

Direct Signal Interference Suppression and Target Detection for Low-cost SDR-based Passive Bistatic Radar

Oskar Jonsson

Master of Science Thesis in Electrical Engineering

**Direct Signal Interference Suppression and Target Detection for Low-cost
SDR-based Passive Bistatic Radar**

Oskar Jonsson

LiTH-ISY-EX--22/5503--SE

Supervisor: **Daniel Bossér**
ISY, Linköping University
Thomas Sjögren
Swedish Defence Research Agency, FOI
Axel Tryblom
Swedish Defence Research Agency, FOI

Examiner: **Gustaf Hendeby**
ISY, Linköping University

*Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden*

Copyright © 2022 Oskar Jonsson

Abstract

Passive radar is a technology for detection of targets using echoes of existing radio transmitters, such as FM-radio. Since only receivers are needed for operation, a passive radar system has the possibility of being implemented using low-cost hardware. Using lower cost implementations to cover blind-spots of other, more sophisticated systems, could be a viable solution for full radar coverage. To achieve this, an understanding of the effects such low-cost systems have on the performance of a radar is needed.

A prominent problem for passive radar is that the interference caused by the direct signal from the transmitter used and reflections from uninteresting terrain, called clutter, can drown out the echoes from targets. This thesis compares the direct signal interference (DSI) suppression algorithms: ECA, ECA-S, ECA-B, NLMS and FBNLMS when run on data from a low-cost receiver called KerberosSDR.

It is found that the low ADC resolution of 8 bits is a limiting factor for KerberosSDR. Random noise in the receiver can also limit the performance.

None of the tested algorithms are any more or less affected by the ADC resolution or the noise. The first difference appears when comparing the execution times, where FBNLMS is 10–20 times faster than the other algorithms. However, the slower rate of convergence for FBNLMS and NLMS causes them to lose performance in environments where the DSI and clutter are considerably stronger than the target echoes. The algorithms FBNLMS and NLMS also lose performance due to their inability to model frequency shifted echoes as unwanted. The main disadvantages of ECA, ECA-B and ECA-S are their long execution time.

It is concluded that FBNLMS would be the best candidate in most cases for low-cost hardware, as it allows execution on slower hardware and the main disadvantages not being too prominent in the use case of covering blind-spots of other systems.

Acknowledgments

I would like to thank my supervisor from FOI, Thomas Sjögren, for all valuable discussions and help I received in the beginning of this project. I would also like to thank everyone else involved from FOI, especially Axel Tryblom and Rolf Ragnarsson, for all their input and support during my time here. I am also grateful to my supervisor Daniel Bossér and examiner Gustaf Hendeby from Linköping University for their input during my work on this thesis.

Linköping, June 2022
Oskar Jonsson

Contents

Notation	ix
1 Introduction	1
1.1 Problem formulation	1
1.1.1 Background	1
1.1.2 Aim	3
1.1.3 Research questions	3
1.1.4 Limitations	3
1.2 Outline	3
1.3 System overview	4
2 Passive radar signal processing	7
2.1 Geometry	7
2.2 Different illuminators	9
2.3 Range-Doppler matrices	10
2.4 Direct signal interference and clutter	11
2.5 Target detection	13
2.5.1 Constant false alarm-rate detector	13
2.5.2 Peak to noise floor ratio	15
3 DSI suppression algorithms	17
3.1 Extensive cancellation algorithm	18
3.1.1 Properties of ECA solution	19
3.1.2 Batched version of ECA	19
3.1.3 Sliding version of ECA	20
3.2 Least mean square adaptive filters	21
3.2.1 LMS and Normalized LMS	24
3.2.2 Block LMS	25
3.2.3 Fast block normalized LMS	25
4 Hardware limitations and tests	29
4.1 Hardware limitations	29
4.2 Test of DSI suppression algorithms	30

4.2.1	Test metrics and purpose	30
4.2.2	Test scenarios	31
4.2.3	Test construction	32
4.3	Test of impact of hardware limitation	33
4.4	Simulation of surveillance signal	33
5	Results	35
5.1	Parameters for DSI suppression algorithms	35
5.2	DSI suppression for KerberosSDR	37
5.2.1	General results	37
5.2.2	Results for target with low bistatic velocity	42
5.2.3	Results for far away targets	46
5.2.4	Parameter sensitivity analysis	47
5.3	DSI suppression for PXIe-5644	51
5.4	Impact of hardware limitation results	52
5.4.1	Quantization	52
5.4.2	Additive noise and DSI	54
6	Conclusion and future work	59
6.1	Conclusions	59
6.2	Future work	60
	Bibliography	63

Notation

NOTATIONS

Notation	Description
\mathbb{Z}	The set of all integers

ABBREVIATIONS

Abbreviation	Description
ADC	Analog-to-digital converter
BLMS	Block least mean square
CAF	Cross-ambiguity function
CA-CFAR	Cell averaging constant false alarm rate
CFAR	Constant false alarm rate
DFT	Discrete Fourier transform
DSI	Direct signal interference
ECA	Extensive cancellation algorithm
ECA-B	Batched extensive cancellation algorithm
ECA-S	Sliding extensive cancellation algorithm
FBLMS	Fast block least mean square
FBNLMS	Fast block normalized least mean square
FFT	Fast Fourier transform
FIR	Finite impulse response
IoO	Illuminator of opportunity
LMS	Least mean square
MSE	Mean squared error
NLMS	Normalized least mean square
PNFR	Peak to noise floor ratio
SDR	Software-defined radio

1

Introduction

This chapter presents background and the main aim of this master's thesis. The problem is also summarized as two research questions.

1.1 Problem formulation

This section presents the problem investigated in this thesis and the motivation for investigating it.

1.1.1 Background

A passive radar is a system for estimating position and velocity of different targets based on the reflection of radio waves. Unlike an active radar, the passive radar does not transmit any radio waves on its own. It instead relies on listening to waves from existing transmitters, such as those used for FM-radio and Digital Video Broadcasting (DVB). The transmitters are in this context referred to as illuminators of opportunity (IoO). Since the system does not transmit its own signal, it does not use up any spectrum nor does it reveal its position. The lack of a transmitter also allows for more cost-efficient implementations [1].

Since the transmitters used by the passive radar system are separated from the receiver by a distance in space, the term passive bistatic radar is also used. The system measures the signal received directly from the transmitter to use as a reference signal. It also measures the time delay and Doppler shifts of waves that have reflected from potential targets before reaching the receiver. Based on the time delay and known position of the transmitter, the system can calculate possible positions of the target. The Doppler shift of the reflected signal gives information about the target's velocity. By fusing the measurements from multiple

transmitter-receiver pairs, it is possible to estimate the position and velocity of a target.

A side effect of the passive radar's bistatic nature is the range resolution get worse when the target is close to being in between the passive radar and the transmitter it uses. The area with low range resolution is effectively a blind-spot for that transmitter and receiver pair. Like for an active radar, there can also be terrain blocking either the radio-waves from the transmitter or the echo from the target, creating more blind-spots where targets cannot be detected. A way to remedy this, is to let the receiver use multiple transmitters, creating more transmitter-receiver pairs where the blind-spots ideally are not overlapping. There is however a limit to using more transmitters, as the placement of transmitters cannot be controlled by the designer of the passive radar system and terrain blocking target echoes from reaching the receiver is still a problem. To guarantee full coverage and more information for estimating target positions, passive radar receivers need to be placed in multiple strategic locations [1]. For this solution to be economically viable, the cost of each passive radar receiver system need to be low.

Low-cost software-defined radio (SDR) receivers are available on the market and have been shown to function in passive radar applications [2]. A candidate that have shown promise for this application is the KerberosSDR receiver [3]. Due to the device's low-cost, it could make it economically viable to place in regions where it would cover blind-spots of more sophisticated and expensive setups. It is however unknown how such low-cost hardware affects the performance of passive radar.

An important aspect of the low-cost KerberosSDR is its limited analog to digital converter (ADC) resolution of 8-bits, giving the input-signal a bit-depth of 8-bits. The effect of quantization noise is therefore interesting for evaluating this hardware solution. It is also interesting in general to know the effect of ADC resolution when choosing hardware for passive radar, as lower ADC resolution often translate to lower hardware cost or a higher sampling rate. In fact, many ADC circuits can be configured to either output higher bit-depth signals or signals with a higher sampling rate and therefore bandwidth. A higher bandwidth can, in turn, allow listening to more transmitters simultaneously, leading to better estimations of the target's position and velocity. In other words, choosing the minimum required bit-depth may allow for better coverage for the passive radar.

A signal processing problem for passive radar is the separation of the direct signal and reflected signals of interest. The signals are referred to as reference signal and surveillance signal, respectively. Since the direct signal is orders of magnitude stronger than the surveillance signal, it is important to remove it before doing any further signal processing. It is also of interest to remove reflections from uninteresting objects in the environment, such as buildings. The problem is known as direct signal interference (DSI) suppression [4].

In order to implement low-cost passive radar systems, it is essential to have algo-

rithms that can fulfill the requirements of:

- Effectively removing the unwanted DSI and clutter.
- Not removing echoes from interesting targets.
- Being unhindered by a lower bit-depth of the sampled signals.
- Having a low computational burden.

There exist multiple candidate algorithms for this filtering, e.g. [5–7]. Different approaches yield different performance at the cost of different amounts of computational power. It is still an active field of research.

In the end, there is a need for low-cost systems in order to make it feasible to achieve full coverage using passive radar technology. There is therefore also a need to investigate what shortcomings and limitations low-cost hardware imposes on passive radar systems, and how effective different signal processing algorithms are at dealing with the limitations.

1.1.2 Aim

The focus of this master thesis is on comparing signal processing algorithms for increased probability of detection of targets by a passive bistatic radar, specifically on direct signal interference suppression. The comparison takes into account the effect of hardware limitations related to using lower-cost SDR hardware, with a focus on signal bit-depth, processing power and thermal noise.

1.1.3 Research questions

To fulfill the aim in section 1.1.2 the thesis investigates the following questions:

- What algorithms for direct signal interference suppression are feasible to implement on a low-cost passive bistatic radar system?
- What are the main limiting constraints put on a passive bistatic radar by using low-cost hardware, and how do they affect the target echo to noise floor ratio?

1.1.4 Limitations

Passive radar can use a variety of different radio signals as illuminators of opportunity, this thesis however only considers the use of FM-radio signal.

1.2 Outline

The thesis is outlined as follows. The rest of chapter 1 gives a short description of the hardware and setup used to investigate the problem described in section 1.1. Chapter 2 describes the signal processing used for passive radar and the problem of direct signal interference. Chapter 3 describes the algorithms for direct signal

interference suppression tested in this thesis. Chapter 4 presents the hardware limitations imposed by low-cost hardware and how the effects are simulated and how performance is evaluated. Chapter 5 then presents the results from the test described in chapter 4. Finally, chapter 6 concludes the thesis and contains possible future work.

1.3 System overview

This section gives a short introduction to the low-cost system and antenna setup used in the thesis.

The main passive radar system in this thesis is based on a low-cost hardware receiver called KerberosSDR [3]. The system utilizes four directional antennas placed so that they receive signals from the north-east, south-east, south-west and north-west, respectively. Each antenna is connected to one channel of a KerberosSDR receiver that in turn communicates with a Raspberry Pi 4B over USB. The Raspberry Pi has software that synchronizes samples from the four channels with respect to time and saves the data. See Figure 1.1 for an illustration of the basic components and how the radio signals ideally travel.

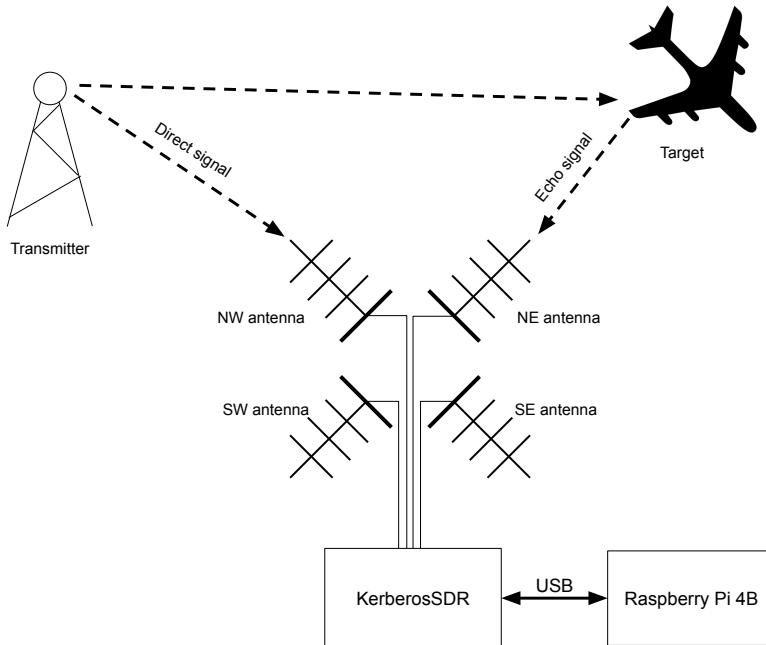


Figure 1.1: Overview of the system and illustration of possible ideal signal path.

The main component that is to be investigated in this thesis, the KerberosSDR, consist of four separate SDR-receivers driven by a common clock and oscillator. The receivers are commonly referred to as RTL-SDR (RTL being a short for the RTL2832U demodulator used) and are based on hardware that was originally intended for receiving terrestrial digital video broadcasting (DVB-T) as a USB-dongle, but where the driver has been modified to allow SDR-like operation where raw I/Q-values can be read from the device [8]. As the RTL-SDR was not intended to be used this way, there exist no official public datasheet or specification for it. All documentation available is either gathered from testing, reverse-engineering or taken from leaked documents intended for internal use for its manufacturer. The mass production of the RTL-SDR hardware has however made it cheap.

The RTL-SDR used in the KerberosSDR consist of two main components, a R820T2 tuner and a RTL2832U demodulator. The tuner is used for band-pass filtering and extracting the frequency content around a specified center frequency, and the demodulator is used to then sample the signal.

The RTL-SDRs' in the KerberosSDR are intended for single channel use and while their common clock allows all four to have the exact same sample rate, the generated data has no indication of the time when the sample was taken. Since the time it takes to start the sampling is not deterministic, each channel has an unknown time delay in the sampled signal. To estimate this time delay and allow for synchronization of the samples from different channels, a common noise generator is connected to all channels. See Figure 1.2 for an overview of the internal components of the KerberosSDR.

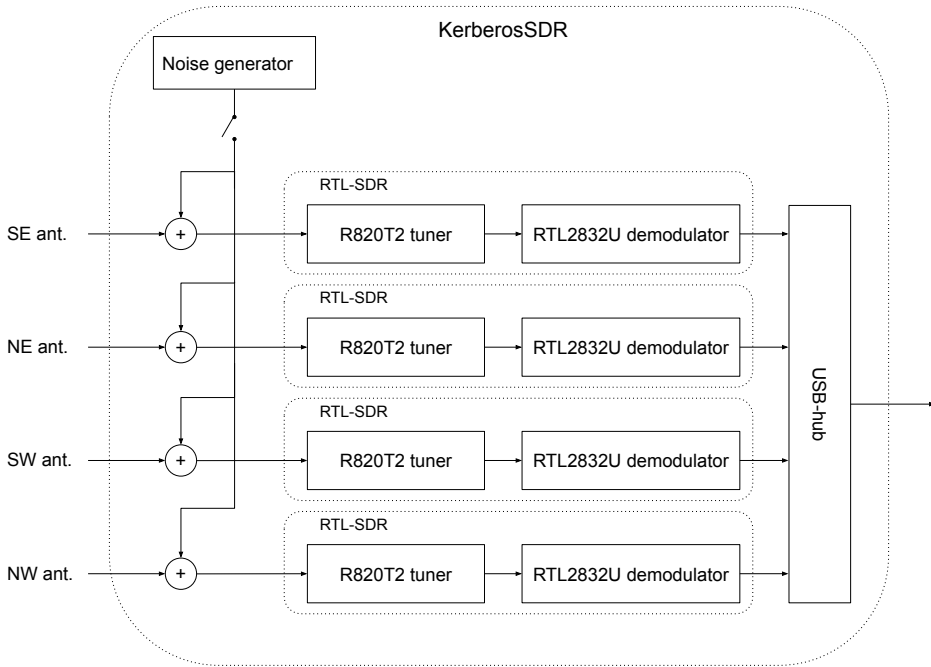


Figure 1.2: Overview of KerberosSDR’s internal components and the signals’ path through them. Note that the RTL2832U demodulators are only used to sample the analog radio signal in this thesis, no demodulation is done by them.

2

Passive radar signal processing

This chapter gives a theoretical background to how data from a passive radar is processed in order to detect possible targets and measure the targets' bistatic distance and speed. Furthermore, the chapter presents the problem from direct signal interference and unwanted clutter and demonstrates why they need to be suppressed before targets can be detected. It also provides an understanding of how the detection performance in the following chapters is measured.

2.1 Geometry

As the working principle of a passive radar system is to let uncooperative illuminators of opportunity (IoO) create the radio-signals that are reflected on targets back to the radar system, see Figure 2.1, it is necessary for the system to know what the original signal sent from the IoO is. This is typically solved by using one directional antenna dedicated to receiving the radio signal sent directly from the IoO. It can alternatively be solved by utilizing multiple omnidirectional antennas and beamforming to get a clear signal from the direction of the IoO as done in [9]. The focus in this thesis is however on the first approach of using a directional antenna. In either case, the estimation of the direct signal is referred to as the reference signal or channel. An arbitrary number of other directional antennas can then be used to listen for the echo generated when the radio signal is reflected on targets. These signals are referred to as surveillance signals or channels. For the basic case, only one such antenna and channel is considered.

With the direct signal known, it is possible to find the time delay between the direct signal and a reflection. Using the known speed of radio waves (the speed of light) this time delay can be converted to a distance, the bistatic range, R_B ,

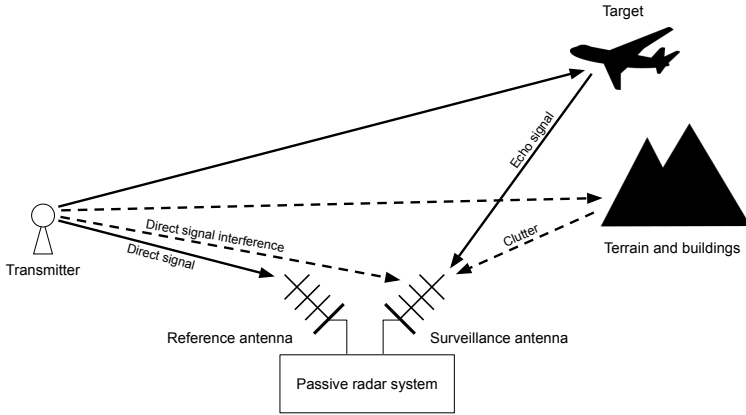


Figure 2.1: Possible paths for the radio signals for the basic case with one surveillance channel. Dashed lines mark unwanted signals.

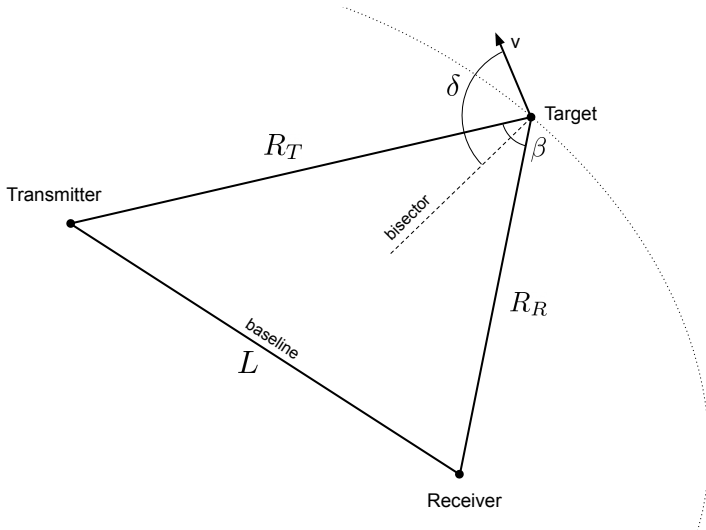


Figure 2.2: The geometry of a (passive) bistatic radar.

consisting of:

$$R_B = R_T + R_R - L, \quad (2.1)$$

where R_T is the distance between the target and the transmitter, R_R the distance between target and receiver (the radar) and L the distance between transmitter and receiver, see Figure 2.2. The velocity of the target can also create a Doppler shift in the frequency of the echo. The Doppler shift, f_D , depends on the rate of change in the transmitter-target-receiver path, and the wavelength, λ , of the used FM-carrier frequency, and is given by:

$$f_D = \frac{1}{\lambda} \frac{dR_B}{dt}. \quad (2.2)$$

This can also be written as:

$$f_D = \frac{2v}{\lambda} \cos(\delta) \cos(\beta/2), \quad (2.3)$$

where v is the speed of the target, δ the angle of the target's velocity relative to the bisector of the transmitter-target and the target-receiver paths and β the angle between the transmitter-target and the target-receiver paths [1].

With the location of the IoO and the receiver known, a measurement of the bistatic range means that the position of the target is on an ellipsoid. In the 2D plane, the possible target locations create an ellipse with focal points where the transmitter and receiver are located, see the dotted line in Figure 2.2. To narrow down the location of the target, the receiver can listen for signals from multiple transmitters to create multiple ellipsoids. The intersection points of the ellipsoids then becomes the possible target locations. Finally, the bistatic velocity, here defined as the rate of change of the bistatic range:

$$v_B := \frac{dR_B}{dt}, \quad (2.4)$$

can be compared between transmitters to remove false target locations [1].

2.2 Different illuminators

As the passive radar relies on non-cooperative transmitters, the choice of IoO, and the waveform associated with it, have a large impact on the system's performance. Some common choices of IoO are digital radio/TV, Wi-Fi and FM-radio transmitters [10].

Digital radio (DAB, digital audio broadcast) and television (DVB, digital video broadcast) have successfully been used for passive radar purposes [11]. The broadcast waveforms use coded orthogonal frequency division multiplex (COFDM) modulation. The COFDM modulation allows for high information content for the bandwidth and the ability to reject multipath propagation. The modulated waveform is noise-like and has a quickly decreasing autocorrelation function that allows for high range resolution. Its digital nature also allows for reconstruction of the signal from the digital content, which can reduce the noise input to the

passive radar system [12].

Wi-Fi radio waves have also been exploited as illuminators for short range passive radar applications [13]. While the waves does not travel far, the number of Wi-Fi transmitters still allow good coverage in a populated environment.

Finally, FM-radio is one of the earliest used illuminators, and the one that will be used by the system investigated in this thesis. There are many systems that have demonstrated its usefulness for long range passive radar applications [14, 15]. The transmitters' high transmit power and placement in high points allows the waves to reach far and create large areas of coverage. With the data content having an effective bandwidth of approximately 50 kHz, a passive radar system based on the illuminator can have a range resolution of approximately 3 km [1]. One weakness of the selected IoO is that the analog modulation used can result in an effective bandwidth, and therefore width of autocorrelation peak, that changes with the content. For instance, silence or people talking usually have a low effective modulated signal bandwidth and will give poor range resolution for the passive radar. On the other hand, intense rock music is an example of content that can have a wider bandwidth and give better range resolution [10].

2.3 Range-Doppler matrices

In order to estimate any time delays and Doppler shifts from possible echoes, the surveillance channel is cross-correlated with the reference signal and Doppler shifted versions of the reference signals. If the surveillance channel contains an echo, the cross correlation will give a peak value at the time delay and frequency shift corresponding to the time delay and Doppler shift of the echo. As a time delay can be converted to distance, peaks in the cross-correlation give information about bistatic range and bistatic velocity of targets.

The correlation for different time delays and Doppler shifts is given by the cross-ambiguity function (CAF). When using a complex baseband representation of the signals, a model for the received echo, $x_i(t)$, from the target i at bistatic range $R_B^{(i)}$ and with a Doppler shift of $f_D^{(i)}$, as a function of time, t , is [16]:

$$x_i(t) = C_i \cdot x_r \left(t - \frac{R_B^{(i)}}{c} \right) \cdot \exp \left(j2\pi f_D^{(i)} t \right), \quad (2.5)$$

where C_i is a complex constant, $x_r(t)$ is the reference signal and c is the speed of light. The signal received by the surveillance antenna, $x_e(t)$, can then be modeled as:

$$x_e(t) = \sum_{i \in I_{\text{targets}}} x_i(t) + \omega(t), \quad (2.6)$$

where I_{targets} is the set of indices for all visible target and $\omega(t)$ is all uninteresting signal content such as noise and reflections from terrain. To be able to estimate a cross-correlation, processing of passive radar data is done in blocks containing

data collected during a time denoted as the integration time.

Using the model (2.6), the cross-ambiguity function for the integration time of T_{int} becomes [16]:

$$\psi(R_B, f_D) = \int_{-T_{int}/2}^{T_{int}/2} x_e(t) \cdot x_r^* \left(t - \frac{R_B}{c} \right) \cdot \exp(-j2\pi f_D t) dt, \quad (2.7)$$

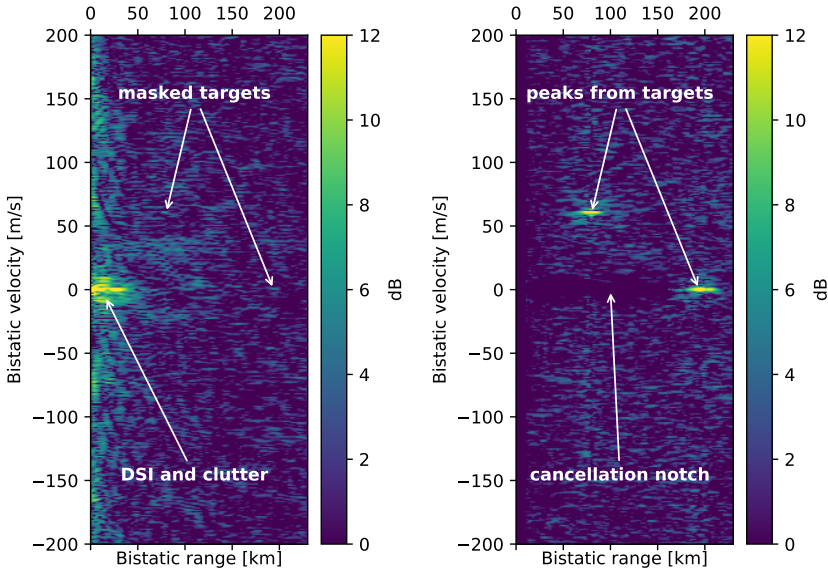
where $*$ denotes complex conjugation. This can also be seen as filtering the surveillance channel with a matched filter. As the signals are sampled and the processing happens in the digital domain, a discrete-time version of (2.7) is used instead:

$$\psi[m, k] = \sum_{n=0}^{N-1} x_e[n] \cdot x_r^*[n - m] \cdot \exp\left(-j \frac{2\pi}{N} kn\right), \quad (2.8)$$

where m is the time delay in an integer number of samples, k is the Doppler shift as an integer number of frequency bins (one bin is approximately $1/T$ Hz) and N is the number of samples taken during the integration time T_{int} . The function in (2.8) is evaluated for all values of m and k in an interval in order to find the peaks originating from targets. With the evaluated values for (2.8) organized as elements in a matrix, the resulting matrix is referred to as the range-Doppler matrix. The calculation of the matrix is commonly done in the frequency domain for faster execution time [16]. Finally, the absolute value of the cross-ambiguity function is used when looking for peaks. Throughout this thesis, range-Doppler matrixes are illustrated by drawing them as images where the color of each pixel correspond to the decibel value of the corresponding element in the matrix, $|\psi[m, k]|$, see Figure 2.3 for an example. All matrix elements are normalized so that 0 dB is equal to the median of all elements.

2.4 Direct signal interference and clutter

While the cross-ambiguity function in (2.8) should give peaks when m and k corresponds to an echo from a target of interest, they are almost always masked by correlation sidelobes from strong direct signal interference (DSI) and clutter [16]. Even if a directional antenna is used, there is still a significant signal power of the direct signal present in the surveillance channel. In the cross-ambiguity function, this manifests as a peak at zero Doppler and distance. The radio-waves can also reflect on terrain and building, creating unwanted peaks around zero Doppler-shift called clutter. Elements in the terrain, like trees moving in the wind, can also cause clutter at small Doppler-shifts. Both of the unwanted interference types are shown as dashed lines in Figure 2.1. In order to get information from the range-Doppler matrix, the interference first has to be suppressed before calculating the matrix elements.



(a) Unfiltered range-Doppler matrix (b) Range-Doppler matrix where DSI and clutter is removed using ECA-B

Figure 2.3: Illustration of a range-Doppler matrix. In (a) the correlation between the signal from the reference antenna is seen as yellow pixels around low bistatic range and velocity. The targets are also masked by the sidelobes from the DSI and clutter. In (b) the echoes from targets are visible as bright peaks in the image. The dark area that is the cancellation notch is also visible.

DSI and clutter is modeled as components of $\omega(t)$ from (2.6):

$$\omega(t) = x_{DSI}(t) + w(t) \Rightarrow x_e(t) = \sum_{i \in I_{\text{targets}}} x_i(t) + x_{DSI}(t) + w(t), \quad (2.9)$$

where $x_{DSI}(t)$ is both clutter and DSI while $w(t)$ is the remaining unknown noise. Algorithms for suppressing the DSI and clutter, i.e. removing the $x_{DSI}(t)$ term from $x_e(t)$, are described in chapter 3. Exactly how each algorithm models $x_{DSI}(t)$ is described in their corresponding section. In all cases, the resulting filtered signal from the surveillance antenna is denoted $x_{ef}(t)$.

Figure 2.3 illustrates how DSI and clutter manifests in the range-Doppler matrix and how they mask targets. Note that there are effects from the DSI and clutter, visible as brighter pixels, at bistatic velocities between -200 m/s and 200 m/s for short bistatic ranges. These effects are caused by properties of the FM-modulated signals and do not correspond to any Doppler shifted echos. Figure 2.3 also shows a cancellation notch that is created when a DSI and clutter removing algorithm is applied.

2.5 Target detection

To determine when a peak is a target or not, a common strategy is to use a constant false alarm-rate (CFAR) detector.

2.5.1 Constant false alarm-rate detector

The purpose of the detector is to determine whether a target echo is present or not in each element in the range-Doppler matrix. This can be expressed with two hypotheses:

$$\begin{aligned} H_0 : x_{ef} &= w, \\ H_1 : x_{ef} &= x + w, \end{aligned} \quad (2.10)$$

where x_{ef} is the filtered signal from the surveillance antenna, x is a target echo and w is noise. In other word, the two possibilities are that the signal only contains noise or that the signal contains an echo and noise.

For hypotheses H_0 the range-Doppler equation (2.8), where x_e is replaced with x_{ef} , becomes:

$$\psi_0[m, k] = \sum_{n=0}^{N-1} w[n] x_r^*[n - m] e^{-j \frac{2\pi}{N} kn}, \quad (2.11)$$

which can be modeled as the sum of random values. Since the sum is done over many samples, N , the central limit theorem states that the resulting sum can be seen as a realization of a complex stochastic variable where the real and imaginary parts are Gaussian distributed.

For hypotheses H_1 , the same range-Doppler equation, using a time discrete version of the model for a target echo (2.5), becomes:

$$\begin{aligned} \psi_1[m, k] &= \sum_{n=0}^{N-1} (C x_r[n - m] e^{j \frac{2\pi}{N} kn} + w[n]) x_r^*[n - m] e^{-j \frac{2\pi}{N} kn} \\ &= C \sum_{n=0}^{N-1} |x_r[n - m]|^2 + \psi_0[m, k], \end{aligned} \quad (2.12)$$

which is the sum of a complex constant, related to the amplitude of the echo and the power of the reference, and the same expression as in (2.11). The problem of detecting a target echo can therefore be seen as the problem of detecting a complex constant in complex Gaussian distributed noise [16].

The CFAR detector marks a detection when the absolute value of the element in the range-Doppler matrix is above a threshold. The threshold is chosen to make the probability of a false alarm, P_{FA} , constant. In other words, the threshold is set to make the probability of detection from only noise constant. Mathematically,

this means the threshold, D , is given by:

$$P_{FA} = \int_D^\infty f_{|\psi_0|}(z) dz, \quad (2.13)$$

where $f_{|\psi_0|}(z)$ is the probability density function for the value of $|\psi_0[m, k]|$. Since $\psi_0[m, k]$ tends to a complex Gaussian distribution, $|\psi_0[m, k]|$ will tend to a Rayleigh distribution. Similarly, $|\psi_1[m, k]|$ will tend to a Rice distribution. Using the absolute value of the element in the range-Doppler matrix results in an incoherent, linear-law detector. An alternative is to use the squared value instead, resulting in an incoherent, square-law detector, with different distributions for the signal and noise. However, only the linear-law is considered in this thesis. The threshold from (2.13) is therefore given by [16]:

$$D = \sigma \sqrt{-2 \ln P_{FA}}, \quad (2.14)$$

where σ is a parameter of the Rayleigh distribution of the noise. A detection is therefore made if:

$$|\psi[m, k]| > \sigma \sqrt{-2 \ln P_{FA}} = \sigma \alpha_D, \quad (2.15)$$

where α_D is a constant. In other words, a detection is made if the absolute value of an element in the range-Doppler matrix is more than a threshold equal to the level of the noise floor multiplied with a constant.

Since the properties of the noise are not known, they have to be estimated. A common way to estimate the level of the noise floor is to use the average of the absolute value of surrounding elements in a reference window, see Figure 2.4. However, as a peak from a target echo commonly affects more than one matrix element, the elements closest to the current element being tested has to be discarded. These ignored elements are referred to as guard elements or guard cells and makes up the guard window. The final estimate of the noise floor level around an element at range index m and Doppler index k , is given by:

$$\begin{aligned} \hat{\sigma}[m, k] = & \frac{1}{(2M_w + 1)(2K_w + 1)} \sum_{m_i=m-M_w}^{m+M_w} \sum_{k_i=k-K_w}^{k+K_w} |\psi[m - m_i, k - k_i]| \\ & - \frac{1}{(2M_g + 1)(2K_g + 1)} \sum_{m_i=m-M_g}^{m+M_g} \sum_{k_i=k-K_g}^{k+K_g} |\psi[m - m_i, k - k_i]|, \end{aligned} \quad (2.16)$$

where M_w and K_w are the size of the reference window in the Doppler- and range domain respectively, similarly M_g and K_g are the size of the guard window. The resulting detector is referred to as the cell averaging CFAR (CA-CFAR) detector [16].

There exists other ways to estimate the noise floor, such as using the median of surrounding elements or splitting the surround cells into groups and using the minimum or maximum of the averages of the groups [16]. However, only cell

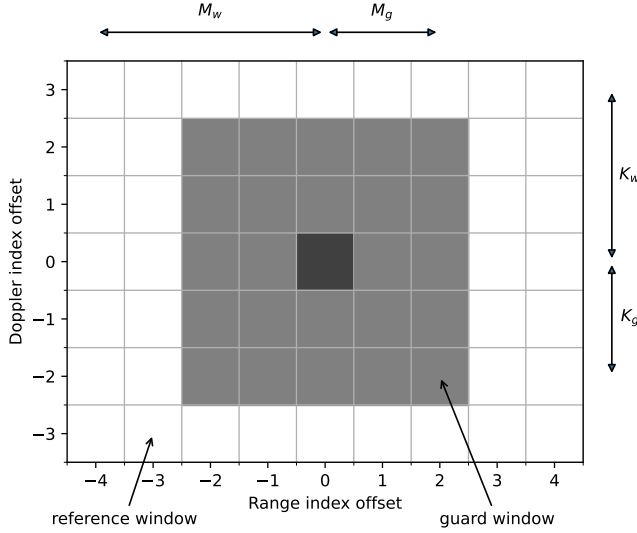


Figure 2.4: Illustration of the reference elements (white) used to estimate the noise floor. The elements closest to the tested element are guard elements (gray) and are not used for the estimate. Here the following parameters values are illustrated: $M_w = 4$, $M_g = 2$, $K_w = 3$ and $K_g = 2$.

averaging is used in this thesis.

Note that while the estimate $\hat{\sigma}$ describes the power of the noise floor, it is not necessarily a good estimate of σ , hence the exact value of the threshold factor, $\tilde{\alpha}_D$ is seen more as a design parameter rather than something that can be calculated directly from P_{FA} . The actual range-Doppler matrix also contain other sources of interference than noise, e.g. other target echoes and effects from the DSI suppression, that $\tilde{\alpha}_D$ needs to be tuned for.

Detections with the CA-CFAR detector are therefore made if:

$$|\psi[m, k]| > \hat{\sigma}[m, k] \tilde{\alpha}_D, \quad (2.17)$$

where $\tilde{\alpha}_D$ is a design parameter controlling the false alarm rate of the detector. A higher $\tilde{\alpha}_D$ result in fewer false alarms, but also fewer correct detections.

2.5.2 Peak to noise floor ratio

From (2.17) it can be seen that a constant false alarm rate detection can be rewritten as a comparison of the ratio of the element in the range-Doppler matrix and the level of the noise floor, with a constant, i.e.:

$$\frac{|\psi[m, k]|}{\hat{\sigma}[m, k]} > \tilde{\alpha}_D. \quad (2.18)$$

With the estimate, $\hat{\sigma}[m, k]$, relating to the power of the surrounding noise floor in the range-Doppler matrix, the ratio $\frac{|\psi[m, k]|}{\hat{\sigma}[m, k]}$ is denoted the peak to noise floor ratio (PNFR) in this thesis. The PNFR is a measure that is used throughout the thesis for measuring detection performance. A higher PNFR for a target means that it could be detected with a lower probability of false alarm detections.

3

DSI suppression algorithms

This chapter presents relevant algorithms for suppression of the direct signal and clutter described in section 2.4. Implementations of the algorithms are used in later chapters to find which ones are feasible for a low-cost passive bistatic radar system. Two main categories of algorithms are described:

- Extensive cancellation algorithm (ECA).
- Least mean square (LMS) based adaptive filters.

The category of the ECA variants are block processing algorithms with a flexible model that allows more freedom in choosing which echoes are to be considered as unwanted DSI and clutter. They are investigated in this thesis as they prioritize DSI suppression performance over computational load. The intent is that they can describe how a low-cost receiver can perform if paired with more capable processing hardware.

On the other end of the spectrum is the LMS based algorithms. LMS belongs to a group of iterative algorithms with a DSI and clutter model with less freedom compared to ECA. The algorithm have a low computational complexity at a cost of a slow rate of convergence, representing an alternative that would allow the use of lower-cost processing hardware for a passive radar system

If the results for ECA and LMS are understood, then it should be possible to predict the performance of other categories based on what properties they share.

3.1 Extensive cancellation algorithm

The first method presented here is ECA [5]. The method assumes the received signal in the surveillance channel can be described by the model:

$$x_e[n] = \sum_{m=0}^{M-1} \sum_{k=-K_m}^{K_m-1} C_{m,k} x_r[n-m] \cdot \exp\left(j \frac{2\pi}{N} kn\right) + w[n], \quad (3.1)$$

where x_r is the reference signal, M and K are arbitrary large constants representing the maximum number of sample delays and frequency bins of Doppler shifts of the reference signal, $C_{m,k}$ is a constant amplitude and w is noise. In other words, the surveillance signal is a sum of delayed and frequency shifted copies of the reference signal with some additive noise.

Using the knowledge of DSI and clutter from section 2.4 it is reasonable to assume that most unwanted interference (the direct signal and clutter) consist of time delayed signals with small to no shifts in frequency. That is, interference consists of the terms in (3.1) where the absolute value of index k is small. The exact indexes that are considered as unwanted becomes parameters that can be changed to tune the ECA filter or determined by iteratively calculating the range-Doppler matrix and finding uninteresting peaks. By denoting the set of unwanted delays as \mathbf{M}_{DSI} and the set of unwanted frequency bins of Doppler shift for the sample delay m as $\mathbf{K}_{DSI}(m)$, the unwanted influence of the DSI and clutter can be described by:

$$x_{DSI}[n] = \sum_{m \in \mathbf{M}_{DSI}} \sum_{k \in \mathbf{K}_{DSI}(m)} C_{m,k} x_r[n-m] \cdot \exp\left(j \frac{2\pi}{N} kn\right). \quad (3.2)$$

The filtered signal is then the remaining part of the signal, that is $x_{ef}[n] = x_e[n] - x_{DSI}[n]$. Finding the targets in $x_{ef}[n]$ is done by calculating the range-Doppler matrix, see section 2.3.

Equation (3.2) can also be written on matrix form:

$$\mathbf{X}_{DSI} = \mathbf{X}_r \mathbf{C}, \quad (3.3)$$

where $\mathbf{X}_{DSI} = [x_{DSI}[n] \ x_{DSI}[n+1] \ \dots \ x_{DSI}[n+N-1]]^T$, \mathbf{X}_r is a matrix where the columns are time delayed and frequency shifted copies of the reference signal, i.e.:

$$\text{column of } \mathbf{X}_r = \begin{bmatrix} x_r[n-m] \exp\left(j \frac{2\pi}{N} kn\right) \\ x_r[n+1-m] \exp\left(j \frac{2\pi}{N} k(n+1)\right) \\ \vdots \\ x_r[n+N-1-m] \exp\left(j \frac{2\pi}{N} k(n+N-1)\right) \end{bmatrix}, \quad (3.4)$$

and \mathbf{C} is a column matrix that contains the amplitude constants in the same order as the corresponding columns in \mathbf{X}_r . When creating the \mathbf{X}_r matrix, any missing value from $x_r[n]$, i.e. before the start and after the end, is replaced with a zero.

The problem of removing DSI and clutter can then be expressed as the least

squares (LS) minimization problem:

$$\min_{\mathbf{C}} \|\mathbf{X}_e - \mathbf{X}_r \mathbf{C}\|_2^2, \quad (3.5)$$

where $\mathbf{X}_e = \begin{bmatrix} x_e[n] & x_e[n+1] & \dots & x_e[n+N-1] \end{bmatrix}^T$. The value of \mathbf{C} that solves the problem (3.5) is given by:

$$\hat{\mathbf{C}} = (\mathbf{X}_r^H \mathbf{X}_r)^{-1} \mathbf{X}_r^H \mathbf{X}_e, \quad (3.6)$$

where H indicates the Hermitian matrix. Meaning that the output of the filter, $\mathbf{X}_{ef} = \begin{bmatrix} x_{ef}[n] & x_{ef}[n+1] & \dots & x_{ef}[n+N-1] \end{bmatrix}^T$, is given by :

$$\mathbf{X}_{ef} = \mathbf{X}_e - \mathbf{X}_r \hat{\mathbf{C}} = \mathbf{X}_e - \mathbf{X}_r (\mathbf{X}_r^H \mathbf{X}_r)^{-1} \mathbf{X}_r^H \mathbf{X}_e = \left(\mathbf{I} - \mathbf{X}_r (\mathbf{X}_r^H \mathbf{X}_r)^{-1} \mathbf{X}_r^H \right) \mathbf{X}_e. \quad (3.7)$$

The filtered signal, \mathbf{X}_{ef} , is the projection of the surveillance signal on the subspace that is orthogonal to the disturbance subspace (created by the columns in \mathbf{X}_r) [5].

3.1.1 Properties of ECA solution

From (3.7) it is possible to see that the projection matrix does not depend on the surveillance signal. When using multiple surveillance channels, the projection matrix only need to be calculated once. It is even possible to extend the matrix \mathbf{X}_e with more columns containing different surveillance channels and the resulting \mathbf{X}_{ef} matrix will become a column matrix where each column contains the respective surveillance signal where the interference has been suppressed. This allows for faster calculations for systems with multiple surveillance antennas.

Furthermore, it is noted that the calculation of matrix inverse is computationally intensive and can result in numerical problems. A more numerically stable approach is therefore to rewrite the equation containing the matrix inversion to a linear equation system and use LU-decomposition followed by forwards and backwards substitution to solve it. The expression in (3.7) can in this way be rewritten as:

$$\begin{aligned} \mathbf{X}_{ef} &= \mathbf{X}_e - \mathbf{X}_r \hat{\mathbf{C}}, \\ \text{where: } \mathbf{X}_r^H \mathbf{X}_r \hat{\mathbf{C}} &= \mathbf{X}_r^H \mathbf{X}_e. \end{aligned} \quad (3.8)$$

3.1.2 Batched version of ECA

For a long time window, like the 1 second commonly used for FM [5, 14], the fact that the clutter and radio wave characteristics, can change during the integration time can negatively impact the performance of ECA. To combat this, a batched version of ECA (ECA-B) is used [5]. ECA-B modifies the matrices \mathbf{X}_r and \mathbf{X}_e to only contain a batch of the rows used in the original matrices. The length of a batch is denoted as T_B and must fulfill $0 \leq T_B \leq T_{int}$ and $T_B \cdot B = T_{int}$ where B is the integer number of batches. For batch $b \in \{b \in \mathbb{Z} | 0 \leq b < B\}$ the modified

matrices therefore contains rows indexed (starting at index 0) $b \cdot \frac{N}{B}$ to $(b + 1) \cdot \frac{N}{B}$ of the original \mathbf{X}_e and \mathbf{X}_r . The LS problem is solved for each batch, and the results are then concatenated into a single filtered output. By splitting the time window into smaller batches, the algorithm can give a better filtering performance for a time-varying environment.

However, the smaller batches also cause the algorithm to remove more content in the Doppler dimension around the Doppler frequencies specified as DSI and clutter. In other words, it increases the Doppler width of the filter notch. The width of the notch is related to the batch length as $1/T_B$ Hz, translating to $\frac{\lambda}{2T_B}$ m/s bistatic velocity for carrier frequency with wavelength λ [5].

3.1.3 Sliding version of ECA

The ECA-B approach has been shown to have problems when used in an environment with highly time-varying interferences and targets that generate a low frequency shift in the Doppler domain [6]. The fact that neighboring batches are created using non-overlapping signal data can create unwanted effects in the regions where the batches change, more specifically it creates discontinuities in the filtered signal from the surveillance antenna. These periodic discontinuities can create periodic repetitions of the target echo along the Doppler dimension in the range-Doppler matrix, possibly generating false detections. The false peaks will repeat every $1/T_B$ Hz and are only noticeable when the true target echo is affected by the ECA-B filtering, in other words when it is inside the cancellation notch. A more thorough description of the problem can be found in [6].

To make the filter output smoother, a sliding version of ECA (ECA-S) can be used. This algorithm is similar to ECA-B but introduces a new parameter T_S , $T_S \geq T_B$ that specifies the length of the window used to calculate the amplitude constants in $\hat{\mathbf{C}}$. The output of one batch of ECA-S is therefore:

$$\begin{aligned} \bar{\mathbf{X}}_{ef, T_B} &= \bar{\mathbf{X}}_{e, T_B} - \bar{\mathbf{X}}_{r, T_B} \hat{\mathbf{C}}, \\ \text{where: } \bar{\mathbf{X}}_{r, T_S}^H \bar{\mathbf{X}}_{r, T_S} \hat{\mathbf{C}} &= \bar{\mathbf{X}}_{r, T_S}^H \bar{\mathbf{X}}_{e, T_S}, \end{aligned} \quad (3.9)$$

or

$$\bar{\mathbf{X}}_{ef, T_B} = \bar{\mathbf{X}}_{e, T_B} - \bar{\mathbf{X}}_{r, T_B} (\bar{\mathbf{X}}_{r, T_S}^H \bar{\mathbf{X}}_{r, T_S})^{-1} \bar{\mathbf{X}}_{r, T_S}^H \bar{\mathbf{X}}_{e, T_S}. \quad (3.10)$$

The bar over a matrix in (3.9) indicate that it is batched and the second index, T_B or T_S , indicate how many samples from x_r or x_e are used in the matrix columns. See Figure 3.1 for a graphical illustration of the matrices for a simple case.

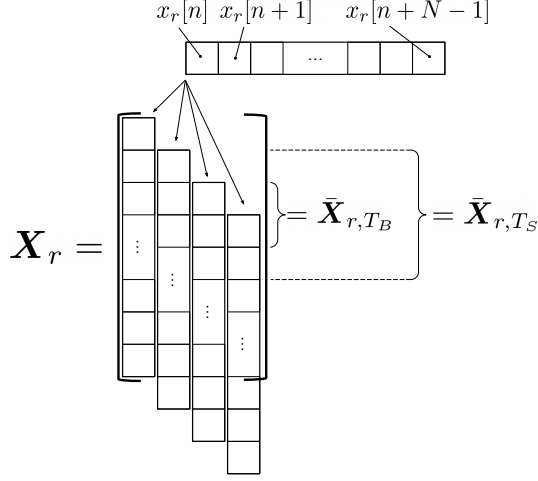


Figure 3.1: Graphical illustration of the matrices used in the ECA variants. The illustrated case is when $\mathbf{M}_{DSI} = \{0, 1, 2, 3\}$ and $\mathbf{K}_{DSI}(m) = 0$. The shown matrixes $\bar{\mathbf{X}}_{ef,T_B}$ and $\bar{\mathbf{X}}_{ef,T_S}$ represents the second batch with $T_B = 2T$ and $T_S = 4T$, where T is the sample time. The empty area in the matrix is filled with zeros.

3.2 Least mean square adaptive filters

Variants of the least mean square (LMS) adaptive filters can be used to remove DSI and clutter [4]. The idea is to continuously estimate the transfer function from the reference signal to the surveillance signal and use this to subtract the expected direct signal influence from the surveillance signal, see the block diagram in Figure 3.2.

The common factor for all LMS algorithms used in this thesis is that they use a finite impulse response FIR filter for modeling the DSI and clutter, i.e:

$$x_{DSI}[n] = \sum_{i=I_s}^I h_i^*[n] x_r[n-i], \quad (3.11)$$

where $h_i[n]$ is the FIR tap with index i at time represented by sample n , $*$ is the complex conjugate operator, I_s is the sample delay for the first FIR tap, I is the sample delay of the last FIR tap. As I corresponds to the maximal sample delay that is modeled as unwanted, it also controls how long the resulting cancellation notch reaches along the bistatic range domain the range-Doppler matrix. The

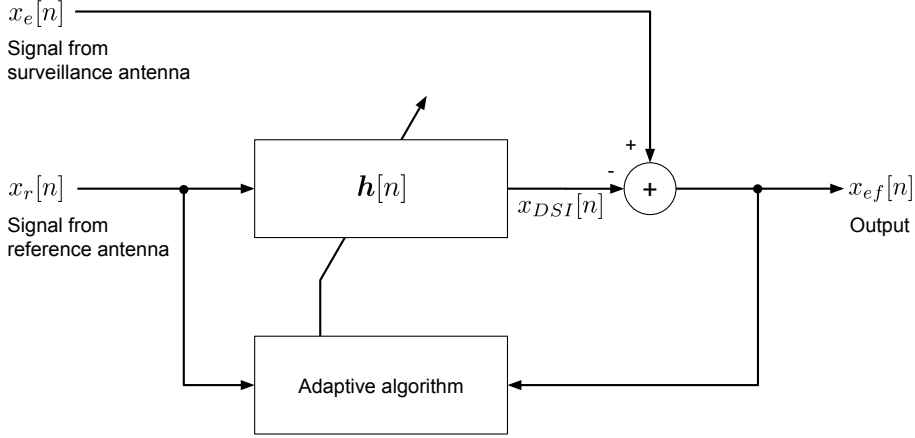


Figure 3.2: Block diagram of how an adaptive filter is used to remove DSI and clutter.

model (3.11) can be written on matrix form as:

$$x_{DSI}[n] = \mathbf{h}^H[n] \boldsymbol{\varphi}[n], \quad (3.12)$$

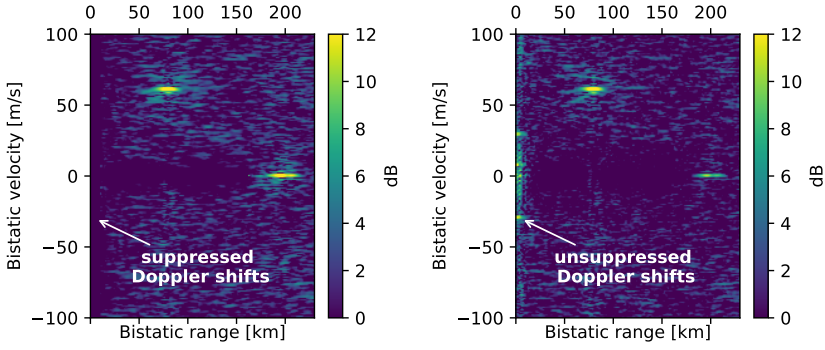
$$\text{where: } \mathbf{h}[n] := [h_{I_s}[n] \ h_{I_s+1}[n] \ \dots \ h_I[n]]^T$$

$$\boldsymbol{\varphi}[n] := [x_r[n - I_s] \ x_r[n - I_s - 1] \ \dots \ x_r[n - I]]^T.$$

The noticeable difference between the FIR model and the model (3.2) used by the ECA variants, is the lack of frequency shifted versions of the reference signal in the FIR model. See Figure 3.3 for an illustration of what the effects of not modeling, and therefore suppressing, Doppler shifts means in terms of the range-Doppler matrix. On the other hand, the FIR model, as it is written here, allows influence of future reference values (if I_s is negative) in the modeled DSI and clutter. This non-causal property was shown to be beneficial in [17], where the non-causal filter taps were attributed to allow interpolation for better modeling of clutter with time delays not evenly divisible with the sample time.

To motivate how non-causal filter taps help with modeling time delays, the scheme for creating an arbitrary time delay is described. The signal, sampled with sample time T , is first converted to continuous time using ideal reconstruction (pulse amplitude modulation) [18]:

$$x_r(t) = \sum_{i=-\infty}^{\infty} x_r[i] \text{sinc}\left(\frac{t - iT}{T}\right), \quad (3.13)$$



(a) Range-Doppler matrix where unwanted Doppler shifts are modeled

(b) Range-Doppler matrix where unwanted Doppler shifts are not modeled

Figure 3.3: Illustration of the difference between removing Doppler shifts at low bistatic range and not removing them. In (a) all content at low bistatic range is removed, while in (b) only content at zero bistatic velocity is removed. In (b) remains from the DSI can be seen as bright pixels at non-zero bistatic velocity.

where:

$$\text{sinc}(t) := \begin{cases} 1 & t = 0 \\ \frac{\sin(\pi t)}{\pi t} & \text{otherwise} \end{cases}. \quad (3.14)$$

The delayed signal, $x_{r,t_d}(t)$, with time delay t_d , is introduced:

$$x_{r,t_d}(t) := x_r(t - t_d) = \sum_{i=-\infty}^{\infty} x_r[i] \text{sinc}\left(\frac{t - t_d - iT}{T}\right). \quad (3.15)$$

The signal is then sampled again:

$$\begin{aligned} x_{r,t_d}[n] &:= x_{r,t_d}(nT) = \sum_{i=-\infty}^{\infty} x_r[i] \text{sinc}\left(\frac{nT - t_d - iT}{T}\right) \\ &= \sum_{i=-\infty}^{\infty} x_r[i] \text{sinc}\left(n - i - \frac{t_d}{T}\right). \end{aligned} \quad (3.16)$$

From (3.16) the operation of an arbitrary time delay can be realized by a discrete convolution of the signal and a time delayed sinc function. Since convolution is commutative, (3.16) can be written on a similar form to the model used by LMS (3.11):

$$x_{r,t_d}[n] = \sum_{i=-\infty}^{\infty} \text{sinc}\left(i - \frac{t_d}{T}\right) x_r[n - i]. \quad (3.17)$$

In other words, time delays can be approximated by letting the impulse response, $h_i[n]$ approximate a time delayed sinc function. For the special case when the time delay is evenly divisible with the sample time, the sinc gets sampled at its zero-crossings, except for when $i = t_d/T$, meaning only one filter is need to describe it. However, for all other cases an infinite number of non-zero filter taps are needed to sample the sinc, meaning the FIR model can only be an approximation. For those cases, the sinc function has non-zero values for both positive and negative times, providing motivation to why negative time delays in $h_i[n]$ can help approximate the sinc function and therefore allow better interpolation.

3.2.1 LMS and Normalized LMS

The LMS filter estimates the FIR coefficients, $\mathbf{h}[n]$, by minimizing the cost function:

$$J(\mathbf{h}[n]) = E \{ |e[n]|^2 \}, \quad (3.18)$$

$$\text{where: } e[n] := x_e[n] - \mathbf{h}^H[n] \boldsymbol{\varphi}[n].$$

Note that $E \{ \cdot \}$ denotes the expected value operation. The minimization is done iteratively using a stochastic gradient descent method, i.e. for each sample in the signal the current estimate of the FIR coefficients, $\hat{\mathbf{h}}[n]$, is adjusted according to the gradient of the cost function. Since the expected value of the cost function $J(\mathbf{h}[n])$ is unknown, it is approximated as:

$$J(\mathbf{h}[n]) \approx J_s(\mathbf{h}[n]) := |e[n]|^2 = e^*[n]e[n], \quad (3.19)$$

which means that the gradient of the estimated cost function is:

$$\nabla J_s(\mathbf{h}[n]) := \frac{\partial J_s(\mathbf{h}[n])}{\partial \mathbf{h}^H[n]} = 2\boldsymbol{\varphi}[n]e^*[n]. \quad (3.20)$$

The estimate of the FIR coefficients is thereby updated as:

$$\begin{aligned} \hat{\mathbf{h}}[n+1] &= \hat{\mathbf{h}}[n] - \frac{1}{2}\mu \nabla J_s(\hat{\mathbf{h}}[n]) \\ &= \hat{\mathbf{h}}[n] - \mu \boldsymbol{\varphi}[n]e^*[n], \end{aligned} \quad (3.21)$$

where μ is a small positive constant used to control the step size [19].

The step-size constant μ must be small enough for the LMS filter to be stable. Exactly how small depends on the maximum value of the correlation matrix for the input, $x_r[n]$ [19]. The power of the input also affect the magnitude of the gradient and therefore also how fast the algorithm converges to the optimal solution. To make the tuning of μ less dependent on power of the input, the step-size can be normalized, resulting in the Normalized LMS or NLMS. The normalized step-size is given by [19]:

$$\tilde{\mu}[n] = \frac{\mu}{\alpha + \boldsymbol{\varphi}^H[n]\boldsymbol{\varphi}[n]}, \quad (3.22)$$

where α is a small positive constant meant to prevent $\mu[n]$ from becoming too

large or undefined if the input, $\varphi[n]$, contains little or no energy. The value of $\tilde{\mu}[n]$ is then used instead of μ in (3.21).

Finally, the full NLMS algorithm for DSI and clutter suppression is described by the steps below:

1. Initialize $\hat{\mathbf{h}}[0]$ with zeros and set $n = 0$.
2. Filter sample:

$$x_{ef}[n] = x_e[n] - \hat{\mathbf{h}}^H[n]\varphi[n] \quad (3.23)$$

3. Update filter coefficients:

$$\hat{\mathbf{h}}[n+1] = \hat{\mathbf{h}}[n] - \frac{\mu}{\alpha + \varphi^H[n]\varphi[n]} \varphi[n]x_{ef}^*[n] \quad (3.24)$$

4. Increase n with 1 and go to step 2.

When all samples have been processed, $x_{ef}[n]$ contains the filtered surveillance signal. If a value of $x_r[n]$ or $x_e[n]$ is unknown for any sample, n , they are substituted with zeros, like in ECA.

As LMS need to be tuned differently depending on the input signal and has no significant benefits over NLMS, only NLMS will be evaluated in this study.

3.2.2 Block LMS

A variation of the LMS adaptive filter is the block LMS (FBLMS) adaptive filter [19]. BLMS filters multiple samples between each update of the filter coefficients. The number of samples filtered with the same filter is denoted the block size, L . The filter equation is the same as for LMS and NLMS, except $\hat{\mathbf{h}}[n]$ is constant for the entire block. The accumulated error for the block is then used when updating the filter coefficients, i.e. the update in (3.21) becomes [19]:

$$\hat{\mathbf{h}}[l+1] = \hat{\mathbf{h}}[l] - \mu_B \frac{1}{L} \sum_{i=0}^{L-1} \varphi[lL+i]e^*[lL+i], \quad (3.25)$$

where l is the index, starting at 0, of the block being processed and μ_B is the step size for BLMS. Note that in order for the BLMS algorithm to have the same rate of convergence as LMS the step size has to be increased as $\mu_B = L\mu$, since the number of steps taken is reduced. It should be noted that increasing the step size like this for BLMS have a higher risk of instability issues compared to LMS with the same rate of convergence [19].

3.2.3 Fast block normalized LMS

One of the prominent benefits of LMS and NLMS is that their simplicity gives them a low computational complexity. They are therefore of interesting for applications where execution time should be minimized. It can be seen that both the

filtering and update of filter coefficients in BLMS is similar to convolution, which can be done faster in the frequency domain. The fast block LMS algorithm is therefore introduced, where fast convolution is done using the fast Fourier Transform (FFT). The fast convolution is done using the overlap-save method, where two signals are transformed using the discrete Fourier transform (DFT), multiplied with each other and then transformed back [19].

According to [19] the fastest implementation of FBLMS uses a block size equal to the number of FIR filter taps, i.e. $L = I - I_s$, therefore this is the only case that will be considered.

The FBLMS algorithm suffers from the same difficulties in tuning μ as LMS, which is why the normalized step size, $\tilde{\mu}$, from (3.22) is used [4, 7]. The resulting algorithm is referred to as the fast block normalize LMS (FBNLMS) algorithm. Note that this algorithm never uses the matrix $\hat{\mathbf{h}}[n]$. It instead represents the filter coefficient with:

$$\hat{\mathbf{H}}[l] = \text{FFT} \left\{ \begin{bmatrix} \hat{\mathbf{h}}[lL] \\ \mathbf{0} \end{bmatrix} \right\}, \quad (3.26)$$

where $\mathbf{0}$ is a single column matrix with L zeros and $\text{FFT} \{ \cdot \}$ is the operation of applying the discrete Fourier transform (with a FFT algorithm). The appended zeros, $\mathbf{0}$, are needed to avoid circular convolution when calculating the convolution in the DFT domain.

The final FBNLMS algorithm is described by the steps below:

1. Initialize $\hat{\mathbf{H}}[0]$ with zeros and set $l = 0$.

2. Filter block l :

$$\boldsymbol{\varphi}_B[l] = \text{FFT} \left\{ \begin{bmatrix} x_r[lL - L] & x_r[lL - L + 1] & \dots & x_r[lL + L - 1] \end{bmatrix}^T \right\} \quad (3.27)$$

$$\mathbf{x}_{ef}[l] = \mathbf{x}_e[l] - (\text{last } L \text{ elements of } \text{IFFT} \{ \hat{\mathbf{H}}[l] \odot \boldsymbol{\varphi}_B[l] \}) \quad (3.28)$$

$$\text{where: } \mathbf{x}_e := \begin{bmatrix} x_e[lL] & x_e[lL + 1] & \dots & x_e[lL + L - 1] \end{bmatrix}^T,$$

$$\mathbf{x}_{ef} := \begin{bmatrix} x_{ef}[lL] & x_{ef}[lL + 1] & \dots & x_{ef}[lL + L - 1] \end{bmatrix}^T,$$

$\text{IFFT} \{ \cdot \}$ is the operation of applying the inverse DFT

and \odot denotes the element-wise product.

3. Update DFT of filter coefficients:

$$\boldsymbol{\phi}[l] = \text{first } L \text{ elements of } \text{IFFT} \left\{ \boldsymbol{\varphi}_B^*[l] \odot \text{FFT} \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x}_{ef}[l] \end{bmatrix} \right\} \right\} \quad (3.29)$$

$$\hat{\mathbf{H}}[l + 1] = \hat{\mathbf{H}}[l] + \frac{\mu}{\alpha + \boldsymbol{\varphi}^H[lL + L] \boldsymbol{\varphi}[lL + L]} \cdot \text{FFT} \left\{ \begin{bmatrix} \boldsymbol{\phi}[l] \\ \mathbf{0} \end{bmatrix} \right\} \quad (3.30)$$

4. Increase l with 1 and go to step 2.

When all blocks have been processed, $x_{ef}[n]$ contains the filtered surveillance signal. Like for NLMS, if a value of $x_r[n]$ or $x_e[n]$ is unknown, it is replaced with a zero.

FBNLMS should be faster while yielding the same results as block normalized LMS. Thus, only FBNLMS is tested in this thesis.

4

Hardware limitations and tests

This chapter presents the possible hardware limitations related to low-cost hardware and how they can be simulated. It also describes the test and evaluation metrics used to fulfill the thesis aim of determining the limitations' effects on the system's detection performance.

4.1 Hardware limitations

The effect of hardware limiting constrains is investigated by comparing the direct signal interference suppression measure when using input data from a high performing system. To isolate the effect of one limitation, the signal from the high performing system is modified to emulate the hardware constraint. The main limitations and ways to simulate them are:

- Limited computational power of the processor used for signal processing. This is analyzed by measuring the execution time of each algorithm.
- Additional noise in ADC readings. The design and placement of ADC in the circuit can cause each sample to contain a variation in the form of noise. Generated random noise is added to the digital signal to emulate this effect.
- Interference between channels. Antenna type and placement affects how much DSI is present in the surveillance channel. A cheaper directional antenna might for instance have large sidelobes that allow more of the direct path signal to leak into the surveillance channel. This effect is simulated by generating the signal from the surveillance antenna as the sum of the reference signal and a time and frequency shifted version of the reference signal, used as a target echo. The relative signal strength of the simulated

echo is varied to test different levels of DSI.

- Low-resolution ADC. To evaluate this effect, tests are run on signals where the bit-depth of the sampled signal is artificially lowered to different degrees. The simulated quantization with quantization step, Δ , is described (assuming no saturation) as [18]:

$$x_q[n] = \Delta \left\lfloor \frac{x[n]}{\Delta} + \frac{1}{2} \right\rfloor, \quad (4.1)$$

where $x[n]$ is the original signal, $x_q[n]$ is the resulting quantized signal and $\lfloor \cdot \rfloor$ is the operation of rounding down to the closest integer. For complex signals, the real and imaginary part is quantized separately. The quantization step is chosen as:

$$\Delta = \frac{2}{2^b} \max_n |\max \{ \Re(x[n]), \Im(x[n]) \}|, \quad (4.2)$$

where b is the wanted bit-depth, $\Re(x[n])$ and $\Im(x[n])$ are the real and imaginary parts of $x[n]$, respectively. It should be noted for further comparisons that quantization noise can often be modeled as white additive noise with intensity $\frac{\Delta^2}{12}$ [18], i.e. the variance of a uniform distribution with minimum and maximum $\mp \Delta/2$. When the quantization is done on both the real and imaginary part independently, the modeled additive noise has double the intensity, $\frac{\Delta^2}{6}$.

4.2 Test of DSI suppression algorithms

The following section describes the test used for evaluating the different DSI suppression algorithms.

4.2.1 Test metrics and purpose

The output of each algorithm, described in chapter 3, is used to create range-Doppler matrices, as described in section 2.3.

For the first evaluation metric, ground truth data consisting of latitude, longitude, altitude and velocity for a specific target is used to determine the location of the target peak in the range-Doppler domain. The generated range-Doppler matrix is searched for echoes from the target around the location given by the ground truth data. The target echo is considered to be the element with the maximum amplitude in a search area, centered on the ground truth location. The search area is a rectangle, 9 elements (approximately 16 km) in the bistatic range dimension and 15 elements (approximately 20 m/s) in the bistatic velocity dimension. The reason for the large search area is to make sure no target peak is ever missed due to low accuracy or missing ground truth data. The target is followed during a time interval where it, based on visual inspection, is seen in the collected data and the PNFR, described in section 2.5.2, is plotted as a function of time.

A single value quantitative measure of the performance with a certain algorithm is the number of detections made of the target with the CA-CFAR detector, presented in section 2.5, with a threshold factor, $\tilde{\alpha}_D$, of 7 dB and 9 dB. The 9 dB threshold factor was chosen as it gives next no false detections after visually inspecting the range-Doppler matrices. With a threshold factor of 7 dB, the number of correct detections is higher, but there are also a number of false detections. This measure gives an indication of how many measurements could be fed to a possible tracking filter. A plot of the number of detection as a function of the threshold factor is also used to give a better understanding of how the threshold factor affects the results.

A second measure estimated during this test is the execution time of each DSI suppression algorithm. The execution time is measured as the difference between the time when the algorithm is started and the time when it is finished. The distribution and variance of the execution time for each segment of DSI suppression can show how reliable the algorithm can be in a real-time scenario. As the tests are run on a computer with an operating system, there might be outliers in the processing time, caused by other processes temporary taking priority over the benchmarking process. To reduce the influence of those outliers, the median execution time (rather than the mean execution time) is used when comparing algorithms.

More qualitative evaluations of the algorithms are done by inspecting a selection of visualizations of the generated range-Doppler matrices with markers representing ground truth data of targets.

The sensitivity to the design parameter, the step-size μ , for the LMS based algorithm is also investigated. The sensitivity is evaluated in terms of suppression notch shape, mean PNFR and how fast the algorithms converges.

4.2.2 Test scenarios

To get a full understanding of the performance of the passive radar system and the difference between the DSI suppression algorithms, the test metrics are evaluated for a set of different scenarios.

The first scenario is the general one, where a target is visible for a long time. In this case, the result provide a general idea of how the system could be expected to perform. It also provides a baseline useful for comparing with other special scenarios.

The second scenario consists of a target with a low bistatic velocity, v_B , and is therefore close to the cancelation notch in the range-Doppler matrix. Since the shape of the cancelation notch is determined by the DSI suppression algorithms, the properties of the target close to the notch can indicate important differences between the algorithms. The notch may not only reduce the target peak, but also cause additional effects like false target peaks for some algorithms.

The final scenario examines a far away target. The echo from the target is weak

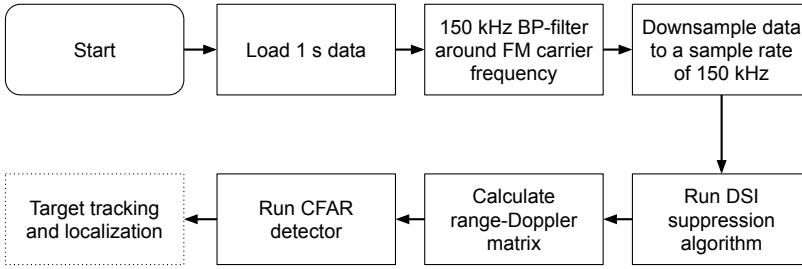


Figure 4.1: Flowchart of the indented data-processing scheme for the passive radar system. The final step of tracking and localization (in the Cartesian domain) is not considered in this thesis.

in this case, which put high requirements on the suppression algorithm to reduce the noise floor in order for the target to be detectable.

4.2.3 Test construction

The construction of the test is meant to be similar to the first steps of the data-processing scheme that would typically be used for a passive bistatic radar, see the flowchart in Figure 4.1. The scheme works with segments of data collected during a time interval (integration time) of approximately one second (the exact integration time depends on the system used). For each segment, the following steps are taken:

1. Load a block of data representing approximately one second.
2. Band-pass filter and downsample the signal to only contain a bandwidth of 150 kHz, centered on the carrier frequency of the FM-radio signal from a chosen transmitter. The resulting signal-vector contains approximately 150000 samples.
3. Run the DSI suppression algorithm on the signal-vector, $x_e[n]$, and measure execution time.
4. Calculate the range-Doppler matrix from the filtered signal-vector, $x_{ef}[n]$.
5. Save an illustration of the range-Doppler matrix with markers for possible targets given by the ground truth for visual inspection.
6. Run the CFAR detector on the range-Doppler matrix where the ground truth data shows the target should be and record the resulting PNFR.

Tests are run on data from both the low-cost system (KerberosSDR) and a higher performing system based on a PXIe-5644 transceiver. The recorded data from the PXIe-5644 consists of one second segments with a total bandwidth of 20 MHz

and an ADC resolution of 16-bits. It is assumed that the PXIe-5644 does not suffer from any of the hardware limitations described in section 4.1.

The low-cost system is set to record 1.024 seconds data blocks with a bandwidth of 2.048 MHz. Note that the current implementation of the interface with the KerberosSDR is limited to recording blocks containing a power of two number of samples, hence why 1.024 seconds data blocks are used instead of 1 seconds data blocks. The R820T2 tuner is also configured to dampen frequencies between 97.2 MHz and 97.6 MHz, corresponding to an unused weaker transmitter located in a water tower close to where the receiver is located. The filtering happens before sampling and therefore allows more of the ADC's dynamic range to be used for the transmitters used for passive bistatic radar. The system is configured and data collected using a modified version of the KerberosSDR demo software.

The algorithms are implemented in the programming language Python 3 and the package Numpy is used for faster linear algebra operations. The package Numba is also used to optimize the main loop in the NLMS algorithm. The computational times are recorded when the algorithms run on a laptop with an Intel Core i7-10750H CPU. Finally, the range-Doppler matrices are calculated with the help of the pyAPRiL package.

4.3 Test of impact of hardware limitation

To benchmark the performance impact of a specific hardware limitation, data from the high performing system is used and the effect of hardware limitations simulated as described in section 4.1. The data is then fed to the same data processing scheme as described in section 4.2.

4.4 Simulation of surveillance signal

In order to emulate the additional noise in ADC readings and interference between channels limitations, the noise, DSI and clutter levels in the surveillance signal $x_e[n]$ has to be controllable. This is done by discarding the collected signal from the surveillance antenna and instead simulating it as the sum of modeled DSI and clutter, a target echo and random noise. Specifically, the simulated surveillance signal, $\hat{x}_e[n]$, is calculated as:

$$\hat{x}_e[n] = \underbrace{c_{DSI} \sum_i h_{DSI}[i] x_r[n-i]}_{\text{DSI and clutter}} + \underbrace{x_r[n-m_t] \exp\left(j \frac{2\pi}{N} k_t n\right)}_{\text{target echo}} + \underbrace{c_w w[n]}_{\text{noise}}, \quad (4.3)$$

where $h_{DSI}[i]$ is a model for the DSI and clutter, m_t represents the bistatic range to the target, k_t represents the bistatic velocity of the target and $w[n]$ is random Gaussian noise. The constant c_{DSI} is changed to control the power of the simulated DSI and clutter content in the surveillance signal. Similarly, c_w is used to control the noise power. For the tests, a simple DSI and clutter model is used.

For index $i = 0$, $h_{DSI}[i]$ has an absolute value of 1, representing the strong direct path signal. The weaker clutter reflections are simulated as $h_{DSI}[i] = 2 \cdot 10^{-3}$ for index $i = 1, 2, \dots, 6$ and $h_{DSI}[i] = 1 \cdot 10^{-4}$ for index $i = 10, 11, \dots, 19$. For all other indexes, $h_{DSI}[i]$ is zero. This represents clutter content up to a bistatic range of approximately 38 km. It should be noted that this clutter model is limited as it assumes all clutter is static. It should, however, be good enough to demonstrate any effects from the intended hardware limitations.

5

Results

This chapter presents the parameters and data used for the tests described in chapter 4. The results from the tests and processing scheme developed during this thesis are also documented here.

The chapter's first section describes the choice of parameters used for all algorithms.

The chapter's second section presents general performance results with the low-cost KerberosSDR and then investigates a set of special cases (see section 4.2.2) to find which DSI suppression algorithms are feasible for the system. The section is concluded with a parameter sensitivity analyses done with data from KerberosSDR.

The third section presents results when using the high performing PXIe-5644 system. The final section then applies the simulated effects of hardware constraints and presents their effects.

5.1 Parameters for DSI suppression algorithms

The default parameters used for each algorithm is shown in Table 5.1. The parameters used for calculating the PNFR are: $M_w = 14$, $K_w = 4$, $M_g = 4$ and $K_g = 4$, see section 2.5.2 for an explanation of the parameters.

The choice of parameters is mostly based on the assumption of what is and is not clutter. For the ECA variants, all content at zero-Doppler shift, up to a time delay of 86 sample times or approximately 0.56 ms, is removed ($m = 0, 1, \dots, 86$ for $k = 0$). This represents a bistatic range of approximately 172 km. No reflections are expected from terrain beyond this range, as it is beyond the radio horizon. For the

Table 5.1: Default parameters used for DSI suppression algorithms. See chapter 3 for an explanation of the parameters.

Algorithm	Parameters
ECA	$M_{DSI} = \{m \in \mathbb{Z} 0 \leq m \leq 86\},$ $K_{DSI}(m) = \begin{cases} \{-200\frac{2f_c}{c}, -190\frac{2f_c}{c}, \dots, 200\frac{2f_c}{c}\}, & m \leq 5 \\ \{0\}, & \text{otherwise} \end{cases}$ where f_c is the center frequency and c the speed of light
ECA-B	$M_{DSI}, K_{DSI}(m)$ same as ECA, $T_B = 0.05$ s
ECA-S	$M_{DSI}, K_{DSI}(m)$ same as ECA, $T_B = 0.05$ s, $T_S = 0.2$ s
NLMS	Number of FIR taps = 128 ($I_s = -10, I = 117$), $\mu = 0.01, \alpha = 10^{-10}$
FBNLMS	Number of FIR taps = 128 ($I_s = -10, I = 117$), $\mu = 0.01, \alpha = 10^{-10}$

LMS algorithms, a similar reasoning is used. However, the range is rounded up to make the number of filter coefficients a power of two, in order to allow better use of the FFT algorithm in FBNLMS. By rounding the range up, the cancelation notch will be longer, but should not affect detection performance unless the peak from the target is located in the notch (see section 5.2.2). The number of filter coefficients with a negative time delay, controlled by I_s , for LMS is chosen based on experimental results showing it giving decent performance.

It should be noted that there is nothing preventing the use of negative time delays in the ECA variants, i.e. letting M_{DSI} contain negative integers. However, since the literature used for this thesis does not mention this approach, and because the ECA variants may compensate for this imposed limitation with the additional freedom from the ability to suppress frequency-shifted content, it is not tested in this thesis.

The parameter α , used in the LMS algorithms, is set to an arbitrary positive non-zero value significantly lower than the average power of the reference and surveillance signals. Small variations of the parameters had no noticeable effect on the results.

As for what Doppler shifts the ECA variants removes, determined by the parameter $K_{DSI}(m)$, the intent is to suppress most content at low bistatic range where the effect of the DSI is visible at up to 200 m/s bistatic velocity in the range-Doppler matrix. These effects are pointed out in section 2.4.

The step-size, μ , for LMS is set to the value that gave the best performance in the parameter sensitivity analysis in section 5.2.4. The batch length, T_B , for ECA-B and ECA-S creates a similar width of the cancelation notch as the one achieved with NLMS with the selected μ . Finally, the width of the overlapping time windows, T_S , used in ECA-S is set to make the false peaks analyzed in section 5.2.2 not visible.

Table 5.2: Median computation time and number of detections (out of 282) for BE9L target with a 9 dB and 7 dB PNFR threshold factor. Data is recorded with KerberosSDR.

Algorithm	Detections (7 dB)	Detections (9 dB)	Computation time
ECA	178	86	0.99 s
ECA-B	205	119	1.3 s
ECA-S	205	129	4.7 s
FBNLMS	230	160	0.054 s
NLMS	232	162	0.44 s
None	13	1	≈ 0 s

5.2 DSI suppression for KerberosSDR

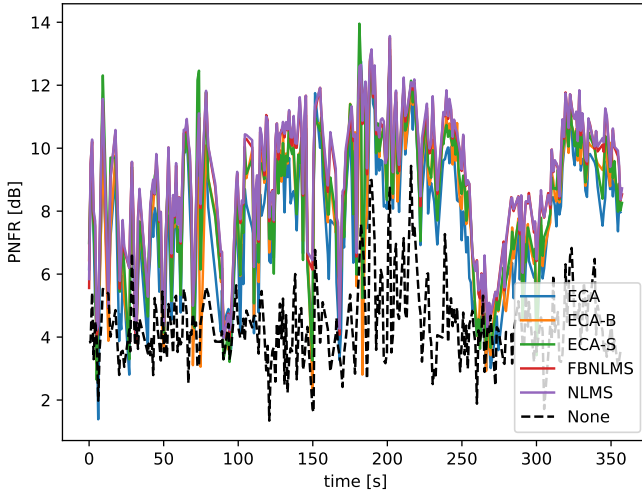
This section presents the inputs and results from the evaluation of suppression of DSI and clutter, described in section 4.2, when using data recorded by the low-cost system called KerberosSDR.

Here the IoO used was “Lillhärads masten” close to the Swedish city Västerås. The center frequency used was 98.0 MHz, corresponding to the radio channel “Sveriges Radio P3”. The chosen target was a propeller aircraft of the type Beech C90 King Air (type code BE9L). The target kept a bistatic distance of approximately 60 km during the recorded data of 356 seconds. The ground truth data was recorded using an ADS-B receiver.

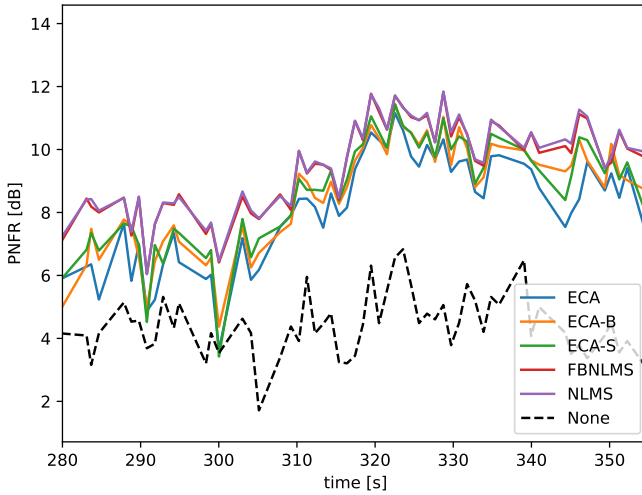
5.2.1 General results

A plot of the PNFR over time for different algorithms can be seen in Figure 5.1. The number of detections and execution time are presented in Table 5.2. The execution time is also presented as boxplots in Figure 5.2. For comparison, the results when not using any DSI suppression algorithm is also presented.

To make a more qualitative analysis, range-Doppler matrices, taken at the time 172 s, are plotted in Figure 5.4. The data at 172 s is used as it has a clear target. The target’s true bistatic range and velocity is also marked as a red ellipse in the images. Note that the amplitude is normalized so that the global noise-floor (estimated using median of matrix elements) is at 0 dB. To better illustrate the suppression notch around zero Doppler, a cross-section of some range-Doppler plots was also taken at the same time, see Figure 5.5.



(a) The complete recorded time.



(b) Time interval with clearer results.

Figure 5.1: PNFR over time for BE9L target recorded with KerberosSDR. In (a) the entire time interval between 0 s and 356 s is shown, while (b) only shows the PNFR around the time 320 s in order to more clearly see the results. FBNLMS and NLMS appear to give better results in both plots.

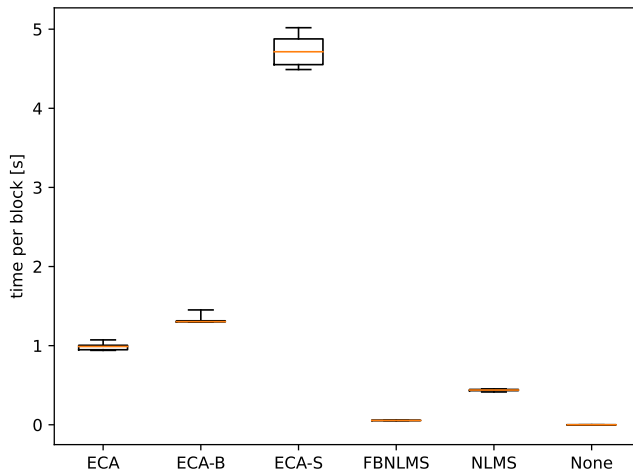


Figure 5.2: Boxplots showing the distribution of how long each algorithm takes at processing 153 600 samples of complex I/Q-data. Whiskers show 10-90% range of values. Note that the difference between ECA-B and ECA-S is influenced by the parameter choice for ECA-S.

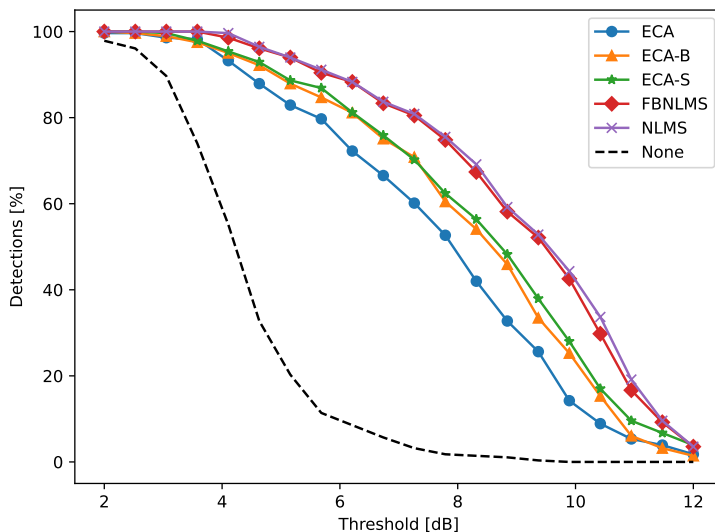


Figure 5.3: Percentage of seconds where the PNFR for the BE9L target is above the threshold factor. The unmodified ECA performs worse of the algorithms, while FBNLMS and NLMS show a clear advantage.

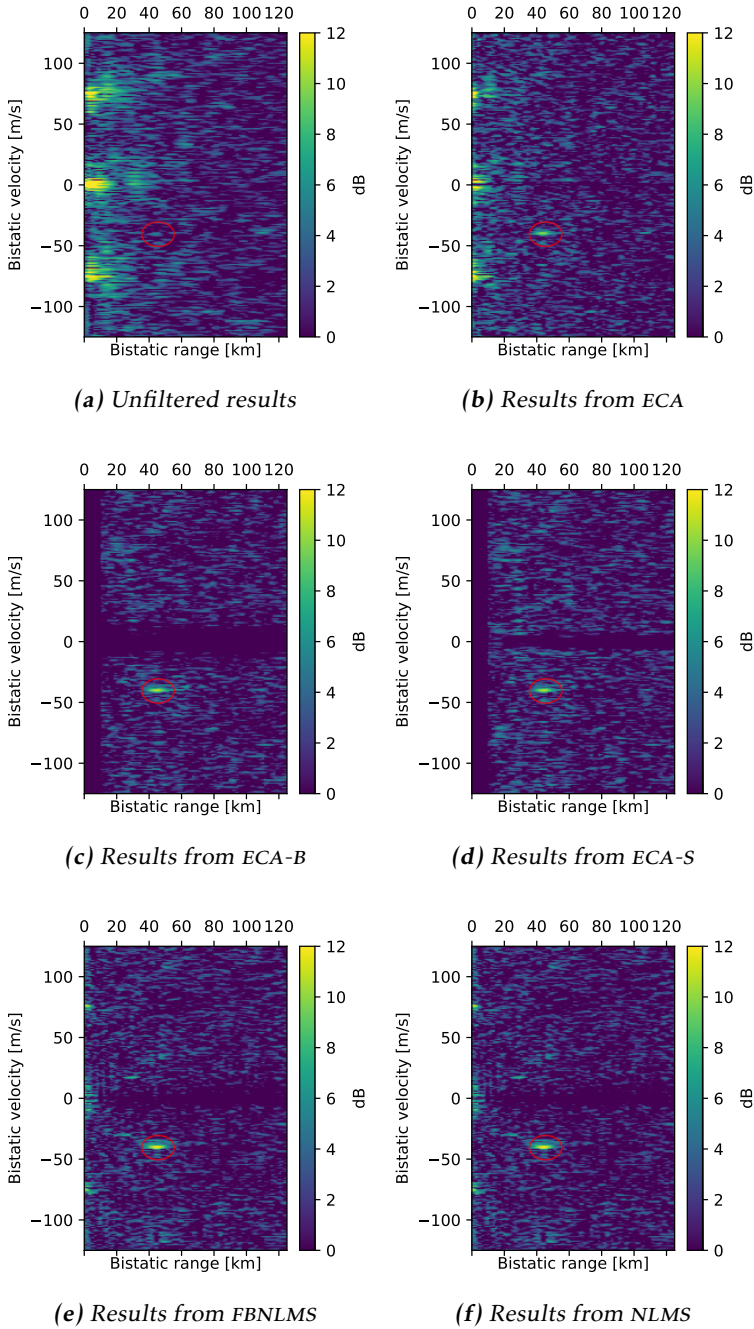


Figure 5.4: Range-Doppler matrix with a known target (BE9L) marked with a red ellipse in each image. The target peak is hard to detect in (a), but clearer in (b)–(f). Noise contributions at short range and certain bistatic velocities can be seen in (a), (b), (e) and (f), these have however been suppressed in (c) and (d).

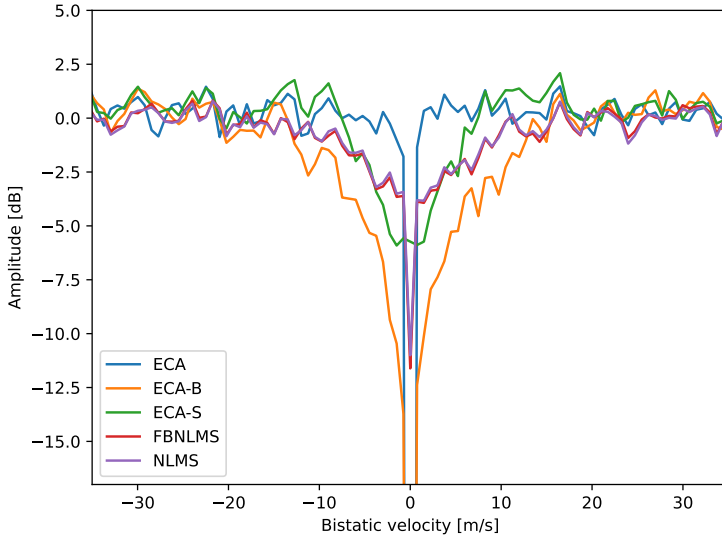


Figure 5.5: Cross-section of some Range-Doppler matrices from Figure 5.4. To get a clearer illustration of the notches, each line is the mean cross-section from 60 km to 160 km bistatic range. Note that the ECA-B and ECA notches continues down to -70 dB and -140 dB, respectively.

Discussion

From the plot of PNFR over time (Figure 5.1) it can be seen that NLMS and FBNLMS appear to have a higher PNFR than the other algorithms. It is also clear that all algorithms give better PNFR than when using no DSI suppression.

The execution time in Figure 5.2 show that the LU-decomposition approach of implementing ECA, ECA-B and ECA-S can yield varying execution times. As seen in Table 5.2, this implementation of the ECA variants is also slow compared to NLMS and FBNLMS.

The number of detection made as a function of the PNFR threshold factor in Figure 5.3 gives a better understanding of the overall performance of the algorithms. Here it is clear that unmodified ECA performs worse than the other four algorithms. The difference is approximately 10 percent units compared to ECA-B and ECA-S. The best performing algorithms are FBNLMS and NLMS. The reason for the better performance is likely connected to that the LMS algorithms allow filter taps with negative time delay, i.e. better approximation of clutter with a time delay not evenly divisible with the sample time, see section 3.2. Considering the sample-time is $T \approx 1/150000$, meaning that the waves can travel $T \cdot c \approx 2$ km during the sample time, it is not unreasonable that there exist such clutter.

The range-Doppler matrices in Figure 5.4 show qualitative differences between the algorithms. While all suppression algorithms results in similar peaks for

the target, ECA-B and ECA-S removes significantly more of the sidelobes at short bistatic range than the other algorithms. This is reasonable, as the model they use (see (3.1)) for DSI and clutter explicitly includes Doppler shifted versions of the reference at short time delays.

The NLMS and FBNLMS filter used here is limited to a FIR filter for modeling DSI and clutter. The FIR works for modeling simple time delay, as can be seen in Figure 5.4f where there is a clear notch at zero-Doppler. There is however no way of explicitly covering frequency shifts, as can be realized when considering that a FIR filter is a linear filter and linear filters cannot move signal energy to new frequencies. It is the update step in NLMS and FBNLMS that allows the removal of low frequency-shifts to be possible.

It should be noted that the strong peaks around ± 75 m/s at low range are not caused by a frequency shift of the direct signal, but instead due to properties of the FM-radio signal itself, meaning that modeling of large frequency shift may not be necessarily to fully remove the DSI and clutter. Nevertheless, while NLMS and FBNLMS reduces the peaks compared to the unfiltered results, they do not remove them completely in this test.

As for the unmodified version of ECA, while it does include the same frequency shifts as ECA-B and ECA-S, its notch of suppression only affects one frequency bin in the Doppler dimension, see Figure 5.5. This means that it will not remove the sidelobes unless every frequency bin is accounted for in the signal model (3.1), i.e. the parameter $\mathbf{K}_{DSI}(m) = \{k \in \mathbb{Z} | |k| \leq f_B\}$ for $m \leq 5$ where f_B is the maximum bin where sidelobes are visible.

A concern might be that the sidelobes left by NLMS, FBNLMS and ECA can result in false detections. For the data used here, this can be avoided by ignoring any detections at short bistatic range. This creates a blind spot for the radar but since ECA-B and ECA-S suppresses all signal content, both DSI and target echoes, at low bistatic range the resulting detection ability might not differ between the algorithms because of this specific limitation.

Since none of the algorithms can distinguish what causes correlation with the reference signal, be it DSI, clutter, the target echo or random noise, they will always create the notch seen in Figure 5.5. This means that if a target is in the notch, it will be suppressed. Similarly, if the signal from the surveillance antenna contains nothing but noise, there will still be a similar cancellation notch. Therefore, the shape and depth of the notch can not be directly associated with how much DSI and clutter is present in the signal.

5.2.2 Results for target with low bistatic velocity

One important special case for DSI and clutter suppression is when a target echo has close to zero Doppler-shift, i.e. a low bistatic velocity. The BE9L target echo moves from a negative to a positive bistatic velocity around the time 183 s. Therefore, a zoomed in plot of the PNFR over time is shown in Figure 5.6. Range-

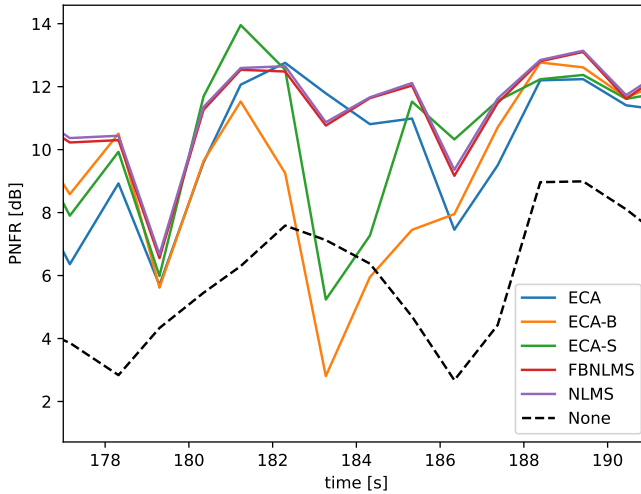


Figure 5.6: PNFR over time for BE9L target when passing zero-Doppler at approximately 183 s. It is evident that ECA-S and ECA-B performer worse than the other algorithms at 183 s. This may be explained by them having a wider and deeper cancellation notch around zero-Doppler, as can be seen in Figure 5.5.

Doppler matrices for the different algorithms at the time 183 s are illustrated in Figure 5.7.

Discussion

From the plot of PNFR over time in Figure 5.6, the effect of the wider and stronger cancellation notch of ECA-B and ECA-S becomes apparent as the echo from the target gets suppressed when passing through zero-Doppler. While both ECA-B and ECA-S loses PNFR around 183 s, ECA-S recovers quicker, confirming previous results that the smoothing makes the notch slightly narrower. Another important observation is that PNFR for ECA-S only goes down to approximately 5 dB while ECA-B falls all the way to 3 dB, this coincides with the plot in Figure 5.5 showing that the ECA-S notch is not as deep as the ECA-B notch.

The fact that normal ECA performs well in this case shows that the Doppler width of the clutter influence is small, and a narrow cancellation notch is not a real problem here. While ECA-S perform slightly better than ECA-B around zero-Doppler, the increase in performance comes with increased computational cost.

A second difference between ECA-B and ECA-S may however be more significant. As noted in section 3.1.2, and visible in the range-Doppler illustration in Figure 5.7a, ECA-B creates a number of false peaks when a true target echo

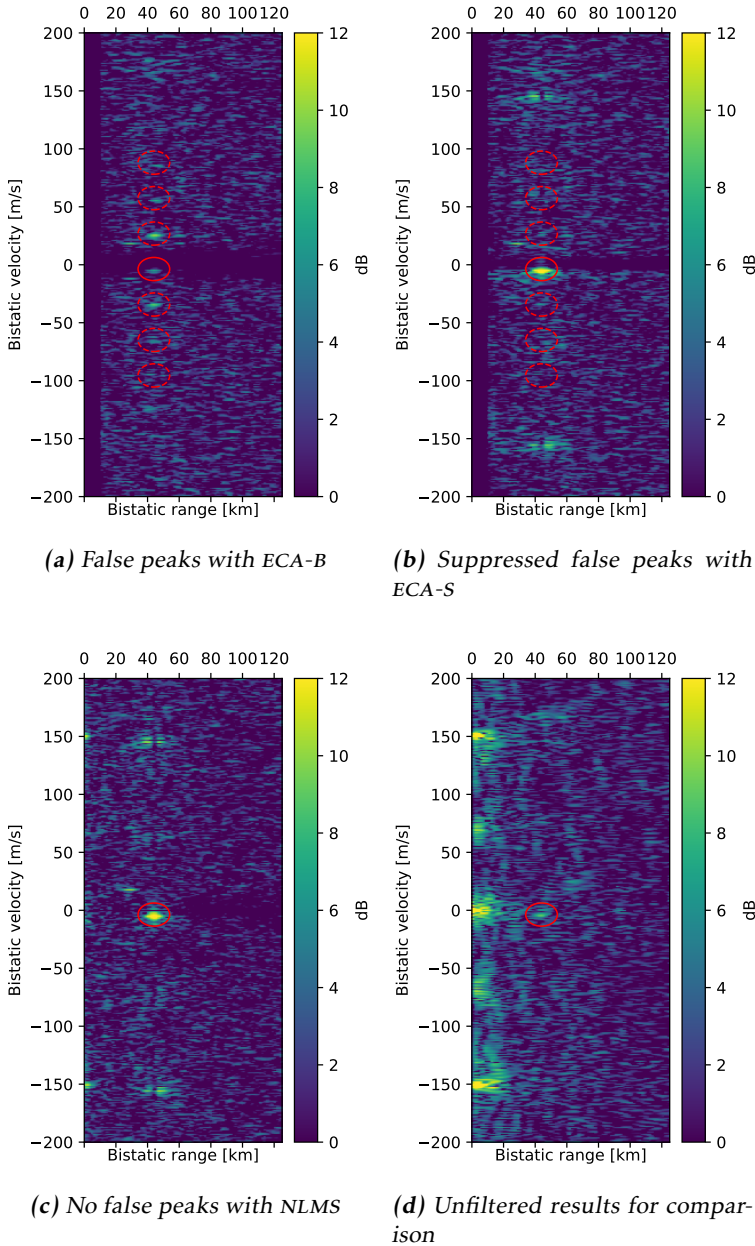


Figure 5.7: Range-Doppler matrix when the marked target (solid red ellipse) crosses zero-Doppler. False peaks are indicated with dashed red ellipses. The echo is strong at this time, making it visible in the unfiltered range-Doppler matrix.

is close to the cancellation notch. With the carrier frequency, $f_c = 98.0$ MHz, and $T_B = 0.05$ the expected spacing, v_s , in bistatic velocity of the false peaks is $v_s = \frac{\lambda}{2T_B} = \frac{c}{2f_c T_B} \approx 30.6$ m/s, where c is the speed of light. This expected spacing is indicated in Figure 5.7 using red ellipses with dashed lines. With ECA-B at least six false peaks are noticeable. The false peaks do not affect the PNFR of the target, but they can generate false detections. Based on the fact that the false peaks are only generated when a target is in the cancellation notch and that their placement and amplitude in the range-Doppler matrix is predictable, it could be possible to ignore them during detection stage. However, it is still desirable to not have them.

With ECA-S the amplitude of the false peaks are reduced to a level below the noise floor in Figure 5.7b. In this test, ECA-S is configured with the same length of data being updated at a time as ECA-B, specifically $T_S = T_B = 0.05$, therefore the spacing of any theoretical false peaks is also approximately 30.6 m/s for ECA-S, but as can be seen in Figure 5.7b the false peaks are not visible. As a side note, the target echo is strong at this time, seen by the fact that the true peak is visible in the unfiltered data. If the echo was weaker, less of the false peaks would have been visible above the noise floor.

Finally, it is noted that the NLMS algorithms does not suffer from the false peaks, as it does not work with batches. The blocks used in FBNLMS are not fully independent of each other and not big enough to generate any visible effect.

False peaks caused by properties of the FM-signal

Figure 5.7b and Figure 5.7c appears to have some repetitions of the target peak at ± 150 m/s. This is however related to ambiguities in the signal used as reference, noticeable as peaks around ± 150 m/s at zero bistatic range in the unfiltered range-Doppler matrix in Figure 5.7d. In fact, these repetitions visible at ± 150 m/s in Figure 5.7b and slightly in Figure 5.7c can be identified as false peaks by noting the notch that passes through them at a bistatic range equal to the true target's bistatic range. In other words, these false peaks in Figure 5.7 have a dark vertical line going through them.

This vertical notch can be explained by investigating the equation for the range-Doppler matrix, given knowledge of the FM-signal. Assuming that a single target echo is received by the surveillance antenna, the signal is:

$$x_e[n] = Cx_r[n - m_t]e^{j\frac{2\pi}{N}kn}, \quad (5.1)$$

where $x_r[r]$ is the reference signal, assumed to only consist of the transmitted FM-modulated signal, m_t relates to the target's bistatic-range and k_t to the target's bistatic velocity. A FM-modulated signal can be modeled by [18]:

$$x_r[n] = Ae^{jM[n]}, \quad (5.2)$$

where A is the arbitrary constant amplitude of the signal and $M[n]$ is the current phase of the modulated signal.

With (5.1) and (5.2) inserted into the equation for the range-Doppler matrix (2.8) the result is:

$$\begin{aligned}
 \psi[m, k] &= \sum_{n=0}^{N-1} x_e[n] \cdot x_r^*[n-m] e^{-j \frac{2\pi}{N} kn} \\
 &= \sum_{n=0}^{N-1} \left(C x_r[n-m_t] e^{j \frac{2\pi}{N} kn} \right) \cdot x_r^*[n-m] e^{-j \frac{2\pi}{N} kn} \\
 &= \sum_{n=0}^{N-1} C \left(A e^{jM[n-m_t]} e^{j \frac{2\pi}{N} k_t n} \right) A e^{-jM[n-m]} e^{-j \frac{2\pi}{N} kn} \\
 &= C A^2 \sum_{n=0}^{N-1} e^{j(M[n-m_t]-M[n-m])} e^{j \frac{2\pi}{N} (k_t-k)n}.
 \end{aligned} \tag{5.3}$$

For the elements in the matrix that has the same bistatic-range as the true target, $m_t = m$ is true, resulting in the simplified expression:

$$\psi[m_t, k] = C A^2 \sum_{n=0}^{N-1} e^{-j \frac{2\pi}{N} (k-k_t)n} = \begin{cases} C A^2 N, & \text{if } k_t - k = iN, i \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}. \tag{5.4}$$

Based on (5.4), for the bistatic-range of the true target, the only peak in the visible range-Doppler matrix is when $k_t = k$, i.e. at the true target's bistatic velocity. However, when $m_t \neq m$ the same can not be said. Indeed, as mentioned, Figure 5.7 shows that there can be other peaks not at the true target's bistatic velocity. It should be noted that there is no guarantee that the true bistatic range of the target will exactly line up with the integer value of m , but assuming that the true time contiguous phase of the FM-signal does not change too fast, it can work as an approximation.

5.2.3 Results for far away targets

All tests so far are done with a relative close target that is visible for a long period of time. To get a better view of overall performance, the PNFR for a second, less visible target at a longer bistatic distance away will now be investigated. The target is an Airbus A321-212 (type code A321), a larger passenger jet airliner moving at a bistatic distance of approximately 350 km. A different IoO was also used, "Orrbergenmasten" close to the city Norrköping (center frequency is 98.7 MHz, still corresponding to the channel "Sveriges Radio P3"). Data covering a time interval of approximately 36 s was used in this test.

The PNFR over time is plotted in Figure 5.8.

Discussion

From Figure 5.8, it is clear that ECA-B and ECA-S outperforms all other algorithms in this case. The common property of ECA-B and ECA-S is that they can fully

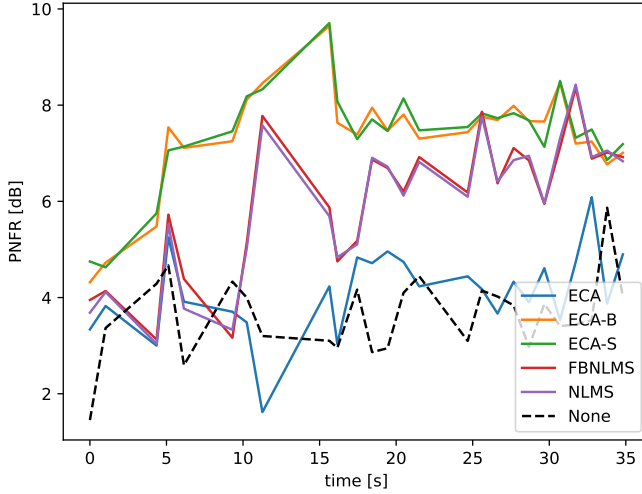


Figure 5.8: *PNFR over time for A321 target at a bistatic distance of 350 km. ECA-B and ECA-S is outperforming the other algorithms.*

remove the ambiguities in the Doppler dimension at low bistatic distances, see previous results in Figure 5.4 for an illustration of this.

Whether or not the ambiguities get suppressed depends on the width and spacing of the notches. As the notch width in Hz for ECA-B can be expressed as $1/T_B$, it is clear that the notches also have to be placed with a spacing of at most $1/T_B$ Hz, or $\lfloor T/T_B \rfloor$ frequency bins, for complete coverage. For ECA-S the notches have to be slightly denser as ECA-S does not create as wide notches. As the unmodified ECA has a narrow notch of only one frequency bin, notches have to be placed at every frequency bin along the Doppler dimension. While this is not impossible, it creates a high computational burden, which is why it was not tested.

5.2.4 Parameter sensitivity analysis

This section presents results aimed at evaluating how sensitive the previous detection performances are with respect to the design parameters. As the behavior, i.e. effects on noise floor, cancellation notch width and placement, for the ECA variants is well described by the theory in section 3.1 no further tests are performed on them in this thesis.

The NLMS and FBNLMS adaptive filters are both controlled by two parameters, the step size, μ , and the number of filter taps. While the number of filter taps correspond to how long the cancellation notch reaches in the bistatic range domain, the exact relation between the parameter μ and the resulting PNFR and cancellation notch width is not known and is analyzed here. Data for the BE9L target is

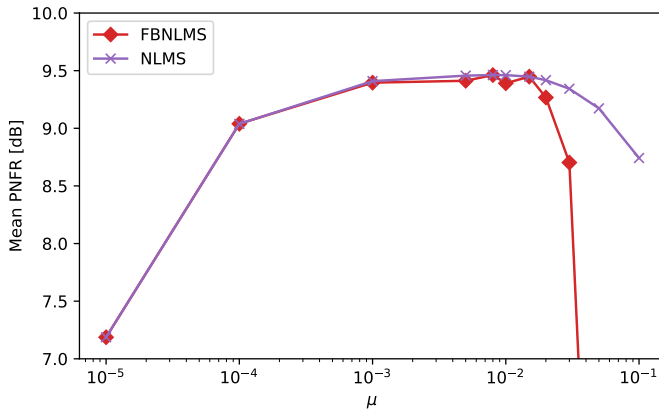


Figure 5.9: Mean PNFR for BE9L target with varying value of the step-size μ . Both algorithms perform best around $\mu = 0.01$. FBNLMS loses PNFR fast with μ higher than 0.015.

again studied between the times 295 s and 356 s as the target is able to generate a consistently good PNFR there. The parameter μ is varied for both FBNLMS and NLMS, and the resulting mean PNFR for the time interval is plotted in Figure 5.9. Note that the FBNLMS algorithm failed for $\mu \geq 0.1$ due to numerical problems. A cross-section of the cancellation notch at time 172 s can be seen in Figure 5.10 and the signal power, smoothed with a median filter of order 299, plotted as a function of time for the same second of data is shown in Figure 5.11.

Discussion

Based on Figure 5.9 both NLMS and FBNLMS appears to have similar performance for small μ . The difference happens after $\mu = 0.015$, where the mean PNFR for FBNLMS plummets due to instability problems.

As for the cancellation notches in Figure 5.10, other than the much wider notch for FBNLMS with $\mu = 0.05$ (again attributed to instability problems), the difference between the algorithms is not noticeable. A higher value of μ creates a wider and deeper notch. Showing that, as the filter is able to adapt faster, it is able to remove more of the signal power from the filtered signal. This becomes apparent in Figure 5.11 where the signal power of the filtered signal is clearly lower for higher μ . However, if the step-size μ is too high, the filter can overshoot the correct solution and create worse or unstable results.

Figure 5.11 shows that the filter has to re-adapt at multiple times, seen as the signal power going up after being low for a while, e.g. at 0.3 s. The higher values of μ allows for faster change of the filter coefficients. Once again, FBNLMS appears to have problems with $\mu = 0.05$, as the power peaks around 0.3 s and 0.93 s.

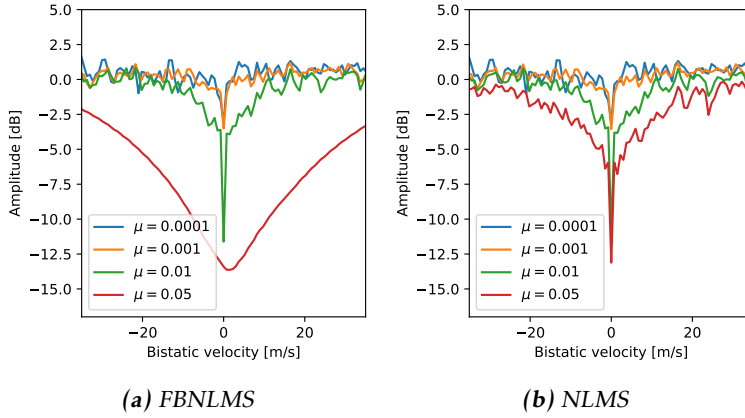
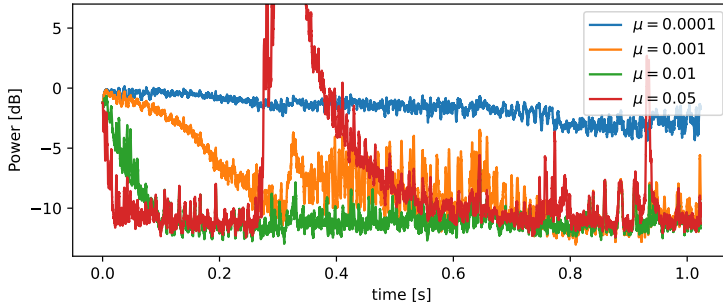
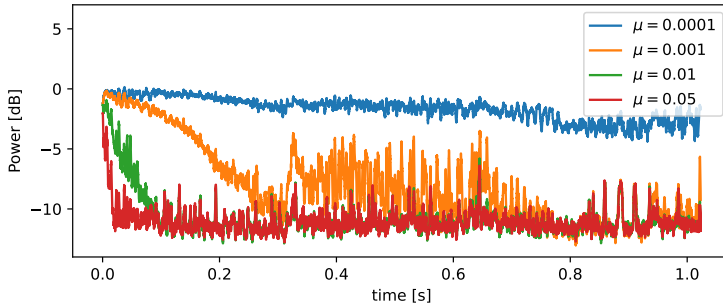


Figure 5.10: Cross-section of cancellation notch. A higher value of μ gives a wider and deeper notch. The wide notch for $\mu = 0.05$ in (a) is attributed to instabilities in the FBNLMS algorithm.

Another observation is that the both algorithms with $\mu = 0.01$ can converge to their best solution in approximately 0.1 s, without any prior data about the FIR coefficients. The total signal power is only reduced by approximately 12 dB in this case, so convergence might take longer if there is more DSI and clutter that can be removed.



(a) FBNLMS



(b) NLMS

Figure 5.11: The power of the filtered signal, relative to the average power of the unfiltered signal. Lower power corresponds to more DSI and clutter removed. A higher value of the step-size μ increases how fast the algorithms adapt to clutter (how fast the power is reduced). Signs of instability can be seen for $\mu = 0.05$ in (a) as the power unexpectedly increases by 60 dB at time 0.3 s.

5.3 DSI suppression for PXIe-5644

To get the baseline for performance, the suppression algorithms are tested on the high performing system using the PXIe-5644. As the data was collected at a different time, a different target and IoO was used in this test. The IoO was “Billingenmasten” in Skövde. The center frequency was 97.5 MHz, again corresponding to “Sveriges Radio P3”. This time, a cooperative target with its own measurement of location and velocity was used. The target was a North American Sabreliner. The target kept a bistatic range of approximately 70 km during the 100 s recording. The resulting PNFR over time is plotted in Figure 5.12.

Discussion

This time, all algorithms are able to suppress the DSI and clutter. Overall, the performance is a lot more stable for this system compared to KerberosSDR, possibly due to less influence from random noise and quantization distortion. It is still noted that the PNFR sometimes drops considerable, e.g. around 39 s. As these drop in PNFR happens for this high performance system, it is unlikely to be related to any hardware constraints. The exact reason for the drop is not investigated in this thesis. However, a possible explanation is that the FM-radio signal temporary has such a low bandwidth that there is no range resolution, see section 2.2.

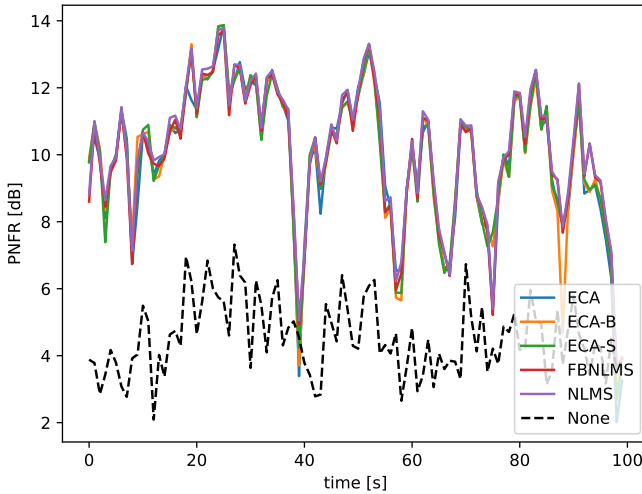


Figure 5.12: *PNFR over time for the North American Sabreliner target recorded with the high performing system PXIe-5644. All algorithms have a similar suppression performance.*

5.4 Impact of hardware limitation results

This section describes the results when testing how the algorithms from chapter 3 performs under different hardware limitations. The same data from the test in section 5.3 was used.

5.4.1 Quantization

This test aims at investigating the effects of quantization on the PNFR. Data from the PXIe-5644 system taken at time 24 s is used as it gave the highest PNFR for all algorithms in Figure 5.12. The quantization of the sampled signal is simulated as described in section 4.3. See Figure 5.13 for how the bit-depth of the sampled signal affected the PNFR of the target in the range-Doppler matrix.

Discussion

As can be seen from Figure 5.13 all algorithms handled the error from quantization similarly. Since no algorithm suffers more than any other, it is reasonable to assume that all algorithms are able to successfully remove the DSI and clutter and the PNFR performance is instead limited by other factors for this data. The plot instead provides how the quantization distortion limits the PNFR overall. It can be seen that there are close to no gains with an ADC resolution higher than 10 bits. It is also noticeable that the performance quickly deteriorates with less

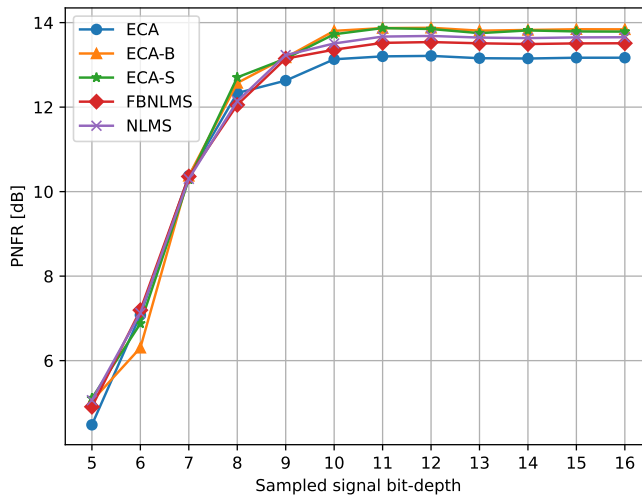


Figure 5.13: PNFR for North American Sabreliner target where the sampled signal has different bit-depth. From this result, there is no difference between the algorithms in how they handle quantization noise. For this integration time of 1 s there appears to be no further improvement with higher bit-depth after 10 bits. With less than 8 bits, the performance deteriorates fast.

than 8 bit ADC resolution. This does mean that the KerberosSDR with a sampled signal bit-depth of 8 bits (with only 7 bits being effective) is limited by its ADC resolution.

The loss of PNFR with less than 8 bits is somewhat linear in the decibel scale, with approximately 2.67 dB / bit. By modeling the quantization error as additive noise, the intensity becomes:

$$\frac{\Delta^2}{6} = \frac{\left(\frac{2}{2^b} A_{max}\right)^2}{6} = \frac{2A_{max}^2}{3} 2^{-2b}, \quad (5.5)$$

where A_{max} is the maximum amplitude of the signal, independent of the bit-depth. Expressed in dB, the noise intensity is:

$$10 \log_{10} \left(\frac{2A_{max}^2}{3} 2^{-2b} \right) = 10 \log_{10} \left(\frac{2A_{max}^2}{3} \right) - 2b \cdot 10 \log_{10} (2). \quad (5.6)$$

This means that the modeled noise intensity increases with $20 \log_{10}(2) \approx 6.02$ dB for every bit of the signal bit-depth lost. For future comparison, this is an approximate $2.67/6.02 \approx 0.444$ dB reduction of the PNFR for each dB increase in the modeled noise intensity.

5.4.2 Additive noise and DSI

This test aims to evaluate the impact of different noise and DSI levels in the surveillance signal. For the test, a recorded signal from a reference antenna is used and the signal from the surveillance antenna is simulated as described in section 4.4 with $m_t = 40$ and $k_t = 29$. The reference signal is from the KerberosSDR recording described in section 5.2.1, taken at the time 172 s. The data at 172 s is used as the results in Figure 5.4 shows that this reference signal can give good PNFR for a target.

See Figure 5.14 for how different noise and DSI intensities (equivalent to echo to noise ratio and echo to DSI ratio respectively) affects the PNFR. The noise is simulated as white Gaussian noise and is meant to represent phenomenon such as thermal noise and quantization noise.

As the results are close to identical for all ECA variants (ECA, ECA-B, ECA-S) only one of them are plotted. The same is true for the LMS based algorithms (NLMS and FBNLMS).

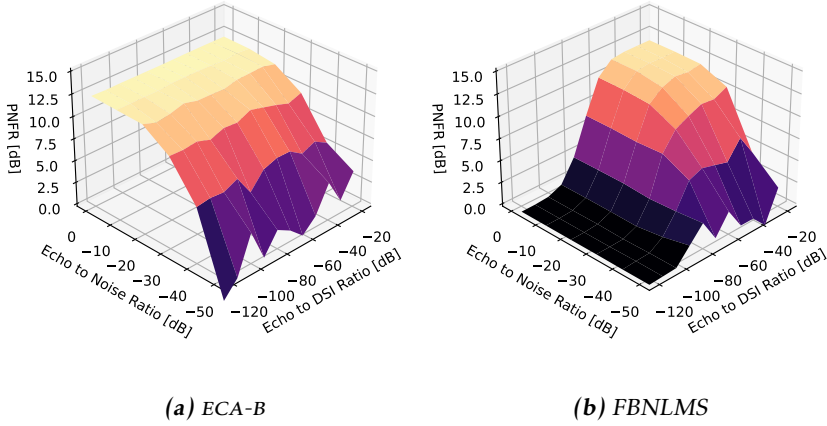


Figure 5.14: The PNFR as a function of echo to noise ratio and echo to DSI ratio. The ECA variants can better suppress DSI and clutter, while all algorithms are similarly affected by the noise intensity.

Discussion

The results in Figure 5.14 provides two important realizations, that none of the algorithms are any better or worse at handling random noise and that the LMS-variants struggle when the DSI and clutter influence is strong.

The fact that all algorithms are equally limited by the echo to noise ratio can be explained as the increase in the noise-floor dominates any other effects that the noise might have on the algorithm's DSI and clutter suppressing performance. That is, it is the noise-floor that limits the PNFR. It is noticeable that there appear to be a point of diminishing returns when increasing the echo to noise ratio. This point is approximately at -20 dB. This is consistent with the results in Figure 5.13 considering that quantization distortion can be well modeled by additive white noise. As the signal strength in the test with quantization was unknown, the values of the echo to noise ratio with generated noise and the signal to distortion ratio with the quantization cannot be compared directly.

To be able to better understand the reason for FBNLMS and NLMS losing performance when the DSI is stronger (when the Echo to DSI ratio is lower), despite the echo to noise ratio not changing, the mean squared error (MSE) between the true DSI and clutter is investigated. Since the goal of the algorithms is to create a version of the surveillance signal without the DSI and clutter, the estimated DSI and clutter signal, $\hat{x}_{DSI}[n]$, is given by:

$$\hat{x}_{DSI}[n] = x_e[n] - x_{ef}[n], \quad (5.7)$$

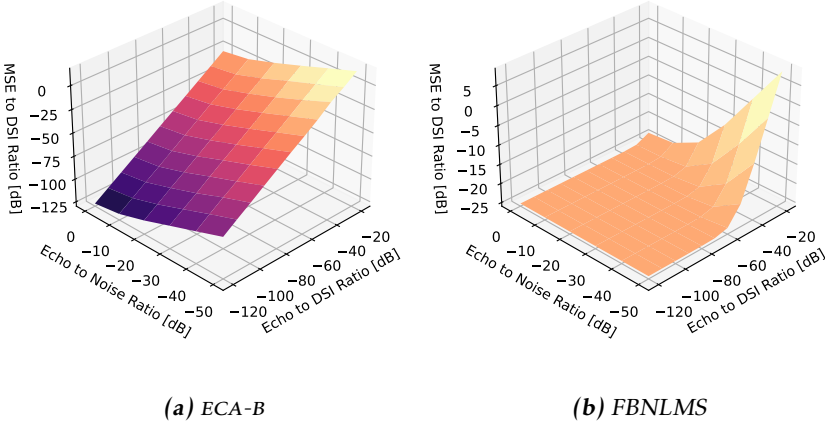


Figure 5.15: The ratio between the mean squared error for the estimated DSI and the true DSI as a function of echo to noise ratio and echo to DSI ratio. For the ECA variants the error is only affected by the DSI to noise ratio while the errors for the LMS algorithms has a minimum of approximately -25.3 dB.

where $x_e[n]$ is the signal from the surveillance antenna containing the DSI and $x_{ef}[n]$ is the filtered signal produced by the algorithm. As $x_e[n]$ is completely simulated in this test, the true DSI and clutter influence, $x_{DSI}[n]$ is known. The mean squared error, normalized with the power of the DSI, is given by:

$$\text{Normalized MSE} = \frac{\frac{1}{N} \sum_{n=0}^{N-1} |x_{DSI}[n] - \hat{x}_{DSI}|^2}{\frac{1}{N} \sum_{n=0}^{N-1} |x_{DSI}[n]|^2}, \quad (5.8)$$

where N is the number of samples in the processed segment of the signal. The normalized mean squared errors (5.8) are plotted in Figure 5.15.

From Figure 5.15a, it is apparent that the normalized MSE, in decibel, scales linearly with echo to DSI ratio minus the echo to noise ratio, both expressed in decibel. That is,

$$\begin{aligned} \text{Norm. MSE} &= \alpha((\text{Echo} - \text{DSI}) - (\text{Echo} - \text{Noise})) \\ &= \alpha(\text{Noise} - \text{DSI}), \end{aligned} \quad (5.9)$$

where Echo, DSI and Noise are measured in dB and α is a positive constant. In other words, the normalized MSE increases when the noise to DSI ratio does. It is to be expected that more noise causes the estimation of the DSI and clutter to be worse.

While Figure 5.15b shows similar scaling for higher noise to DSI ratios, the error in this plot has a lower limit around -25.3 dB. To compare this error with the

power of the target echo, the echo to (not normalized) error ratio is:

$$\begin{aligned} \text{Echo} - \text{MSE} &= (\text{Echo} - \text{DSI}) + \text{DSI} - ((\text{MSE} - \text{DSI}) + \text{DSI}) \\ &= (\text{Echo} - \text{DSI}) + 25.3 \text{ dB}, \end{aligned} \quad (5.10)$$

i.e. the echo to error ratio is equal to the echo to DSI ratio plus 25.3 dB. Based on Figure 5.14b, FBNLMS starts losing PNFR when the echo to DSI ratio is less than approximately -50 dB, which translates to a echo to error ratio of -50 dB + 25.3 dB = -24.7 dB which in turn is not far from the limit of around -20 dB echo to noise ratio where all algorithms begin losing PNFR due to the noise. The decline in PNFR then happens at similar rates when comparing the losses from strong noise and strong DSI. From this, it is reasonable to come to the conclusion that it is the lower limit of the mean squared error for the LMS variants that is the major cause of the loss of PNFR when the DSI gets stronger. Based on the assumption that the MSE increases the noise floor the same way as noise of the same power (the MSE of a signal is equal to its power).

As for the reason behind why the normalized MSE has a lower limit with the LMS algorithms, it is linked to the rate of convergence. Consider the special case where there is no noise and no echoes. The signal from the surveillance antenna only consist of DSI and clutter, simulated as the reference signal filtered with a time invariant FIR filter, i.e. simulated as something that the LMS algorithms can perfectly model. The normalized MSE for these conditions is again -25.3 dB, i.e. the minimum value seen in Figure 5.15b. By letting the FBNLMS algorithm estimate the filter on multiple runs thought the same simulated signal, the normalized MSE is reduced according to Table 5.3.

Table 5.3: Table showing how the normalized MSE decreases when FBNLMS is given more time to converge.

Extra time for convergence	Normalized MSE
0 s	-25.3 dB
1 s	-27.8 dB
2 s	-29.3 dB
3 s	-30.5 dB

Considering that the ECA variants does not need time to converge, it is reasonable that they do not have a similar lower bound for the normalized MSE. The final result is that algorithms with a faster rate of convergence will be able to handle stronger DSI and clutter.

It should be noted that this difference between the LMS algorithms and ECA variants, when handling strong DSI, cannot explain the differences seen when analyzing the far away A321 target (see section 5.2.3). The unmodified variant of ECA performs well with strong DSI in this test, but gives bad result for the A321 target.

6

Conclusion and future work

The aim of this thesis was to compare signal processing algorithms for direct signal interference suppression and investigate how different hardware constraints affect the detection performance. This chapter concludes the results found and answers the research questions put forth in section 1.1.3. The next steps for future work are also discussed.

6.1 Conclusions

To be able to fulfill the aim of comparing algorithms for DSI suppression, a selection of algorithms used for comparison were made. The selection consisted of the high performing ECA algorithm and its variants ECA-B and ECA-S, together with the faster LMS-variants, NLMS and FBNLMS.

To answer the first research question on what DSI suppression algorithms were feasible to implement on a low-cost passive radar system, benchmark on the execution time, as well as detection performance in different scenarios, for each candidate were made. It was found that in terms of execution time, FBNLMS was the most prominent contender, being almost 10–20 times faster than the other algorithms. FBNLMS fast run-time did however come with an increased risk of instability for high μ compared to the normal NLMS with the same convergence speed, seen in the parameter sensitivity analysis in section 5.2.4. Both FBNLMS and NLMS were also found to be unable to handle strong DSI and clutter.

In comparison, the ECA variants had no problems with the power of the DSI. The fastest variant, unmodified ECA, had an execution time of less than one second, making it feasible to use for real-time operation as is. However, considering that multiple IoOs are needed to get an estimate of target position and velocity, its

slower execution time becomes a hindrance as it needs to be run once per IoO.

The other two variants, ECA-B and ECA-S showed a benefit over FBNLMS for the far-away target in section 5.2.3. Here, the ability to model and therefore remove frequency shifted DSI lowered the noise floor enough to make the target visible, where no other algorithm could. The complete cancellation of the DSI was possible due to the wider and deeper cancellation notch of ECA-B. This notch did however also cause problems, as it also removed targets close to it, as well as created multiple false target detections when the target was close to the notch. ECA-S avoided the false detections, but at a cost of higher execution time.

From the results, the conclusion is that FBNLMS is the most feasible algorithm to use in a low-cost passive radar system, especially if real-time operation is wanted. This conclusion is mostly based on FBNLMS' better execution time and lack of disadvantages to outweigh it. For the purpose of covering blind-spots of more sophisticated radar system, the low-cost system would unlikely have to deal with weak and far-way targets. While strong DSI can be a disadvantage for this purpose, an assessment have to be made depending on the direct signal levels in the intended location of the low-cost system. For blind-spots close to mountains and in valleys, the same terrain blocking sight to the main system might, in some cases, also be utilized to block and reduce the strongest DSI.

Continuing on to the second research question on what the main limitations and effects on PNFR the use of low-cost hardware caused, it was found that all the DSI suppression algorithms were equally effected by limitations imposed by hardware limitations. The exception being the lower performance of LMS algorithms for strong DSI if this was seen as a consequence of a choice of a cheaper directional antenna, with the idea that a cheaper antenna would have bigger sidelobes allowing more leakage of the direct signal into the surveillance channel. A similar reasoning works if beamforming is used instead, where the quality of the directional antenna is replaced with the number of omnidirectional antennas and channels used for beamforming.

Considering the ADC resolution, it was found that the PNFR started to degrade with less than 10 bits. With a bit-depth of 8 or less, the quantization noise became dominant and reduced the PNFR with 6.02 dB for every bit lost. Similarly, it was seen that the PNFR started to fall with an echo to noise ratio of -20 dB or less. As the used low-cost SDR, KerberosSDR, had an ADC resolution of less than 10 bits, it was clear that the bit-depth of the signal was a limiting factor for this system.

6.2 Future work

There exist multiple modifications of the DSI and clutter suppressing algorithms investigated in this thesis. Some examples include the modifications to the ECA variants where Fast Fourier transform is used to faster calculate the matrix multiplications and faster ways of solving the main least squares problem related to Levinson–Durbin recursion. Another possibility could be to allow the inclusion

of echoes with negative time delays in the model used in ECA. While such an echo does not exist, it could allow better modeling of echoes with a time delays not evenly divisible with the sample time, similar to FIR taps with a negative time delay used for LMS. For possible modifications of FBNLMS, there could be gains in performance if the rate of convergence could be increased without risking problems with instability. Furthermore, there also exist other algorithms like Recursive Least Squares (RLS) and Least Square Lattice that can yield a better rate of convergence.

Considering that the ADC resolution was a limiting factor for KerberosSDR, it can be interesting to test using multiple channels for the same surveillance antenna. The idea being that the quantization noise can be reduced through averaging of the channel data, assuming that there is enough noise in the ADC to make the quantization noise not equal between the two channels.

One of the main benefits with ECA-B and ECA-S is that they can suppress frequency shifted versions of the direct signal. This allowed them to suppress the peaks and content along the Doppler-dimension at short bistatic range. As mentioned, this content in the range-Doppler matrix is caused by properties of the FM-signal but further studies can be made to understand exactly what makes them appear and more importantly, if any prepossessing can be done to remove them. Removing them would not only allow better performance for the LMS algorithms and unmodified ECA, but also remove the false peaks caused by properties of the FM-signal mentioned in this thesis.

While the detection performance has been investigated in this thesis, there is still an important next step of fusing the detections from multiple IoOs into estimates of position and velocity of the targets. There is also the possibility of applying tracking in the bistatic range and Doppler domain to further increase estimation accuracy and to allow better rejection of false alarms, i.e. ignoring detections that do not follow a realistic motion model.

Bibliography

- [1] H. D. Griffiths and C. J. Baker, *An Introduction to Passive Radar*. Boston, MA, USA: Artech House, 2017.
- [2] “Building a passive radar system with rtl-sdr dongles.” RTL-SDR.com. URL <https://www.rtl-sdr.com/building-a-passive-radar-system-with-an-rtl-sdr/>. (Accessed: 2022-01-24).
- [3] “KerberosSDR Quickstart Guide.” RTL-SDR.com. URL <https://www.rtl-sdr.com/ksdr/>. (Accessed: 2022-01-27).
- [4] J. L. Garry, C. J. Baker, and G. E. Smith, “Evaluation of direct signal suppression for passive radar,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3786–3799, 2017. doi: 10.1109/TGRS.2017.2680321.
- [5] F. Colone, D. W. O’Hagan, P. Lombardo, and C. J. Baker, “A multistage processing algorithm for disturbance removal and target detection in passive bistatic radar,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, pp. 698–722, 2009, doi: 10.1109/TAES.2009.5089551.
- [6] F. Colone, C. Palmarini, T. Martelli, and E. Tilli, “Sliding extensive cancellation algorithm for disturbance removal in passive radar,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 3, pp. 1309–1326, 2016, doi: 10.1109/TAES.2016.150477.
- [7] Y. Zhao, Y. Zhao, X. Lu, and M. Xiang, “Block NLMS cancellation algorithm and its real-time implementation for passive radar,” in *IET International Radar Conference 2013*, (Xi’an, China), Apr. 2013. doi: 10.1049/cp.2013.0341.
- [8] “About RTL-SDR.” RTL-SDR.com. URL <https://www.rtl-sdr.com/about-rtl-sdr/>. (Accessed: 2022-02-28).
- [9] P. Tamás and S. Rudolf, “Quad channel software defined receiver for passive radar application,” *Archives of Electrical Engineering*, vol. 66, no. 1, pp. 5–16, 2017, doi: 10.1515/aee-2017-0001.

- [10] H. D. Griffiths and C. J. Baker, "Passive bistatic radar," in *Principles of Modern Radar: Radar Applications*, Raleigh, NC, USA: SciTech Publishing, 2014, ch. 11.
- [11] D. Poullin, "Passive detection using digital broadcasters (DAB, DVB) with COFDM modulation," *IEE Proceedings-Radar, Sonar and Navigation*, vol. 152, no. 3, pp. 143–152, 2005.
- [12] J. E. Palmer, H. A. Harms, S. J. Searle, and L. Davis, "DVB-T Passive Radar Signal Processing," *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp. 2116–2126, 2013, doi: 10.1109/TSP.2012.2236324.
- [13] T. Martelli, F. Colone, C. Bongioanni, D. Pastina, and P. Lombardo, "Short-range passive radar for small private airports surveillance," in *2016 European Radar Conference (EuRAD)*, pp. 165–168, 2016.
- [14] D. W. O'Hagan and C. J. Baker, "Passive Bistatic Radar (PBR) using FM radio illuminators of opportunity," in *2008 New Trends for Environmental Monitoring Using Passive Systems*, pp. 1–6, 2008, doi: 10.1109/PAS-SIVE.2008.4787000.
- [15] M. Malanowski, K. Kulpa, J. Kulpa, P. Samczynski, and J. Misiurewicz, "Analysis of detection range of FM-based passive radar," *IET Radar, Sonar and Navigation*, vol. 8, no. 2, pp. 153–159, 2014, doi: 10.1049/iet-rsn.2013.0185.
- [16] M. Malanowski, *Signal Processing for Passive Bistatic Radar*. Boston, MA, USA: Artech House, 2019.
- [17] Z. Yaodong, L. Xiaode, L. Jichuan, and X. Maosheng, "Noncausal least mean square adaptive interference canceller for passive radar," in *2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2013)*, (KunMing, China), pp. 1–3, 2013, doi: 10.1109/IC-SPCC.2013.6664051.
- [18] M. Olofssen, *Signal Theory*. Lund, Sweden: Studentlitteratur, 2011.
- [19] S. Haykin, *Adaptive Filter Theory*. London, United Kingdom: Pearson Education Limited, 5 ed., 2014. International edition.