $\Omega$  equal to the chirp rate. It is also shown to be asymptotically optimal at the centre of the data record, so that our interest is only at  $n = 0$  (the centre of the data). Figure 3.3a shows that the choice of  $n$  does not matter much, so  $n = 0$  is fine. The chirp rate is then found from

$$\hat{\alpha} = \arg \max_{\Omega} CP(0, \Omega). \quad (3.19)$$

The paper states that the algorithm works from about -4dB SNR for a 515-point signal. This is comparable with the results obtained in this thesis, which are plotted in figure 3.4a (denoted by O'Shea). Upon testing the algorithm, it turned out the number of samples used has great impact on the performance of the estimator. If too few or too many samples are used, the performance degrades rapidly. In fact it appeared that the number of samples required depends on the chirp rate. In order to find out if that was the case and what the optimal number of samples would be, some simulations were run. For a fixed SNR of 10dB, 2000 Monte Carlo simulations were run where the number of samples fed to the algorithm was proportional to the chirp period. This proportion was then varied from 0.1 to 5 periods. The results are plotted in figure 3.3b. The horizontal axis shows the proportion and the vertical axis shows the mean square error (as defined in (3.24)) of the estimator on a dB scale. It can be seen that the estimator actually performs best when it receives around 0.7 chirp periods.

This algorithm also works on real chirps as long as the number of samples used is chosen properly. This requires knowing the chirp period beforehand, meaning that an additional algorithm is needed to provide this period. The O'Shea algorithm only finds the chirp rate, so no information about initial frequency, phase or the bandwidth and centre frequency are obtained.

### 3.3.2. (modified) Discrete Chirp Fourier Transform

A rather straightforward approach to finding the initial frequency and chirp rate is the so called *Discrete Chirp Fourier Transform* (DCFT), which is in fact just a matched filter. This method correlates the received signal  $x$  with chirps for varying initial frequency and chirp rate. It basically works like a Discrete Cosine Transform, but extended with a chirp rate. The parameters are found by maximising the correlation for the unknowns,  $f_0$  and  $\alpha$  [22, 24, 25]:

$$\arg \max_{f_0, \alpha} \left| \sum_{n=0}^{N-1} x[n] \exp \left[ -j2\pi \left( f_0 n + \frac{\alpha}{2} n^2 \right) \right] \right|. \quad (3.20)$$

In theory the matched filter, which this is, is optimal for a single signal in white noise [26]. [25] claims to attain the CRLB for around 15dB SNR and [24] requires around 0dB SNR to use this technique

properly. This does require the number of samples to be chosen as a prime number [24], however this problem can be overcome by using a slightly modified version: the mDCFT [27]. It is also shown by [28] that the technique can be combined with a compressed sensing technique obtaining proper results down to around -7dB SNR.

Another problem with the DCFT is that it requires a search over two dimensions to find the global maximum. This makes it a computationally demanding task. It is however possible to speed up these computations by employing an algorithm like Newton's method.

This algorithm estimates both the initial frequency and the chirp rate. When applied to a band-limited chirp, the accuracy (MSE) is far worse in most cases than when applied to a full-band chirp (remember section 3.1.2). This is due to the fact that the chirp used in the correlation in (3.20) is also a full-band chirp. A way to overcome this problem is to apply the DCFT to carefully selected window of the original signal. This window should be exactly one chirp period long and aligned in such a way that the chirp does not wrap around within the window. This will however require additional algorithms to find the chirp period and also when the chirp wraps back around.

### 3.3.3. Phase unwrapping

In 1989 Lang and Musicus [29] and in 1990 Djurić and Kay [30] proposed similar methods to find the chirp rate. They assume that the SNR is sufficiently high so that the additive noise may be approximated by phase noise. They then rely on the fact that the phase is the integral of the frequency or the double integral of the chirp rate. This relation can be expressed as

$$\phi = \mathbf{G}\theta = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ \vdots & \vdots & \vdots \\ 1 & N & N^2 \end{bmatrix} \begin{bmatrix} \phi_0 \\ f_0 \\ \alpha \end{bmatrix}. \quad (3.21)$$

Here  $\phi$  is the vector containing the phase of each sample of  $\mathbf{x}$ . This phase is unbounded, i.e. it does not wrap. The parameters  $\phi_0$ ,  $f_0$  and  $\alpha$  are the initial phase, initial frequency and chirp rate respectively.

Because the phases of the samples of  $\mathbf{x}$  are bounded on the interval  $[-\pi/2, \pi/2)$  they first need to be unwrapped. In [30] this is achieved by first taking the second derivative and then integrating twice.

The estimates for the parameters can then be found from (3.21) as

$$\hat{\theta} = (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T \phi \quad (3.22)$$

where  $\hat{\theta}$  contains the estimates of the parameters.

This method works relatively well for SNR larger than around 8 dB. For SNRs from 12 dB the algorithm actually attains the CRLB, as can be seen in figure 3.4a (denoted Djuric). Because the estimator only works for SNRs  $>8$  dB it is not very suitable in this thesis.

### 3.3.4. Hankel Rank Reduction

DiMonte and Arun proposed a Hankel Rank Reduction method in 1990 [31] to find the instantaneous frequency of a signal at different time instants. It creates a Hankel matrix from the data with  $c$  columns, where  $c$  is chosen to be on the order of three times the number of real sinusoids in the signal. The frequencies are found only from the principal singular vectors and values.

Contrary to most other algorithms, this method does not assume a constant chirp rate for the whole signal, but rather estimates the instantaneous frequency over time. This gives fuzzy results and does not directly give an estimate of the chirp rate. It is however possible to fit a sawtooth function to the estimated instantaneous frequencies and find the chirp rate from the slope of the sawtooth. This means an additional and computationally burdensome step, making the algorithm less suited for estimating chirps with constant chirp rate. If however, nonlinear chirps needed to be estimated, this could be an interesting approach.

Another approach is given by Völcker and Ottersten in 1998 [32] and 2001 [33]. They are not after the chirp rate or initial frequency, but rather after the bandwidth and centre frequency. These are found by again constructing a Hankel matrix from the signal and from this get a covariance matrix. By applying root-MUSIC or a similar algorithm to the covariance matrix to estimate two frequencies. The

centre frequency and bandwidth are then found from those frequencies. According to some Matlab simulations, the algorithm works from around 10 dB SNR.

A similar method can be applied to a tensor of order 3, which is constructed in a similar way as a Hankel matrix. The decomposition into subspaces is then performed by Tucker's decomposition and it the algorithm works fairly the same as the previous methods. The results appear to be slightly better according to [34].

### 3.3.5. Radon transform of Ambiguity Function

Wang et al. presents a totally different approach in 1998 [35]. This approach applies the ambiguity function to the received signal. In the continuous-time domain this function is

$$\text{AF}_x(\tau, \omega) = \int_{-\infty}^{\infty} x(t + \tau/2)x^*(t - \tau/2)e^{-j\omega t} dt. \quad (3.23)$$

Discrete implementations exist, such as Matlab's `ambgfun`<sup>2</sup>. After applying this function, the chirp is transformed into a line that passes through the origin and the slope of the line is proportional to the chirp rate. To find the slope of the line, the Radon transform is applied. The Radon transform is maximal for the angle that corresponds with the slope of the line and so the chirp rate is found. The accuracy of this transform is outstanding, yielding very accurate estimates at SNR as low as -12dB. The algorithm however only provides the chirp rate, not the initial frequency, initial phase or bandwidth and centre frequency. Another drawback of the algorithm is that it is extremely computationally demanding, as can be seen in figure 3.4b (denoted Wang).

### 3.3.6. Fractional Fourier Transform

Song et al. [36] suggest using the *Fractional Fourier Transform* (FrFT) to estimate the chirp parameters. The FrFT effectively rotates the time-frequency plane over a certain rotation angle. There will be a maximum for a certain rotation and offset in the transformed domain. These values are proportional to the initial phase, initial frequency, chirp rate and amplitude of the chirp. This algorithm has high accuracy from roughly -9 dB SNR, as can be seen in figure 3.4a. The algorithm however takes a long time to compute, which can be seen in figure 3.4b. This transform might be useful in filtering, because it can transfer the signal into a domain where the chirp is sparse. This has not been tested in this thesis however.

### 3.3.7. Conclusion on estimators

A number of different estimators have been presented that work based on different principles. Table 3.1 summarises the list and shows which parameters are estimated.

The estimators have been implemented in Matlab to verify their accuracy and complexity. Not all algorithms find the same set of parameters of the chirp, however most of them find at least the chirp rate. For this reason a more thorough comparison of the chirp rate estimation performance has been made.

Each of the estimators has been tested using 1000 Monte Carlo simulations. For each Monte Carlo simulation a new chirp of 1000 samples was generated with a random chirp rate. This chirp rate was uniformly distributed between 0.54 MHz/ $\mu$ s and 10.8 MHz/ $\mu$ s, with a sampling frequency of  $f_s = 52$ MS/s. Furthermore, each chirp had a full band or  $B = f_s$  and the initial frequency and initial phase were chosen randomly. For each simulation also a new noise signal was generated consisting of unit variance complex Gaussian noise. The chirp was then multiplied by an amplitude  $A$  and added to the noise. The amplitude  $A$  was calculated from the desired SNR as  $A = \sqrt{10^{\frac{\text{SNR}}{10}}}$ , where SNR is in dB.

For each of the Monte Carlo runs the difference between the estimated chirp rate and the actual chirp rate was calculated and used to determine the *root mean squared error* (RMSE), which is defined as

$$\text{RMSE}(\alpha, \hat{\alpha}) = \sqrt{\frac{1}{N_{\text{mc}}} \sum_{i=1}^{N_{\text{mc}}} (\alpha - \hat{\alpha}_i)^2} \quad (3.24)$$

<sup>2</sup><https://nl.mathworks.com/help/phased/ref/ambgfun.html>

Estimator	Papers	Parameters
Cubic Phase function	<ul style="list-style-type: none"> <li>O'Shea (2002) [23]</li> </ul>	<ul style="list-style-type: none"> <li>Frequency rate</li> </ul>
DCFT	<ul style="list-style-type: none"> <li>Abatzoglou (1986) [25]</li> <li>Irkhis and Shaw (2019) [28]</li> <li>Pingyi Fan and Xiang-Gen Xia (2000) [27]</li> <li>Xiang-Gen Xia (2000) [24]</li> </ul>	<ul style="list-style-type: none"> <li>Frequency</li> <li>Frequency rate</li> </ul>
Phase unwrapping	<ul style="list-style-type: none"> <li>Lang and Musicus (1989) [29]</li> <li>Djurić and Kay (1990) [30]</li> </ul>	<ul style="list-style-type: none"> <li>Phase</li> <li>Frequency</li> <li>Frequency rate</li> </ul>
Hankel reduction	<ul style="list-style-type: none"> <li>DiMonte and Arun (1990) [31]</li> <li>Völcker and Ottersten (1998 [32], 2001 [33])</li> <li>Ge et al. (2014) [34]</li> </ul>	<ul style="list-style-type: none"> <li>Instantaneous frequency</li> <li>Centre frequency</li> <li>Bandwidth</li> </ul>
Radon TF of AF	<ul style="list-style-type: none"> <li>Minsheng Wang et al. (1998) [35]</li> </ul>	<ul style="list-style-type: none"> <li>Frequency rate</li> </ul>
FrFT	<ul style="list-style-type: none"> <li>Song et al. (2013) [36]</li> </ul>	<ul style="list-style-type: none"> <li>Phase</li> <li>Frequency</li> <li>Frequency rate</li> <li>Amplitude</li> </ul>

Table 3.1: Overview of different estimation algorithms.

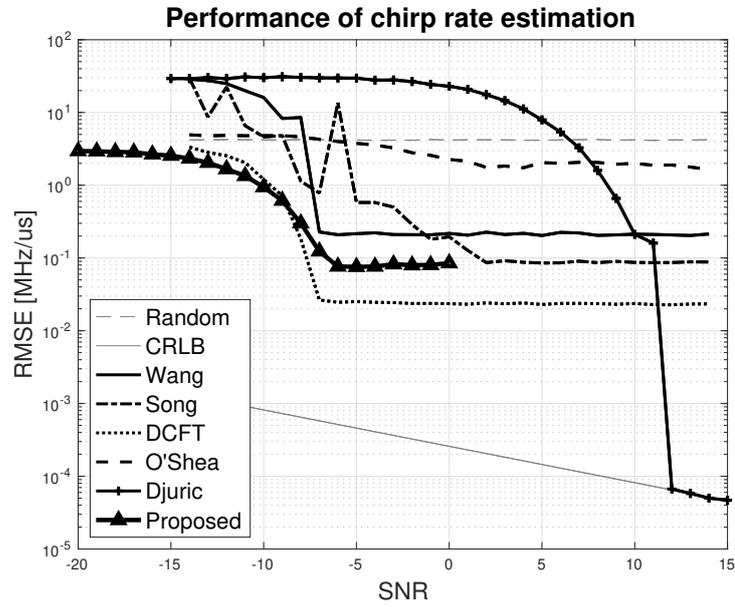
where  $\alpha$  are the actual chirp rates and  $\hat{\alpha}$  is the vector containing the estimates for all monte carlo runs. The results have been plotted in figure 3.4a on a logarithmic scale for better visibility. Since this is a metric for the *error*, a lower RMSE value means a better estimate. An upper and lower bound have also been plotted. The lower bound is the CRLB as discussed in section 3.2. The upper bound is a *random* estimator. The random estimator does not take into account any signal, but just chooses an estimated chirp rate at random (again within 0.54 MHz/ $\mu$ s and 10.8 MHz/ $\mu$ s). Anything above that line clearly has no meaning and only when an estimator gets below that line it starts to add information.

The consumed CPU time for the different estimators is plotted in figure 3.4b. This was run on a consumer grade laptop. Note that the absolute time obviously depends on used hardware, so only the difference between the estimators is of interest.

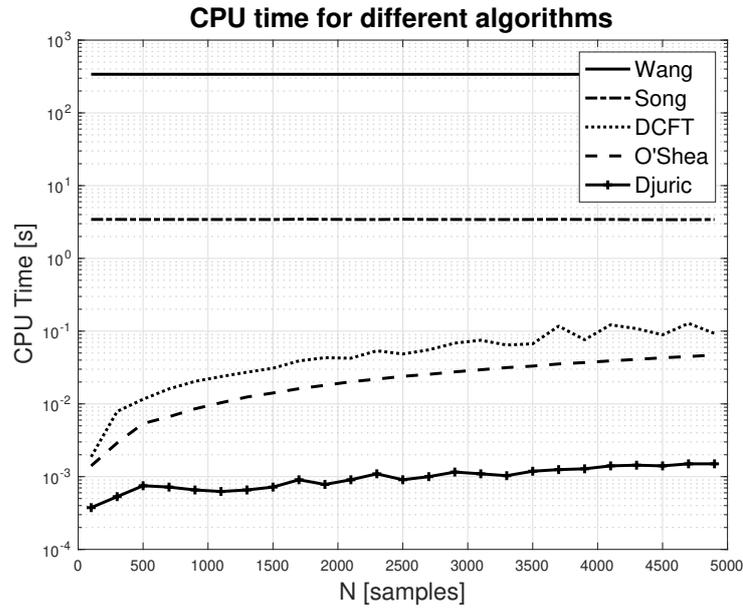
Looking at both figures a few conclusions can be drawn. With respect to computation time, Wang and Song are orders of magnitude slower than the other three. This renders them impractical. Turning now to the accuracy, it turns out Djuric is the only one that attains the CRLB. This, however, happens only for SNRs higher than around 12 dB. Below that the accuracy gets worse very rapidly. An overall performer is the DCFT, which has the best overall accuracy even for low SNRs and requires moderate CPU time.

### 3.4. Proposed estimation algorithm

In the previous sections some estimators have been discussed that provide estimates for different parameters. They however do not provide the total estimate of an arbitrary chirp, i.e. for a signal that has multiple chirp periods and is band-limited. In order to find all necessary parameters of such a signal with sufficient accuracy a combination is needed. In this section the proposed algorithm to find all parameters is presented.



(a) Performance comparison of  $\alpha$  estimators,  $N = 1000$  in the simulations.



(b) Comparison of CPU time for the different algorithms. Note that they are plotted on a logarithmic scale.

Figure 3.4: Estimation algorithms compared

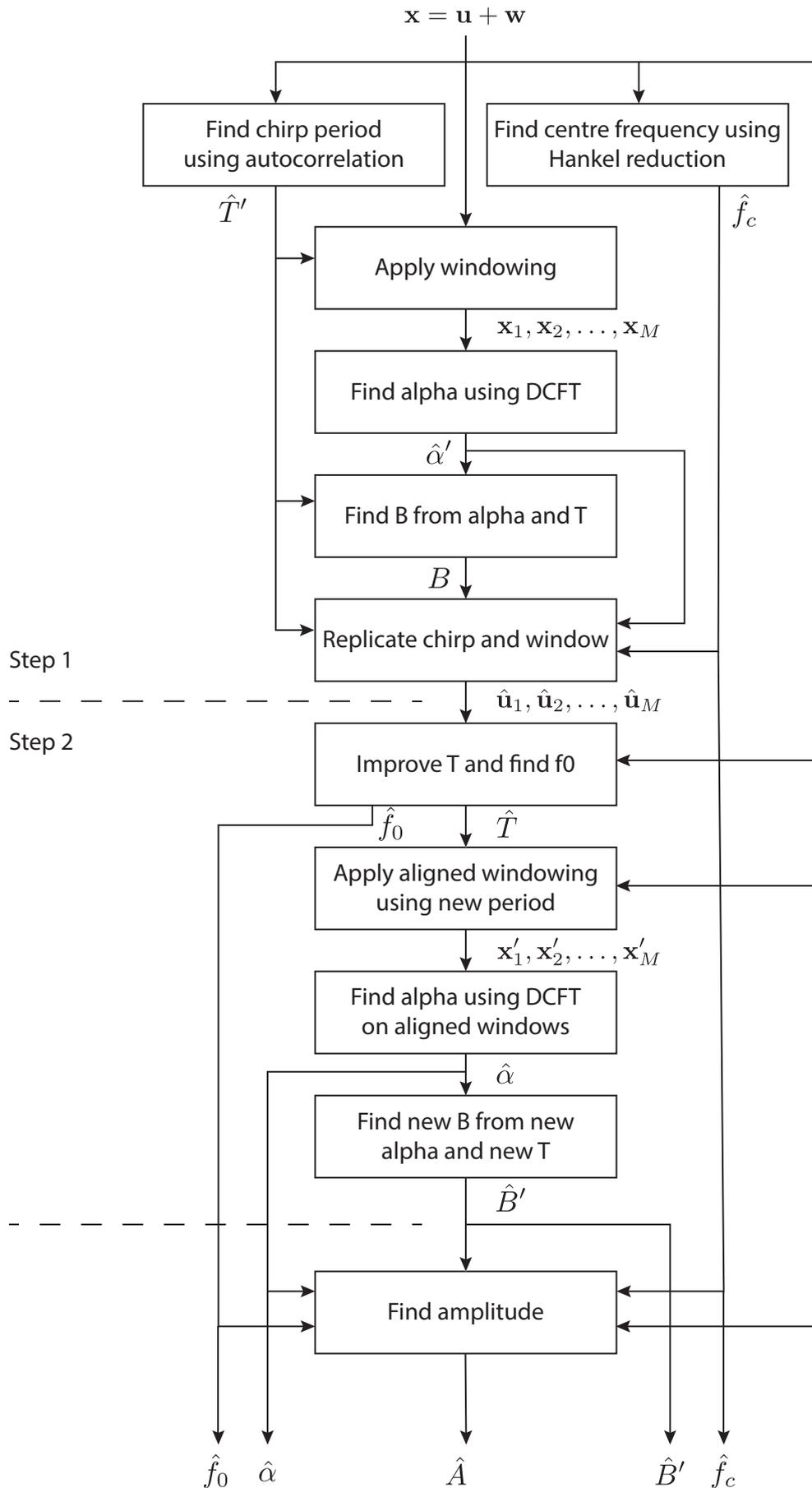


Figure 3.5: Proposed algorithm, schematically

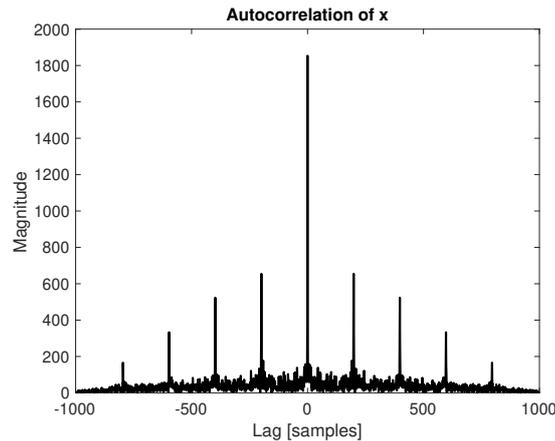


Figure 3.6: Autocorrelation of  $x$

Figure 3.5 shows schematically how all the pieces work together in this algorithm. The algorithm takes only the input signal  $x$  as input and outputs estimates for the chirp rate, bandwidth, initial frequency, centre frequency and amplitude. Signal  $x$  consists of noise ( $w$ ) and the chirp ( $u$ ), only the chirp is of interest. The general idea of the algorithm will be discussed first and then the details about each block in figure 3.5 will be investigated.

The algorithm consists of two main steps. In the first step the signal is windowed on a rough chirp period estimate and from these windows the chirp rate and bandwidth are found. In the second step, the error in the chirp period estimation is corrected and the windows are aligned in such a way that the chirp does not wrap around within the windows. The chirp rate estimate and bandwidth estimate are then improved using the new windows. The estimation of the centre frequency is performed over the entire input signal  $x$ , so this happens in parallel to the two-step approach. The amplitude is estimated last, because it requires the other estimated parameters.

**Coarse period estimation** In this step a coarse estimate is found for the chirp period. This is done using the autocorrelation of  $x$ , which is found as

$$r_x[n] = x[n] * x^*[-n] \quad (3.25)$$

where  $x^*[-n]$  is the complex conjugate of the time reverse of  $x$  and  $*$  denotes the convolution operator.

The autocorrelation typically looks like the one in figure 3.6. In this figure a very low number of samples is used to get a clearer plot. The chirp period corresponds with the distance between the high peaks that can be seen in the figure. Especially for low SNRs and for long data records, the peaks are far more prominent at the centre of the autocorrelation. Therefore only the middle 80% of the data is used.

The peaks are found using Matlab's `findpeaks` function. It is important to find the right peaks, i.e. the ones corresponding to the chirp period and not accidental peaks which are part of the noise. This can be achieved by constraining `findpeaks` to only find peaks above a certain threshold. Choosing the threshold is difficult however, because the peaks get lower further away from the centre of the autocorrelation. To overcome this, the autocorrelation is split into windows of 5001 samples (the whole signal is assumed to be at least 100 000 samples). The sample length is chosen such that a chirp period of 2500 samples (which corresponds to the slowest assumed chirp period) has two peaks within each window. In each of these windows the threshold can be set to 0.7 times the magnitude of the second highest peak. The second highest is chosen instead of the highest because the peak at zero lag is so much higher than the other peaks, that this would be a bad reference point. For the other windows it does not matter much if the highest or second highest is chosen.

For each window the time difference (in samples) between the peaks is found and they are stacked in a list. The final decision on the chirp period is then found from the mode of the time differences and

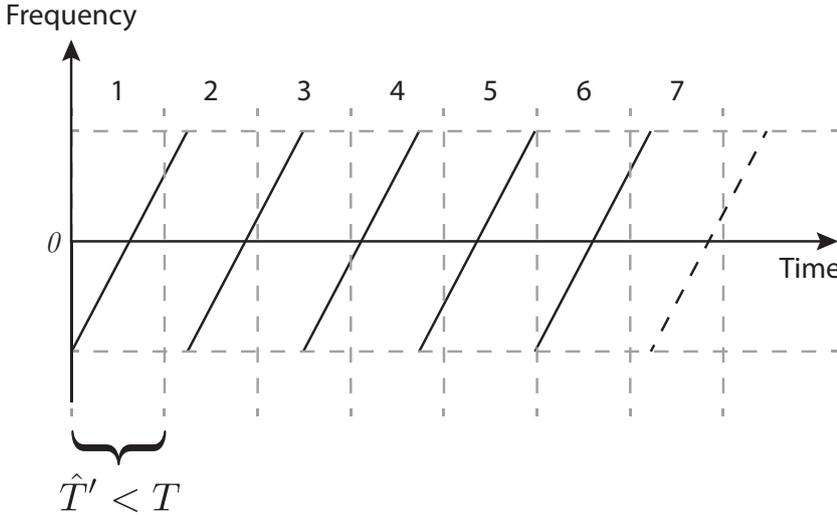


Figure 3.7: Window drift because of inaccurate window length.

will be referred to as  $\hat{T}'$ . The mode is chosen, because the estimates are all integer and there can be some serious outliers. The mode is robust against these problems.

**Windowing using coarse period** Having found the coarse period, which is an integer number of samples, the signal  $x$  can be divided into windows of 1 period length, the number of windows will be called  $M$ . One window is then simply defined as

$$\mathbf{x}_i = \begin{bmatrix} x[i\hat{T}'] \\ x[i\hat{T}' + 1] \\ \vdots \\ x[(i+1)\hat{T}' - 1] \end{bmatrix}, \quad i = 1, \dots, M \quad (3.26)$$

and contains  $\hat{T}'$  samples. Please note that due to the rounding of the chirp period and thus the window length, there will be a certain drift in the windowing. This is illustrated in figure 3.7, it can be seen that window 1 is very different from window 4 for example. This problem is solved in step two of the algorithm.

**Find coarse chirp rate** Based on the conclusion drawn in section 3.3.7, the DCFT is used as an estimator for the chirp rate. This estimator is applied to each window and then the mean over all the estimates is taken as the final estimate for the chirp rate.

Please note that since the windowing here is coarse and the windows are not aligned with the beginning of each chirp, the DCFT does not perform at its best. The reason is that a wrapping may occur within the window, which is not taken into account by the DCFT. Therefore this is only the coarse chirp rate estimate, denoted by  $\hat{\alpha}'$ .

**Find coarse bandwidth** Although Völcker and Ottersten's algorithm ([32], [33]) theoretically also provides an estimate of the bandwidth, this estimator has proven to work poorly, based on Matlab simulations. Therefore an alternative approach is taken.

The assumption is made that the chirp falls entirely within the receiver bandwidth and there are no pauses between chirps. This means that the chirp wraps around immediately from its highest to its lowest frequency (or the other way around). If this assumption holds, then the chirp rate, chirp period and bandwidth are related and the bandwidth can be found as

$$\hat{B}' = \hat{T}' \hat{\alpha}'. \quad (3.27)$$

Since the bandwidth is based on the coarse estimates, it is itself also a coarse estimate, hence the prime.

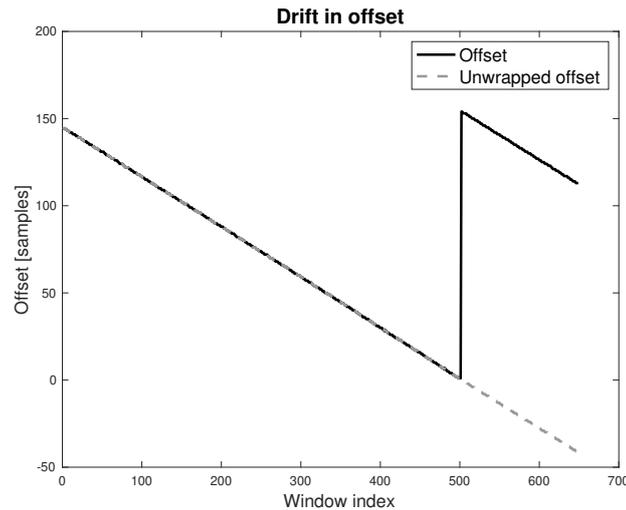


Figure 3.8: Offset as function of window index, showing window drift.

**Centre frequency** Referring back to table 3.1 the options for centre frequency estimation are limited and thus the algorithm by Völcker and Ottersten is implemented. The estimator is performed over the entire signal  $x$  as opposed to the windows.

**Coarse replica** Using the found coarse parameters  $\hat{B}'$  and  $\hat{\alpha}'$  and the centre frequency estimate  $\hat{f}_c$  a replica of the chirp can be generated by filling in these parameters in the chirp model of section 3.1.2. This replica will be called  $\hat{u}$  as it is a replica of the chirp only and not the noise. The same windowing is applied to this replica as was applied to the original signal  $x$ .

**Improving  $\hat{T}$  and finding  $\hat{f}_0$**  As mentioned before, the coarse period estimation leads to drift in the windows. Now that a replica is built, this drift can very precisely be determined and corrected for, while at the same time finding the sample offset of the replica with the original signal  $x$ . This step marks the beginning of step 2 of the algorithm.

For each window, the replica and original are cross correlated to find the offset in samples between them. Because of the drift, this offset increases or decreases linearly for increasing window index. This is plotted with the solid line in figure 3.8. It can happen that the offset wraps around, which needs to be unwrapped. This is shown by the dashed line in figure 3.8. A linear function of the form  $\Delta n'(i) = \hat{d}i + \hat{\Delta}_n$  can be fitted against the unwrapped offset.  $\Delta n'(i)$  denotes the offset in samples as function of the window index,  $i$  denotes the window index and  $\hat{d}$  and  $\hat{\Delta}_n$  are the parameters of the function. The slope  $\hat{d}$  of the function gives the drift in the chirp period estimate and the intercept of the function  $\hat{\Delta}_n$  gives the offset in samples of the chirp in  $x$ .

The chirp period is corrected as

$$\hat{T} = \hat{T}' + \hat{d}. \quad (3.28)$$

The sample offset  $\hat{\Delta}_n$  will later be useful for finding the initial frequency  $\hat{f}_0$  as well as for aligning the windows such that no wrapping occurs within them.

**Windowing using new period** Similarly as before, the original signal  $x$  is windowed. This time however, the fine chirp period estimate is used. Also, the windows are aligned this time so that the chirps start at the start of each window. Since the estimate  $\hat{T}$  this time is not an integer, some rounding off has to be done. In order to not drift like earlier, some overlap between windows is allowed as well as samples being left out. The windows are defined as

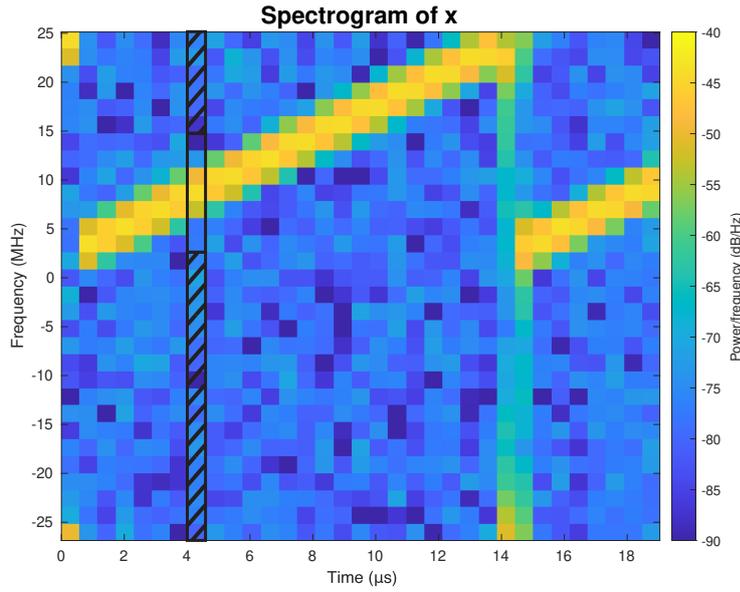


Figure 3.9: The shaded part is only noise. This area is what's left after removing the maximum of the column and 4 pixels up and down from the maximum.

$$\mathbf{x}_i = \begin{bmatrix} x \left[ \lfloor (i-1)\hat{T} \rfloor \right] \\ x \left[ \lfloor (i-1)\hat{T} \rfloor + 1 \right] \\ \vdots \\ x \left[ \lfloor (i-1)\hat{T} \rfloor + \lfloor \hat{T} \rfloor - 1 \right] \end{bmatrix}. \quad (3.29)$$

**Find fine chirp rate** The DCFT is again applied to the windows, exactly the same as before. The estimate is expected to work better this time since the windows are aligned with the chirp. This produces an estimate  $\hat{\alpha}$

**Find fine bandwidth and initial frequency** The bandwidth can be found the same way as before, but this time from the fine estimates, i.e.

$$\hat{B} = \hat{T}\hat{\alpha}. \quad (3.30)$$

The initial frequency can now also be found from  $\hat{\Delta}n$  as mentioned before:

$$\hat{f}_0 = \hat{\alpha}\hat{\Delta}n \quad (3.31)$$

**Amplitude estimation** It seems that it would be easy to find the amplitude of the chirp, since all its other parameters is known. This could be achieved by correlating  $\mathbf{x}$  with  $\hat{\mathbf{u}}$  as  $\hat{A} = \mathbf{x}^H \mathbf{u} (\mathbf{u}^H \mathbf{u})^{-1}$ . It turns out, however, that in practice this estimate produces very poor results if the estimates of the parameters are only slightly off. Therefore another method was chosen.

The amplitude of the chirp is found from its power. Since the chirp, noise and satellite signals are uncorrelated, the following holds:

$$P_x = P_s + P_u + P_w. \quad (3.32)$$

Due to the low SNR for the satellites and the way the signal looks and since only the power of the chirp needs to be known, the signal and noise power are combined as

$$P_x = P_{w'} + P_u. \quad (3.33)$$

This allows to find  $P_u$  from finding  $P_x$  and  $P_{w'}$ .  $P_x$  is readily found as the sample variance of the signal  $x$ , i.e.

$$P_x = \frac{1}{N-1} \sum_{i=1}^N |x[i] - \bar{x}|^2, \quad (3.34)$$

where  $\bar{x}$  is the sample mean.

If it is assumed that the noise is white, its power will be evenly distributed over the spectrum. Also, if it is assumed to be wide-sense stationary (WSS), then it will not change over time. In this scenario looking only at a part of the spectrum of the noise will give information on the noise everywhere else. This property can be leveraged to find the noise power from the spectrum of  $x$  by finding the parts where there is only noise.

Looking at figure 3.9, there is always a large part of the spectrogram that is not occupied by the chirp. Because the chirp parameters are known from the other estimators, its instantaneous frequency is also known at any given moment. This allows to carefully determine which *pixels* of the spectrogram belong to the noise. This is done per column. It can be seen from the figure that there is some *bleeding* around the chirp, i.e. its power is not confined to a single pixel per column. As stated earlier, only a part of the noise spectrum is needed to find the noise power. Therefore some more pixels above and below the chirp centre are regarded as part of the chirp. The spectrogram is created without overlapping windows and without zero padding in the FFTs so no extra smoothing/bleeding is introduced. Figure 3.9 shows that the shaded area is the part of that specific column that is regarded as noise. This is done for each column.

The associated power densities for the selected pixels are then averaged to get the estimated noise power density. Multiplying by the sampling frequency gives the total noise power.

Now the jammer power is simply found from (3.33). The amplitude is then found from the power as

$$\hat{A} = \sqrt{\hat{P}_u}. \quad (3.35)$$

In practice, it might be the case that the noise is not white. This will have some effect on the estimation performance. Within the band that the chirp is in, there are less pixels that contribute to the estimate. This means that this frequency band has less effect on the average than the part outside this band. To illustrate: assume the noise has more power in the same band as where the chirp is. This part has less influence on the average, so the estimated noise power would be too low. In this case the resulting estimated chirp power would be too high.

This problem could be mitigated by at least two ways. The first way would be to weigh the average in such a way that counteracts the bias that is introduced. This could be done by assigning weights to each row in the spectrogram for example. It is known how many of the pixels are removed for each row. If more pixels are removed, it should receive a higher weight so eventually all rows contribute the same amount.

Another approach would be to apply a whitening filter to the signal first. This however requires knowing the noise shape beforehand, which might not always be available.

For this thesis the noise is assumed to be approximately white, so that the effect of the estimator bias is minimal.

### 3.4.1. Measurement setup

In order to assess the accuracy of the estimator, it has been tested on simulated data using 1000 Monte Carlo simulations for different jammer-to-noise-ratios. For each simulation a chirp with random parameters is generated. These random parameters are generated with uniform probability within their allowed bounds. These bounds are shown in table 3.2. There is quite some coupling between the parameters, making the bounds not very straightforward. For example the chirp rate together with the bandwidth determine the period of the chirp. Similarly the bounds of the centre frequency depend on the bandwidth of the chirp. Also, the amplitude is not chosen randomly as it is determined by the jammer-to-noise-ratio.

Parameter	Lower bound	Upper bound
Chirp rate ( $\alpha$ )	0.54 MHz/ $\mu$ s	10 MHz/ $\mu$ s
Bandwidth ( $B$ )	10.4 MHz and resulting period $\geq 2.9 \mu$ s	31.2 MHz and resulting period $\leq 48 \mu$ s
Centre frequency ( $f_c$ )	Such that the chirp lies entirely within the receiver bandwidth	
Initial frequency ( $f_0$ )	0	$B$
Initial phase ( $\phi_0$ )	0	$2\pi$

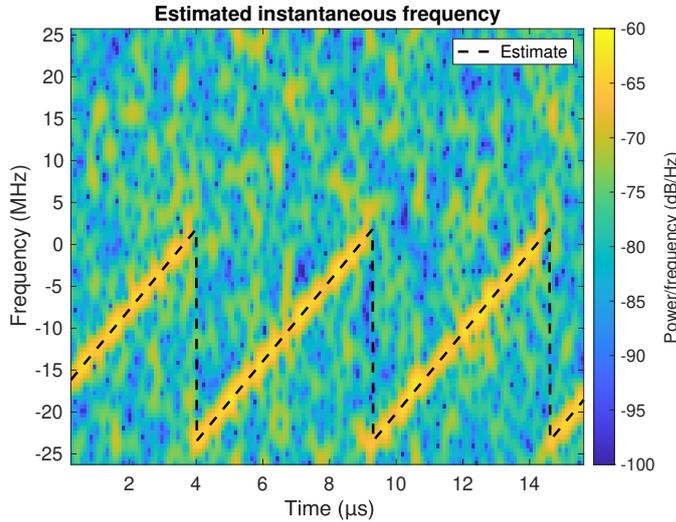
Table 3.2: The allowed bounds for the different parameters, assuming a sampling frequency of  $f_s = 52\text{MS/s}$ 

Figure 3.10: Chirp spectrogram and estimated instantaneous frequency for a chirp with an SNR of 0 dB

For the randomly chosen parameters and the set jammer-to-noise-ratio a chirp is generated as well as complex white Gaussian noise. The two are superimposed and then fed to the proposed estimator. For each Monte Carlo simulation then the difference between the estimated parameters and the actual parameters is calculated to find the RMSE of the estimator.

### 3.4.2. Results

The proposed estimator has been implemented in Matlab and is indeed able to estimate all the parameters. An example is shown in figure 3.10.

The results are shown in figures 3.12a-3.12d for the different parameters. The *random* estimator, which has been described in section 3.3.7 has also been plotted. It means that if the proposed estimator's error is above the error of the random estimator it does not give any information at all.

The performance of the chirp rate estimator has also been plotted in figure 3.4a to compare it to the chirp rate estimators from literature. It can be seen that it has roughly the same shape as that of the DCFT, which is to be expected as the proposed estimator relies on the DCFT for the chirp rate estimation. The exact parameters used for the DCFT in the proposed estimator were slightly different than those from the DCFT itself which was used in figure 3.4a. This explains why they are not exactly the same. Also, in the proposed estimator, the DCFT is applied to aligned windows instead of the whole signal at once. This might explain why it performs better for low JNRs.

Although most of the parameters are estimated better for increasing JNR, the amplitude estimation has an optimum around  $-10$  dB JNR. After this point the estimates get worse. This might be because if the amplitude gets higher it tends to pollute the noise a bit. This can be seen for extreme cases as shown in figure 3.11. In this figure the noise for both cases have unit variance, however due to the very strong chirp in the right hand plot the noise level appears much higher. This effect might cause a decrease in estimation accuracy for higher JNRs.

The bottleneck of the proposed estimator is clearly the bandwidth estimation. This needs around  $-9$  dB JNR to start functioning properly.

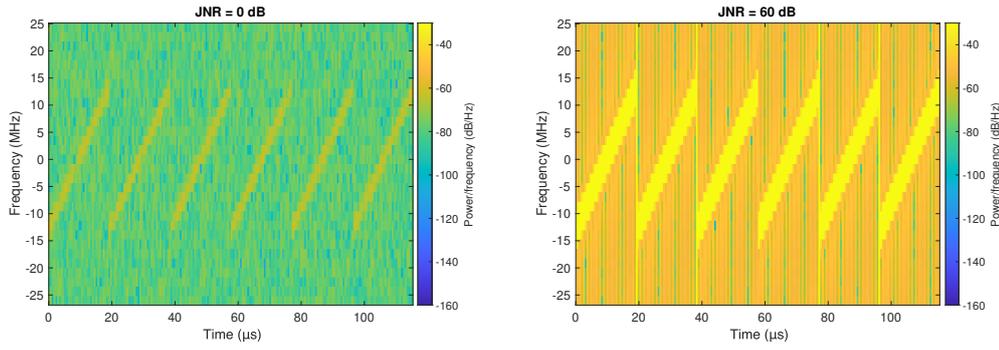


Figure 3.11: Effect of very strong chirp on surrounding noise. In both cases the noise has variance 1.

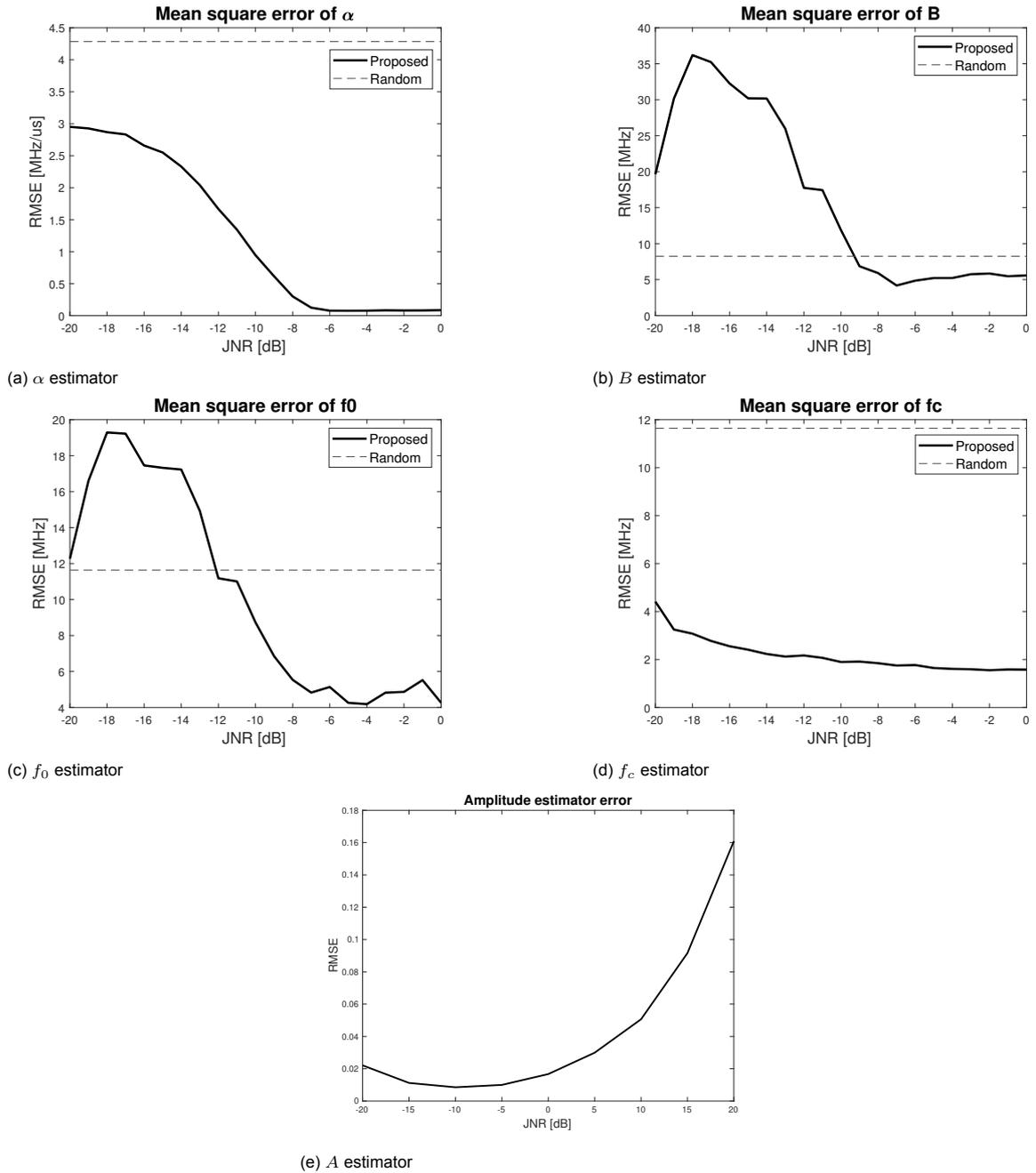


Figure 3.12: Accuracy of proposed estimator

### 3.5. Conclusion

In this chapter many different chirp estimators from literature have been investigated and compared with respect to computation time and estimation accuracy. It turned out however that no single estimator from literature estimated all required parameters. This led to the proposal of a new chirp estimation algorithm. This new algorithm incorporates the DCFT and Hankel reduction methods from literature and is able to estimate all required parameters of a chirp, namely centre frequency, chirp rate, amplitude, bandwidth and centre frequency.

It works well for jammer-to-noise-ratios as low as  $-9$  dB. Below that the bandwidth estimation fails first. Since the bandwidth is estimated from the chirp rate and chirp period combined and since the chirp rate estimation does work for lower JNRs, it seems that it is in fact the period estimation that fails.



# 4

## Filtering chirp interference

The goal of this chapter is to define a filter that can be used to remove chirp type RF interference such that the reception of GNSS signals can be improved. First three different filtering techniques from literature are reviewed. Then a new filtering technique is proposed and implemented that makes use of the earlier proposed estimator. At the end of the chapter the performance of the proposed filter is tested on simulations, these are then compared to other filtering techniques.

### 4.1. Filters from literature

A number of different filtering techniques have been proposed in literature for filtering chirp interference in GNSS applications. In this section three of those techniques are briefly introduced to give an idea of the possible directions. In this thesis a different method is investigated, which will be introduced afterwards in section 4.2.

#### 4.1.1. Filter bank pulse blanking

Rügamer [37] proposed a very simple technique to filter chirp interference. It makes use of a technique called pulse blanking and combines it with a filter bank. Pulse blanking is a time-domain filtering technique where the signal is *blanked*, i.e. made zero, when interference is detected. This can be a useful technique for mitigating against pulsed interference. During the pulses the signal is blanked and the rest of the time the signal is clean. This in itself however does not help mitigating against chirp interference, especially if the chirp bandwidth is smaller than the receiver bandwidth. In this case the interference is always present and it is clearly not useful to blank the whole signal all the time.

To overcome this problem the paper proposes to split the signal up into three bands using a filter bank. Each of these bands can be blanked when the chirp passes through that specific band. This way at most one third of the total receiver bandwidth is blanked at any given moment. This filter is able to improve the  $C/N_0$  ratio by about 10 dB for a  $J/N$  ratio of 15 dB.

#### 4.1.2. Transform domain processing

Another way of filtering chirp interference from GNSS signals is by transforming the received signal to a transform domain, such techniques are proposed in for example [38, 39]. This transform domain should be chosen in such a way that in that domain the chirp interference has a sparse representation. It is also important that the transform can be inverted so that it is possible to transform back to the time domain. An example could be the Short Time Fourier Transform domain.

In the transform domain the chirp can then be removed by getting rid of its sparse non-zero values after which the signal is transformed back to the time domain. It is important to make sure that the underlying GNSS signals do not suffer from the transformations, otherwise the filtering has no beneficial effect.

This technique works quite well according to the literature, for example [38] reports about 5 dB increase in  $C/N_0$  at a  $J/N$  of 6.5 dB.

### 4.1.3. Adaptive Notch Filter

A third way of filtering proposed in literature such as [40] uses an adaptive notch filter to filter out chirp interference. This requires continuously estimating the instantaneous frequency and updating the filter coefficients of the notch filter so that it tracks the chirp. The difficulties are especially in the frequency discontinuities, when the filter has to instantly change. The notch filter also introduces a delay in the reception, but this does not pose problems as researched in [41].

The notch filter appears to work well and can increase the correlation by up to 20 dB for a  $J/N$  of 18.2 dB according to [40].

## 4.2. Proposed filter

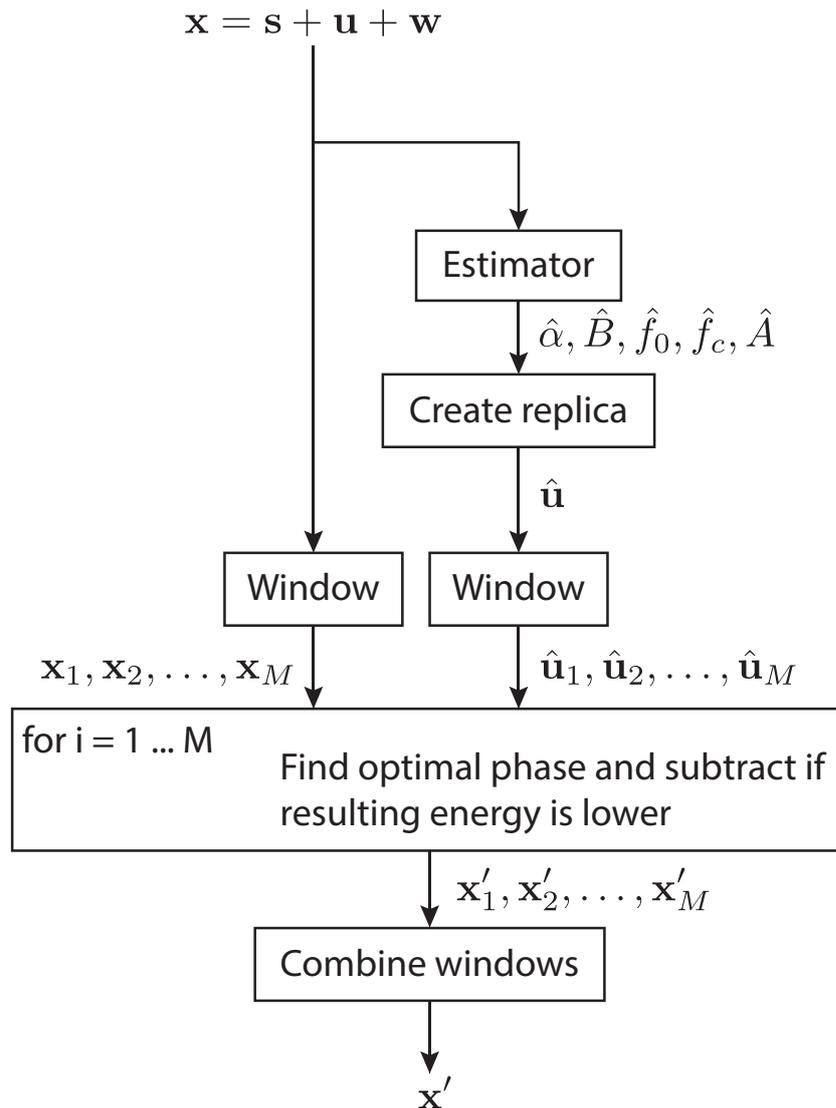


Figure 4.1: Overview of the filtering algorithm

The previous section showed different methods that can be used to filter chirp interference. However, none of the mentioned estimators really use the fundamental mathematical structure of a chirp. Because an estimator has been proposed in this thesis that is capable of finding very accurately the parameters that constitute the chirp, this opens up the possibility to leverage that information. Theoretically if the interfering signal was known exactly and the interfering signal is strictly additive (i.e. a superposition) then it would be possible to remove it without changing the rest of the signal at all. In this thesis, this idea is pursued by making use of the proposed estimator.

A method is proposed that filters in the time-domain, taking full advantage of the chirp estimator. The filter is composed of a few different blocks. At the core of the algorithm is a subtraction of a replica in the time domain. The filter is schematically shown in figure 4.1.

The idea of the filter is to perform a subtraction in the time domain. The received signal is a superposition of the satellite signals, noise and the unwanted chirp, i.e.  $\mathbf{x} = \mathbf{s} + \mathbf{u} + \mathbf{w}$  (see section 3.1). This means that if  $\mathbf{u}$  was known exactly, it could be subtracted from  $\mathbf{x}$  to received at  $\mathbf{x}' = \mathbf{x} + \mathbf{u} + \mathbf{w} - \mathbf{u} = \mathbf{x} + \mathbf{w}$ . However,  $\mathbf{u}$  is not known exactly and needs to be estimated. The better the estimate, the better the chirp can be removed from  $\mathbf{x}$ .

For this reason, the signal  $\mathbf{x}$  is first fed to the chirp parameter estimator as proposed in chapter 3. This will give estimates for the chirp rate, bandwidth, initial frequency, centre frequency and amplitude or  $\hat{\alpha}$ ,  $\hat{B}$ ,  $\hat{f}_0$ ,  $\hat{f}_c$  and  $\hat{A}$  respectively.

Using these parameters, a replica of the chirp is constructed using the chirp definition from section 3.1.2. This replica will be called  $\hat{\mathbf{u}}$ .

If the replica was accurate enough, it would be possible to simply subtract it at this point. However, the phase of  $\hat{\mathbf{u}}$  needs to be correct at all times. Subtraction is essentially adding with a phase shift of  $180^\circ$ . It is easy to see that if  $\hat{\mathbf{u}}$  is  $180^\circ$  out of phase, its amplitude is effectively doubled instead of removed.

Having a perfect estimate with perfect phase is however infeasible in practice. Any small errors in the estimation of the instantaneous frequency are accumulated to large errors in the phase. This accumulation stems from the fact that the phase is the integral of the instantaneous frequency, therefore the errors continue to add up. To illustrate this, imagine that the frequency estimate is off by 10 kHz. In this case every  $50 \mu\text{s}$  the real chirp and its replica are  $180^\circ$  out of phase. Since one period of the L1 C/A signal from GPS is 1 ms, this happens 20 times per period. This example shows how quickly the mismatch occurs, confirming that the problem must be addressed. Figure 4.2 shows the effect for two sinusoids that are 10 kHz apart. This effect is also known as beat.

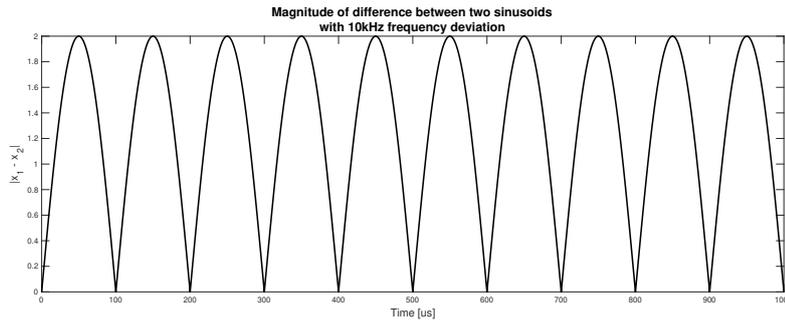


Figure 4.2: Illustration of how subtracting one unit amplitude sinusoid from another with frequencies 10 kHz apart leads to a beating pattern.

Although the illustration is with sinusoids, the same holds for chirps as they are essentially sinusoids with linearly changing frequency. The effect can however be even worse, since frequency errors can stem from errors in the chirp rate estimate. In the same way as the phase, the frequency is in turn the integral of the chirp rate, so errors here also add up. Fortunately, due to the construction of the proposed estimator, this problem is somewhat limited. Because the chirp period is estimated quite accurately, the chirp will start from the bottom again each period. Because of this the chirp rate error can only induce cumulative frequency errors within one chirp period.

Because the phase drifts so drastically, it is not possible to subtract the replica of the chirp from the received signal in one operation. Instead, both the original and the replica are sliced up into short windows (e.g. 20 samples per window) and operations are performed on those windows. For each window it is possible to vary the phase of the replica. For each phase shift the energy of the signal after subtraction is calculated. Then the phase shift for which the energy is minimal is chosen. Mathematically this is represented as

$$\phi_i^* = \arg \min_{\phi_i} E[\mathbf{x}_i - \hat{\mathbf{u}}_i e^{j\phi_i}], \quad (4.1)$$

where  $E[\cdot]$  denotes the energy of the signal.

Because it can happen that the resulting energy for each phase shift is actually higher than the energy of  $\mathbf{x}_i$  itself, this is first verified before the subtraction is done. Only when it is lower the subtraction is done:

$$\mathbf{x}'_i = \begin{cases} \mathbf{x}_i - \hat{\mathbf{u}}_i e^{j\phi_i^*} & E[\mathbf{x}_i - \hat{\mathbf{u}}_i e^{j\phi_i^*}] < E[\mathbf{x}_i] \\ \mathbf{x}_i & E[\mathbf{x}_i - \hat{\mathbf{u}}_i e^{j\phi_i^*}] \geq E[\mathbf{x}_i] \end{cases}. \quad (4.2)$$

After these operations have been performed on all windows, the windows are recombined. No overlap is used in the windows, so the recombining is as simple as stacking them together.

### 4.3. Simulations

The described filter has been implemented in Matlab. In order to quantify its performance a number of simulations have been run using the filter to see how it performs in various situations and with various parameters. The following section describes how the performance is measured and after that the results are presented.

#### 4.3.1. Measuring performance

Measuring the performance of the filter is not as straightforward as measuring how much of the chirp is removed. The ultimate goal is not to remove the chirp, but to arrive at a better GNSS receiving performance. This is assumed to be improved if the chirp is removed, but it is important that the underlying satellite signals are not distorted. If they would be distorted by the filtering operation, this might actually reduce the performance. It is therefore important to measure the performance in terms of GNSS reception.

To this end the simulations are performed on signals that contain GPS L1 satellite signals as well as white Gaussian noise and a chirp. L1 signals use 1023 bit gold codes to spread the signal over the spectrum. This happens at a bit rate of 1.023 Mbit/s, so one code (or one period) takes exactly 1 ms. These gold codes are chosen in such a way that they appear random and cross correlation between codes of different satellites is minimal, i.e. they are approximately orthogonal. Each satellite is assigned such a *Pseudorandom Noise* (PRN) code and they are numbered 1 to 32. The satellites themselves are often referred to as SV xx, where xx is the PRN number. During the acquisition stage, GPS receivers lock on to the signals of the various satellites by correlating with those PRN codes. If the SNR is sufficient the receiver is able to do so for a number of satellites and can start calculating its position.

The correlation needs to be performed for a range of frequencies due to Doppler shift of the signal. Since the satellites travel at great velocity, with in most cases at least some component away from or towards the receiver, a Doppler shift is introduced. Furthermore the receiver itself may be attached to a moving object, also introducing a Doppler shift. The correlation itself is a function of time. These two variables, namely time and frequency, form a plane in which a correlation peak is found if the satellite is within range. This correlation in the time-delay/doppler-shift plane is referred to as the *Cross Ambiguity Function* (CAF).

Due to the low SNR that is typical for the satellite signals, multiple periods of this 1023 bit code might be needed to get a clear correlation peak. These correlations then coherently sum up so that as more periods are added, the peak becomes more prominent. This is illustrated in figure 4.3. In this figure the effect of using multiple periods in the correlation is shown. The figure also shows the delay-Doppler plane.

For each frequency within the range of Doppler frequencies, the correlation is calculated. This gives a matrix of correlation values for the different frequencies and delays. This matrix will be called  $\mathbf{R}_i$  for the  $i$ 'th period with a total number of periods  $M$ .  $\mathbf{R}_i$  has dimensions  $L \times N_d$ , where  $L$  is the number of samples in one period, which is  $f_s/1000$  and  $N_d$  is the number of Doppler shifts that are evaluated. The element  $[\mathbf{R}_i]_{jk}$  then gives the the correlation for a delay of  $j$  samples and  $k$ 'th index of the Doppler shifts.

The values of the matrices for the different periods are squared and multiplied, so the final autocorrelation matrix has values  $[\mathbf{R}]_{jk} = \sum_i [\mathbf{R}_i]_{jk}^2$ .

Let  $j^*$  and  $k^*$  denote the indices for which the autocorrelation is maximum, i.e. the peak. Then the correlation strength of the peak is defined as the ratio between the peak itself and the average of the non-peak values:

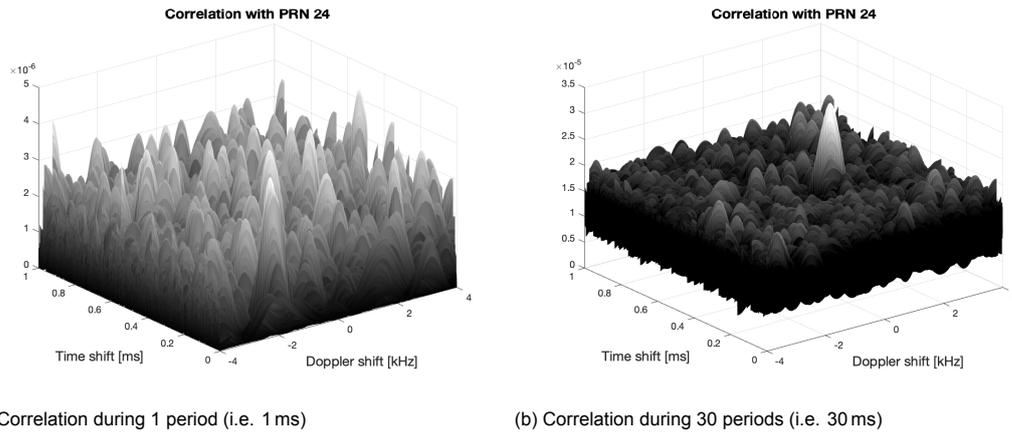


Figure 4.3: Correlation peak for a specific satellite using one PRN period and 30 periods. In this simulation  $C/N_0 = 40 \text{ dB} - \text{Hz}$

$$\text{corrstr} = \frac{[\mathbf{R}]_{j^*k^*}}{\frac{1}{LN_d - 1} \sum_{j \neq j^*, k \neq k^*} [\mathbf{R}]_{jk}}. \quad (4.3)$$

This metric should be used with care. It is inherently biased since it will always be greater than one. Even if cross ambiguity function is only noise, there will always be one value that is highest. So even in the noise only case, the correlation strength will be greater than one. It is therefore best to compare the values for different satellites to see which ones stand out.

This correlation strength can then be computed for a number of satellites that are known to be within range. The correlation strength is lower if a chirp is applied to the received signal, this corresponds with a GPS receiver having difficulty locking to the satellites if chirp interference is present. The goal is then that after applying the proposed filter to the received signal with chirp, the correlation strength goes up. The results will be shown next.

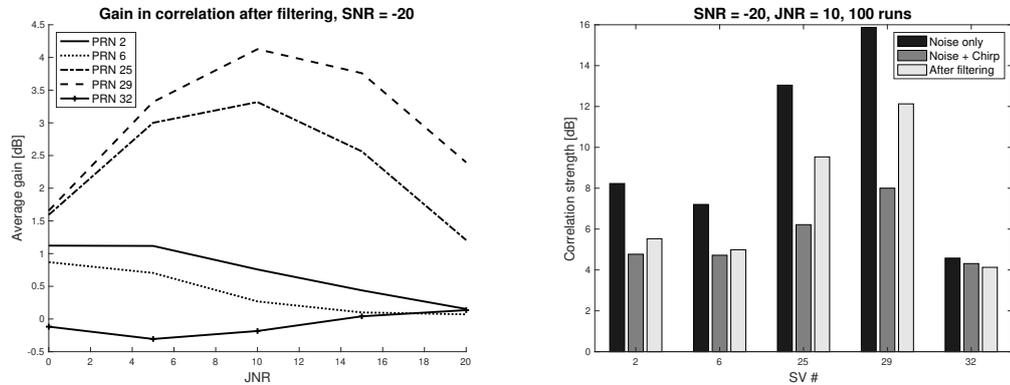
### 4.3.2. Results

The proposed filter has been simulated for different values for the SNR and JNR as well as for different window lengths. The simulations use a predefined signal that contains the contributions of the simulated satellites and adds white noise and a chirp with randomly chosen parameters. The simulations are performed similar to the ones as described in section 3.4.1. For each setting multiple Monte Carlo simulations are performed. Each Monte Carlo simulation has a random chirp and newly generated white noise. This means that the performance seen in this section is for a wide variety of possible chirps.

Figure 4.4 shows the correlation performance for 5 selected satellites. The SNR in this case was  $-20 \text{ dB}$ . Figure 4.4a shows the increase in correlation performance after filtering for different values of the JNR. It can be seen that especially for certain satellites the gain can be as much as 3 to 4 dB. Another thing to be seen is that there appears to be an optimum for which JNR the gain is best. In this scenario this is around 10 dB of JNR. The rising part from 0 dB to 10 dB might be because for lower JNRs the estimation of the chirp is more erroneous, leading to poorer replicas and thus poorer filtering. As the JNR goes up the estimates improve and hence the filtering works better. Then beyond 10 dB the amplitude estimation starts to become worse, which was seen in figure 3.12e. This effect might take over here reducing the effectiveness of the filter. It could also be due to other effects. Remember from (4.2) that a subtraction is only performed if the energy is reduced. Because the estimate can be somewhat off in some cases, every now and then a window remains unfiltered. If the JNR is much higher, the effect of the little bits of chirp left in those windows might contribute more, worsening the performance of the filter.

The effect of the filter is further illustrated in figure 4.4b. This figure shows the correlation strength for the same SNR and for the optimal JNR, which is 10 dB. In this plot the effect of the different stages can be seen. Note that with noise only, the correlation levels differ. This is because the satellite simulations take into account an assumed antenna radiation pattern, which attenuates satellites depending on their position in the sky. The first thing to observe in the plot is that adding a chirp to the signal severely

degrades the correlation performance. Then the second thing to observe is that in some cases, the filtering brings back a lot of the performance. However, the effect is not the same for all satellites. Some are barely recovered, while others are recovered greatly. Another thing to notice is that the filtering brings the performance back to some extent, but not all the way back up to the level it was before the chirp was added.

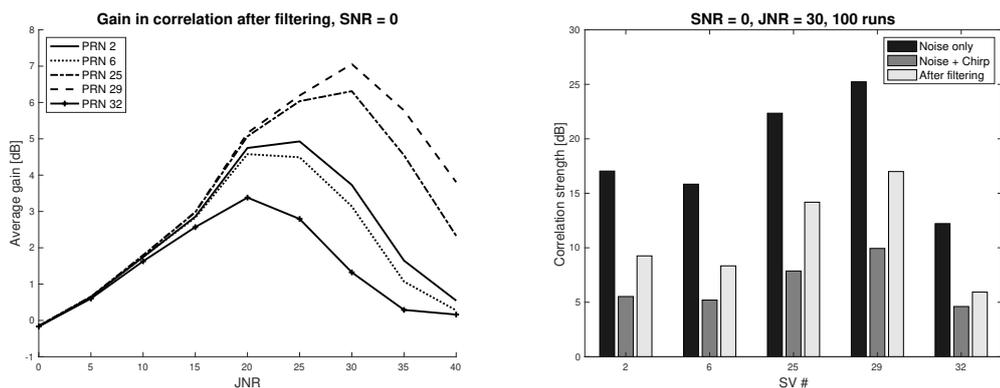


(a) The gain for 5 different satellites in range as a function of the jammer-to-noise-ratio. (b) The correlation peak strength for the case with only noise, noise and chirp and noise and chirp after filtering. JNR = 30 dB

Figure 4.4: Gain in correlation peak for SNR = -20 dB

Figure 4.5 shows similar plots, but for an SNR of 0 dB. In this scenario the performance of the filter appears to be better. First of all, all satellites benefit from the filter as opposed to the previous scenario where the gain was only for a few satellites. Additionally, the gains can be much higher, up to 7 dB. Another observation is that the JNR values for which the filter works best have shifted up, to somewhere between 20 and 35 dB depending on the satellite. This behaviour is somewhat unexpected. The JNR does not depend on the SNR, so the amount of chirp filtered for higher SNR should be the same. Nonetheless, the increase in performance here is higher. It could be that the relation between the correlation strength and the SNR is nonlinear in such a way that the correlator suffers more if the SNR is high. This does correspond with the effect seen in figure 4.5. The satellites for which the initial correlation strength was higher (see figure 4.5b) profit more from the filtering. It is also seen that as the initial correlation strength was higher, the optimum for the JNR is also higher. This also corresponds with the fact that the optimal JNRs here are higher than in figure 4.4 for an SNR of -20 dB.

Figure 4.5b was added here again to show the gain for a particular JNR value.



(a) The gain for 5 different satellites in range as a function of the jammer-to-noise-ratio. (b) The correlation peak strength for the case with only noise, noise and chirp and noise and chirp after filtering. JNR = 30 dB

Figure 4.5: Gain in correlation peak for SNR = 0 dB

The previous results were obtained using windows of 20 samples. It was stated in section 4.2 that the subtraction needs to be done in windows. Each of these windows uses a phase shift that works

best for that window. Figure plot 4.6 shows the phase shift applied to each window in a simulation, its resolution is dictated by the window size. It can be seen that over the course of 200  $\mu\text{s}$  the phase shifts vary greatly. There is also a clear periodicity in the shifts. This corresponds with the smaller period of roughly 4  $\mu\text{s}$  that can be observed. This periodicity can be attributed to the period of the chirp. The large periodicity of around 25  $\mu\text{s}$  is only due to the phase wrapping around  $360^\circ$  to get a more practical graph, it carries no physical meaning. This result should justify the windowing to some extent.

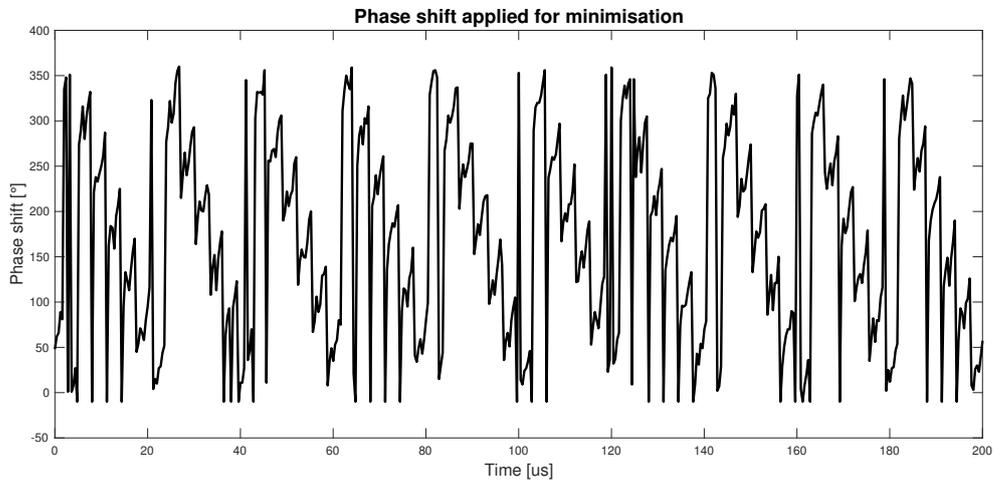
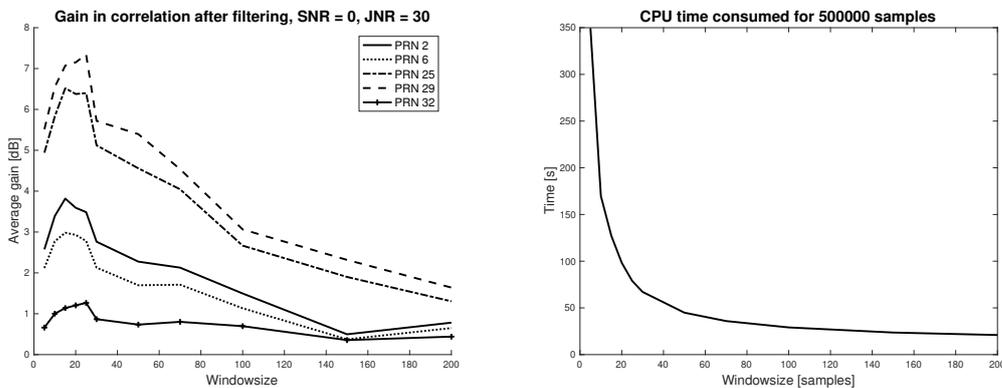


Figure 4.6: Phase shift required for proper filtering, using 20 sample windows

In order to justify the window size of 20 samples, it is necessary to examine the effect of the window size on the filter performance. Figure 4.7a shows the effect of the window on the filter performance. It can be seen that around 20 samples gives the best performance of filtering. The small window size comes at a cost. This is because for each window the optimal phase shift needs to be found by trying all possible phase shifts. The CPU time required to do the filtering in Matlab is plotted in figure 4.7b<sup>1</sup>. Depending on the implementation that is desired, a trade off between correlation gain and CPU time has to be made.



(a) The gain for 5 different satellites in range as a function of the window size used for filtering (b) The CPU times needed to filter 500 000 samples using different window sizes.

Figure 4.7: The effect of the window size on the filter

### 4.4. Conclusion

In this section chirp interference filters from literature have been reviewed. Then another type of filter has been proposed by leveraging the mathematical model of a chirp combined with the estimator that

<sup>1</sup>Simulations were run on an Intel Core i7 920 (from 2008) using only a single core.

was proposed earlier in this thesis. This is in contrast to the filters from literature that do not take this model into account.

It has been shown that the proposed filter works on simulations. It is able to improve the GNSS correlation by removing the chirp from a signal. The amount by which the filter improves the correlation strength depends on the SNR and the JNR and was at best around 7 dB.

If this value is compared to the values from literature as reported in section 4.1 it is not the highest, some report improvements up to 20 dB! It should be concluded that the proposed filtering method performs worse than those proposed in literature. Nonetheless an attempt will be made in the next chapter to apply the filter to real signals to see how it performs in application.

# 5

## Application

In this chapter, the proposed estimator and filter are applied to actual recorded signals. These are signals that have been captured at the NLR monitoring station (sec. 2.3.4). First an overview is presented of the different types of chirp signals that have been observed. Then the proposed estimation algorithm as well as the proposed filter are applied to actual recorded chirps, to see how well they work and how that compares to the simulations.

### 5.1. Types of chirps

As already mentioned in section 2.1.4 chirps come in many different shapes. They are all in some way frequency modulated signals, i.e. they are sinusoids with frequency that changes over time. In this section an overview is given of actual chirp signals that can be found and can be attributed to jammers. An overview of received signals and their relative occurrence at the monitoring station at NLR Amsterdam is presented.

#### 5.1.1. From the NLR monitoring

The monitoring station at NLR Amsterdam (see section 2.3.4) does some automatic classification. The dataset used for the following overview consists of all the interferers measured by the system and classified as chirp. This totals to 349 signals. They have then been further inspected manually to classify them further into eight different classes.

**High chirp rate, wide-band - fig. 5.1a** This class is distinguished by the very high chirp rate, on the order of 3us. The signal sweeps most of the L1/E1 band.

**High chirp rate, side-band - fig. 5.1b** This class also has a high chirp rate, but the chirp is confined only to a small side band. This band is often around 15 MHz offset from the L1 centre frequency.

**Medium chirp rate - fig. 5.1c** These signals usually span a good portion of the L1 band. In some cases they exceed the sampling band of 52MHz on one side or both sides. The chirp period is typically on the order of 10us.

**Slow chirp rate - fig. 5.1d** Like the medium chirp ones, these also mostly span a good portion of the bandwidth. The period is on the order of 40us.

**Triangular - fig. 5.1e** These signals have a triangular shape in the time-frequency domain instead of a sawtooth. They mostly span the lower half of the sampling band and fall just outside of it. Their period is usually on the order of 20us, making their slope comparable to that of the slow chirps.

**Pulsed - fig. 5.1f** This type of signal is quite rare. It is a pulsed chirp with relatively low duty cycle and a narrow band around the L1 centre frequency.

**Strange pulses - fig. 5.1d** These signals are quite strange. They exhibit some pulses at very long intervals. Sometimes they appear in groups, spaced around 5ms apart and then it is silent for 25ms. At this point the source of these signals is unknown, but they have been reported very frequently.

**Miscellaneous** Apart from the signals that are easily categorised, there are also a number of signals that do not represent anything recognisable. It is unknown what the sources of such signals are.

### Relative occurrence

The relative occurrence of the aforementioned signals is plotted in figure 5.2. Note that the high chirp rate, both wide-band and side-band, medium chirp rate and low chirp rate are all essentially the same signal type with different parameters and they account for 64% of the chirps. This makes these type of signals a good place to start working on estimation and filtering techniques.

## 5.2. Parameter estimation on real data

Validating the accuracy of the estimator on real signals is a difficult task. Two options are possible, both with their limitations.

- The first option is to use the signals recorded by the detector at the NLR. This gives real-world examples of noisy measurements of chirps. The problem with using these signals is that there is no knowledge of the ground truth. In other words, the *real* parameters of the chirp are unknown. This makes it difficult to quantify the accuracy of the estimator. An attempt will be made in the next paragraph however.
- The second option would be to acquire a number of jammers and make measurements with those jammers. In order to get meaningful results however, it is necessary to have a wide variety of jammers. If not, the test only focuses on a few types and it is easy to overfit based on those jammers and be incorrectly tricked into thinking the accuracy is very high. It is also more difficult to get real-world noise scenarios.

In order to get the most realistic signals it was decided to select the first option.

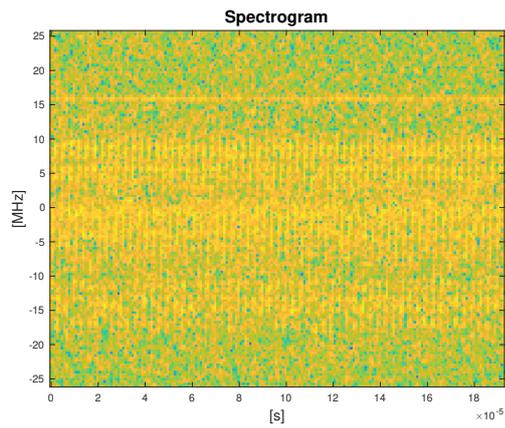
From the recorded chirps at the NLR 57 were found to meet the requirement for the estimator. This requirement is that the chirp lies entirely within the receiver bandwidth. For all of these signals, the parameters have been estimated by visual inspection. Then for the same signals the estimator has attempted to estimate the parameters and the parameters were compared afterwards. Please note that the human error in visually finding the parameters must be taken into account.

Figure 5.3 shows the errors that are made by the estimator for four of the parameters. The amplitude has not been tested because visually determining the amplitude is infeasible. The errors are taken with respect to the values from visual inspection. The plots in figure 5.3 show that at times there is considerable error. The root mean square error has been plotted as well. It was mentioned in the previous paragraph that there is human error in the visual inspection. In order to quantify this error, the parameters of one signal have been estimated by hand for 20 different segments. The standard deviation of the results are shown with the dotted line. The signal used here was a reasonably easy signal to estimate by hand, some other signals were more difficult to estimate.

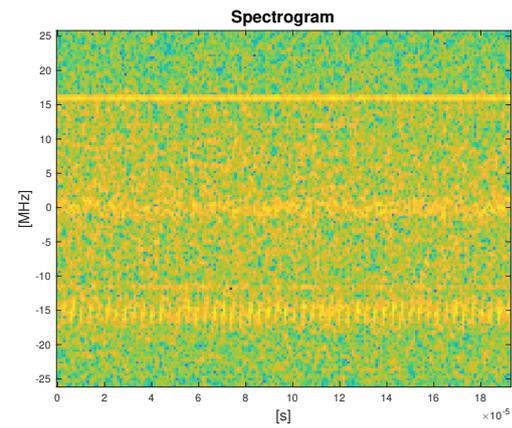
The plots from figure 5.3 do not tell the whole story however. After visually comparing the estimates by hand to the estimator estimates, by looking at both overlaid on the spectrogram, 38 out of 57 times (66.7%) the proposed estimator did as well or better than by hand. Six examples are shown in figure 5.4. One particularly interesting example is the one from figure 5.4d. In this example the bandwidth and centre frequency estimates are off. However, the chirp rate is still very well estimated and the chirp is also properly aligned. With the goal in mind of filtering the chirp through subtraction, this will work properly in the part where the estimate overlaps with the actual chirp. Due to the way the filter is constructed, i.e. it only performs a subtraction if the energy is reduced by it, the part where the estimate extends beyond the actual chirp should not lead to significant problems.

## 5.3. Filtering on real data

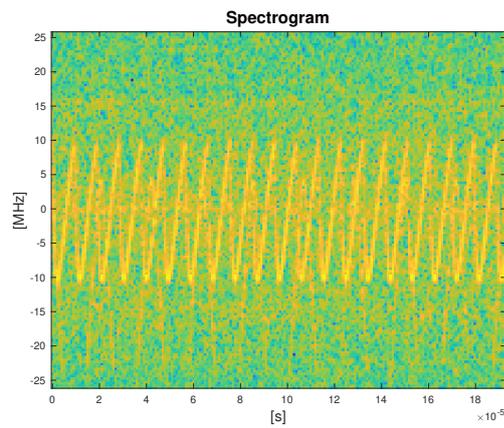
Using recordings of real interference events, it is possible to try out the proposed filter and examine the results. Because there is no knowledge of the *ground truth* in the recorded data, a quantitative



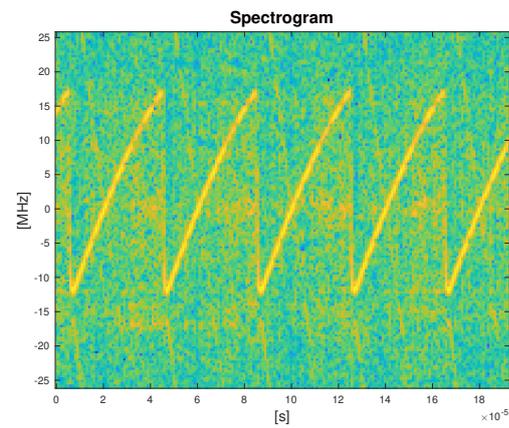
(a) High chirp rate, wide-band



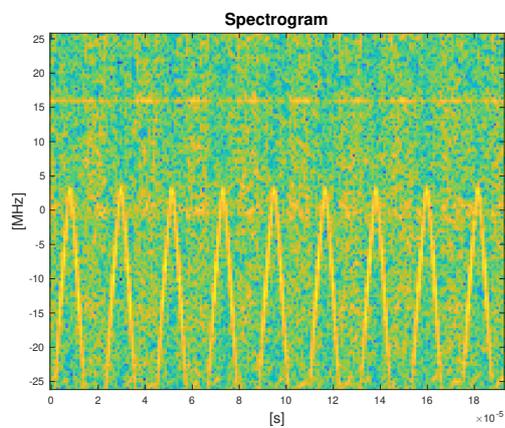
(b) High chirp rate, side-band



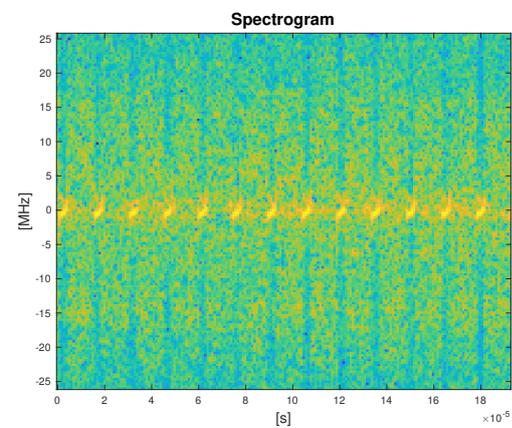
(c) Medium chirp rate



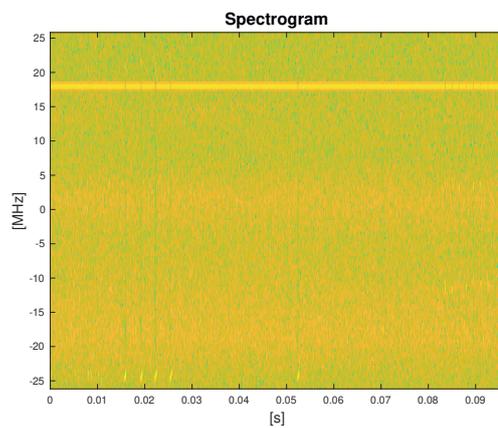
(d) Slow chirp rate



(e) Triangular signal



(f) Pulsed signal



(g) Strange pulse behaviour. Note that the time axis of this plot is scaled significantly

Figure 5.1: Received chirp-like interference signals.

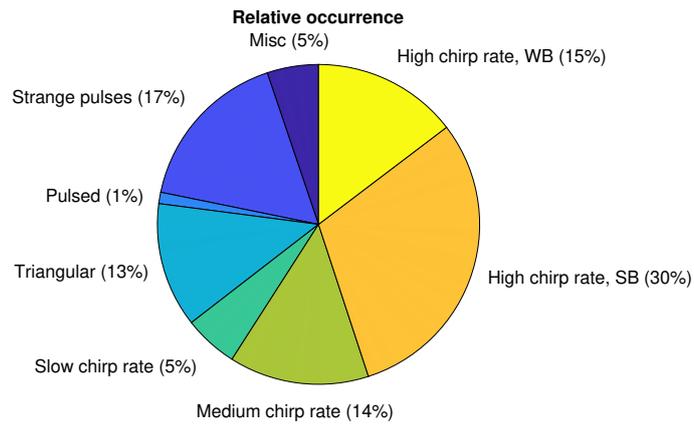
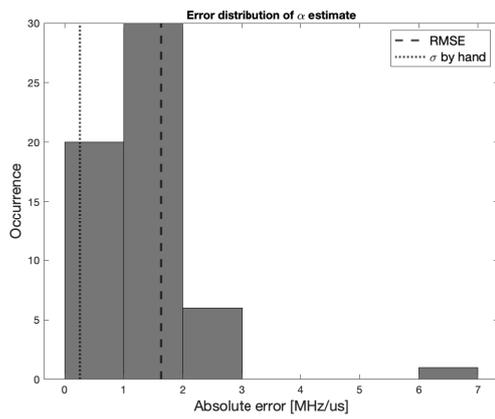
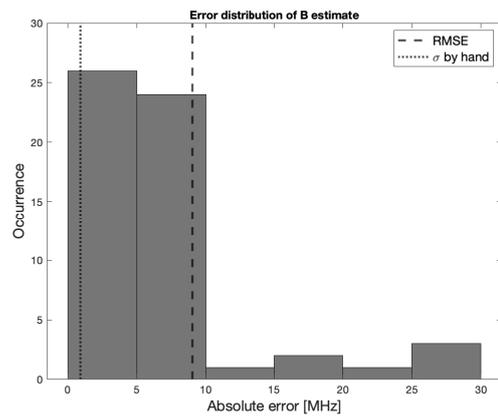


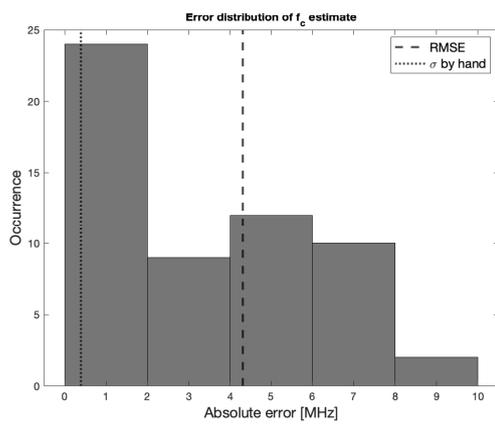
Figure 5.2: Relative occurrence of the different chirp types.



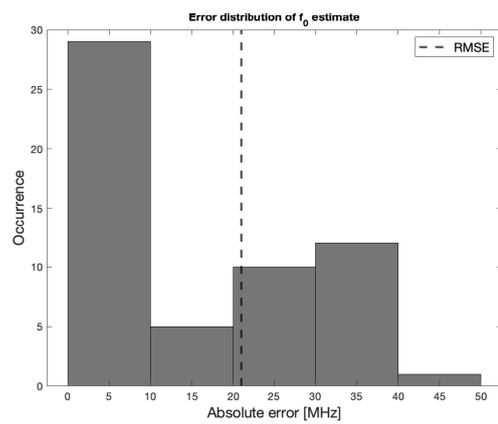
(a) The distribution of the estimation errors of the chirp rate.



(b) The distribution of the estimation errors of the bandwidth.

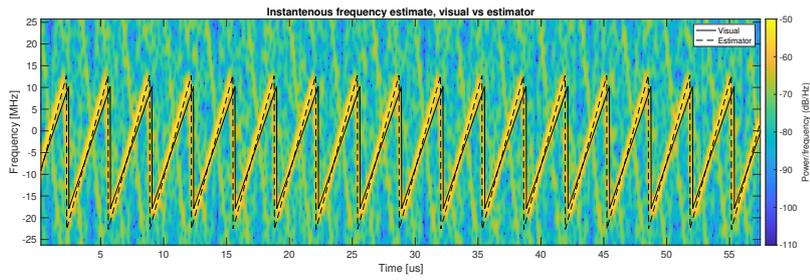


(c) The distribution of the estimation errors of the centre frequency.

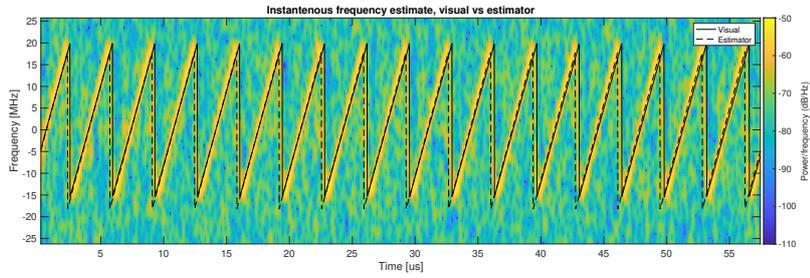


(d) The distribution of the estimation errors of the initial frequency.

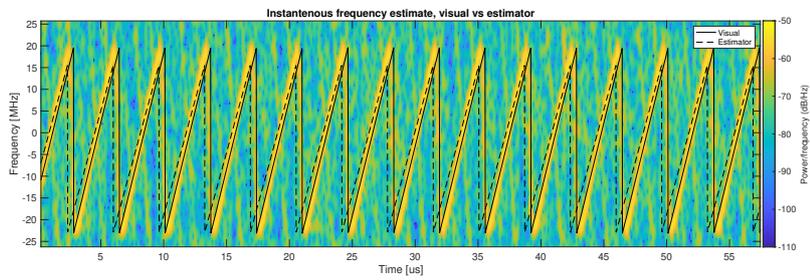
Figure 5.3: Estimator results on real data. The  $\sigma$  by hand values indicate the standard deviation resulting from estimating parameters by hand.



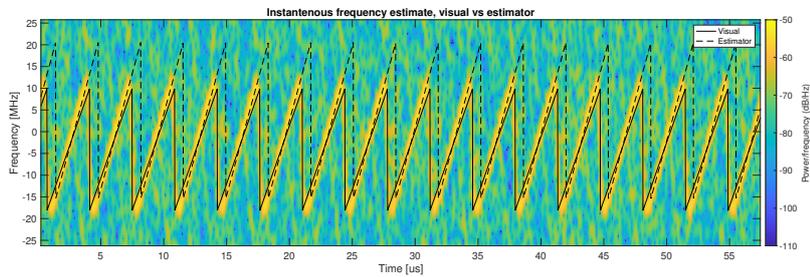
(a) For this signal, the estimator did a better job of finding the chirp than the visual estimate did.



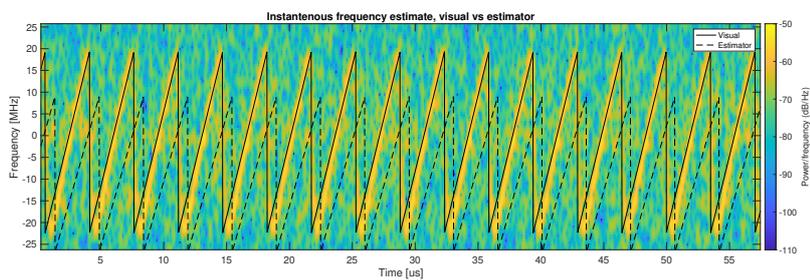
(b) For this signal, it is a close call which of the two is the better estimate.



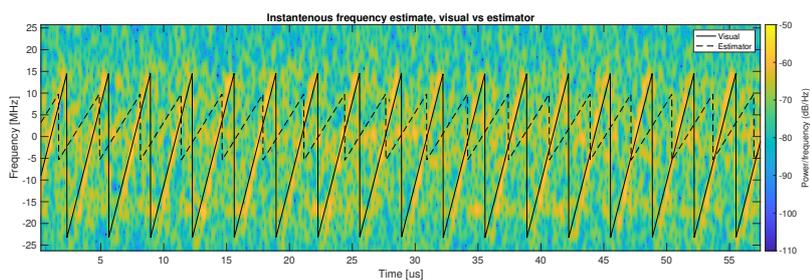
(c) For this signal, the estimator just did not do a proper job in finding the estimate. It is not extremely far off, but definitely incorrect.



(d) For this signal, although the bandwidth and centre frequency estimates are off, the chirp is tracked very well for the most part.

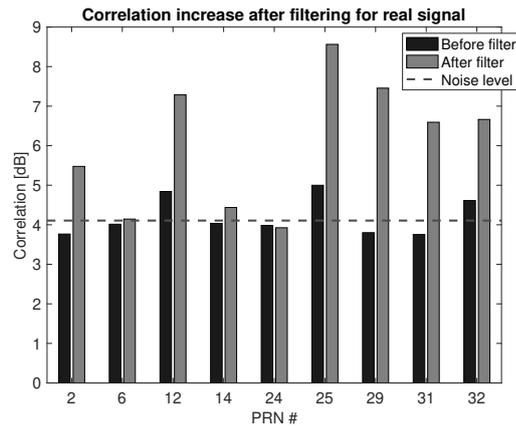


(e) For this signal, the estimator completely failed to find the initial frequency, resulting in a time shift.

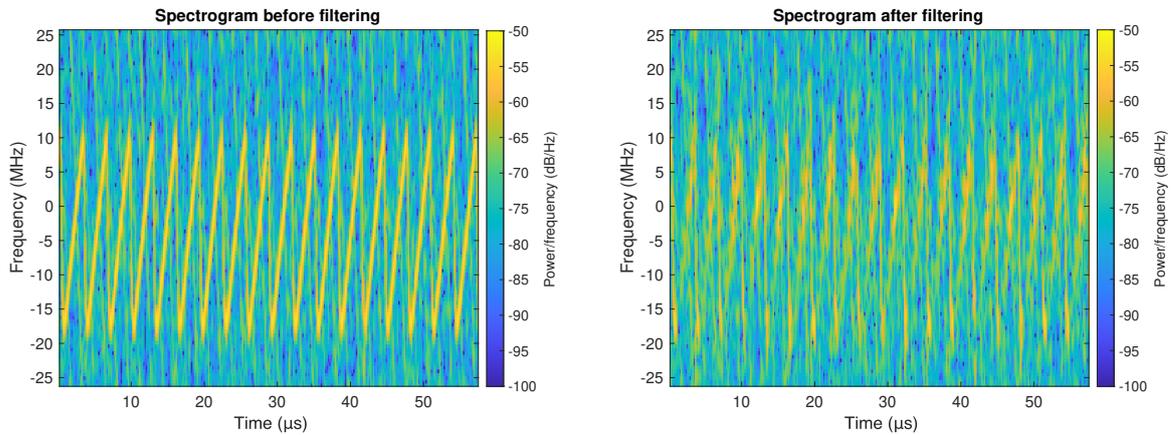


(f) For this signal, the estimator failed entirely to find the chirp. This is a complete miss.

Figure 5.4: This figure shows some examples of the errors that the estimator might make and how that compares to a visual estimate.



(a) Correlation performance before and after filtering



(b) Spectrogram before filtering

(c) Spectrogram after filtering

Figure 5.5: Filtering results for fast chirp

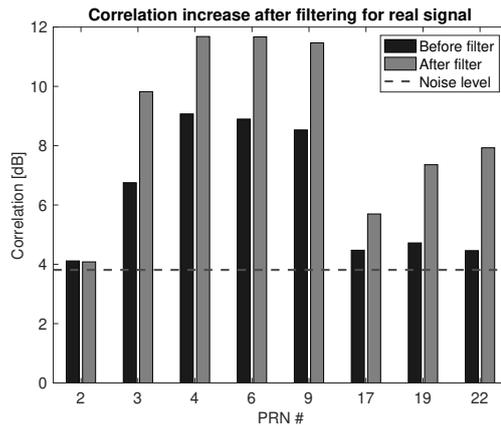
evaluation is somewhat difficult. Nonetheless an attempt is made, but first a qualitative examination is presented in the following subsection.

### 5.3.1. Qualitative results

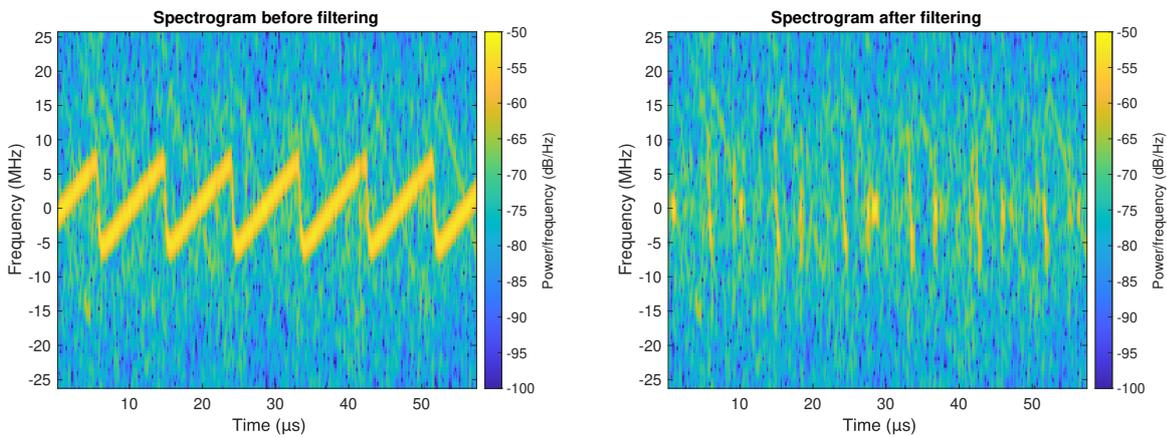
In this section the filter is applied to a few selected recordings from the detector at the NLR. These show what the filter does and how the GNSS reception is improved. It does not however quantify how well the increase is overall. First three examples are shown where the filter works well. Then some examples are shown where the filter fails. The goal is to get an understanding of what determines when the filter does or does not work properly.

The first example is a chirp from the *fast* category. It has a chirp period of  $3.2 \mu\text{s}$ , a bandwidth of 34 MHz centered around  $-5.6 \text{ MHz}$ . The spectrogram before filtering is shown in figure 5.5b. Using 20 periods of the L1 signal, there is hardly any correlation with any satellite as can be seen by the dark bars in figure 5.5a. Only the satellites that were in view at that moment are shown. This chirp is neat in the sense that it appears to be linear and its power is distributed more or less evenly across its spectrum. The spectrogram of this signal after filtering is shown in figure 5.5c. It can be seen that there is still some of the chirp left, primarily at the points where it wraps around. In the linear part of the chirp, there is barely any chirp energy left however. Returning now to the correlation performance in figure 5.5a, it can be seen from the light bars that 6 satellites have risen from the noise of which 5 with great performance, about 2 dB to 3 dB.

The second chirp is one with medium speed. It has a period of  $9.2 \mu\text{s}$  and a bandwidth of 14 MHz centered around 0.0 MHz. Compared to the previous example, it is much slower and also has a much narrower band. Like the previous one, this chirp is very linear and has an even power distribution. It is plotted in figure 5.6b. The filtered result is plotted in figure 5.6c. It can be seen that the chirp is almost



(a) Correlation performance before and after filtering



(b) Spectrogram before filtering

(c) Spectrogram after filtering

Figure 5.6: Filtering results for medium chirp

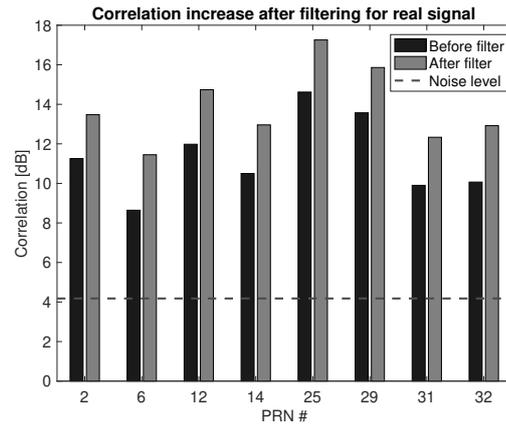
completely removed. There is only a few artifacts left that stem from the chirp. This performance is again reflected in the correlation performances before and after filtering as is plotted in figure 5.6a. For 6 of the satellites there is an increase in correlation of about 2.5 dB.

The third chirp is of the *slow* category. It has a period of 37  $\mu$ s and a bandwidth of 36 MHz centered around 5.0 MHz and shown in figure 5.7b. It can be seen that the chirp is not exactly linear, but has a slight curve. This makes it more difficult for the filter to remove the chirp completely. It can be seen in figure 5.7c that indeed a part of the chirp is missed in the removal. Fortunately, the majority of the chirp was removed. This leads to some increase in the correlation performance, but the effect is not as big as in the previous examples.

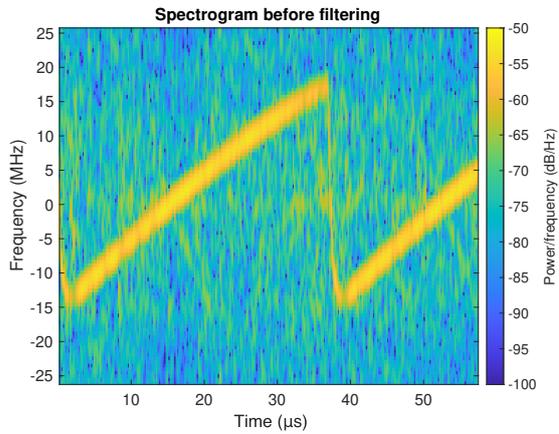
In some cases, the filter fails to properly remove the chirp. This is most likely due to the chirp varying too much with time, not being linear enough or because the energy is not spread evenly over the spectrum. The following example shows that in some cases filtering does not increase performance, but in fact decreases performance.

A slow chirp is plotted in figure 5.8b. The time axis in this plot spans a longer period than in the previous examples. A few things must be noticed from the chirp. First of all the chirp is again not very linear, but slightly curved. Additionally, there is either a reflection or there was another chirp present during recording with a negative chirp rate. This chirp is clearly far from a clean chirp. Consequently, the filter achieves a poorer result as can be seen from figure 5.8c. In the low frequency region of the chirp, it is hardly removed. This leaves quite a bit of the chirp energy in the signal. Looking now at figure 5.8a it can be seen that the correlation performance is in fact lower after filtering.

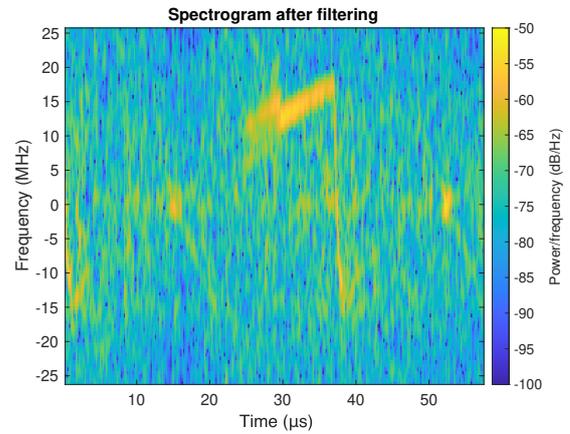
It must be stressed that any chirp that is highly nonlinear, pulsed or not contained within the receiver bandwidth will not work. Chirps with such properties are not shown in this section. This and other shortcomings of the filter will be discussed later in section 6.2.1.



(a) Correlation performance before and after filtering

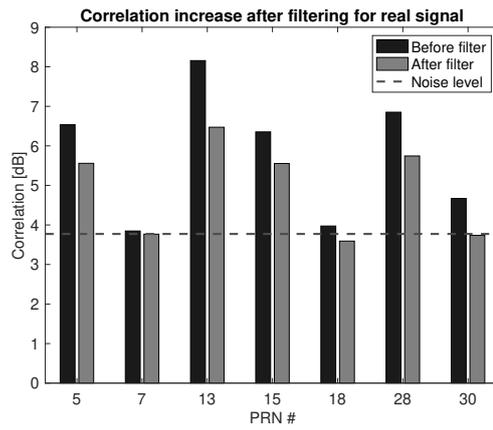


(b) Spectrogram before filtering

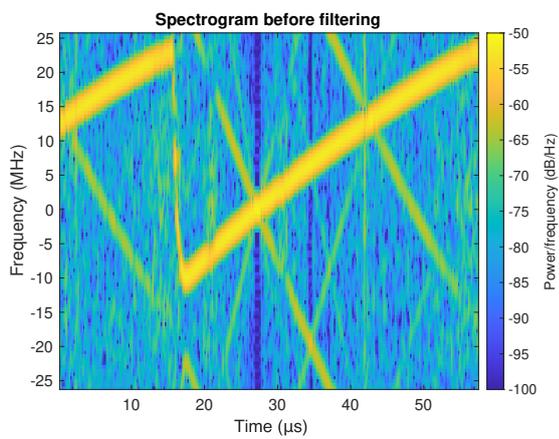


(c) Spectrogram after filtering

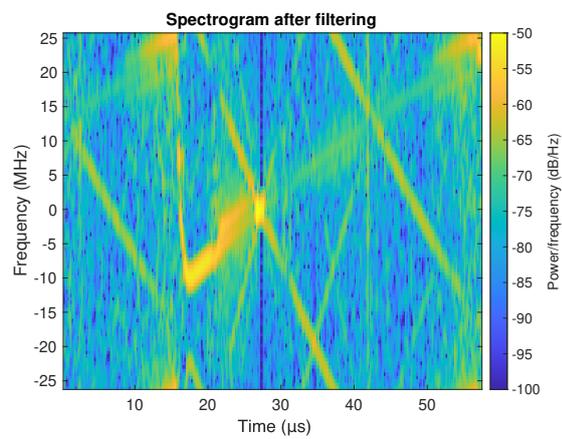
Figure 5.7: Filtering results for slow chirp



(a) Correlation performance before and after filtering



(b) Spectrogram before filtering



(c) Spectrogram after filtering

Figure 5.8: Filtering results for slow chirp

Category	Performance increase	Performance decrease	Total
Fast	23	20	43
Medium	2	9	11
Slow	1	2	3
Total	26	31	57

Table 5.1: This table lists for how many signals the correlation performance was increased by applying the filter.

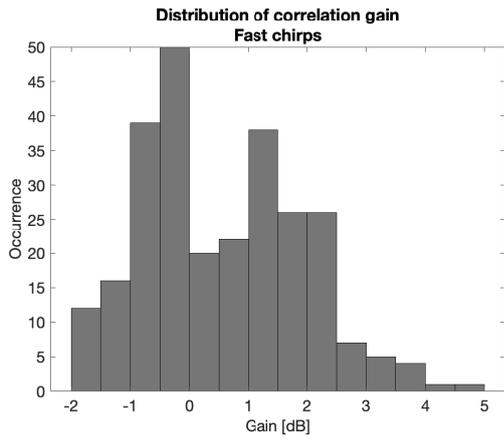
### 5.3.2. Quantitative results

The previous section provided some visually appealing results, but was unable to address the question: *how well does the filter work?* This section discusses results from more recordings.

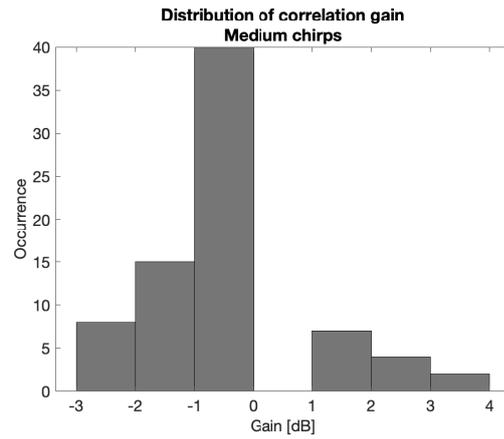
After carefully choosing the chirps that meet the requirement of having a smaller bandwidth than the receiver bandwidth, 57 chirps were left. Most of the chirps were from the *fast* category, a few from *medium* and only 3 from *slow*. For all of these signals first the correlation with respect to all 32 possible PRNs was calculated over 20 L1 periods. Then the filter was applied and afterwards the correlations were calculated again. For all PRNs that had a correlation significantly higher than the noise level, the ratio between the correlation after filtering and before filtering was taken. Per signal, the gain in correlation ratio has been averaged. After these operations, it turns out that for some signals the correlation had increased on average and for some it had decreased. Table 5.1 shows how many signals had benefited from the filter and how many had not.

Unfortunately the results are not very striking. In many cases the performance gets worse after applying the filter. For the medium and slow categories that even happens much more often than improving the performance. Over all the chirps there are also more signals that had decreased correlation performance. Some more detailed insights can be drawn from figure 5.9. It can be seen from figure 5.9d that although there are more signals with decreased performance, the amount by which it decreases is less than in the cases where the performance is increased. In fact the overall mean gain is just above unity at 1.1297. This is mainly due to the contribution of the fast chirps, which can be seen from the similarity with figure 5.9a. The medium and slow chirps on average do not benefit from the filter unfortunately as can be seen from figures 5.9b and 5.9c respectively. This is probably because the slower the chirps, the more they tend to become nonlinear. This leads to inaccuracies because the estimator attempts to estimate a linear chirp and the filter also subtracts a linear chirp.

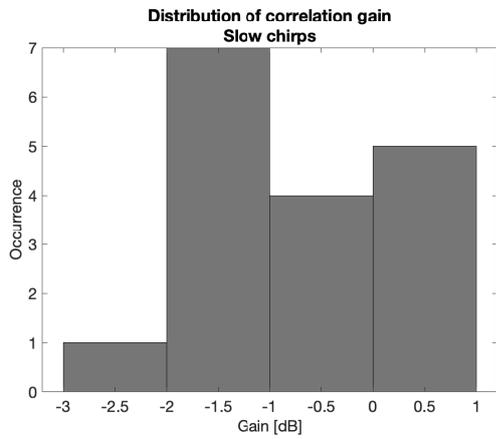
For the cases the filter did increase performance, it did so by approximately 1.5 dB on average.



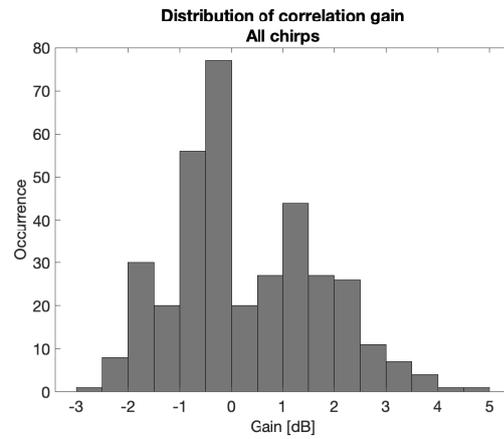
(a) Distribution of amount of gain was achieved for all fast chirps. Overall mean is 0.78 dB



(b) Distribution of amount of gain was achieved for all medium chirps. Overall mean is -0.22 dB



(c) Distribution of amount of gain was achieved for all slow chirps. Overall mean is -0.61 dB



(d) Distribution of amount of gain was achieved for all chirps. Overall mean is 0.53 dB

Figure 5.9: Filtering results for slow chirp



# 6

## Conclusion

The goal of this thesis was to study ways to effectively estimate and filter out chirp interference from GNSS frequency bands. This research question consists of two important parts. The first part concerning the estimation and the second one concerning filtering. These two parts have been addressed in this thesis and will now be concluded upon.

### 6.1. Estimation

For the estimator a variety of options from literature have been researched. They have been implemented in Matlab to test their capabilities with respect to estimation accuracy as well as computation time. It was concluded that the estimators from literature were incapable of estimating all required parameters and so a new estimator had to be designed.

The proposed estimator uses the DCFT and Hankel reduction methods from literature, but adds functionality to find all required parameters. This estimator has been extensively tested on simulated chirps in different noise levels using 1000 Monte Carlo simulations for each noise setting. It was shown that the estimator performs well for chirps with JNR of  $-9$  dB or higher. Visual inspection of the estimates also show that the chirp is indeed tracked correctly by the estimator.

The proposed estimator has also been tested on real chirps that have been recorded by the detector at the NLR. The results for the estimation of real chirps were slightly worse than the simulations. It was difficult however to get accurate metrics of its performance, because the ground truth of the chirp parameters were unknown and had to be estimated by hand. After visually comparing the results of the estimator with estimates done by hand 38 out of 57 times the proposed estimator did a better job than was done by hand. The estimator worked particularly well for fast chirps with chirp period of around  $3 \mu\text{s}$ .

This novel chirp estimator is capable of accurately finding the parameters that constitute a chirp in noisy conditions. This can be used for classification or identification of chirps, but also for filtering as was shown in the filtering part of this thesis.

### 6.2. Filtering

The second part of this thesis concerned filtering. The goal here was to design a filter to remove chirp interference from the GNSS frequency band in such a way that the underlying satellite signals were not distorted. The system was constrained to consist of only one antenna and it was assumed that the raw IQ samples were available. No constraint was placed on the required time to process the filtering.

First three existing filtering techniques with GNSS applications from literature were reviewed. It was concluded that they did not take into account the fundamental structure of a chirp. It was also argued that in theory, if the exact interferer was known, it would be possible to remove it perfectly without any distortion to the underlying signals. Because an estimator of the chirp interference was already built, it was possible to investigate how well such a time-domain subtraction would work in practice.

To this end a filter was constructed which used the proposed estimator to create an accurate replica of the chirp interference. The replica was then subtracted from the contaminated received signal. This subtraction was performed in small windows of 20 samples to correct for phase drifts.

The filter has been tested for a variety of SNR and JNR values. The best results were obtained for an SNR of 0 dB and JNR between 25 dB and 30 dB. In this scenario the correlation of the received signal with GPS satellites could be increased as much as 7 dB for some satellites.

The filter had later been tested on the same real chirps from the detector at the NLR that were used to test the estimator. It worked best on fast chirps, for which the filter improved the correlation in 23 of the 40 cases. For the medium and slow chirps the filter performed poorly and was only able to increase performance in 3 of the 14 cases. It was seen however that for the cases that the filter did improve the performance, it improved the performance by an average of 1.48 dB

### 6.2.1. Filter limitations

This filter is limited in the sense that it only works for chirps that meet certain requirements. The chirp has to lie entirely within the receiver bandwidth and it has to be linear. Both of these requirements are often not met. Especially if receivers with smaller bandwidths are used, many jammers will emit chirps that extend beyond this bandwidth. Also, especially slower chirps tend to exhibit nonlinear behaviour which the filter is unable to deal with. The chirp also needs to have stationary or at least very slowly changing parameters.

Another limitation is that the computation time required by the filter is very high. This was not a problem in this thesis as no constraint was placed on the computation time of the filter. If it is desired, however, to use this filter in a real-time scenario, it has to be sped up significantly.

### 6.2.2. Comparison

In chapter 4 different filtering techniques were mentioned. Making a proper quantitative comparison with those filters is difficult, because the reported results are based on different experimental setups. Roughly looking at the reported results however, it is clear that all of the mentioned filters in section 4.1 outperform the one proposed in this thesis by quite a bit. They report up to 20 dB increase in correlation performance, whereas the filter proposed here achieves about 4 dB at best with real world signals and up to 7 dB in simulations.

The biggest drawback of the proposed filter compared to the ones from literature might be its lack of robustness. The proposed filter in this thesis can work quite well under the right circumstances, but if the chirp interference is slightly distorted the filter breaks down quite heavily. Although the filters from literature do not report on this specifically, it seems they suffer less from this problem.

Lastly, at least some of the filters from literature can work in real-time, introducing only a slight delay to the received signal. This could be advantageous, but has not been achieved with the filter from this thesis.

Looking at these differences, it is safe to say that the proposed filter here is not a good contestant with respect to the alternatives. There are many opportunities for improvement however, as will be addressed in the next section. If these improvements are made, it might be a better competitor to the alternative filters.

## 6.3. Future work

The proposed estimator and filter certainly work under circumstances, but there are ways to improve both. The improvements can be roughly classified in three categories: non-linear chirps, out-of-band chirps and computational complexity.

### 6.3.1. Non-linear chirps

It was seen both for the estimator and the filter that whenever the chirps became slightly curved, the performance broke down significantly. This is not surprising as they both rely on a chirp model that does not allow any curvature. It could be researched however if this problem can be solved by adding a term in the polynomial of the chirp description. This allows for some curvature which could significantly improve the performance of the filter for especially slower chirps, as they tend to be less linear. Adding an extra variable to the chirp model of course increases its complexity and it might be more difficult to fit the parameters.

### 6.3.2. Out-of-band chirps

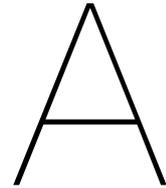
One of the biggest problems with the current estimator and filter is that they cannot cope with chirps that extend beyond the receiver bandwidth. This problem should be addressed in a further research. One approach might be to allow a chirp with pauses, because in essence that is what happens when the chirp extends beyond the band. If a chirp with pauses can be estimated, this can be achieved. Estimating where the pauses are should be relatively easy for chirps with high JNR since the overall power of the signal will clearly show when the chirp is present. Then the parts where the chirp is not present can be disregarded in the estimation process of the other parameters. It will be more difficult for chirps with low JNR or for chirps that have bandwidths which are much wider than the receiver bandwidth. In the last case the chirp reduces to vertical lines in the spectrogram, or just pulses which span the full band. In this case it is probably wise to not treat them as chirps, but use a pulse blanking technique instead of this filter.

### 6.3.3. Computational complexity

Although the estimator and filter do work, they need considerable processing time. This makes it impossible to use them in a real-time application. This might be overcome by performing estimation and filtering in parallel and updating the estimates as well. This might be achieved by using a Kalman filtering approach or similar. This will also make the filter adapt to changes in the chirp over time, which is currently not possible.

The process of finding the correct phase shift is currently also very time consuming and might be improved by understanding better how the phase shifts are introduced. If these can be estimated or predicted more quickly, the filter will be much faster. Even without understanding the nature of the phase shift mismatch, finding the optimal one might be improved by employing a gradient descent-like method.





# Jammer measurements

During this thesis a question arose about the stability of GPS jammers. Most PPDs are bought cheaply, sometimes for as little as \$10. For these retail prices the electronics inside those jammers have to be very simple. The simplicity can lead to poor stability of the chirp parameters such as centre frequency and chirp rate. For the effectiveness as a jammer poor stability is not a problem. A certain randomness in the output might even increase the jamming effectiveness. However, for the estimation and filtering of these interferers the stability of the parameters can have a large impact. If chirp parameters are estimated over the first 100 ms, and used for the next second, it is important to know that the parameters do not change.

Another situation in which the stability is of importance can be fingerprinting devices. If at a certain location chirp interference is measured at different times, for example different days, it might be tempting to compare the chirp parameters. One might assume that if the parameters are the same, they might come from the same physical jammer. This assumption only holds, however, if the parameters do not change from one moment to the next.

Although there is literature on the signals of GPS jammers, such as [5, 6], they do not address this question. The literature focuses on the types of signals as well as their powers, but not the stability of their parameters. To address this question a typical PPD was characterised.<sup>1</sup>

The jammer that was bought is shown in figure A.1, the circuit board is shown in figure A.2. It cost roughly \$10 and is meant for use in a car. The jammer plugs in to the car's 12V power output and is equipped with a dipole antenna to transmit the RF jamming signals. For the tests the antenna was removed and the test equipment was connected to the SMA RF output connector.

The device without antenna measures around  $80 \times 20 \times 20$ mm and the supplied antenna has a length of 60 mm. The device has no buttons, but it has a green LED which is assumed to illuminate when plugged in. It also has a glass fuse which can be replaced by unscrewing the tip of the 12V connector. The rating of this fuse is unknown. Electrical specifications of the device are unknown as they were not specified by the vendor.

## A.1. First glance

Before conducting any tests it was important to get familiarised with the device and to see if it worked at all. It was powered by a 12V laboratory power supply. The output of the jammer was connected via a cable to a spectrum analyser. Upon turning the power on the jammer showed power in the L1 band (around 1.575 GHz) and some harmonics above that, see figure A.3 for the main lobe and A.4 for the harmonics. The RF output power in the L1 band was roughly 45  $\mu$ W or  $-13.5$  dB mW.

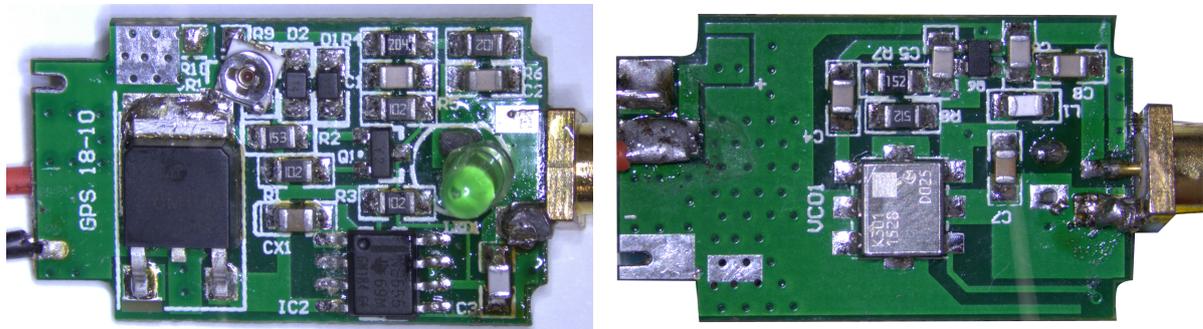
Another observation was that the output of the jammer was a chirp. This was already assumed from its PSD, but the spectrogram confirms it. It is shown in figure A.5. It has a period of roughly 2.5  $\mu$ s and spans approximately 49 MHz centered neatly around 1.575 GHz.

Furthermore it was noticed that the device is very sensitive to its surroundings. Placing your hand in the vicinity of the device ( $< 10$  cm) changes the output of the jammer a lot. The power will change,

<sup>1</sup>The jammer has only been switched on inside the anechoic chamber of the NLR EMC laboratory to avoid interference to any GNSS receivers.



Figure A.1: The jammer



(a) Top view of the jammer circuit

(b) Bottom view of the jammer circuit

Figure A.2: The circuit board of the jammer. On the right is the SMA connector that connects to the antenna.

but also the chirp parameters.

## A.2. EM Leakage

Before conducting any tests it was interesting to know how well the power of the jammer is transmitted through the SMA connector and how much of it radiates somewhere else from the electronics. If this electromagnetic radiation leaks even if it is connected via a cable, the device cannot be tested like this unless the environment is chosen carefully and no one will be harmed by the jamming signal. In order to see if this was the case, the antenna was removed and a  $50\ \Omega$  load was connected to it. If the jammer was carefully matched to  $50\ \Omega$  and properly shielded, no EM should leak.

The jammer along with its load was placed inside an anechoic chamber at the NLR premises. It was powered with 12 V and the electric field strength was measured at a distance of 1 m with a variety of antennas, depending on the frequency. The results are plotted in figure A.6. It can be seen that in the low frequency range, 25 MHz to just below 100 MHz there is a clear line pattern. The lines are approximately 400 kHz apart. This frequency has turned out to correspond to the chirp period of the jammer, i.e.  $2.5\ \mu\text{s}$ . Then the main lobe is seen at the L1 centre frequency along with its harmonics.

It must be stressed that the results from figure A.6 are with a  $50\ \Omega$  load and no antenna! It was tried to swap the load for the antenna that came with the jammer and the difference was around 5 dB. The conclusion that must be drawn from this experiment is that it is not safe to do any testing with the device using a guided medium. A lot of EM will still be radiated by the device, so this must be taken into account when choosing a suitable location for the tests.

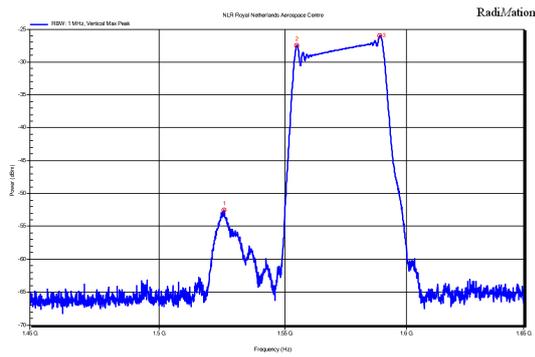


Figure A.3: The main lobe of the jammer, centered around 1.575GHz

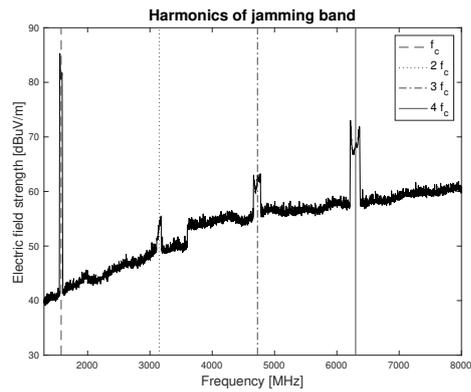


Figure A.4: The main lobe is at  $f_c = 1.575\text{GHz}$  and there are clear harmonics at  $2f_c$ ,  $3f_c$  and  $4f_c$ .

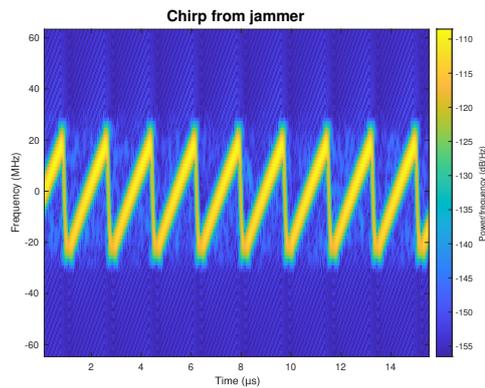


Figure A.5: The spectrogram of the output of the jammer, it is a chirp.

## A.3. Stability

As described, the real question was how stable the jammer is. To answer this question three tests have been conducted. The first test investigates how the chirp parameters over prolonged periods of time. The second test investigates the influence of the ambient temperature and the last test the influence of the supply voltage and the last test

### A.3.1. Long term

The first stability test considered the variation of parameters as a function of time. The jammer was turned on and during 1 hour, a 10 ms IQ recording of its output was saved<sup>2</sup>. For each of the recordings the parameters have been estimated and they are shown in figure A.7. It can be seen that all parameters vary quite a bit over time. There is also a noticeable dip in the bandwidth and chirp rate while at the same time a increase of the centre frequency between around 200 and 1300 s. Upon further inspection it was seen that the lower bound of the chirp band moved up, therefore reducing the bandwidth, increasing the centre frequency and decreasing the chirp rate (as the chirp period remained the same). The cause of this clear dip is unknown.

### A.3.2. Temperature dependence

Jammers like the one examined here are usually used from the car. The temperature inside a car can differ greatly from day to day, so it was interesting to see what the effect of ambient temperature was on this jammer. To find out, it was placed inside a temperature controlled room at the NLR. Here the temperature was brought down to  $-10^\circ\text{C}$  and then increased in steps of  $5^\circ\text{C}$ , with a 5 minute pause after each increase to allow the jammer to settle to the new temperature.

<sup>2</sup>Used hardware was a Rohde & Schwarz FSV3030 spectrum analyser/IQ sampler, sampling at 128 MS/s with 32 bit per phase.

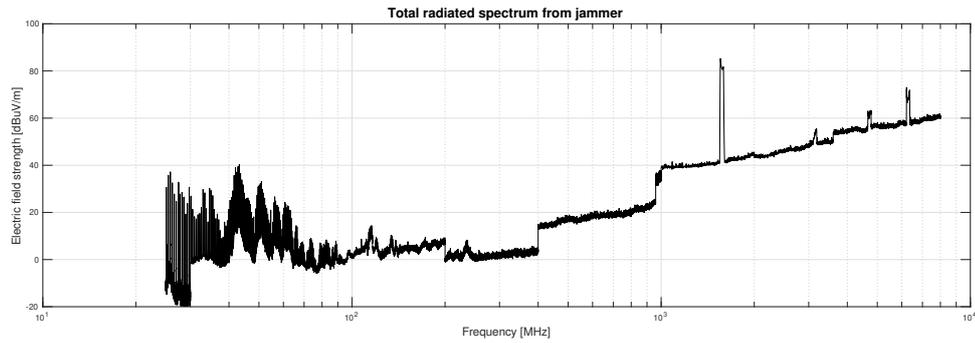
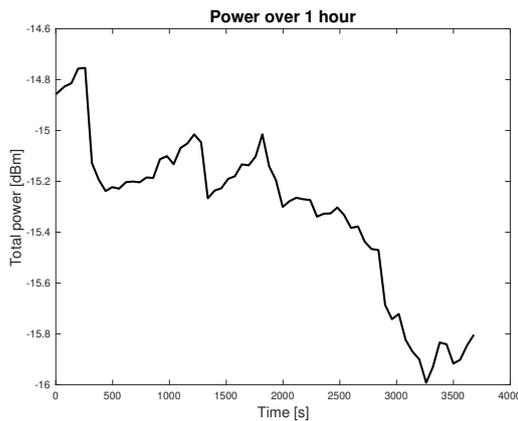
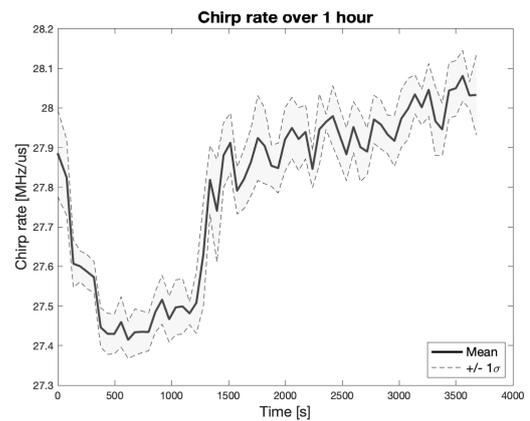


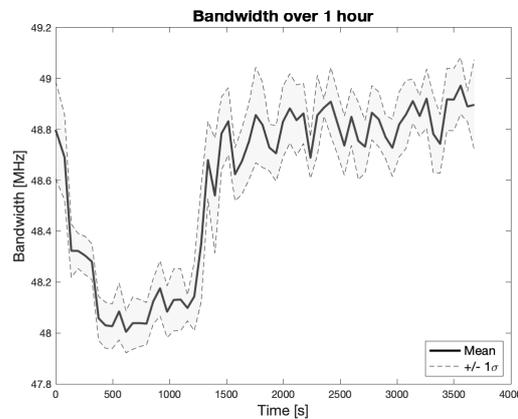
Figure A.6: The full spectrum from 25 MHz to 8 GHz



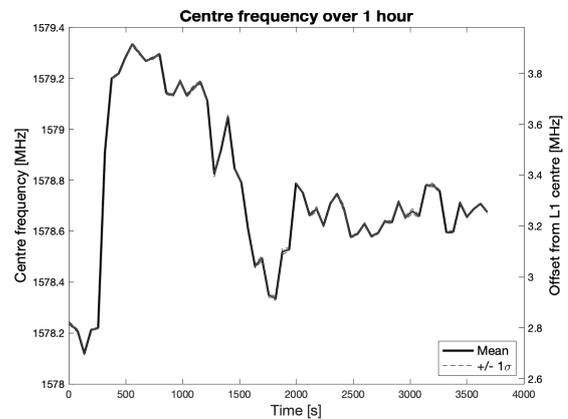
(a) Measured power over 1 hour



(b) Estimated chirp rate over 1 hour



(c) Estimated bandwidth over 1 hour



(d) Estimated centre frequency over 1 hour

Figure A.7: Estimated parameters of the chirp over 1 hour.

Looking at the spectrogram for different temperatures, it turned out that for temperatures below  $10^{\circ}\text{C}$  the chirp no longer looked like a chirp, but became severely distorted. The spectrogram is shown in figure A.8.

The results of the test are shown in figure A.9. Note that due to the signal not being a chirp, there are no estimates for its parameters below  $10^{\circ}\text{C}$ . Only the power has been estimated, because it does not require a chirp. It can be seen that there is a very clear dependence on the temperature. It can be seen that the chirp rate goes down, as well as the centre frequency and the bandwidth goes up. Upon closer inspection of the spectrograms it was seen that the lower bound of the chirp band became lower.

Another remark must be made with respect to figure A.9a as the power during this measurement

was between 13 dB mW and 18 dB mW. Compared to all the other measurements this is insanely high. In all the other measurements the power is around  $-15$  dB mW. No explanation has been found as to why this was the case.

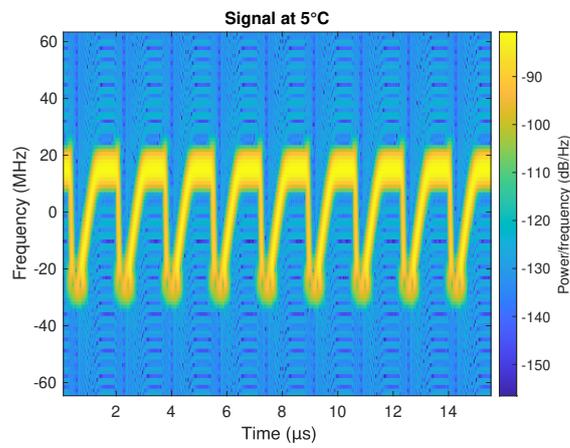


Figure A.8: The signal of the jammer at 5°C

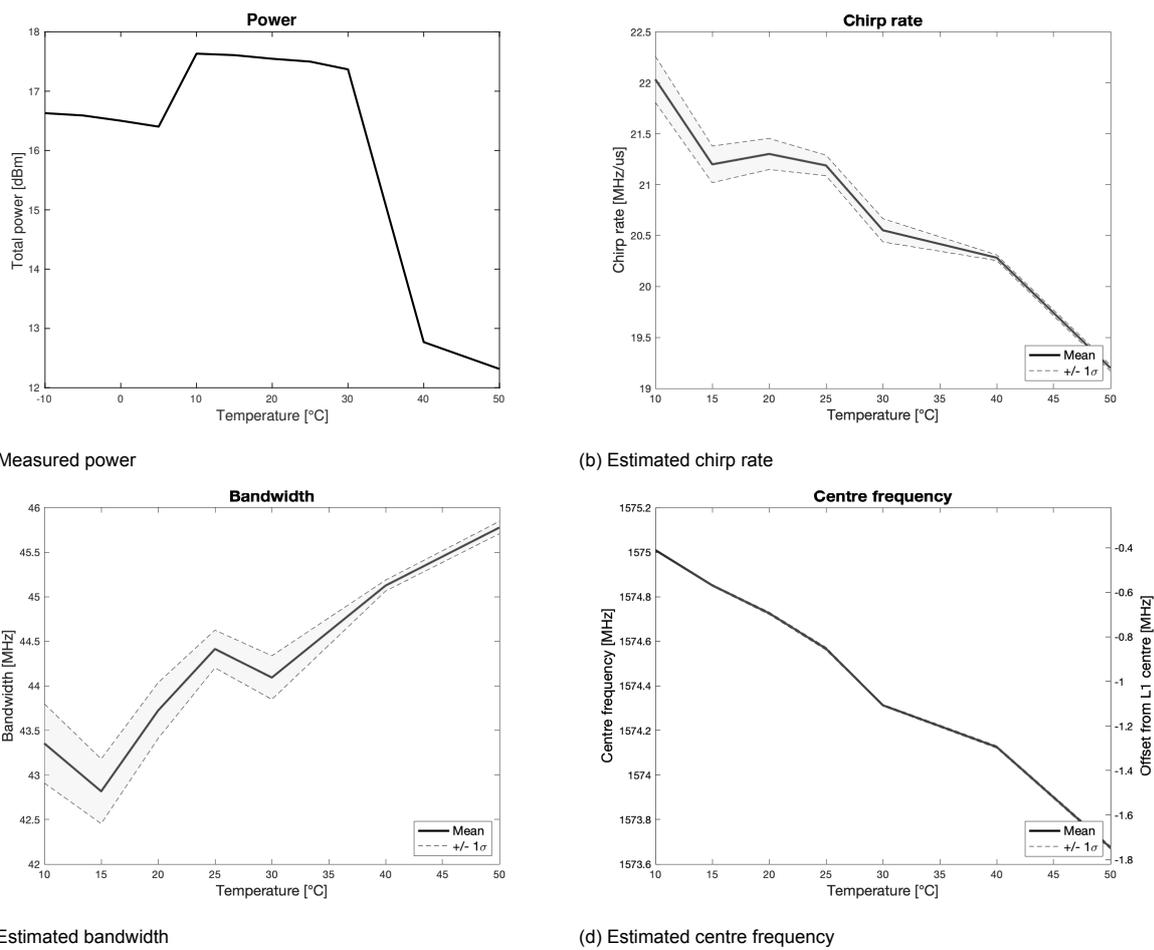


Figure A.9: Estimated parameters of the chirp for different temperatures.

### A.3.3. Supply voltage dependence

The last stability test was with respect to the supply voltage. Here again it must be taken into account that the jammers are usually powered by a cars battery, which can easily swing between 12 V and 14 V. To test the dependence, the jammer was connected to a laboratory power supply and the voltage was varied between 11 V and 14 V with 0.2V increments. The test was conducted twice, first going up and then going back down. During the first run, while going up, until 13.2 V the jammer exhibited the same behaviour as it did during the temperature test below 10 °C, i.e. it emitted a distorted signal.

The results of the test are shown in figure A.10. Run 1 is only shown for the part where the jammer emitted a chirp. By carefully looking at the axes it can be seen that only the power changes significantly, which goes up by around 1 dB. The other parameters change only barely and the data is not convincing enough to make any assumption on the dependence.

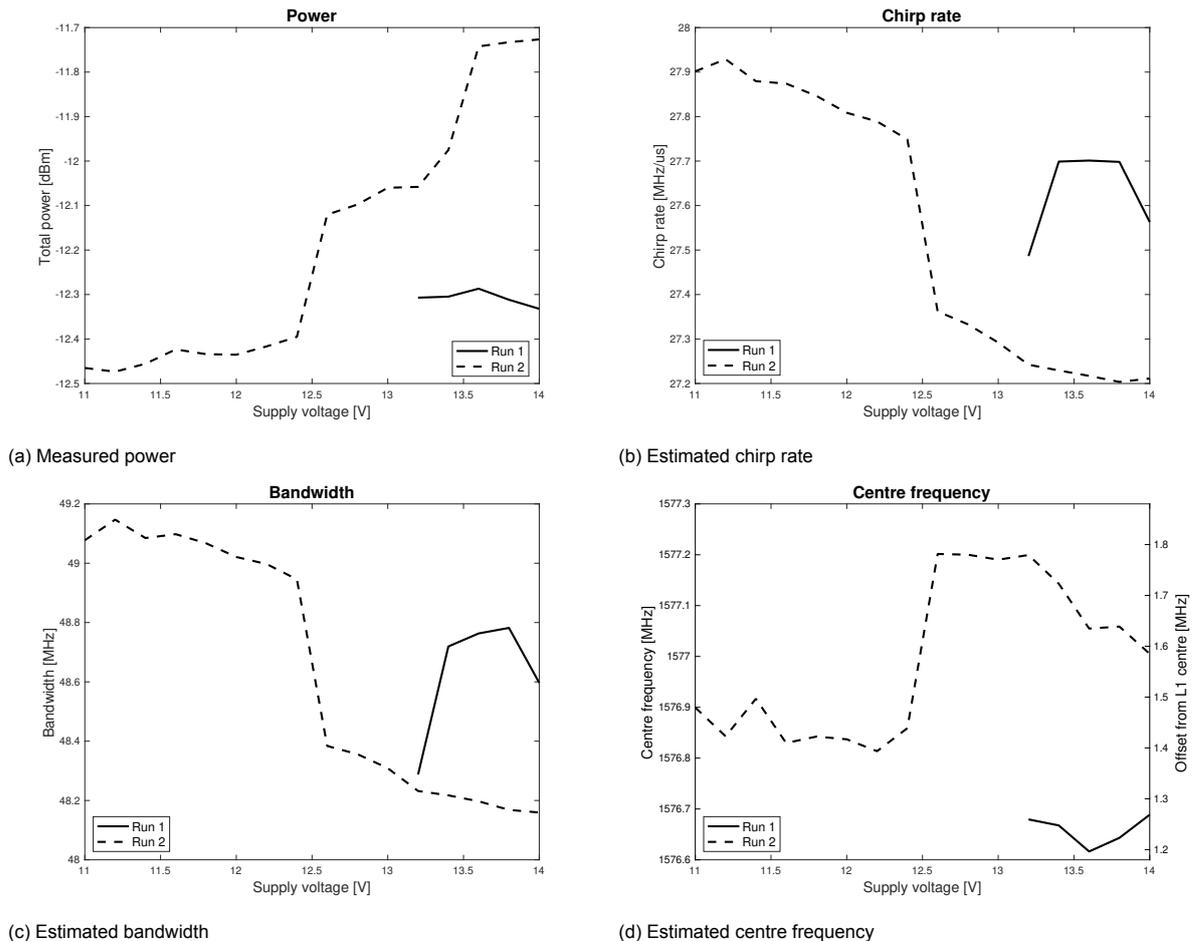


Figure A.10: Estimated parameters of the chirp for different supply voltages.

It was also tried to power the jammer with only 5 V and, as expected, the jammer did no longer produce a chirp. What was left of the signal is shown in figure A.11.

### A.3.4. Conclusion

The main question in this research was how stable cheap jammers are and if they are dependent on temperature and supply voltage. Before any of the tests were conducted, it was already clear that the jammer was far from stable. Just waving your hand around the jammer affected its output greatly and the power from the jammer fluctuated quite a bit. This observation was confirmed by the measurements. Over the course of 1 hour, the parameters vary greatly. It also turned out that the temperature has a direct influence on the parameters of the chirp.

It must be concluded that there is no guarantee that chirps from the same device recorded at different instances are actually the same. It might thus happen that a single jammer is advertently registered as

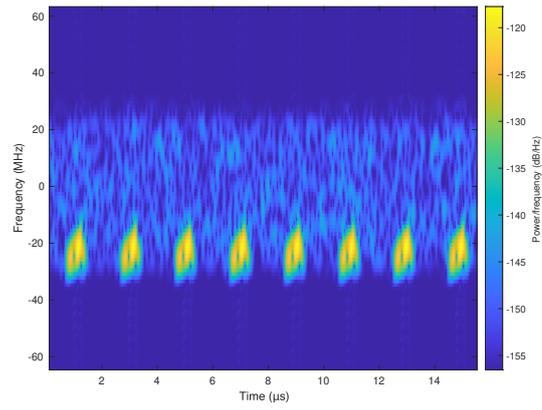


Figure A.11: The signal from the jammer when only 5 V is applied.

two different devices.



# Bibliography

- [1] J.A. Ávila Rodríguez, “Galileo Signal Plan - Navipedia (ESA),” 2011. [Online]. Available: [https://gssc.esa.int/navipedia/index.php/Galileo\\_Signal\\_Plan](https://gssc.esa.int/navipedia/index.php/Galileo_Signal_Plan) [Accessed: 2020-11-18]
- [2] “GPS.gov: GPS Performance.” [Online]. Available: <https://www.gps.gov/systems/gps/performance/> [Accessed: 2020-10-12]
- [3] E. Kaplan and C. Hegarty, *Understanding GPS Principles and Applications, Second Edition*. Artech, 2005.
- [4] A. Boonman and H. U. Schnitzler, “Frequency modulation patterns in the echolocation signals of two vesperilionid bats,” *Journal of Comparative Physiology A*, vol. 191, no. 1, pp. 13–21, 2005.
- [5] R. H. Mitch, R. C. Dougherty, M. L. Psiaki, S. P. Powell, B. W. O’Hanlon, J. A. Bhatti, and T. E. Humphreys, “Signal Characteristics of Civil GPS Jammers,” *Proceedings of the 24th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2011)*, pp. 1907–1919, Sep. 2011.
- [6] T. Kraus, R. Bauernfeind, and B. Eissfeller, “Survey of In-Car Jammers - Analysis and Modeling of the RF Signals and IF Samples (Suitable for Active Signal Cancelation),” *Proceedings of the 24th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2011)*, pp. 430–435, Sep. 2011.
- [7] D. Borio, F. DAVIS, H. Kuusniemi, and L. L. Presti, “Impact and Detection of GNSS Jammers on Consumer Grade Satellite Navigation Receivers,” *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1233–1245, Jun. 2016.
- [8] M. Dumville, “Initial Findings from the STRIKE3 GNSS Interference Monitoring Network,” May 2018. [Online]. Available: <https://www.gps.gov/governance/advisory/meetings/2018-05/dumville.pdf> [Accessed: 2020-10-30]
- [9] A. Rødningsby, A. Morrison, N. Sokolova, N. Gerrard, and C. Rost, “RFI Monitoring of GNSS Signals on Norwegian Highways,” *Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2020)*, pp. 3536–3549, Sep. 2020.
- [10] H. Meikle, *Modern Radar Systems*, 2nd ed. Artech House, 2008.
- [11] G. X. Gao, “DME/TACAN Interference and its Mitigation in L5/E5 Bands,” *Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation*, pp. 1191 – 1200, 2007.
- [12] E. Anyaegbu, G. Brodin, J. Cooper, E. Aguado, and S. Boussakta, “An Integrated Pulsed Interference Mitigation for GNSS Receivers,” *Journal of Navigation*, vol. 61, no. 2, pp. 239–255, Apr. 2008.
- [13] F. DAVIS, *GNSS Interference Threats and Countermeasures*. Artech House, 2015.
- [14] F. Bastide, D. Akos, C. Macabiau, and B. Roturier, “Automatic gain control (AGC) as an interference assessment tool,” *ION GPS/GNSS 2003, 16th International Technical Meeting of the Satellite Division of The Institute of Navigation*, pp. 2042–2053, Sep. 2003.
- [15] E. Axell, “GNSS interference detection.” Feb. 2014. [Online]. Available: <https://www.foi.se/rest-api/report/FOI-R--3839--SE> [Accessed: 2020-09-17]

- [16] A. Balaei, A. Dempster, and J. Barnes, "A Novel Approach in Detection and Characterization of CW Interference of GPS Signal Using Receiver Estimation of C/No," in *2006 IEEE/ION Position, Location, And Navigation Symposium*, Apr. 2006, pp. 1120–1126, iSSN: 2153-3598.
- [17] European Global Navigation Satellite Systems Agency, "STRIKE3 Project." [Online]. Available: <http://gnss-strike3.eu/> [Accessed: 2021-01-11]
- [18] C. Curry, "Sentinel Project Report on GNSS vulnerabilities," Chronos Technology, Tech. Rep., Apr. 2014. [Online]. Available: [https://www.chronos.co.uk/files/pdfs/gps/SENTINEL\\_Project\\_Report.pdf](https://www.chronos.co.uk/files/pdfs/gps/SENTINEL_Project_Report.pdf) [Accessed: 2021-01-11]
- [19] L. Scott, "J911: Fast Jammer Detection and Location Using Cell-Phone Crowd-Sourcings," Nov. 2010. [Online]. Available: <https://www.gpsworld.com/j911-fast-jammer-detection-10720/> [Accessed: 2020-09-16]
- [20] S. Lo, Y. H. Chen, D. Akos, B. Cotts, and D. Miralles, "Tests of Crowdsourced Smartphones Measurements to Detect GNSS Spoofing and Other Disruptions," *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, pp. 373–388, Jan. 2019.
- [21] L. Strizic, D. M. Akos, and S. Lo, "Crowdsourcing GNSS Jammer Detection and Localization," in *Proceedings of the 2018 International Technical Meeting of The Institute of Navigation*, Feb. 2018, pp. 626–641.
- [22] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*. Pearson, 1998, vol. 1.
- [23] P. O'Shea, "A new technique for instantaneous frequency rate estimation," *IEEE Signal Processing Letters*, vol. 9, no. 8, pp. 251–252, Aug. 2002.
- [24] Xiang-Gen Xia, "Discrete chirp-Fourier transform and its application to chirp rate estimation," *IEEE Transactions on Signal Processing*, vol. 48, no. 11, pp. 3122–3133, Nov. 2000.
- [25] T. Abatzoglou, "Fast maximum likelihood joint estimation of frequency and frequency rate," in *ICASSP '86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 11, Apr. 1986, pp. 1409–1412.
- [26] G. Turin, "An introduction to matched filters," *IRE Transactions on Information Theory*, vol. 6, no. 3, pp. 311–329, Jun. 1960.
- [27] Pingyi Fan and Xiang-Gen Xia, "A modified discrete chirp-Fourier transform scheme," in *WCC 2000 - ICSP 2000. 2000 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress 2000*, vol. 1, Aug. 2000, pp. 57–60.
- [28] L. A. A. Irkhis and A. K. Shaw, "Compressive Chirp Transform for Estimation of Chirp Parameters," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5.
- [29] S. W. Lang and B. R. Musicus, "Frequency estimation from phase differences," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, May 1989, pp. 2140–2143.
- [30] P. M. Djuric and S. M. Kay, "Parameter estimation of chirp signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 12, pp. 2118–2126, Dec. 1990.
- [31] C. L. DiMonte and K. S. Arun, "Tracking the frequencies of superimposed time-varying harmonics," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, Apr. 1990, pp. 2539–2542.
- [32] B. Volcker and B. Ottersten, "Chirp parameter estimation using rank reduction," in *Conference Record of Thirty-Second Asilomar Conference on Signals, Systems and Computers (Cat. No.98CH36284)*, vol. 2, Nov. 1998, pp. 1443–1446.
- [33] ———, "Chirp parameter estimation from a sample covariance matrix," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 603–612, Mar. 2001.

- [34] M. Ge, G. Wei, and X. Zhou, "Chirp parameter estimation from tensor decomposition," in *2014 12th International Conference on Signal Processing (ICSP)*, Oct. 2014, pp. 2057–2062.
- [35] Minsheng Wang, A. K. Chan, and C. K. Chui, "Linear frequency-modulated signal detection using Radon-ambiguity transform," *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 571–586, Mar. 1998.
- [36] J. Song, Y. Wang, and Y. Liu, "Iterative Interpolation for Parameter Estimation of LFM Signal Based on Fractional Fourier Transform," *Circuits, Systems, and Signal Processing*, vol. 32, no. 3, pp. 1489–1499, Jun. 2013.
- [37] A. Rügamer, S. Joshi, J. R. v. d. Merwe, F. Garzia, W. Felber, J. Wendel, and F. M. Schubert, "Chirp Mitigation for Wideband GNSS Signals with Filter Bank Pulse Blanking," in *30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, Nov. 2017, pp. 3924–3940.
- [38] D. Borio and P. Closas, "Robust transform domain signal processing for GNSS," *NAVIGATION, the Journal of the Institute of Navigation*, vol. 66, no. 2, pp. 305–323, Apr. 2019.
- [39] L. Musumeci and F. Dovic, "A comparison of transformed-domain techniques for pulsed interference removal on GNSS signals," in *2012 International Conference on Localization and GNSS*, Jun. 2012, pp. 1–6.
- [40] W. Qin, F. Dovic, M. T. Gamba, and E. Falletti, "A Comparison of Optimized Mitigation Techniques for Swept-frequency Jammers," in *2019 International Technical Meeting of The Institute of Navigation*, Feb. 2019, pp. 233–247.
- [41] D. Borio and C. Gioia, "Interference mitigation: impact on GNSS timing," *GPS Solutions*, vol. 25, no. 2, Apr. 2021.