Cascade Matched Filtering & Adaptive Threshold Techniques on Optical Tracking Signals under AWGN

Sebastian Elm

LUND UNIVERSITY

Department of Automatic Control

Hilti AG - BU Measuring

MSc Thesis
TFRT-6076
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2019

Abstract

A cascade matched filtering (CMF) approach is purposed for filtering rectangular pulse train tracking signals, with known signal characteristics, in an embedded signal detection unit. Two adaptive detection algorithms are suggested, adapting to alterations in noise signal characteristics, due to environmental changes. The end product was an embedded tracking detection system, enabled to identify distance and direction to a tracking source, and transmit a well synchronized replica signal back to the tracking source. The suggested filtering and detection method, enhanced the tracking distance of the system significantly.

An introduction to the topic is given in the first section of the thesis, followed by underlying theory on the broad topic. The thesis methodology is explained and tracking detection system with its subcomponents are modeled and described. Then the results from both simulation and embedded real-time data is presented, proceeding with discussions and ultimately conclusions.

Both simulation and embedded real-time data confirms improvements by implementing purposed filtering and detection techniques, by studying the SNR gain from the filter segment, and the ROC of the detection segment. The SNR gain in dB increases logarithmically with each added pulse period to the matched filter, and the linearity of the matched filter is clearly visualized in the filtered output signal. The 3-dimensional ROC curves, illustrates saturations in detection probabilities at a certain threshold, implying a correlation between filter extension and achieving a certain detection probability.

Acknowledgements

I would like to express my deepest gratitude to my supervisor Andreas Winter, advisor Rolf Johansson and my examiner Karl-Erik Årzén for their exceptional support during the Master Thesis. The given confidence from Andreas Winter at Hilit AG have enriched my competence within a broad field of engineering topics. Credit to Ivan Sojic, Thomas Gloor and Giovanni Dandrea that have participated in meetings to elucidate project issues and presenting valuable feedback.

A special appreciation to all employees at Hilti AG - BU Measuring, that have greeted me with enthusiasm and geniality during my work period.

Schaan, Liechtenstein, March 2018
Sebastian Elm

Abbreviations

Digital Signal Processing

AAF	Anti-Aliasing Filter
ADC	Analogue to Digital Converter
ALE	Adaptive Line Enhancer
AMF	Adaptive Matched Filter
AOI	Area Of Interest
AWGN	Additive White Gaussian Noise
CMF	Cascade Matched Filter
CFRFT	Concise Fractional Fourier Transform
DAC	Digital to Analogue Converter
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
FDAMF	Frequency-Domain Adaptive Matched Filter
FFT	Fast Fourier Transform
FIR	Finite-duration Impulse Response
MF	Matched Filter
MOAMF	Multiple Observations Adaptive Matched Filter
NLMS	Normalized Least-Mean-Square
PFA	Probability of False Alarm
RLS	Recursive Least-mean-Squared
ROC	Receiver Operating Characteristic
SAADC	Successive Approximation Analogue to Digital Converter
SNR	Signal to Noise Ratio
SOI	Signal Of Interest
TOC	Transmitter Operating Characteristic
TRC-IS	Time Reversal Convolution and Interference Suppression

Embedded System

CMSIS	Cortex Microcontroller Software Interface Standard
DMA	Direct Memory Access
EOC	End Of Conversion
RTCS	Real-Time Control System
RTOS	Real-Time Operating System

Miscellaneous

EMR	Electromagnetic radiation
ESD	Energy Spectral Density
LED	Light Emitting Diode
LTI	Linear Time-Invariant
MAP	Maximum *A posteriori* Probability
NEP	Noise equivalent power
NIR	Near-Infrared
PDF	Probability Density Function
RMS	Root Mean Square

Contents

1

Introduction

Automation has a great potential in the ongoing digitalization of the construction industry, recognized as ”Construction 4.0”. Leading construction players believe that automation within production and quality management will gain the most of the digitalization in the construction industry [Nölling et al., 2016, p. 6 & 14]. This will create opportunities for intelligent automation solutions for the future construction site. Laying out construction points is a time consuming operation, which requires high accuracy. Extending the autonomous functions in this field would save time and money for construction companies. Intelligent tracking of active mobile targets used for measuring and positioning will extend the operating portfolio, contribute to faster and more stable solution. A simple model of the tracking solution is illustrated in Fig 1.1, which is a reduced system model of later demonstrated model in Fig. 4.3.

The operating environments for optical tracking differs and are prone to various optical interferences. This influences the performance of optical signals used for tracking in construction environments. Optical interferences differ in characteristics and could be categorized as intrinsic noise, signal noise, external noise, processing noise and signal interruption [Kitchin, 2013, p. 55]. This categorization separates, e.g., background radiation from electrical measurement noise in detectors apart from each other. Smart filtering techniques will contribute to stable operation, increasing detection distance and allow the tracking system to use its full potential.

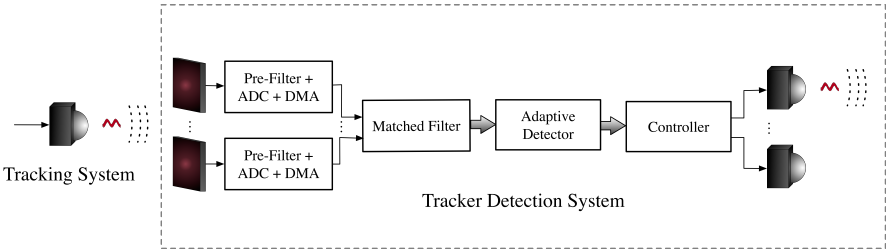

Figure 1.1 A simplification of a survey system tracking a mobile active target, each of the blocks in the tracking detection system will be explained and studied in this paper.

1.1 Problem Formulation

Tracking signals used in various surveying systems can have disparate signal characteristics, e.g., different frequency, continuous or pulsating signals with different pulse widths. The bandwidth of a receiver capable to operate with several surveying systems must have a broader bandwidth. However, by increasing the bandwidth one will decrease the signal to noise ratio (SNR) [Hamamatsu, 2011, p. 6] [Mahafza, 2013, p. 30]. Decreasing the SNR will resolve in reduced detection distance. To maintain a relatively feasible SNR level, the bandwidth of the detector must be narrowed as much as possible to restrain the noise power. This contradicts the requirements of the tracking systems, which operates in different frequencies and pulse widths to change its tracking resolution [Mahafza, 2013, p.13,171]. A fixed analogue filter would need an overly wide bandwidth and increasing the noise power to impractical levels. To still fulfill the requirements, a narrowband digital adaptive filter could be used. In this thesis a semi-adaptive Matched Filter (MF) was developed called Cascade Matched Filter (CMF). The adaptive characteristics is narrowed down to some known signal characteristics, were practically different digital filters are used in a search phase to detect the best performing filter and then apply this filter template to the incoming signal.

1.2 Delimitations

Different types of disturbances could be filtered out in different ways - an incoming optical noise signal could be filtered with an optical filter and an electrical interference with an analogue filter. These two mentioned filtering steps are part of the received signal that is converted in the ADC for this master thesis.

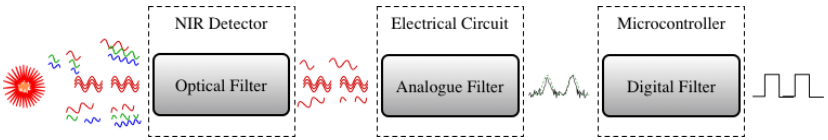

Figure 1.2 Incoming radiation from a tracking system incorporated with external noise are optically filtered as a first step with a bandpass filter (left side). An analogue anti-aliasing filter (AAF) filters out the higher frequencies in the induced electrical signal from the photodiode. Ultimately digital filtering is applied after the analogue to digital converter (ADC).

Studies are limited to the optical spectrum of Near-Infrared (NIR) radiation fixed frequency pulses [*Optics and photonics - Spectral bands*]. The NIR radiation spectrum is detected with optical filtered photodiodes, where the incident energy from the NIR photons will create a measurable photocurrent [Hamamatsu, 2011, p. 5 & 6]. No further details of optical filtering will be included nor will any studies be performed within this topic.

The extracted analogue output voltage from the photodiode will be amplified and converted with an analogue to digital converter (ADC) to measure and specify tracking characteristics. An anti-aliasing filter (AAF) is implemented to remove signals above the Nyquist frequency. The AAF consist of an analogue low-pass filter and filters the signal before it gets sampled by the ADC [Wittenmark et al., 2016, p. 12]. All filtering will be digital and performed with digital signal processing (DSP). The main scope is to filter out Additive White Gaussian Noise (AWGN), signal interruption and external spurious signals from the desired signal. Hence, all different types of noise will not be studied in this master thesis.

Research was made on finding suitable filtering techniques for RADAR signals. Two adaptive filtering techniques were compared, recursive least-mean-square (RLS) and least-mean-square (LMS). Some insightful discoveries were made and some lessons were gained from this research [Salminen, 2013, p. 35]. A different filtering approach was developed for this thesis and also some signal detection techniques. The embedded C code will not be disclosed in this thesis, but the underlying concept and results will be described, displayed and discussed.

1.3 Purpose

The purpose is elucidate the contrast and importance of different detection techniques and pulse period extensions of a matched filter. This thesis covers one out of many approaches on how to filter and detect an incoming signal. A Matched Filter (MF) was selected, since it is the optimal linear filter for instantaneously maximizing the SNR output [Liu et al., 2017, p. 1][Mahafza, 2013, p. 143]. A concept for improving the algorithm was developed to increase the detectivity of incoming pulse

train signals, this modification was named cascade matched filter (CMF). The intent is also to illustrate how detectivity algorithms can improve the detectivity and minimize errors in the detection process.

1.4 Outline

Section 2 - Theory
Background theory of signals, detectors and filters with corresponding equations and algorithms will be provided in this section.

Section 3 - Methodology
Explaining the steps and methods of choice, to prove and establish the system truth, design and develop filters and detection algorithms.

Section 4 - Modeling
Deriving the tracking system model, signal and noise characteristics, the cascade matched filter algorithm concept and detection algorithms.

Section 5 - Simulation
Simulation and performance analysis of the cascade matched filter and detection system.

Section 6 - Real Time Operation
Real-time performance analysis of the embedded cascade matched filter and detection system.

Section 7 - Discussion
Discussion including comparison of the results from simulation and real-time operation.

Section 8 - Conclusion
Conclusions from the thesis, results and discussion section along with further investigation and future development aspects.

2

Theory

2.1 Continuous and Discrete Time Signals

Continuous-time signals have aperiodic spectra and *discrete-time* signals have periodic spectra [Proakis and Manolakis, 2014, p. 275]. Thus the frequency range is finite for discrete-time signals and infinite for continuous-time signals. The frequency range for discrete-time signals is bounded between $\omega = [-\pi, +\pi]$ radians, where $\omega = \pi$ corresponds to the highest rate of oscillation. *Periodic signals* have discrete spectra and are described by Fourier series. The lines of the discrete spectrum represents the Fourier series coefficients, where the line spacing is $\Delta f = 1/T_p$ for continuous-time signals and $\Delta f = 1/N$ for discrete-time signals. Some analogy of the equations holds for discrete time signals, where the fixed discrete time representation k is equal to a fixed time t. The variable denotation including continuous and discrete time simultaneously is refereed by first the continuous time representation and then the discrete time, i.e, $x(t \vee k)$. In this master thesis the desired input signal is denoted with $x(t)$, noise $n(t)$ and the combined measurable signal $y(t) = x(t) + n(t)$ for continuous signals. To understand the signal behavior in this thesis some elementary equations are required. Lets first start with the *instantaneous power* $p_{xx}(t \vee k)$ for a signal $x(t \vee k)$ at a fixed time t for continuous time signals and k for discrete time signals, this is defined by

$$p_{xx}(t) = x(t) \cdot x^*(t) = |x(t)|^2 \tag{2.1a}$$

$$p_{xx}(k) = x(k) \cdot x^*(k) = |x(k)|^2 \tag{2.1b}$$

where $x^*(t \vee k)$ denotes the complex conjugate transpose of signal $x(t \vee k)$ [Johansson, 1993, p. 40]. The *average power* of a signal $x(t \vee k)$ during a time interval

$[t_0, t_0+T] \vee [k_0, k_0+N]$ is defined as

$$\overline{p}_{xx}(t_0, T) = \frac{1}{T} \int_{t_0}^{t_0+T} x(t) \cdot x^*(t) dt \tag{2.2a}$$

$$\overline{p}_{xx}(k_0, N) = \frac{1}{N} \sum_{k=k_0}^{N+k_0} |x(k)|^2, \tag{2.2b}$$

where the general formulation of the *average power* is

$$\overline{p}_{xx} = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} x(t) \cdot x^*(t) dt \tag{2.3a}$$

$$\overline{p}_{xx} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{N} |x(k)|^2. \tag{2.3b}$$

By taking the *instantaneous power* over time, one will get the signal energy

$$e_{xx} = \int_{-\infty}^{+\infty} p_{xx}(t) dt = \int_{-\infty}^{+\infty} x(t) \cdot x^*(t) dt \tag{2.4a}$$

$$e_{xx} = \sum_{-\infty}^{+\infty} p_{xx}(k) = \sum_{-\infty}^{+\infty} |x(k)|^2 \tag{2.4b}$$

if two signals interact with each other they are said to be correlated. In the measurable signal $y(t \vee k) = x(t \vee k) + n(t \vee k)$ there might be a correlation between the desired signal $x(t \vee k)$ and the noise signal $n(t \vee k)$

$$e_{xn}(\tau) = \int_{-\infty}^{+\infty} x(t) n^*(t-\tau) dt$$

$$e_{nx}(\tau) = \int_{-\infty}^{+\infty} n(t) x^*(t-\tau) dt = e^*_{xn}(-\tau) \tag{2.5a}$$

$$e_{xn}(n) = \sum_{k=-\infty}^{+\infty} x(k) n^*(k-n)$$

$$e_{nx}(n) = \sum_{k=-\infty}^{+\infty} n(k) x^*(k-n) = e^*_{xn}(-n). \tag{2.5b}$$

If the noise signal and the desired signal is uncorrelated then $e_{xn}(t \vee k) = 0$ [Johansson, 1993, p. 40]. This results in a total measured signal energy corresponding to the

summation of the desired energy and the noise energy:

$$e_{yy} = e_{xx} + \underbrace{e_{xn}}_{=0} + e_{nn} = e_{xx} + e_{nn}$$

This event is unlikely to occur for the studied tracking system. For simplicity the *energy spectral density* (ESD) of the signal $x(t \vee k)$ could be defined with the Fourier Transform $X(\omega)$

$$E_{xx} = X(\omega)X^*(\omega) \tag{2.6}$$

and the *cross-energy spectral density* between $x(t \vee k)$ and $n(t \vee k)$ yields

$$\begin{aligned} E_{xn} =& X(\omega)N^*(\omega) = \mathscr{F}\{x\} \cdot \mathscr{F}\{n^*\} \\ =& \mathscr{F}\left\{ \begin{array}{l} \int\limits_{-\infty}^{+\infty} x(t)n^*(t-\tau)dt \\ \sum\limits_{k=-\infty}^{+\infty} x(k)n^*(k-n) \end{array} \right\} = \mathscr{F}\{e_{xn}\} \end{aligned} \tag{2.7}$$

$$\Leftrightarrow E_{xn}(\omega) \overset{\mathscr{F}}{\leftrightarrow} e_{xn}(t \vee k). \tag{2.8}$$

This is better known as the Wiener-Khintchine theorem [Johansson, 1993, p. 41]. This results implies that the same information about the signal exist in the correlation sequence of a signal and its energy spectral density, whereas no phase information exist [Proakis and Manolakis, 2014, p. 291]. The ESD represents the distribution of signal energy over frequency. Since the *Fourier transform* of a signal is

$$X(\omega) = \mathscr{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t)e^{-i\omega t}dt \tag{2.9}$$

$$X(\omega) = \mathscr{F}\{x(k)\} = \sum_{k=-\infty}^{+\infty} x(k)e^{-i\omega k} \tag{2.10}$$

and the *inverse Fourier transform* is

$$x(t) = \mathscr{F}^{-1}\{X(\omega)\} = \frac{1}{2\pi}\int_{-\infty}^{+\infty} X(\omega)e^{i\omega t}d\omega \tag{2.11}$$

$$x(k) = \mathscr{F}^{-1}\{X(\omega)\} = \frac{1}{2\pi}\sum_{k=-\infty}^{+\infty} X(\omega)e^{i\omega k} \tag{2.12}$$

the discrete *cross-energy density spectrum* E_{xn} extension is represented by:

$$E_{xn}(\omega) = \mathscr{F}\{e_{xn}\} = \sum_{n=-\infty}^{+\infty} e_{xn}(n)e^{-i\omega n} = \sum_{n=-\infty}^{+\infty}\left[\sum_{k=-\infty}^{+\infty} x(k)n^*(k-n)\right]e^{-i\omega n}$$

Signals with infinite energy are better observed with the *cross covariance*

$$C_{xn}(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} x(t)n^*(t-\tau)dt \tag{2.13a}$$

$$C_{xn}(n) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{N} x(k)n^*(k-n) \tag{2.13b}$$

and the *power cross spectrum* S_{xn}. They are expressible together as

$$S_{xn}(\omega) = \mathscr{F}\{C_{xn}\} \tag{2.14}$$

The *autospectrum* and *autocovariance* are expressible in a similar manner

$$S_{xx}(\omega) = \mathscr{F}\{C_{xx}\} = \mathscr{F}\left\{ \begin{array}{c} \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{+T} x(t)x^*(t-\tau)dt \\ \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{+N} x(k)x^*(k-n) \end{array} \right\} \tag{2.15}$$

Thus the power spectra and covariance function are related according to the Wiener-Khintchine theorem [Johansson, 1993, p. 42].

2.2 NIR-signals and detectors

Near infra-red (NIR) electromagnetic radiation (EMR) has a wave length of 750 nm - 1400 nm, established as one of the spectrum contiguous with the visible spectra. The NIR spectrum is a small portion of the entire infra-red spectra, which spreads over 750 nm - 1000 μm [D'Amico et al., 2009, p. 21-22]. Near infra-red radiation is utilized by image intensifiers such as night vision goggles. The NIR spectra belongs to the reflected infrared spectrum of 750 nm - 3μm, in contrast to the thermal infrared spectrum of 3 μm - 15 μm. One of the most well known tracking technique within the thermal infrared spectrum is the guided missile technology, more known as "heat seeking" missiles [D'Amico et al., 2009, p. 22].

Infra-red detectors could be categorized from the essential operating mechanisms [D'Amico et al., 2009, p. 25]. Photon sensors that are used for the reflective infra-red consist of the following detection category: photoconductive(intrinsic), photoconductive(extrinsic), photovoltaic and superconducting sensors [D'Amico et al., 2009, p. 25]. Of the mentioned NIR detection techniques, the photovoltaic sensors are the only one that does not need an external power supply [D'Amico et al., 2009, p. 25]. Thus photovoltaic sensors such as photodiodes, could be used in low power consumption tools.

Signal-to-noise ratio (SNR) is a parameter of importance in NIR detectors [Orton, 2008, p. 12]. The quantity of incident light that results in a SNR unity is called

noise equivalent power (NEP). [Hamamatsu, 2011, p. 6].

$$NEP = \frac{I \cdot A_s}{\frac{V_s}{V_n} \cdot \sqrt{\Delta f}} = \frac{I \cdot A_s}{SNR \cdot \sqrt{\Delta f}} \quad [W/\sqrt{Hz}] \tag{2.16}$$

A_s denotes the detector active area [cm^2], I the irradiance [W/cm^2], SNR is given by rms value of the signal and noise voltages and Δf is the noise bandwidth [Hamamatsu, 2011, p.6][D'Amico et al., 2009, p. 10]. The active detector area and irradiance are illustrated in Fig. 2.1.

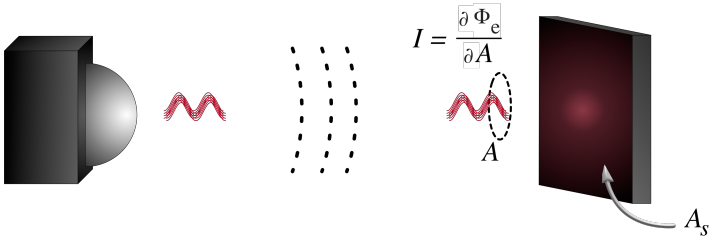

Figure 2.1 Illustrates how a NIR diode produces an incoming irradiance I, which is detected by the detector active area A_s. The irradiance $[W/cm^2]$ is defined as the quotient between the partial derivative of the radiant flux received Φ_e and the partial derivative of the area A.

NEP sets the detection threshold for a desired signal [Orton, 2008, p. 12]. When the tracking NIR signal has reached the ADC, it is incorporated with intrinsic noise, signal noise, external noise and processing noise as mentioned in Section 1. Detection of a NIR signal with a radiation detector is normally limited to the external noise such as background noise. Generally the arising processing noise from amplifiers could be neglected, thus the greatest improvements are performed at the detector output and its surrounding [Orton, 2008, p. 12].
An easier interpretation of how the SNR affects the performance of a NIR detector is the detectivity D^*.

$$D^* = \frac{\sqrt{A_s}}{NEP} = \frac{SNR \cdot \sqrt{\Delta f}}{I \cdot \sqrt{A_s}} \tag{2.17}$$

This is interpreted in a way that the SNR is proportional to the detectivity D^* of the detector [Hamamatsu, 2011, p. 6].

2.3 SNR

The SNR in NIR detectors can be expressed by the detection threshold of the NEP.

$$SNR = \frac{I \cdot A_s}{NEP \cdot \sqrt{\Delta f}} \tag{2.18}$$

This means that if the noise equivalent power (NEP) is reduced, the signal to noise ratio (SNR) will increase. There is a limit on how much the NEP could be reduced, so to further improve the SNR and detectivity there are two considerable actions one can make - increase the desired signal (irradiance) and decrease the bandwidth of the noise. If we can not control the irradiance of the received tracking signal, the best option is to narrow the bandwidth of the noise. A third option could also be considered, namely increasing the active detection area. It is beneficial to not oversize the detection area if the tracking system is able to target the NIR detector. Then the irradiating NIR tracking signal would be aimed at a small surface, hence decreasing the likelihood of detecting external noise. If the detection area is scaled up the detection of external noise increases.
A more general way of measuring SNR is the ratio between signal power and noise power [Johansson, 1993, p. 42]

$$SNR = \frac{e_{xx}}{e_{nn}} \tag{2.19}$$

This approach could also be expressed in dB according to [Kester, 2009, p. 6]

$$SNR = 20log(\frac{e_{xx}}{e_{nn}}) \quad [dB] \tag{2.20}$$

To determine the SNR from experiment and simulation data it is sometimes easier to generalize the SNR from the rms values of the signal and noise

$$SNR = 20log_{10}(\frac{V_{FS\ RMS}}{V_{NOISE\ RMS}}) \tag{2.21}$$

where FS is the full-scale of the ADC value [Kester, 2005, p. 63, Chapter 2]. To avoid signal clipping, all signals must stay within the full-scale of the ADC.

2.4 Correlation and Coherence

SNR is a harsh way of explaining the underlined problem in signal detection. Instead some more explainable factors are the correlation and coherence of the desired signal $x(t \vee k)$ and noise signals $n(t \vee k)$. The correlation aspect is important in this thesis, and the coherence was not investigated.

$$y(t) = x(t) + n(t) = g(t) * u(t) + n(t) \tag{2.22a}$$
$$y(k) = x(k) + n(k) = g(k) * u(k) + n(k) \tag{2.22b}$$

The measurable signal $x(t \vee k)$ is the product of the incoming signal $u(t \vee k)$ convoluted with the system impulse response $g(t \vee k)$ and $n(t \vee k)$ is an incoming external noise signal. The responding SNR from Eq. 2.19 would then be

$$SNR = \frac{e_{xx}}{e_{nn}} = \frac{e_{yy}}{e_{nn}} - 1 \tag{2.23}$$

where the second equality holds if $x(t \vee k)$ and $n(t \vee k)$ is uncorrelated $e_{xn} = 0$. The correlation coefficient ρ between the two signals $x(t \vee k)$ and $y(t \vee k)$ is defined as

$$\rho(\tau) = \frac{C_{xy}(\tau)}{\sqrt{|C_{xx}(\tau)|}\sqrt{|C_{yy}(\tau)|}} \tag{2.24a}$$

$$\rho(n) = \frac{C_{xy}(n)}{\sqrt{|C_{xx}(n)|}\sqrt{|C_{yy}(n)|}}. \tag{2.24b}$$

2.5 Rectangular Pulse Train

In this section the Fourier series and power density spectrum of continuous-time and discrete-time periodic rectangular pulses are established. The rectangular pulse train is a periodic rectangular shaped signal with pulse repetition rate T_p for continuous time and N for discrete time. The Fourier transform of a rectangular pulse changes with the pulse width as according to Figure 2.2.

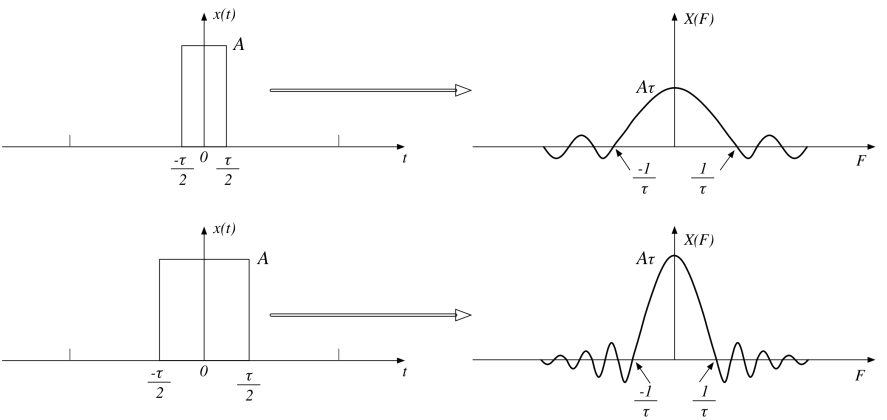

Figure 2.2 More energy is relocated to the higher frequencies as the pulse width decreases. The Fourier transform plots illustrates that the main lobe becomes broader for decreasing pulse widths.

A periodic signal has infinite energy, but finite average power. Equation 2.3 describes in general the average power of a signal. However, for a periodic signal $x(t)$, the average power could be expressed with the Fourier series coefficients.

$$\overline{p}_{xx} = \frac{1}{T_p}\int_{T_p} |x(t)|^2\, dt = \sum_{k=-\infty}^{+\infty} |c_k|^2 \tag{2.25}$$

$$\overline{p}_{xx} = \frac{1}{N}\sum_{k=0}^{N-1} |x(k)|^2 = \sum_{k=0}^{N-1} |c_k|^2 \tag{2.26}$$

When prefiltering the incoming rectangular pulse train, the amount of Fourier coefficients representing the incoming signal will decrease due to signal smoothing. By carefully selecting the anti-aliasing filter, the power of the pulse will be better preserved for the detection filtering. Then the fact, that there is infinite energy within the pulse train could be used to increase the SNR in the signal detection filtering, by including several pulse periods.

Continuous-time Rectangular Pulse Train

An rectangular pulse signal $x(t)$ is an even signal since $x(t) = x(-t)$, which is illustrated in Figure 2.3. Thus, the integration interval of the signal could be applied as $[-T_p/2, T_p/2]$. The Fourier series of an periodic signal $x(t)$, reveals some fundamental characteristics in the signal. The first Fourier coefficient $c_k = c_0$ represents the DC component (average value) of the signal $x(t)$ [Proakis and Manolakis, 2014, p. 238]. Since the periodic signal $x(t)$ is even, the Fourier coefficients c_k are real.

$$c_0 = \frac{1}{T_p}\int_{-T_p/2}^{T_p/2} x(t)dt = \frac{1}{t_p}\int_{-\tau/2}^{\tau/2} A dt = \frac{A\tau}{T_p} \tag{2.27a}$$

$$c_k = \frac{1}{T_p}\int_{-\tau/2}^{\tau/2} Ae^{-ik\Omega t}dt\Bigg|_{\Omega=\frac{2\pi}{T_p}} = \frac{A}{T_p}\left[\frac{e^{-ik\Omega t}}{-ik\Omega}\right]_{-\tau/2}^{\tau/2} \tag{2.27b}$$

$$= \frac{A}{\pi F_0 k T_p}\frac{e^{i\pi k F_0 \tau} - e^{-i\pi k F_0 \tau}}{2i} = \frac{A\tau}{T_p}\frac{sin(\pi k F_0 \tau)}{\pi k F_0 \tau},\ \ k = \pm 1, \pm 2, ...$$

When the Fourier coefficients are inserted in the Fourier series

$$x(t) = \sum_{k=-\infty}^{+\infty} c_k e^{ik\Omega t} \tag{2.28}$$

$x(t)$ resembles the continuous-time periodic train of rectangular pulses. The Fourier series $sinc(\theta) = sin(\theta)/\theta$ appearance shapes the curve on right side in Figure 2.2.

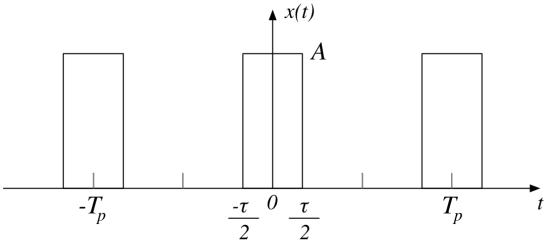

Figure 2.3 An illustration of a continuous-time periodic train of rectangular pulses with amplitude A, period T_p and pulse width τ.

The *power density function* P_{xx} for the continuous-time rectangular pulse train becomes

$$P_{xx} = |c_k|^2 = \begin{cases} \left(\frac{A\tau}{T_p}\right)^2 & ,k = 0 \\ \left(\frac{A\tau}{T_p}\right)^2 \left(\frac{sin(\pi k F_0 \tau)}{\pi k F_0 \tau}\right)^2 & ,k = \pm 1, \pm 2, ... \end{cases} \tag{2.29}$$

The Fourier series coefficients and *power density function* is plotted in Figure A.1 below. Furthermore, different rectangular pulse widths are examined, while keeping the amplitude and pulse repetition rate fixed in Figure B.2 in Appendix B.2.

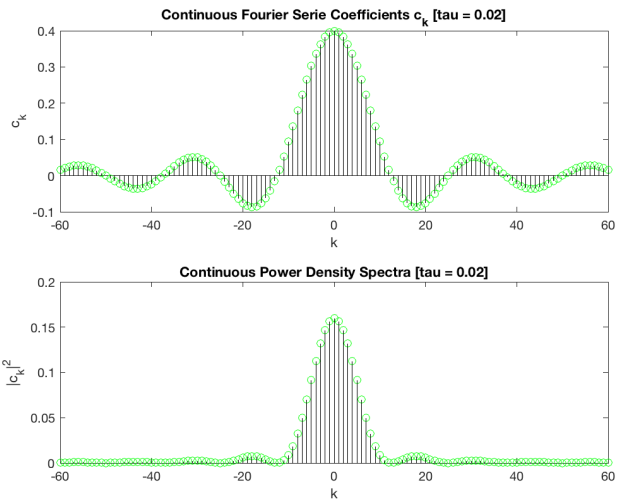

Figure 2.4 The following values are inserted in equation 2.29 $\tau = 0.02$, $A = 5$, $F_0 = 4$, $T_p = 1/F_0$, $k = 0, \pm 1, ..., \pm 60$.

Discrete-time Rectangular Pulse Train

The discrete-time Fourier coefficients is very similar to the continuous-time representation. For continuous-time signals, the *power density spectrum* P_{xx} does not contain any phase information [Proakis and Manolakis, 2014, p. 253]. The spectrum is discrete and periodic, where the periodicity originates from the fundamental signal itself. This fundamental difference between the two *power density spectrum* could be observed by comparison of Figure B.2 and Figure B.3 The Fourier coefficients for discrete-time are

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-ik\Omega n} = \frac{1}{N}\sum_{k=0}^{L-1} Ae^{-ik\Omega n}\Bigg|_{\Omega=\frac{2\pi}{N}}, k = 0,1,...,N-1$$

with some geometric summation the following equation is obtained

$$c_k = \frac{A}{N}\sum_{n=0}^{L-1}\left(e^{-i2\pi k/N}\right)^n = \begin{cases} \frac{AL}{N} & ,k=0 \\ \frac{A}{N}\frac{1-e^{-i2\pi kL/N}}{1-e^{-i2\pi k/N}} & ,k=1,2,...,N-1. \end{cases} \tag{2.30}$$

By using Euler's formula, c_k could be reduced for $k = 1,2,...,N-1$

$$\begin{aligned} c_k &\overset{k\neq 0}{=} \frac{AL}{N}\frac{1-e^{-i2\pi kL/N}}{1-e^{-i2\pi k/N}} = \frac{e^{-i\pi kL/N}}{e^{-i\pi k/N}}\frac{e^{i\pi kL/N}-e^{-i\pi kL/N}}{e^{i\pi k/N}-e^{-i\pi k/N}} \\ &= e^{-i\pi k(L-1)/N}\frac{sin(\pi kL/N)}{sin(\pi k/N)}. \end{aligned}$$

A similar appearance as in the continuous case is then found in the discrete representation

$$c_k = \begin{cases} \left(\frac{AL}{N}\right) & ,k=0,\pm N,\pm 2N,... \\ \left(\frac{A}{N}\right)e^{-i\pi k(L-1)/N}\frac{sin(\pi kL/N)}{sin(\pi k/N)} & ,otherwise \end{cases}. \tag{2.32}$$

To resemble $x(n)$ the discrete-time periodic train of rectangular pulses, the Fourier coefficients are inserted in the series

$$x(n) = \sum_{k=0}^{N-1} c_k e^{ik\Omega n} \tag{2.33}$$

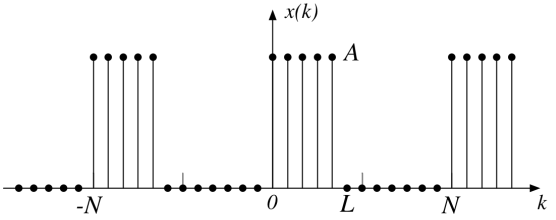

Figure 2.5 An illustration of a discrete-time periodic train of rectangular pulses with amplitude A, period N and pulse width of L descrete-time instances.

The *power density function* P_{xx} for the discrete-time rectangular pulse train is

$$P_{xx} = |c_k|^2 = \begin{cases} \left(\frac{AL}{N}\right)^2 & ,k = 0, \pm N, \pm 2N, ... \\ \left(\frac{A}{N}\right)^2 \left(\frac{sin(\pi kL/N)}{sin(\pi k/N)}\right)^2 & ,otherwise \end{cases} . \tag{2.34}$$

An illustration between the continuous-time and discrete-time power density spectrum.

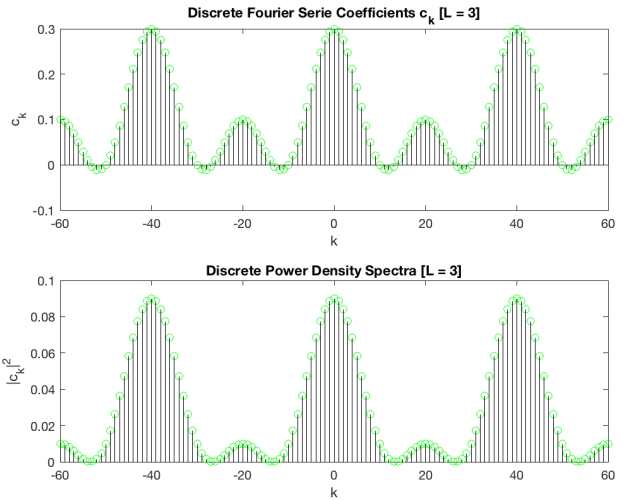

Figure 2.6 The following values are inserted in equation 2.29 $A = 4$, $L = 3$, $N = 40$, $k = 0, \pm 1, ..., \pm 60$.

2.6 Anti-Aliasing Filter and Sampling Rate

The sampling theorem states that the sampling frequency should at least be twice as high as the highest frequency component from the original signal, to avoid aliasing. In practice it needs to be greater than it states, since it assumes periodic input signals which is sampled for an infinite time [Olsson and Rosen, 2005, p. 361]. An analogue signal usually contains high frequency noise, thus making the sampling theorem solely inefficient for avoiding anti-aliasing, especially for real-time embedded digital signal processing (DSP). This would require unnecessarily high sampling frequency, which would increase the computational complexity [Casini et al., 1975, p. 831]. If direct memory access (DMA) is existent on the selected microcontroller, the sampling rate will not occupy any CPU runtime. However, the sampling rate will effect the DSP runtime, hence effecting the CPU runtime. The sampling theory in cooperation with an anti-aliasing filter (AAF) will prevent aliasing in the sampled signal. Unfiltered frequencies above the Nyquist frequency f_N will "fold" into the observable signals, if an anti-aliasing filter is not applied.

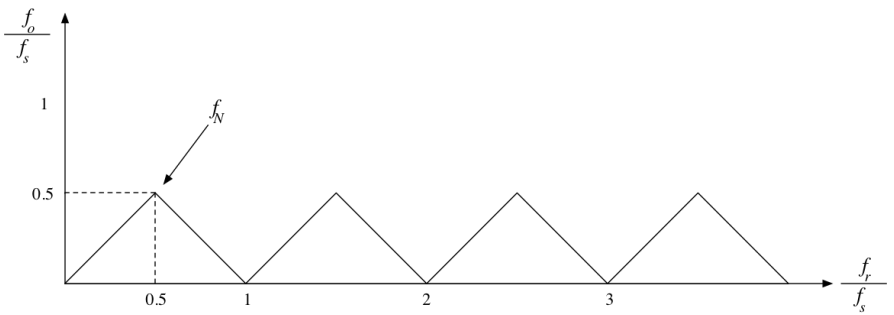

Figure 2.7 The graph represents how frequencies above the Nyquist frequencies F_N folds into the observable frequencies [Olsson and Rosen, 2005, p. 361]. The observable frequencies f_o are limited by the Nyquist frequency on the y-axis and real frequencies f_r is mapped on the x-axis. Frequencies $f_r < f_s/2$ will be observed as their true frequency. The scale represents multiples of the sampling frequency f_s [Olsson and Rosen, 2005].

The observable frequency ω_O after an aliasing occurs with the signal frequency ω_R, is described by the equation

$$\omega_O = |(\omega_R + \omega_N) mod(\omega_S) \ - \ \omega_N| \tag{2.35}$$

where mod is the modulo operator, ω_s is the sampling frequency and the ω_N is the Nyquist frequency [Wittenmark et al., 2016, p. 12]. The AAF should be designed so it removes all frequencies above the Nyquist Frequency before the signal is sampled

[Wittenmark et al., 2016, p.12]. The AFF consist of analogue filters, since after an input signal is sampled, all the aliasing signals can not be separated from original frequency signals. Subsequently the sampled signal could be sampled with a digital filter to improve the signal information for the specified application.

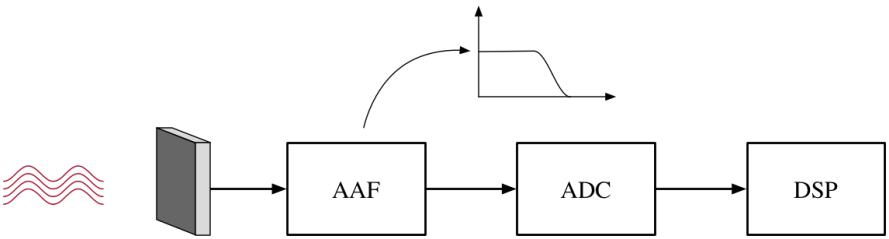

Figure 2.8 The images represents the incoming input signals passing through a low-pass anti-aliasing filter, before the signal is sampled by the ADC and sent to the DSP

Different sampling frequencies will effect the performance of the digital filtering, such as for matched filtering used for target detection. The performance is in relation to multiples of the Nyquist rate ($R_N = 2F_c$), which is two times the bandwidth, not to get confused with the Nyquist frequency ($2f_N = f_s$) [Casini et al., 1975, p. 830]. The SNR is increased considerably with increasing multiples of the Nyquist Rate (R_N) until a certain point, later the performance is no longer remarkably improving in relation to the computational complexity [Casini et al., 1975, p. 830]. A multiple of 8 is considered as a good balance between good performance and low computational complexity [Casini et al., 1975, p. 830].

2.7 Analogue to Digital Converter

The deployed ADC in this thesis is an successive approximation analogue to digital converter (SAADC). The successive approximation ADC is commonly used in medium-to-high-resolution sampling applications with a sampling frequency under 5 megasamples per second (Msps). The bit-resolution is normally between 8 - 16 bits and they have a low power consumption as well as form factor [Integrated, 2001]. The basic structure blocks of the SAADC could be seen in Figure 2.9 below.

The speed and accuracy of the SAADC is dependent on the comparator and the DAC, inside the SAADC block. The comparator must resolve small differences in V_{in} and V_{DAC} within the specified time, meanwhile the DAC must settle within the resolution of the overall converter [Integrated, 2001]. Commonly used DAC within this block is a capacitive DAC, which by default has a sample and hold function.

The SAADC will take N comparison periods to convert an analogue signal into a N-bit digital data, e.g., a 12-bit conversion will need 12 clock cycles to sample the

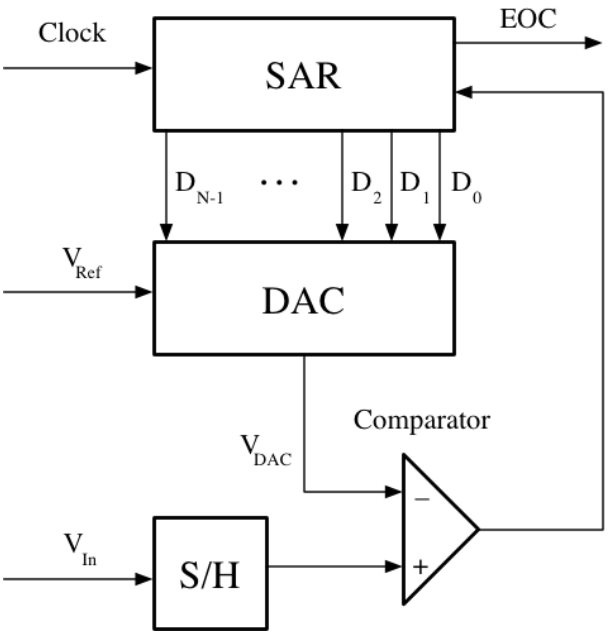

Figure 2.9 Illustrates the basic structure of successive approximation ADC [Integrated, 2001]. The SAR denotes the successive approximation register, DAC the digital-to-analogue converter, S/H is the sample and hold function, EOC signs the end of conversion, Clock is the system clock used for the ADC, V_{Ref} is the reference voltage, V_{DAC} is the output voltage of the DAC and V_{In} is the analogue input voltage [Integrated, 2001].

data. An illustrative image on how the SAR, DAC, S/H and comparator from Figure 2.9 cooperates to successively produce a digital conversion is displayed in Figure 2.11.
The result of a 12-bit ADC sample is mapped into the interval I of

$$I: \{0,1,2,3,...,2^n-2,2^n-1\}|_{n=12(bits)} = \{0,1,2,3,...,4094,4095\}$$

For simplicity, no further differences will be made for ADC's apart. As a result, the SAADC will be mentioned as ADC later on.

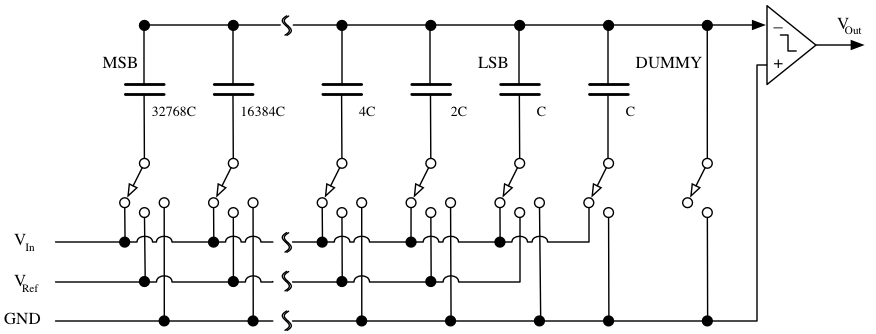

Figure 2.10 As a start position all the capacitors are connected to ground (GND) [Integrated, 2001]. The Most significant bit (MSB) is then connected to the V_{Ref} to test if $V_{In} > \frac{V_{Ref}}{2}$ holds. If this condition holds, the comparator outputs a logical 1, otherwise a logical 0. If a logical 1 was produced the capacitor stays connected to V_{Ref}. Otherwise it connects back to GND again and the next significant bit gets connected for the same condition test, until the least significant bit (LSB) has been evaluated [Integrated, 2001].

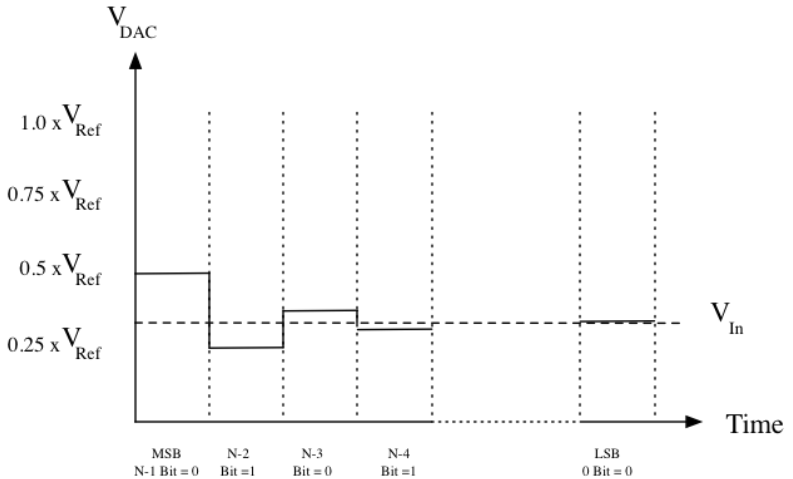

Figure 2.11 This image illustrates how the V_{in} sets an reference line for the comparison process. When $V_{DAC} > V_{in}$ the comparison bit bit results in a logical zero and when $V_{DAC} < V_{in}$ it results in a logical one. Moreover, if a logical bit is set to 1, the value will persist in being high during all of the remaining comparison periods. Thus, a bit summation of the V_{DAC} will occur and used in the iterative comparison with V_{in} [Integrated, 2001].

2.8 Additive White Gaussian Noise AWGN

Additive white Gaussian noise (AWGN) is a normal distributed noise, with uniform power spectral density, mean value μ and variance σ^2. In most cases, noise can be modeled with a Gaussian distribution e.g. shot noise in photon detectors [Liu and Lin, 2013, p. 872]. Equation 2.36 expresses a zero-mean $\mu = 0$ AWGN distribution.

$$f(x) = \mathcal{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \Big|_{\mu=0} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \tag{2.36}$$

AWGN with mean value μ_{AWGN} will add its characteristics to incoming deterministic signals. If the deterministic signal has a mean value μ_{DS}, the combined signal will have its mean-value at $\mu = \mu_{DS} + \mu_{AWGN}$, and the variance will maintain the same properties as for the AWGN [Oppenheim and Verghese, 2010, p 248]. The expected value $E(X)$ or mean value μ is calculated as following

$$E(X) = \mu = \frac{1}{N} \sum_{k=1}^{N} x_k \tag{2.37}$$

Where x_k is the sampled value for samples between 1 to N. From the expected value, the standard deviation can then be determent as

$$\sigma = \sqrt{E\left[(X - E(X))^2\right]} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (x_k - \mu)^2} \tag{2.38}$$

2.9 Digital Filters

FIR Filter

A finite-duration impulse response (FIR) filter could be realized with both analogue filter and digital filter. An input impulse response to a discrete FIR filter, will result in a nonzero output for a finite sample duration, corresponding to the length of the filter. FIR filters are usually implemented in applications where linear-phase characteristics within the passband is required [Proakis and Manolakis, 2014, p. 670]. In order to create a linear-phase FIR filter, the coefficients of the filter should be either symmetric or antisymmetric around the centered filter coefficient(s). Therefore the unit sample response of the linear-phased FIR filter satisfies the condition [Proakis and Manolakis, 2014, p. 677]

$$h(k) = \pm h(N-1-k), \;\; k = 0, 1, ..., N-1 \tag{2.39}$$

A linear-phase characteristic will result in constant delay of any input frequency within the passband, thus no phase distortion occurs. This particular characteristic is advantageous for timing applications, wave shape recognition and detect amplitude values in the time-domain for time-synchronization applications.

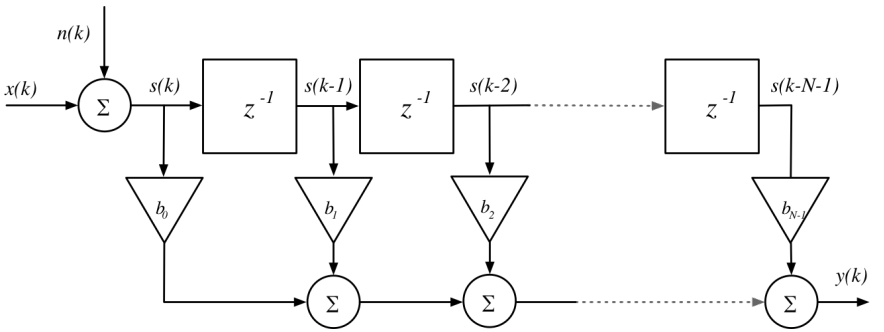

Figure 2.12 A block scheme of a direct form FIR filter were $s(k)$ is the input signal and equal to the summation of noise $n(k)$ and desired signal x(n), y(k) the output signal, z^{-1} corresponds to a delay of one time sample and $\{b_k\}$ denotes the set of FIR filter coefficients.

The output $y(k)$ from the N^{th}-order FIR filter is a summation of current sample up to the N^{th} past samples multiplied with a specified filter coefficients i.e.

$$\begin{aligned} y(k) = \; & b_0 s(k) + b_1 s(k-1) + b_2 s(k-2) + \cdots + b_{N-1} s(k-(N-1)) \\ & = \sum_{n=0}^{N-1} b_n s(k-n) \end{aligned} \tag{2.40}$$

The output sequence could also be expressed by the convolution between the unit sample response $h(k)$ of the system and the input $s(k)$

$$y(k) = \sum_{n=0}^{N-1} h(n) s(k-n) \tag{2.41}$$

The equation form in 2.40 and 2.41 are identical, thus $b_n = h(n), n = 0, 1, ..., N-1$. Since FIR filters have fixed values within a finite space, values outside this space are truncated to zero. This implies that a rectangular window is always existent for a FIR filter [Proakis and Manolakis, 2014, p. 680]. A rectangular window may implement undesirable ringing effects in the FIR frequency impulse response $H(w)$ [Proakis and Manolakis, 2014, p. 682]

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} \tag{2.42}$$

To decrease this kind of behavior, windowing is generally performed in combination with FIR filters. The *Wiener filter* is a well-known optimum linear filter. This filter optimizes the filter impulse response $h(n)$ by minimizing the mean-squared error [Proakis and Manolakis, 2014, p. 881]. The Wiener filter assumes that the system

has knowledge of the spectral characteristics of the desired signal and noise signal, which is often not the case. To mimic the optimum filter without supplying the spectral characteristics of the desired signal and noise signal, an adaptive FIR filter could be used.

Matched Filter - MF

Instead of narrowing an bandwidth to bandpass desirable frequencies with a FIR filter, a correlation filter could be used instead. Matched filter (MF) is a technique where a signal template is convoluted with the input signal to detect the desired signal. Matched filter is a standard filter technique used in radar receiver systems. Even when noise n(k) is applied over the desired signal x(k), the MF produces the maximum achievable SNR at its output $y(t)$ [Mahafza, 2013, p.143]. For proof of optimal SNR output characteristics, see [Mahafza, 2013][p. 143-145]. The optimal characteristics of a MF lies in the property of the filter impulse response $h(k)$, which corresponds to a time-reversed version of the target signal $h[k] = x(-k)$ [Oppenheim and Verghese, 2010, p. 250]. The output $y(k)$ is the summation of the *convolutions* between the impulse response $h(k)$ and the received finite-length L discrete-time input signal $s(k)$. However, since the replica signal is a time-reversed version of the target signal the MF output is the summation of the *correlations* between $s(k) * x(-k)$

$$y(k) = s(k) * h(k) = \sum_{n=-\infty}^{\infty} s(n)h(k-n), \quad where \quad h(k) = \begin{cases} x(-k), & 1 \leq k \leq L \\ 0 & , \text{ elsewhere} \end{cases} \tag{2.43}$$

$$y(k) = s(k) * h(k)|_{h(k)=x(-k)} = s(k) * x(-k) = \sum_{n=1}^{L} s(n)x(n-k)$$

If the LTI system starts and samples the first data at the discrete-time $k = 0$, the input signal $s(k)$ is detectable between $1 \leq k \leq L$. Matched filtering technique requires prior knowledge of the desired tracking signal characteristics in order to design the impulse response as the time-reversed tracking signal. In combination with an matched filter, there is generally a threshold detector to distinguish between different hypothesis decisions H_i. This will be explained in further detail, in Section 2.10.

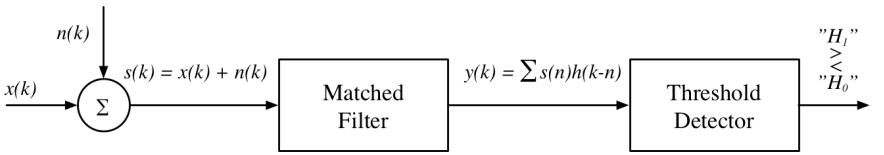

Figure 2.13 A matched filter (MF) receives the input signal $s(k)$ composed of the desired signal $x(k)$ and noise $n(k)$. The input signal $s(k)$ is convoluted with the MF impulse response $h(k)$ producing the output $y(k)$. From this output a threshold decides what information the data represents. In communication, this could be to distinguish between a binary zero and a one, hence the H_0 and H_1 at the threshold detector output.

Equation 2.41 and Equation 2.43 have similar properties, in fact with the convolution commutativity, the matched filter is equal to the FIR filter.

$$y(k) = h(k) * s(k) = \sum_{n=-\infty}^{+\infty} h(n)s(k-n), \left\{ \begin{array}{cc} \hat{n} = k-n & n = k-\hat{n} \\ \lim\limits_{n\to\infty} \hat{n} = -\infty & \lim\limits_{n\to-\infty} \hat{n} = +\infty \end{array} \right\}$$

$$\Rightarrow \sum_{\hat{n}=+\infty}^{-\infty} h(k-\hat{n})s(\hat{n}) = \sum_{\hat{n}=-\infty}^{+\infty} s(\hat{n})h(k-\hat{n}) = s(k) * h(k)$$

The direction of summation does not change the result and a scalar multiplication is commutative, thus a MF is a FIR filter. In Figure 2.12 the internal block scheme of a FIR filter is displayed, this appearance holds for the matched filter as well. In general, there is two different approaches on how to perform the convolution. The convolution could either be executed in the time domain or in the frequency domain, where the convolution in frequency domain is a simple point-wise multiplication between two signals. The fast Fourier transform (FFT) converts the sampled data into the frequency domain and allows the convolution to be performed with a computational complexity of $O(NlogN)$, where N is the data block length. A direct usage of Eq. 2.43 needs to perform L multiplications and $L-1$ additions, where L is the matched filter coefficients length. Therefore, if the sample data length block is of length N, the computational complexity will be $O(LN)$ [Kuo et al., 2013, p. 222]. For higher order matched filters, the computational complexity could be roughly of $O(N^2)$, whereas, the usage of FFT would be faster. Fast convolution techniques utilizes FFT in different ways, a method consists of circular convolution with zero-padding of the shorter sequence between the filter coefficients or the sampled data block. When data gets sampled continuously, the data block size is to big for efficiently perform circular convolution with zero-padding technique, therefore overlap-save or overlap-add methods are used on sample block lengths that is supported by the FFT algorithm.

2.10 Threshold Settings

From a binary communication perspective, a threshold could e.g. determine if the filtered signal $y(k)$ represents a binary 1 or 0. In Figure 2.13, the inequalities representations of H_0 and H_1 represents a binary hypothesis test

$$H_1 : y(k) = x_1(k) * h(k) + n(k) * h(k) = x(k) * h(k) + n(k) * h(k) \tag{2.44a}$$
$$H_0 : y(k) = \underbrace{x_0(k)}_{=0} * h(k) + n(k) * h(k) = n(k) * h(k) \tag{2.44b}$$

where H_1 denotes the hypothesis that the desired signal $x(k)$ is present and H_0 absent, meanwhile the additive noise signal $n(k)$ is always present. The impulse response $h(k)$ of the matched filter changes the spectral shape of the white noise $n(k)$, thus after the filter $n(k)$ is no longer white [Oppenheim and Verghese, 2010, p. 254]. From the hypothesis test the output yields binary 1 if hypothesis H_1 is satisfied or a binary 0 if hypothesis H_0 is satisfied. If an equality occurs, i.e., $H_0 = H_1$, then the signal is equally likely to represent a binary 0 as a binary 1 [Oppenheim and Verghese, 2010, p. 230]. Assumed that the *a priori*, i.e., prior probabilities are known and denoted as

$$p_1 = P(H_1 \text{ is true}) = P(H = H_1) = P(H_1)$$
$$p_0 = P(H_0 \text{ is true}) = P(H = H_0) = P(H_0)$$

Assuming that the additive noise $n(k)$ is independent of the desired signal $x(k)$ for each hypothesis H_i where $i = \{0, 1\}$, these conditional densities are expressed as

$$f_{Y|H}(y|H_1) = f_N(s - x_1)$$
$$f_{Y|H}(y|H_0) = f_N(s - x_0)$$

The maximum *a posteriori* (i.e., posterior) probability (MAP) rule in a combined fashion is represented by [Oppenheim and Verghese, 2010, p.230]

$$P(H_1|S = s) \underset{\text{`}H_0\text{'}}{\overset{\text{`}H_1\text{'}}{\gtrless}} P(H_0|S = s) \tag{2.45}$$

For a pulse amplitude modeling (PAM) in digital communication, this expresses the distribution of the sampled data $s(k)$ under the respective hypothesis H_0 and H_1. The desired signal amplitude x_1 denotes a high/present amplitude and x_0 a low/absent amplitude.

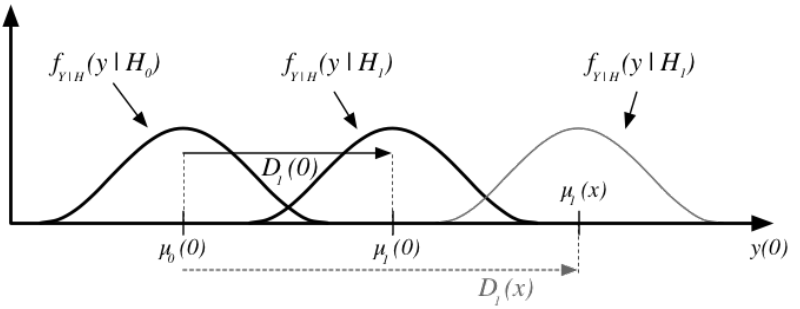

Figure 2.14 Illustration of the probability density functions (PDFs) for hypothesis H_1 and H_0. If the receiving distance is fixed, the PDF's *expected value* μ_1 for hypothesis H_1 will stay fixed. Since the tracking distance varies the *expected value* μ_1 will vary as well. The *expected value* μ_0 of the H_0 distribution will change, if the NEP of the detector is altered. Changes in temperatures is proportional with thermal or Johnson noise, which is a common consideration for changes in NEP [Optoelectronics, 2013].

The combined expression is explained as $P(H_1|Y=y) > P(H_0|Y=y) \overset{yields}{\Rightarrow} H_1$ and vice versa. The minimum probability of error P_e is set by the *likelihood ratio test* [Oppenheim and Verghese, 2010, p. 233]

$$\Lambda(s) = \frac{f_{Y|H}(y|H_1)}{f_{Y|H}(y|H_0)} \begin{matrix} \text{`}H_1\text{'} \\ > \\ < \\ \text{`}H_0\text{'} \end{matrix} \frac{p_0}{p_1} = \lambda \Rightarrow \Lambda(s) \begin{matrix} \text{`}H_1\text{'} \\ > \\ < \\ \text{`}H_0\text{'} \end{matrix} \lambda \tag{2.46}$$

where $\Lambda(r)$ denotes the likelihood ratio and λ the threshold. The threshold λ could be specified for other purposes than to minimize the probability of error P_e. It could, e.g., be set with the probability of false alarm P_{FA} and probability of detection P_D, according to the *Neyman-Pearson lemma*. The essential part of this approach is to acquire the greatest possible P_D, meanwhile ensuring that P_{FA} stays below a predefined limit. Since the *Neyman-Pearson lemma* bases its decisions on the conditional probabilities P_D and P_{FA}, the need for *a priori* probabilities on the hypotheses are avoided. By setting the threshold $\lambda = P_{FA}$, monotonically incrementation of λ will ensure optimal decision ruling for a received signal exceeding the threshold.

Probabilities: $\mathbf{P_D}, \mathbf{P_e}, \mathbf{P_{FA}}, \mathbf{P_M}$

For a binary configuration of hypothesis as in Eq. 2.44, their exist four different probabilities

$$\underbrace{P(\text{`}H_0\text{'}|H_0)}_{\text{Detection } D_0}, \quad \underbrace{P(\text{`}H_1\text{'}|H_0)}_{\text{False alarm } FA}, \quad \underbrace{P(\text{`}H_0\text{'}|H_1)}_{\text{Miss } M}, \quad \underbrace{P(\text{`}H_1\text{'}|H_1)}_{\text{Detection } D_1}$$

where 'H_i' is the "ruled" value and H_i is the actual value during that time instance. There is two situations that could arise for each detection ruling of zeros and ones

$$1 = \underbrace{P(\text{`}H_0\text{'}|H_0)}_{\text{Detection } D_0} + \underbrace{P(\text{`}H_1\text{'}|H_0)}_{\text{False alarm } FA}$$
$$1 = \underbrace{P(\text{`}H_1\text{'}|H_1)}_{\text{Detection } D_1} + \underbrace{P(\text{`}H_0\text{'}|H_1)}_{\text{Miss } M}$$

These probabilities could not be mixed together in a decision ruling, nevertheless the error terms could specify the magnitude of error in the detection process. The summation of wrongfully ruled decisions that could occur when the *a priori* probability condition is met, is called the probability of error (P_e)

$$\begin{aligned} P_e =& P(H_0, \text{`}H_1\text{'}) + P(H_1, \text{`}H_0\text{'}) \\ =& p_0 P(\text{`}H_1\text{'}|H_0) + p_1 P(\text{`}H_0\text{'}|H_1) \\ =& p_0 P_{FA} + p_1 P_M \end{aligned} \tag{2.47}$$

A minimization of P_e could be performed, to reduce errors and optimizing the decision process, if it is possible to determine *a priori* probabilities of p_0 and p_1. However, there is often too little knowledge of establishing these probabilities. Fortunately, optimizations could be performed with solely use of *a posteriori* probabilities, such as in *Neyman-Pearson approach*. The probability of detection (P_D) could be split into two subcategories, detecting a zero P_{D_0} or a one P_{D_1}. However, since a zero defines an absent signal, the probability of detection P_D expresses the detection of an present signal. These probabilities could be derived from their probability density functions

$$P_{D_i} = P(\text{`}H_i\text{'}|H_j) = \int_{D_i} f_{Y|H}(y|H_j)\, ds, \forall\ i = 0{,}1\ \&\ j = 0{,}1 \tag{2.48}$$

The probability of false alarm (P_{FA}) is a detection of a present signal in an absent signal

$$\begin{aligned} P_{FA} = 1 - P_{D_0} =& 1 - P(\text{`}H_0\text{'}|H_0) = 1 - \int_{D_0} f_{Y|H}(y|H_0)\, ds \\ =& P(\text{`}H_1\text{'}|H_0) = \int_{D_1} f_{Y|H}(y|H_0) ds \end{aligned} \tag{2.49}$$

Lastly, the probability of a miss is a non-detection of a present signal

$$\begin{aligned} P_M = 1 - P_{D_1} =& 1 - P(\text{`}H_1\text{'}|H_1) = 1 - \int_{D_1} f_{Y|H}(y|H_1)\, ds \\ =& P(\text{`}H_0\text{'}|H_1) = \int_{D_0} f_{Y|H}(y|H_1) ds \end{aligned} \tag{2.50}$$

An illustration of the relationship between P_{D_1}, P_{FA} and the threshold λ is represented in Figure 2.15.

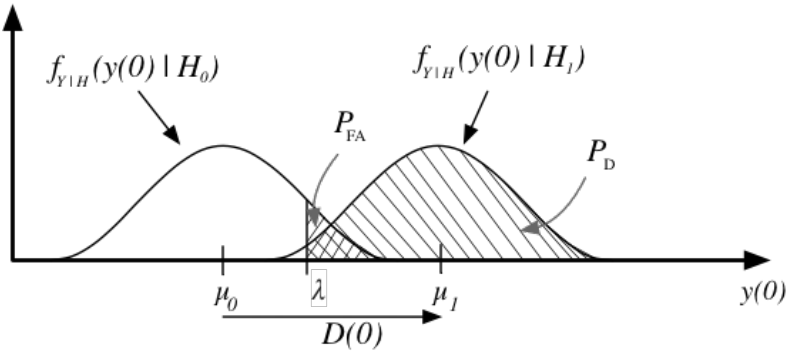

Figure 2.15 Representation on how the threshold λ set the amount of P_{FA} and P_{D_1}. The PDFs $f_{Y|H}(y(0)|H_i)$ with signal $Y = y(0)$ specifies the output of a matched filter at time instance $k = 0$, which corresponds to the respective hypothesis H_i. Vice versa holds for an illustration for P_{D_0} and P_M, whereas with an mirrored appearance in y-axis.

Receiver Operating Characteristics

The receiver operating characteristics (ROC) could be used for classifier performance assessment [Krzanowski and Hand, 2009, p. 6]. Four different joint probabilities are studied, in the case of two populations - *"Positive"* and *"Negative"* population. Even with different naming they are equivalent to the already mentioned P_{D_0},P_{D_1},P_M and P_{FA} in Section 2.10. The analyzed signal is denoted s and the threshold is donated t

$$\begin{aligned} \textit{False positive rate (fp)} = P(s > t|N) \Rightarrow & \quad [P_{FA}] \\ \textit{True positive rate (tp)} = P(s > t|P) \Rightarrow & \quad [P_{D_1}] \\ \textit{True negative rate (tn)} = P(s \leq t|N) \Rightarrow & \quad [P_{D_0}] \\ \textit{False negative rate (fn)} = P(s \leq t|P) \Rightarrow & \quad [P_M] \end{aligned}$$

ROC curves displays and summarizes the performance of the classification rule in a graphical representation [Krzanowski and Hand, 2009, p. 12]. This graph plots true positive rate P_{D_1} on the vertical axis and false positive rate P_{FA} on the horizontal axis, while the classification threshold t varies [Krzanowski and Hand, 2009, p. 12].

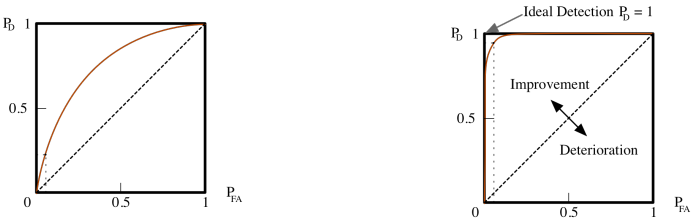

Figure 2.16 General appearance of two different ROC curves, colored brown in each ROC graph. The graph on the right side has superior performance than the graph to the left. Ideal detection performance is when the ROC curve $\rightarrow 1$ on the vertical axis P_D, and simultaneously $\rightarrow 0$ on the horizontal axis P_{FA}.

The variation of threshold value, creates the slope in Figure 2.16. For a well-defined threshold t in a ROC curve, there exist a relationship between the slope and threshold [Krzanowski and Hand, 2009, p. 22]

$$\begin{aligned}\frac{dy(t)}{dx(t)} &= \frac{d(P_D)(t)}{d(P_{FA})(t)} = \frac{d\left(P(S>t|P)\right)}{dt} \cdot \frac{1}{\frac{\left(P(S>t|N)\right)}{dt}} \\ &= \frac{\frac{d}{dt}\left(1-\int_{-\infty}^{t} P(s|P)ds\right)}{\frac{d}{dt}\left(1-\int_{-\infty}^{t} P(s|P)ds\right)} = \frac{-P(t|P)}{-P(t|N)} = \frac{P(t|P)}{P(t|N)}\end{aligned} \tag{2.51}$$

2.11 SNR Gain

To determine improvement of before and after digital signal processing, the method of SNR gain is used. The measure on detection performance based on SNR gain is a well known technique, since the detection power is a monotonically changing with the SNR of $y(k)$ [Martinez and Thomas, 1986, p. 252]. This simplification is justified by assuming that the noise $n(k)$ is Gaussian or nearly Gaussian distributed, so that a linear restriction on $y(k)$ holds. By making these assumptions and proven them to be good estimates, the MF should perform as intended. This SNR gain method is a ratio of improvement, where values greater than 1 is improvement of signal detection, values less than 1 is deterioration of signal detection and no improvements when it is equal to 1. In the decibel case, the last sentence holds but the limit is not equal to 1, instead $log_{10}(1) = 0$ is the comparison value.

$$SNR_{gain} = \frac{SNR_{after}}{SNR_{before}} \tag{2.52}$$

$$= SNR_{after} - SNR_{before}[dB] \tag{2.53}$$

if the signals is converted to *decibels* (dB), then Equation 2.53 could be used.

3

Methodology

A *matched filter* was selected as the DSP algorithm for increasing the detectivity of the incoming signal, since it is the linear filter with maximum instantaneous SNR output for a given signal in noise [Martinez and Thomas, 1986, p.253]. With a symmetric or antisymmetric impulse response, the MF is linearly phased, meaning that no phase distortion occurs and all frequencies within the passband filter has a constant delay [Proakis and Manolakis, 2014, p. 677]. Linear approaches could be made after MF to detect signal characteristics such as frequency and timing of the incoming signal $x(k)$, as long as the noise stays relatively Gaussian distributed.

With a measurement of *SNR gain* and a *ROC plot* for the tracking distance interval, a good measurement on the detection and filtering performance could be determined. The estimated desired signal strength varies with distance and noise levels varies with incoming sunlight, thus two different approaches are used in this thesis to dealt with this issues, in both cases a threshold is determined. One approach was to use the *Neyman-Pearson lemma* where a threshold is designed based on the *probability of false detection* P_{FA}, since keeping this value low is prioritized before *probability of detection* P_{D_1}.

An standard adjective is to minimize the *probability of error* P_e by using the *likelihood ratio test*, which sounds like the optimized detection ruling. Both sides of the hypothesis H_0 and H_1 is included, such that *probability of miss* P_M and *probability of false detection* P_{FA} is unfavorable. Thus the threshold could be set lower than desired, in order to decrease the *probability of miss* P_M by the cost of increasing the P_{FA}. Therefore, the *Neyman-Pearson lemma* was used, where the relation of P_{FA} is set as desired and the P_{D_1} will then be given.

Since the noise characteristics varies with different illumination, adaptive algorithms were developed to handle these changes. One method utilizes the noise distribution characteristic in order to to set a minimum values for threshold λ according to a mapped P_{FA}. A lot of noise data were gathered in different environmental conditions to be able to estimate the noise characteristics, so the threshold could be designed. The gathering of the noise data was performed when the sampling process of the embedded system was working properly. In order to detect if the sampling worked properly, a continuous rectangular wave was generated with a

signal generator and sampled. This procedure was tested for different repetition frequencies. The embedded system used an *ARM Cortex M4* to sample data, DSP data, communication interfaces and to control sensors.

ARM Cortex M4 has several DMA and SAADC inside the unit, which was part of the solution to sample data from several input channels ”simultaneously”. M4 Cortex has a 168 MHz CPU, up to 1 Mbyte flash memory, 192+4 Kbyte SRAM, and for 3 SAADC with a maximum resolution of 12-bits, more information about the system could be found at ARMs and STMs website [ARM, 2018][STM, 2018]. The embedded implementation and system designed will not be part of the thesis, the results however, will be analyzed and discussed. MATLAB and Simulink were used for several tasks including:

- Analyze noise characteristics, with distribution fitting.
- Study tracking signal intensity at different distances, calculating mean values.
- Study tracking signal characteristics such as pulse shape, duty cycle, pulse repetition frequency.
- Recreate and simulate realistic replica of real data.
- Create a model of the embedded system in Simulink, with filtering, detection and analysis tools.
- Design non-adaptive threshold for embedded signal detection.
- Estimate detection distance limitations.
- Serial communication, in order to receive raw and processed data from the embedded system, for debugging the embedded C code.
- Post process data from embedded system to study the similarity of the DSP simulations and real-time DSP.
- Design new adaptive thresholds for signal detection
- Design new algorithm for matched filtering on short duty cycle pulses with several pulse repetitions at fixed pulse repetition frequencies.
- Analyzing the accuracy and deviation of the internal clock frequencies on each side of the tracking system, in order to determine the reliability of repeatability.
- Studying the accuracy of distance estimation and direction estimation, to determine the tracking position.
- Studying theoretical and real SNR gain results from MF, with different filter lengths so that several pulse periods could be included.

- Studying the differences in SNR gain with changing sampling ratio.

Interesting characteristics and results could be made from a lot of offline analysis, which reduced the errors in design of the embedded system. The algorithm debugging was made in MATLAB, where visual output and calculations could verify the algorithm performance. The embedded code was programmed and debugged in ARM KIEL $\mu Vision$ IDE, where CMSIS-RTOS was the main implemented real-time operating system. The real-time embedded system performance was analyzed in MATLAB, by receiving raw as well as processed data from the Cortex M4 chip through serial UART communication. The processed data was compared with the simulated data and the raw data were post processed in MATLAB to compare the results with the embedded solution.

4

Modeling

4.1 Tracking System Model

The optical tracking communication consists of one stationary tracking device that transmits a rectangular pulse train with constant signal strength and one moving receiver. The receiver interprets the received signal, synchronizes to the phase and frequency of the incoming signal to respond to the transmitting system with the same signal characteristics. The receiving part of the receiver is illustrated in Figure 4.1.

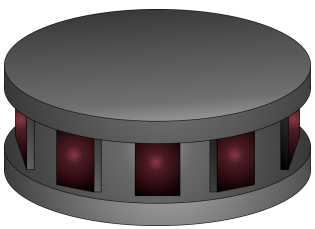

Figure 4.1 A 360 degree NIR signal detection system, with overlapping detection view from the M numbers of photodiodes placed around in a cylindrical shape. The red flat surfaces symbolizes the photodiodes.

Since the receiver is moving during transmission a doppler effect occurs. The moving speed of the receiver is limited to walking speed, which from calculation results in a neglectable frequency change in the system receiving characteristics, explained in Appendix A.1. Since the doppler effect is neglectable, the distance between the transmitter and the receiver D could be mapped quite accurately into a received signal strength.

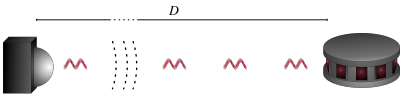

Figure 4.2 The image represents a NIR *tracking system* (NIR-LED), which transmits NIR signals to the NIR *tracking detection system*. The distance *D* between the tracking system and the tracking detection system, sets the intensity of the registered signal at the tracking detection system.

The *tracker detection system* has *M* photodiodes for detecting signals an *M* NIR-LED for responding to the *tracker system*. Each photodiode has an optical bandpass filter (BF) fitted for NIR-signals of 850 nm. The generated analogue signal from the photodiode is filtered with an analogue low-pass anti-aliasing filter (AAF) and converted (ADC) into a digital signal in the microcontroller. Each sampling is performed at a fixed frequency, all *M* input channels are scanned successively from the ADC and transferred into a memory location by the direct memory access (DMA). A matched filter (MF) reads from the current memory location and filters the digital input signal. To detect an incoming signal from the *M* input channels, the filtered signal is observed by an signal detector. The detector sets an threshold for detecting incoming signals and informs the controller if a signal was registered above the threshold, and which channels are receiving the desired tracking signal. The controller then outputs a responding NIR signal with power proportional to the distance *D* towards the *tracking system* position, determined by *distance* and *direction*.

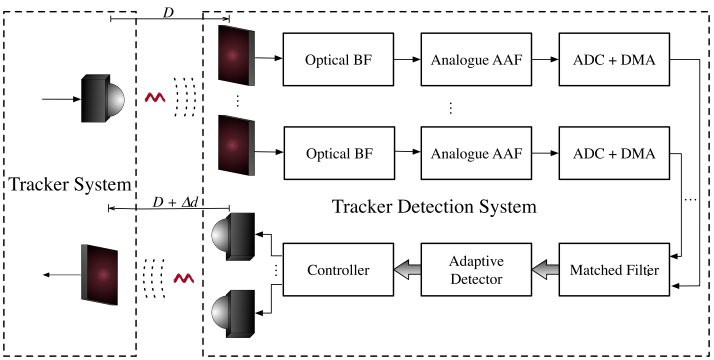

Figure 4.3 The different parts in the tracker detection system displayed in a block scheme overview. The *M* different inputs are first pre-filtered and sampled separately before they are analyzed in the DSP together. The matched filter increases the desired signal before the detector determines if a signal was received, and if so, determine its position.

Distance

The transmitting LEDs are modeled as spherical light sources, which emit a NIR signal everywhere in a three-dimensional space. The intensity of the NIR signal degrades proportionally with the inverse square law, resulting in a specific intensity detectable at distance D for the photodetector. There is a physical limitation on how small the intensity can be before it can not be detected by the photodetector. This is determined by the NEP (2.2), thus when the power of the NIR signal $P_{yy}(k)$ is less or equal to the $P_{yy}(k) \leq NEP\sqrt{\Delta f}$ the signal will not be detectable.

$$I_{PD} = \frac{P_{LED}}{A} = \frac{P_{LED}}{4\pi r^2} \Rightarrow r = \pm\sqrt{\frac{P_{LED}}{I_{PD}4\pi}} \tag{4.1}$$

where I_{PD} is the intensity registered at the photodiode, P_{LED} is the initial transmitting power at the led and r is the distance between the led and the photodiode. By mapping one or more signal amplitudes to a specified distance, the other distances could be calculated with the equation above as.

$$\begin{aligned}
\Delta I =& I_{ref} - I_{new} = \frac{P}{4\pi r_{ref}^2} - \frac{P}{4\pi r_{new}^2} \Rightarrow \frac{P}{4\pi r_{new}^2} = \left(\frac{P}{4\pi r_{ref}^2 - \Delta I}\right) \\
r_{new} =& \pm\sqrt{\frac{P}{4\pi\left(\frac{P}{4\pi r_{ref}^2} - (I_{ref} - I_{new})\right)}} \\
r_{new} =& \pm\sqrt{\frac{C_1}{C_2 + C_3 \cdot I_{new}}} \\
C_1 =& P,\ C_2 = \frac{P}{r_{ref}^2} - 4\pi I_{ref},\ C_3 = 4\pi
\end{aligned} \tag{4.2}$$

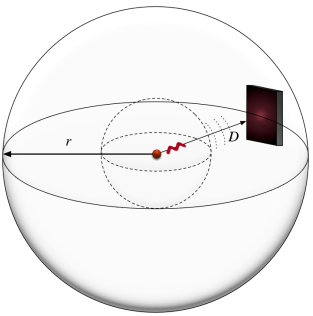

Figure 4.4 A spherical representation of the LED emitting model, where the power of the signal is equally distributed over the spherical mantle area $A = 4\pi r^2$.

The Equation 4.2 determines the distance by using the known value of the LED power P, predefined reference values pair of I_{ref} and r_{ref}. By predefining these constants the computation time will decrease. The distance is calculated with the photodetector that registers the maximum intensity.

Direction

The incoming angle of the NIR signal will effect the registered intensity at the photodetectors, illustrated in Fig. 4.5.

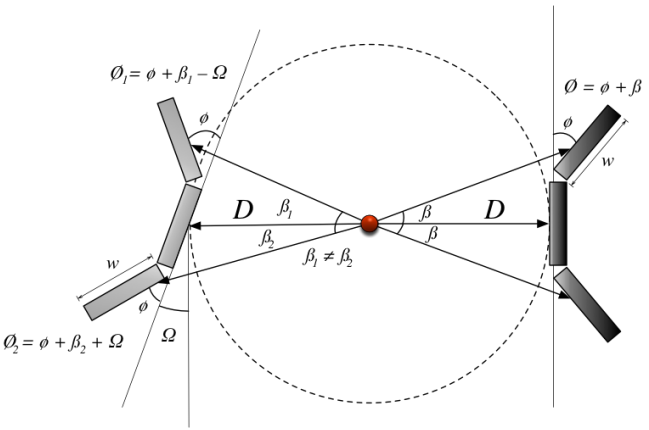

Figure 4.5 The image illustrates an IR-emitting point source in the center of the dotted circle. The three rectangles on each side represents photodetectors in different incoming angles. The centered photodetector on the right-hand-side receives the IR signal along the normal to the surface of the detector, this will further on be recognized as 0° deviation of the incoming angle, resulting in a calculation of the "correct" tracking distance. With an increased incoming angular deviation Ω, the calculated distance error will increase. More on this, see Appendix A.2.

By performing a polynomial fit with values from the photodiode data sheet, an equation could be established to calculate the direction of the incoming signal. In Fig. 4.6, the polynomial fit is plotted. It is fairly straight forward to calculate the relative radiant sensitivity from a measured angle, this could be performed with the 4th-order polynomial

$$I_R(\theta) = 4.946 \cdot 10^{-8}\theta^4 - 7.4704 \cdot 10^{-6}\theta^3 + 1.8930 \cdot 10^{-4}\theta^2 - 2.1 \cdot 10^{-3}\theta + 1.0 \quad (4.3)$$

However, the opposite is quite complicated, to extract an angle from a known relative radiant sensitivity. There is no good polynomial fit for the inverse function, instead a quadratic spline interpolation is proposed to fit polynomial segments to the data

values. The quadratic spline fitting is observed in Fig. 4.6.b. and the equation is described in 4.4.

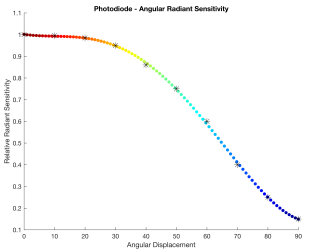

Figure 4.6.a

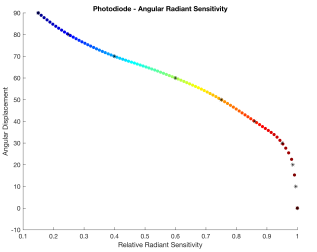

Figure 4.6.b

Figure 4.6 The graph in Fig. 4.6.a displays the relative radiant sensitivity as a function of angular displacement. In order to determine the the angular displacement the inverse function of the parable in Fig. 4.6.a could be used. The appearance of the inverse function is displayed in Fig. 4.6.b., where the parable is determined with quadratic spline interpolation of defined specification data values.

The spline interpolation is in comparison to Eq. 4.3 quite complex. Despite that the polynomial order of each spline segment is lower, therefore less computational heavy, but the measured intensity must undergo some logical test before calculation.

$$\theta(I) = \begin{cases} 171.3708(I-0.15)^2 - 117.1371(I-0.15) + 90 & ,0.15 \leq I \leq 0.2688 \\ 88,5682(I-0.2688)^2 - 76,4365(I-0.2688) + 78,5066 & ,0.2688 \leq I \leq 0.45 \\ -40,5542(I-0.45)^2 - 44,3305(I-0.45) + 67,5621 & ,0.45 \leq I \leq 0.6563 \\ -110,3008(I-0.6563)^2 - 61,0592(I-0.6563) + 56,6937 & ,0.6563 \leq I \leq 0.8050 \\ -2,4980(I-0.8050)^2 - 93,8737(I-0.8050) + 45,1706 & ,0.8050 \leq I \leq 0.9163 \\ -1324,99(I-0.9163)^2 - 94,4295(I-0.9163) + 34,6962 & ,0.9163 \leq I \leq 0.9763 \\ -26684,0(I-0.9763)^2 - 253,4288(I-0.9763) + 24,2605 & ,0.9763 \leq I \leq 0.9938 \\ -108350,7(I-0.9938)^2 - 1187,37(I-0.9938) + 11,6535 & ,0.9938 \leq I \leq 1.00 \end{cases} \tag{4.4}$$

Cubical splines were investigated, but no noticeable improvement in performance was found. In order to determine the direction, sampled data from three diodes are used. First all diodes were scanned to identify which photodiode has the maximum detected signal. This diode and its neighboring two diodes will undergo a relation calculation, where the maximum registered signal will be the reference as $I_{max} = 1$. The two neighboring photodetectors will then have their values relatively to the new I_{max} inserted in the quadratic splines equation. From Figure 4.5 the left hand side will register a stronger signal on two of its channels and a weaker on its third, a direction calculation is needed. On the right hand side, there will be a clear maximum detection channel, but the two others will have approximately the same registered signal intensity, thus the incoming signal is straight in front of the channel that detected the maximum value. Hence, no need to perform any more calculations.

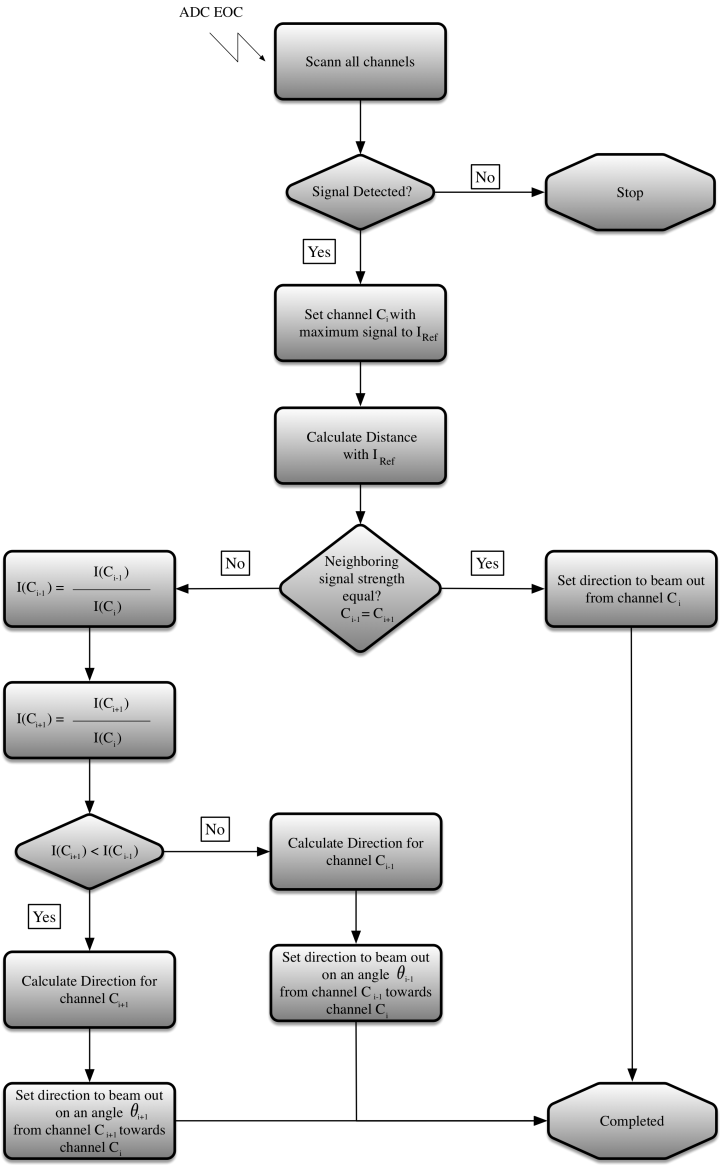

Figure 4.7 Flowchart of the direction and distance calculation. First the distance is calculated with the strongest detected intensity I_{Ref}. Then the direction is calculated either by observing that the neighboring intensity is equal within a tolerance, otherwise by reading out the angle of the weakest photodiode relative to the maximum registered intensity I_{Ref}, see Fig. 4.6 for incoming angles relative to resisted intensity.

From the flowchart in Fig. 4.7, the calculation of the angle is determined by the weakest registered intensity not equal to zero. By selecting the weakest detected neighboring intensity, the angular estimation will be more accurate due to the rough linear appearance of angle deviation from the normal to the detection surface, between 30 - 80 ° (Fig. 4.6). The weakest intensity used in these calculation is greater than 40° and less or equal to 60°, calculations outside this interval will never occur, since the used photodiodes will then be change to diodes whiting this interval.

4.2 Signal Model

Signal Disturbance

To characterize the noise distribution of the background noise, some samples without an active tracking signal was sampled. Different distributions was then matched with the registered samples to find the best fit to the sampled data distribution. A normal distribution was found to be a good accurate approximation. Thus the disturbance in the noise model was selected to consist of AWGN with changing parameters along with changing operating environments. Two different examples are illustrated in Fig. 4.8.

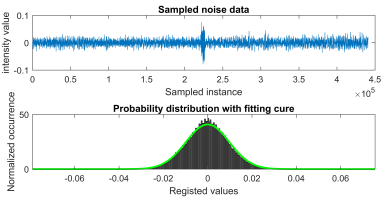

Figure 4.8.a

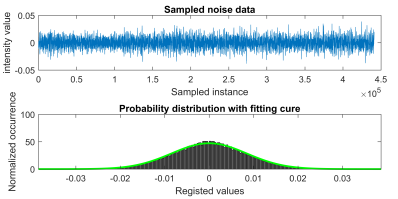

Figure 4.8.b

Figure 4.8 The above subfigure plotted in blue, are the sampled data from background noise radiation. The data was sampled outdoor, with the photodiodes facing directly towards the sun, in fully sunny condition with registered light intensity of 120 000 Lux. The bottom subfigure displays the a histogram of the sampled data in black color, with the green line plotting the normal distribution fit. In Figure 4.8.a a greater disturbance is detected in the middle of the sampled data.

Similar data was sampled in different light conditions as in Fig. 4.8, where the only noticeable difference was the amplitude of the noise signal. The distribution of the signal only changed when other *continuous interferences* was introduced to the system.

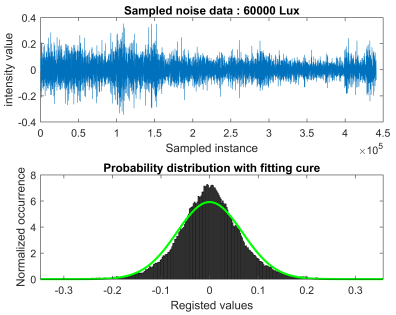

Figure 4.9.a

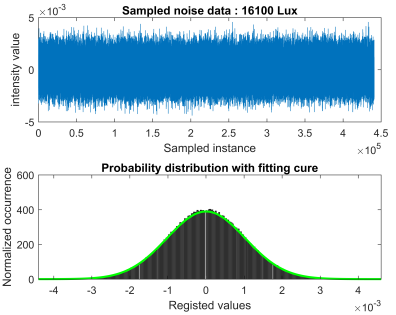

Figure 4.9.b

Figure 4.9 When sampling data in various sunlight conditions, multifarious distribution characteristics is observed. Figure 4.9.a is sampled at 60 kLux and 4.9.b 16.1 kLux. Data sampled at 60 kLux varies in amplitude with changing cloud density during the data collection. The 60 kLux value is the average value during the data collection, thus some extreme points are visible.

By defining a well-conformed noise distribution of the system, in our case a normal distribution, the threshold of the detection system could be set more accurate. However, with outdoor operation the light conditions changes and so does the shape and specifics of the normal distribution describing the background noise.

Table 4.1 Noise characteristics

Lux Value	μ	σ
1000	$3.28 \cdot 10^{-8}$	$9.37 \cdot 10^{-4}$
5000	$1.37 \cdot 10^{-6}$	$9.13 \cdot 10^{-4}$
16100	$2.47 \cdot 10^{-6}$	$1.03 \cdot 10^{-3}$
60000	$-2.60 \cdot 10^{-6}$	$3.11 \cdot 10^{-3}$
120000	$1.44 \cdot 10^{-5}$	$1.08 \cdot 10^{-2}$

Table 4.1 displays the normal probability distribution function characteristics from sampled noise data. From these average values, some thresholds could be purposed. Although, these values are from data where no other interferences occurs. For other types of disturbances, the mean value and standard deviation will change. By comparing Fig. 4.9.a and Fig. 4.9.b, the effect of other interferences are visible. These interferences could occur in areas where multiple reflective surfaces exist and other similar events. Figure 4.9.a with *60 kLux* registered a standard deviation σ of $6.74 \cdot 10^{-2}$, which is roughly 6 times greater than the double illuminance value of *120 kLux*. This sample is therefore not representative for the 60 kLux noise characteristics and not used for extracting the values in Table 4.1. However, Fig. 4.9.a

demonstrates how the noise characteristics changes rapidly with the operational environment over time.

Tracking Signal

Tracking signals with a low duty cycle has been selected for this master thesis, since a low duty cycle signal is more difficult to detect and synchronize to in real-time. The rectangular pulse train makes it difficult to use convenient methods for detecting the signal, such as fast Fourier transform (FFT), more on this in Sec. 4.3.

Since the different tracking signals are known prior to arrival, different searching modes are activated in the beginning, to detect which tracking pulse is active. In practice this search mode consists of various MF templates that searches for their designed pulse train. Different pulse train templates will be tested and ultimately looped around if no signal is jet detected. A fine tuning stage is activated when a signal is detected, to make sure that small deviations in the internal clock system on both side of the tracking process is corrected for. In order to display the system, a predefined pulse characteristics is used, since searching for and synchronizing to different pulses are not part of the thesis report.

The predefined tracking pulse is a rectangular pulse train, with a 0.37% duty cycle, meaning that it is mostly nonactive during a pulse period. The selected frequency was 33.89 Hz with a pulse width of only 108 μs long. This means that the detection of the pulses must be accurate in relative to the discrete time frame, in order to transmit a synchronized response. The time synchronized response signal is a replica of the incoming tracking signal, delayed with one pulse instance, where rising and falling edges of the received and transmitted pulses align in time. Focusing now on the stated tracking pulse characteristics for the simulation and real-time operation of the system:

Table 4.2 Pulse characteristics

Pulse physics	850 nm NIR signal
Pulse shape	Rectangular
Pulse repetition frequency	33.89 Hz
Pulse width	108 μs
Pulse duty cycle	0.37 %

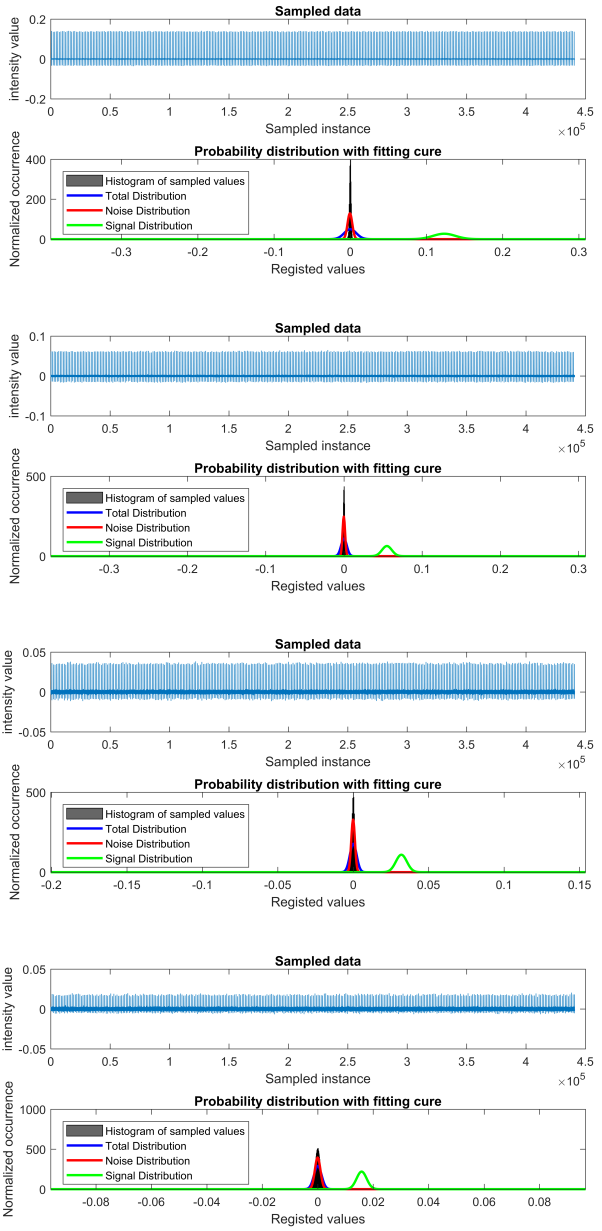

Figure 4.10 Tracking signals received at different distances. The signals strength weakens with distance and the total sampled data distribution mutates towards the noise data distribution. **Observe changes in axis scale**, the zoomed in view is needed to observe the reshaping of the total distribution.

4.3 Matched Filter & Cascade Matched Filter

As explained in Section 2.9, the computational efficiency of the algorithm could be improved in several different ways. There are some problems with the different methods, that needs to be stated. As observed in Fig. 2.2, the energy is relocated to the higher frequencies for shorter pulse widths. This will lead to frequency characterization issues of detected tracking signal. This issue is visualized in Fig. 4.11. The FFT of a rectangular pulse train with low duty cycle, will result in detection of various harmonics and the energy will be shifted to the higher frequencies, resulting in difficulties of tracking the pulse repetition frequency of the signal. This issue impacts the synchronization of the tracking detection system.

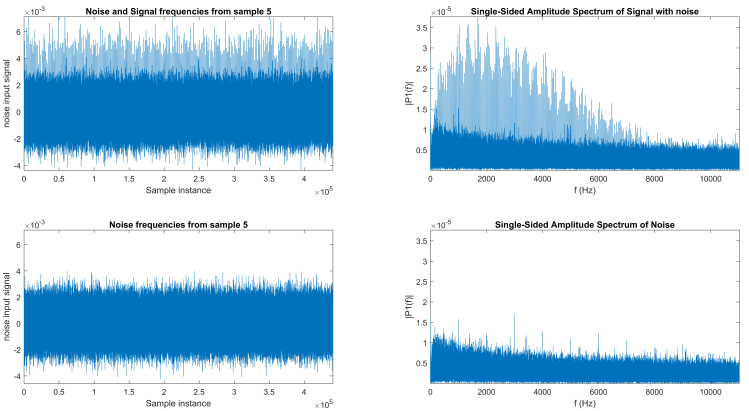

Figure 4.11 On the left hand side is raw sampled data and on the right hand side is single-sided fast Fourier transform of the raw sampled data. Atop is a signal inside of noise and below is sampled noise without a signal at 5000 Lux. The FFT detects several harmonics from the rectangular pulse of 108 μs, the energy is shifted towards the higher frequencies and the pulse repetition frequency of 33.89 Hz is not visible.

In order to utilize the characteristics of the rectangular pulse train that is being continuously transmitted to the tracking detection system, a simplified method in comparison with the FFT, is implemented. To improve the computational complexity, the matched filter coefficients are set to the ideal case of a rectangular shape. Meaning the MF coefficient's length representing one pulse, are set by the number of samples that will be detectable of the rectangular pulse, with the predefined sample rate. These coefficients are set to 1, whereas the Eq. 2.43 will reduce to a simple summation of L sampled data. This optimizes the computational complexity to $O(L)$, where L is the length of the MF template. The detectability could be increased by adding the energy of the rectangular pulse train together. This means that single

pulses from different pulse periods will be used together for cascade matched filtering (CMF). Even when more pulses periods are used for filtering, the main agenda is to detect the phase of the signal in the current pulse period. This is also one of the reasons for optimizing the MF algorithm.

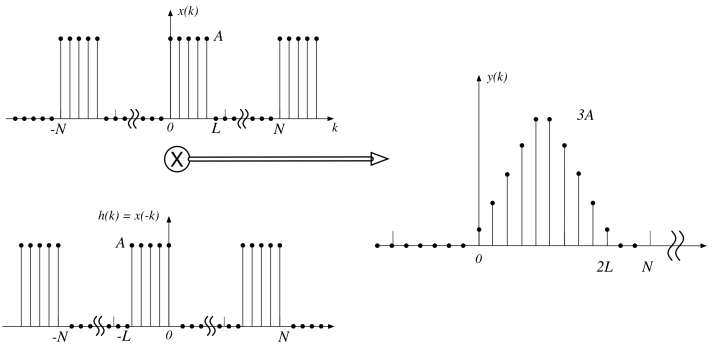

Figure 4.12 The above figure represents a traditional convolution between the desired signal $x(k)$, and the template of three pulse periods, with each pulse of length L. The energy from each pulse is added up to the final CMF output $y(k)$, which is three times greater than the amplitude from each individual pulse.

With a low duty cycle, a rectangular pulse train template will mostly consist of zeros. In order to not extend the computational complexity massively, by multiplying and adding zeros in the system, a smarter algorithm is used, namely cascade matched filtering (CMF). This smart algorithm is adaptable to pulse length, frequency and even number of pulse periods. The CMF algorithm will decrease the usage of memory storage, allowing more data to be sampled and stored in the system. Normally a signal replica is declared in the system storage, allocating memory space. Then by multiplying the MF template with the sampled data and summarizing every time instance together, the MF output $y(k)$ is calculated. If several pulse periods would be used for this method, as for the CMF, the need for data storage would increase and computational complexity would increase massively.

Using pointers in a smart way and setting the MF coefficients to 1, allows the algorithm to only use pointers to map sampled data instances that shall be summarized together. Furthermore, these pointers represents the CMF template but will only increase with the length of a pulse width L, for each pulse period M the algorithm is extended with. Normally a MF would increase to the size of $N \cdot M$, where M is the number of used pulse periods and N is the number of sample instances that represents one complete pulse period.

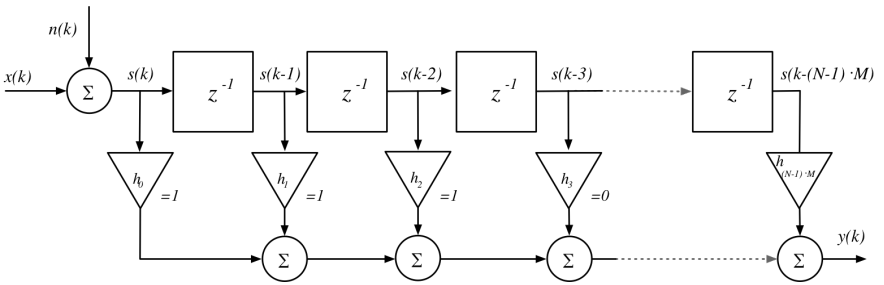

Figure 4.13 A standard MF illustration in a block scheme, where the filter coefficients length is $N \cdot M$, were N is the filter length to filter a complete sampled pulse period, and M is the number of pulse periods that is included in the filtering. Sampled signal instances are denoted by $s(k)$, the impulse response coefficients h_k, output signal $y(k)$ and the delay of one discrete time unit is denoted with z^1. In this example a pulse width L of 3 discrete samples where selected, thus $h_0 = h_1 = h_2 = 1$ and $h_3 = ... = h_{N-1}$. What is not visible is that this sequence will repeat itself for M times, where M is the amount of pulse periods that one desires for filtering with a CMF approach.

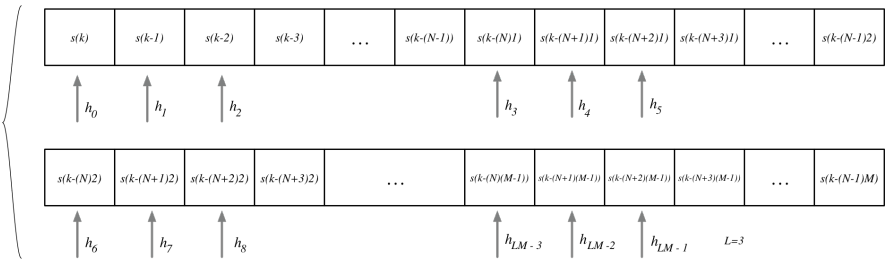

Figure 4.14 The CMF filter has the length of $L = 3$ and M pulse periods are used. Figure 4.13 represents the normal MF, where the filter coefficients length is $N \cdot M$, and the smart filter length in this case has a filter coefficient length of $L \cdot M = 3M$. This reduces memory and also the computational complexity of the CMF, since $L \leq N$, and in this thesis $L << N$ because of the low duty cycle. Sampled signal instances is denoted $s(k)$ and the impulse response coefficients h_k are instead pointers to the elements that should be summed together.

The example in Figure 4.14 of the CMF results in the following computation

$$y(k) = s(k) + s(k-1) + s(k-2) + ... + s(k-(N)(M-1)) + s(k-(N+1)(M-1)) + s(k-(N+2)(M-1))$$

This implementation also means that the frequency, duty cycle and number of pulse periods used for CMF could be changed during real-time operation. The filtering

process includes shifting of the pointers on step to an older sample, until the consecutive pointers has moved to the end of one pulse period. Hence a CMF template with, e.g., 10 pulse periods would always produce a filtered signal within one pulse period, no matter how many pulse periods are used for filtering. Meaning that the M amount of pulse periods are always mapped into 1 pulse period, this reduces the computational time additionally. Instead of performing (NML) multiplications and $M(N-1)(L-1)$ additions, the CMF will only perform $M(N-1)(L-1)$ additions. In this thesis a CMF approach will be used instead of a MF approach. There is not much of a difference between the CMF and MF when filtering pulses of one pulse period. The deviation starts when CMF and MF filters more than one pulse period at the time.

4.4 Non-adaptive Threshold

Different photodetectors/input "channels" will be receiving a signal with either only noise or noise with the desired signal. The detector stage determines which channel has received a signal or not, by comparing registered signal values to a threshold, set by the detector. By defining λ in relation to selected P_{FA}, a fixed threshold could be determined. As seen from Section 4.2, the noise varies in time, when exposed to a time-varying transmitting and receiving medium space, such as an outdoor environment. In order to guarantee a good performing system with use of a fixed threshold, the value of the threshold λ needs to be set as high as the worst case noise environment. The procedure ensures that operation requirements are met even in worst case scenarios, however, this entails that operations are only as good as the worst case.

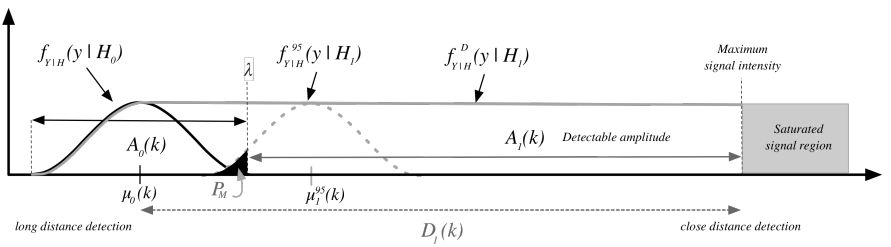

Figure 4.15 The incoming Gaussian distributed signal $x(k)$ is detected if $x(k) > \lambda$. Threshold λ is designed from the noise characteristics, so that the P_{FA} is as specified. The probability of detectability for different distances could then be determined, where signals with distribution outside the threshold will have 100% detectability. For example, transmitted signals $x(k)$ at a certain distance D, which produce a mean value such that $x(k) = \mu_1^{95}(k)$, will have a $P_{D_1} = 0.95$ and $P_M = 0.05$. The amplitude of the detected signal will be saturated at one point, for signal detection this is not a problem, but for determining distance up close this is an issue.

From Table 4.1, different values of standard deviation σ was declared. To address the problem of setting a threshold value λ, the *Neyman-Pearson lemma* was selected. A P_{FA} value of $1-3\sigma = 100\% - 99.73\% = 0.27\%$ was selected for 120 kLux to ensure that a low *False Alarm Probability* was maintained for various types of operating environments. The probability of false alarm P_{FA} is prioritized above probability of detection P_D, so that the probability of false alarm is as low as possible. A synchronization of the pulse is performed continuously, so if no data is detected for a short time, the tracking detection system will still be able to transmit a synchronized pulse train response. The tracking system will receive the synchronized pulse train as usual, unaware of the fact that the tracking detection system is not receiving any signal.

On an outdoor construction site, there will be a lot of blocking interferences, such as people crossing in-front of the system. If the system would synchronize to a "false" detected pulse, the tracking system would not receive the pulse train information, since the exposure/receiving time of the tracking system is matched with its own synchronized transmitting pulse.

4.5 Adaptive Threshold

Noise environments change greatly for outdoor performance, thus utilizing an adaptive threshold could increase the tracking detection performance. On a clear sunny day, the received illuminance will be different from all directions. If a photodetector faces the sun directly, it could receive values of 120 kLux, but the opposite side could receive, e.g., a 10th of that value. If the threshold was set to handle the worst case of 120 kLux, then a lot of detection possibilities on long distances would be missed, if the receiving signal was traveling towards the sun. An adaptive filter could handle changes in signal receiving direction, light conditions and interferences.

The different adaptive thresholds in this thesis is designed from scratch, and utilizes all of the mentioned fact that was stated between the noise signal and the desired signal. The adaptive filter could be adjusted to desired performance by tweaking some parameters, like a PID-controller. In Fig. 4.16, a general view of the detector is illustrated. All of the adaptive algorithms uses post data to determine the threshold characteristics.

The adaptive filters utilizes the memory storage of the output data from the filter, there for it does not use any more memory, but the adaptive filter needs computational time. The main purpose is to analyze the noise characteristic in order to set a threshold, which guarantees at least a predefined P_{FA}. The adaptive filter will try to make the system more robust to future sudden changes in the noise characteristics, hence setting the threshold higher than needed for the predefined P_{FA}. The extra elevation of the threshold is set with assistance of the maximum detected value on the channel. The minimum level of threshold is set by the P_{FA} with a moving standard deviation calculation or a moving maximum for the second adaptive approach.

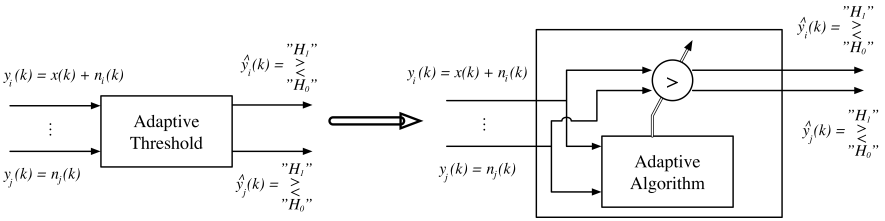

Figure 4.16 Illustration of the to mayor subcomponents for the adaptive threshold. A logical operator the adaptive algorithm. The logical operator operates on the threshold level and simply scrutinizes the channels for a tracking signal. The adaptive algorithm receive information from other channels and stores past sampled data, to adapt to changes in the noise characteristics.

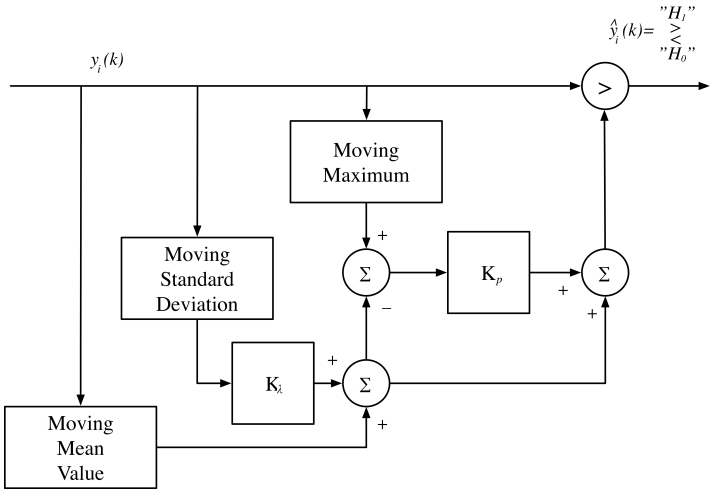

Figure 4.17 The adaptive threshold only observes its own channel, where the moving maximum (MM) is determined, the mean value and the moving standard deviation (MSD). The MM sets the above limit while the MSD is multiplied with a constant to set the P_{FA} as desired. Then the value set by the $K_\lambda MSD + MMV$ is subtracted from the MM to define the region, where only signal should exist. This value is then multiplied with a constant, that could be seen as a percentage of where the threshold should "levitate" in the pure signal region. The value created by the $K_\lambda MSD$, will be added in the end, to offset the "signal region" right.

A constant K_λ set for the first approach the probability of false alarm level by $P_{FA} = 1 - K_\lambda \cdot \sigma$. This value will be added in the last step of the algorithm but also subtracted from the maximum detected amplitude, this value symbolizes the pure

signals region between the lowest threshold level to the highest. By then multiplying this pure signal region with a constant K_p, a elevation of the threshold could be defined. This value is then added in the last step, to set the adapted threshold level.

The adaptive filter approach in Fig. 4.17, works best for signals with low duty cycles and noise distribution close to normal distribution. If the sampled raw data is subtracted with the moving mean value already before filtering, the moving mean value in the adaptive threshold is not needed.

In environment where the noise characteristics changes suddenly, a value of $K_\lambda \approx 0.7$ is tested to give good results. The other approach in Fig. 4.18, calculates the moving maximum on all channels. Then two different channels are compared to set a threshold. The relationship between the comparing channels are specified from the angular intensity seen in Fig. 4.6, where $channel_j = channel_{i+2}$. If then the maximum signal amplitude is detected on $channel_i$ the signal should be barely visible on $channel_j$, thus setting the threshold slightly higher than the maximum noise level. Since the pulse repetition repetition frequency is set to 33.89 Hz it is very likely that next signal will be detected on this channel or one of its neighbor, therefore these channels are inspected first.

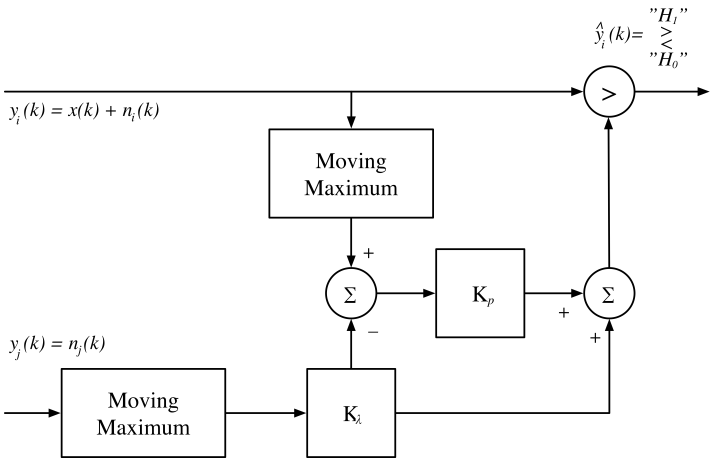

Figure 4.18 This adaptive filter works exactly like the adaptive threshold from figure 4.17. However, the estimation of the noise characteristics is made from another channel. The other filtered channel $y_j(k)$ could sample raw noise data to make the estimate of the noise characteristics better. This method will also work good for signals with greater duty cycle, since increasing the duty cycle will make the total distribution less similar to the noise distribution. Then calculating the maximum value of the noise on a neighboring channel where the tracking signal $x(k)$ is absent, represents the noise on channel $y_i(k)$ better in normal operation.

5

Simulation

The simulation was constructed to match the real system in both functionality and characteristics of the system. The noise and tracking signal characteristics was determined from real data collection, and displayed in Table 4.1 and Table 4.2. Data simulated outside of Simulink was specified as in Table 5.1, presented in Fig. 5.1 - 5.7. Results presented in Figure 5.8 and onward in Section 5, appear from a signal amplitude of $1.2 \cdot 10^{-2}$ and a noise amplitude of $3.2 \cdot 10^{-2}$. These noise values reflects the highest amount of noise and a tracking signal from roughly 50 meters distance.

The effect of increased detectability with adding several pulse periods was observed and the importance of adaptive threshold was studied. SNR gain was used to compare results between different filter lengths and the detector probability was studied. The threshold settings probabilities was determined by simulating several times for time periods of 10 seconds, then counting and calculating the probabilities. The time frame of 10 seconds was analyzed with the selected pulse repetition frequency in mind, since the resolution of $33.89Hz$ during 10 seconds will be $1/338 = 0.3\%$, which would be enough to determine if the probabilities is within $3\sigma = 99.7\%$. Several samples were made to determine the mean value of the probabilities, since this approach is the same as sampling for longer time instances, but it makes it easier to count.

$$P_{tot}(X) = \frac{1}{n}\sum_{i=1}^{n} P_1(x) + \cdots + P_n(x) = \frac{\frac{X_1}{m} + \cdots + \frac{X_n}{m}}{n} = \frac{X_1 + \cdots + X_n}{nm}$$

SNR can be calculated in several ways, the approach of calculating the energy of the signal and noise, and then converting the result into dB was used. This makes it easier to interpret and also compare the different results from simulation and the embedded real-time solution. Since the energy of a pulse increases with increased pulse width and signal amplitude, the terms of SNR should be carefully compared between different studies. Thus for simplifying the analysis comparison, one pulse characteristic was selected as stated in Table 4.2. An SNR value of negative dB values, does not imply that the signal amplitude is buried inside the noise amplitude.

Hence, SNR is not sufficient for determine if the signal is detectable or not. The SNR gain is used in order to get a measurement in how good the performance is increased, since SNR after filtering is not sufficient for determine the performance.

For detectivity another approach is needed, therefore a threshold is needed. If the right threshold settings are determined, the system will be able to know when a pulse is detected or not. Since the incoming signal intensity decreases with the inverse square law by increasing distance, and the filter amplifies this signal linearly, there will be a break point where the filter length cannot be extended further and the signal will not be able to be amplified above the noise amplitude for detection.

Table 5.1 Fixed pulse characteristics for simulation, representing tracking distance of 100 meters on a cloudy day.

Pulse shape	Rectangular
Pulse repetition frequency	33.89 Hz
Pulse width	108 μs
Pulse duty cycle	0.37 %
Signal amplitude	$5 \cdot 10^{-3}$
Noise amplitude	$3 \cdot 10^{-3}$

5.1 Cascade Matched Filter

Performance Analysis

In an early attempt to illustrate the power of the filtering technique of the cascade matched filter (CMF), Figure 5.1 is firstly illustrated in this section. The CMF output when extending with one pulse at the time until 300 pulses are included in the filter. Ramp-up and ramp-down behaviors are visible in the Fig. 5.1. By extending with more pulse periods, a better result is achieved. The peak of the triangle indicates the output where the predefined maximum amount of 300 pulses was used in the MF. If a fixed number of pulse periods are used for CMF, then the ramp up will only happen in the beginning of the process and then a stationary level will occur. When the tracking signal gets lost, the ramp down will occur.

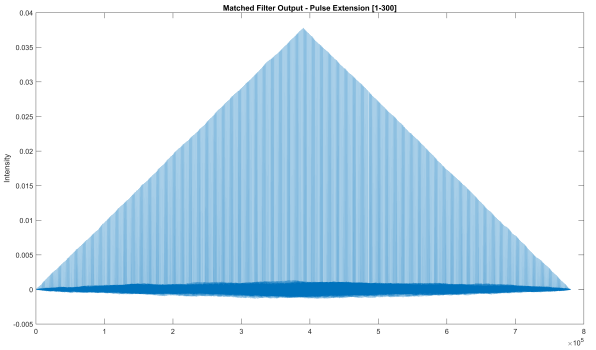

Figure 5.1 Cascade matched filter (CMF) with a filter length extension up to 300 pulse periods.

Figure 5.2 illustrates the filtering performance from Fig. 5.1 in terms of SNR gain. The SNR gain curve appears in a logarithmically scale in y-axis from the conversion into dB, otherwise the linear amplification is visible from the filter output seen in Fig. 5.1.

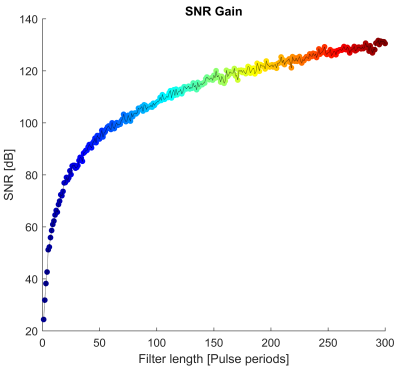

Figure 5.2 SNR gain for signal and noise characteristics representing a tracking distance of 100 meters. The filter output signal to noise ratio increases with each added pulse period used for the CMF template.

The negative and positive values in SNR from before and after filtering is not enough to determine if the desired signal is filtered such that it is detectable by the system. From Fig. 5.3, one can observe that SNR before filtering is negative, even if the tracking signals amplitude is above the noise floor. The desired signal is therefore

already detectable without the need for filtering. After more than 15 filter lengths, the SNR at the output of the filtering is greater than zero. Therefore the SNR gain is solely used to compare the filtering results, presenting the most important aspect of the filtering results.

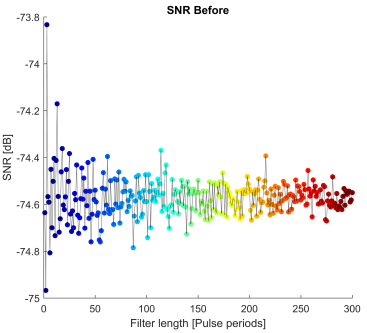

Figure 5.3.a

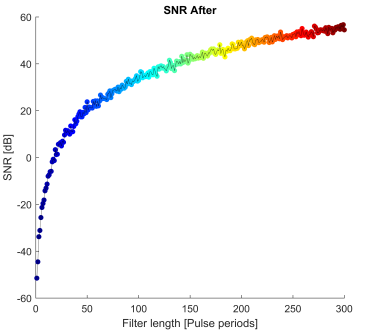

Figure 5.3.b

Figure 5.3 The SNR before and after filtering, where the simulation adds more pulse periods of the incoming signal, in order to extending the filtering length. Small magnitude variations are visible in the plots, as a result of the Gaussian distributed noise energy.

In comparison to Fig. 5.2, the linear representation of the SNR gain performance is illustrated in Fig 5.4. This illustrates the linear behavior in the CMF filter.

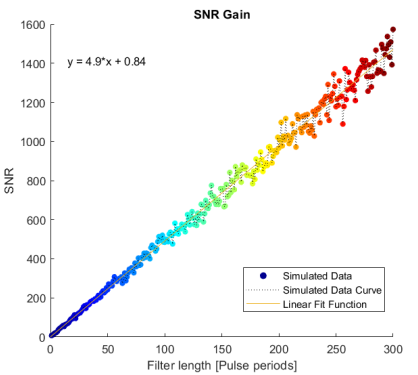

Figure 5.4 By applying Eq. 2.52.a instead of Eq. 2.52.b the conversion into dB is omitted, subsequently the linearity of the filter could be distinguished in the SNR gain plots.

The selected sampling rate effects the CMF results, which makes this parameter interesting to study. The relationship from increasing the sampling frequency and achieving the same SNR gain with fewer pulse periods in the CMF filter is observed in Fig. 5.5. The opposite holds as well, if the system needs lower sampling frequency to function, the filter length can be extended to achieve the same performance as with higher sampling frequency. The enhancement of SNR gain in regards of increasing the sampling rate, in multiples of 2, is illustrated in Fig. 5.5. The sampling rate is doubled in-between the curves, except the start point from 44100 Hz to 100 kHz. Thus, the sampling rates are 44.1 kHz, 100 kHz, 200 kHz, 400 kHz and 800 kHz. The implication of increase sampling rate is furthered illusutrated in Fig. 5.6 and 5.7.

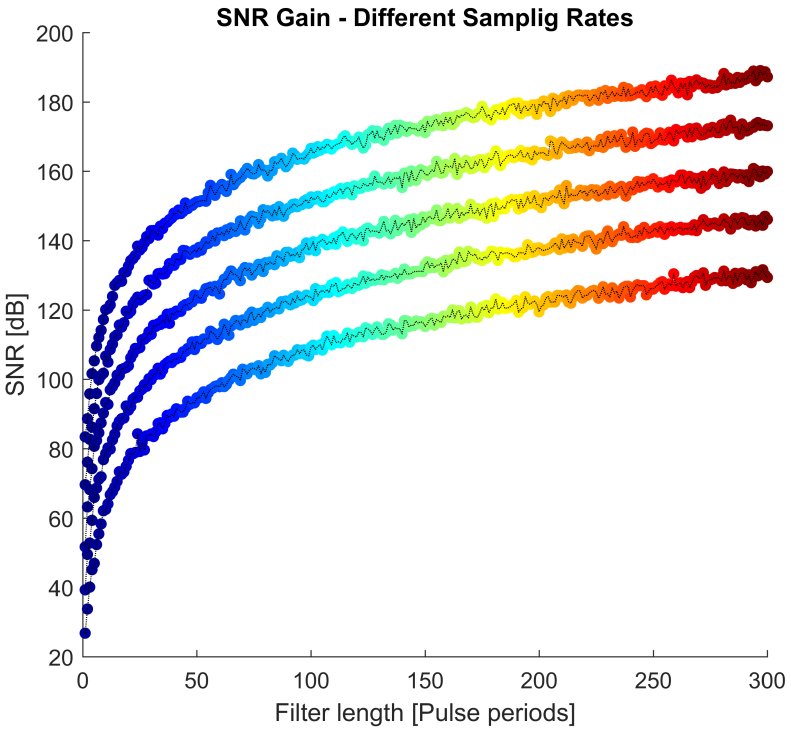

Figure 5.5 SNR gain curves for sampling rates of 44.1 kHz, 100 kHz, 200 kHz, 400 kHz and 800 kHz, where the higher the sampling rate results in greater SNR gain. Thus, the 44.1 kHz sampling rate curve us the bottom curve and the 800 kHz sampling curve is atop.

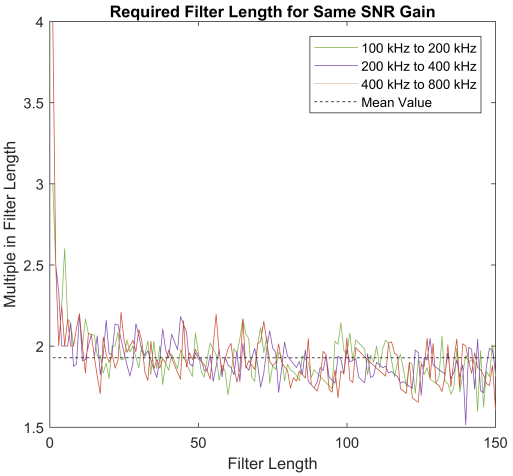

Figure 5.6 The require filter length multiple to achieve the same SNR gain, as data sampled with twice the sampling rate, is displayed in the y-axis and the filter length extension on the x-axis. The observed frequencies of 100 kHz, 200 kHz, 400 kHz and 800 kHz has a mean value of 1.93 multiples needed, to achieve same SNR gain.

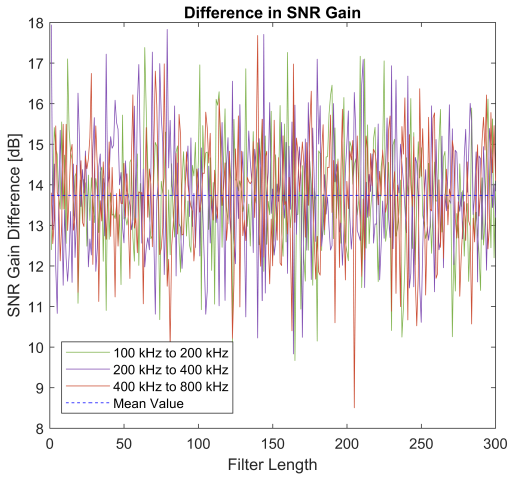

Figure 5.7 The mean value of 13.74 dB difference, between data sections sampled with twice the sampling rate, illustrates the relative filter output performance, by increasing the sampling rate with a factor of 2. By only utilizing one pulse period in the CMF, a system sampling with twice the sampling rate, will result in an improvement of almost 14 dB in the filter output.

Performance Analysis - Simulink

In all of the following parts of Section 5, the simulation was performed with only one pulse period of the rectangular pulse train. This means that the CMF results are the same as for a MF. With this in mind, the performance of the filtering could still get better. However, it illustrates the importance of filtering the incoming signal before detection. In Figure 5.8, both noise and the desired tracking signal is present in the incoming signal, but the tracking signal is undermined by the registered noise.

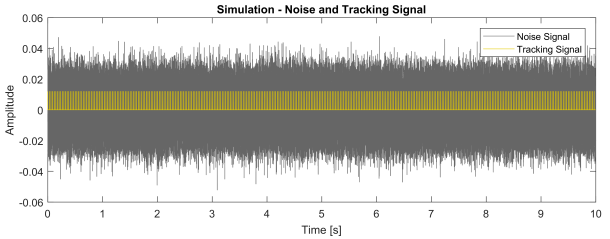

Figure 5.8 The tracking signal (yellow) and noise signal (gray) plotted on-top of each other, where the amplitude of the desired tracking signal is approximately half of the noise signal.

The summation of the two signals is visible in Fig. 5.8 as the incoming signal, were no evident distinction between the two signals can be made, nor a distinction between solely the noise signal in Fig. 5.8 and the incoming signal in Fig. 5.9.

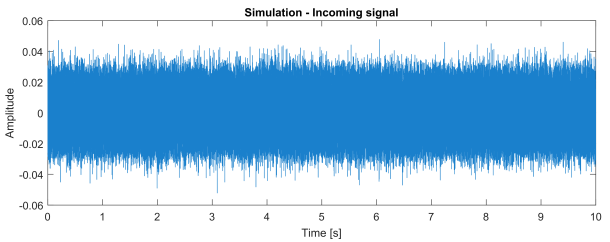

Figure 5.9 The combined signal from Figure 5.8 where the tracking signal is buried inside the noise signal, thus no visible tracking signal could be distinguished.

Filtering the incoming signal with on pulse period, results in the filter output visible in Fig. 5.10. From this output a separation between the former incoming tracking signal and noise signal is made. By feeding this signal forward to a detection stage, a detection of the tracking signal could be made as well as an synchronization to the tracking signal frequency.

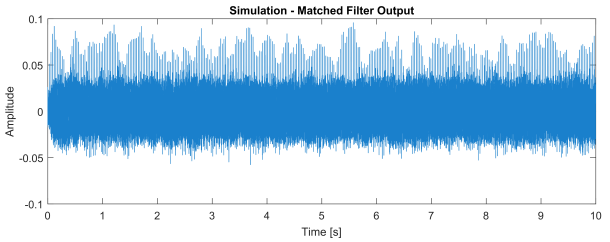

Figure 5.10 After matched filtering the incoming signal displayed in Figure 5.9 with a filter length of one pulse period, the tracking signal is distinguishable from the noise signal by observing the filter output. The signal is influenced by the noise signal, making the amplitude of the filtered tracking signal alternating a lot, sometimes it dips down in the noise signal, making it more difficult to detect an active tracking pulse.

SNR

The performance of the filter in this thesis is measured with SNR gain, since different tracking distances will generate different levels of SNR before and after filtering the incoming signal. This makes it more difficult to compare the performance of the filter, if not the SNR gain is used. The SNR gain will only give the relative improvement, therefore, staying objective in the filter performance for different tracking distances. By comparing the SNR gain before filtering from Fig. 5.3.a and Fig. 5.12, one can observe how different tracking distances of a fixed signal intensity generates different SNR values.

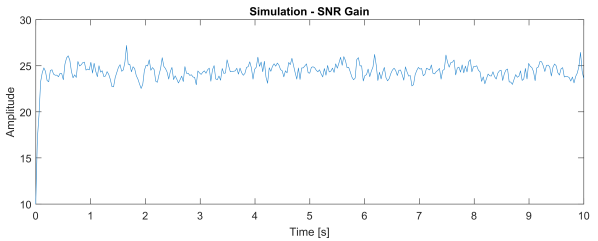

Figure 5.11 Registered SNR gain after filtering the incoming signal, where the filter length consists of one pulse period. The positive value differentiates an improvement after filtering the incoming signal on the order of approximately 25, which is a ratio between the tracking signal and noise signal. This is interpreted as a signal to noise ratio improvement of 25 times the value before filtering.

Dissimilarities are apparent from comparing Fig. 5.3.a and Fig. 5.13 to each other as well.

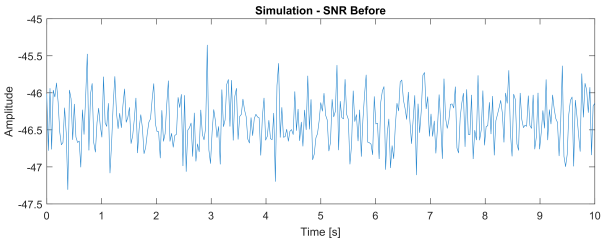

Figure 5.12 The high negative value here in dB, indicates that the energy of the noise is much greater than the tracking signal, but observe that the amplitude is only half of the noise signal.

However, in contrast the SNR gain from Fig. 5.2 and Fig. 5.11 are equal for one pulse period, approximately 25 dB of improvement.

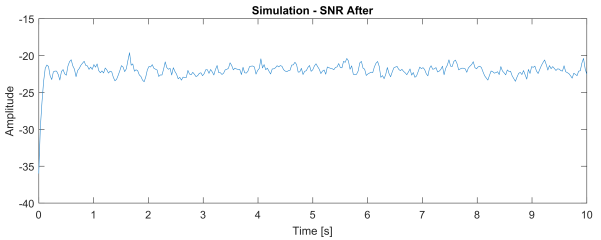

Figure 5.13 The negative value is somewhat misleading, thus the SNR gain is a better measurement in improvement, since the negative value after filtering is only indicating that the energy of the noise signal is still stronger than the tracking signal after filtering.

5.2 Threshold Performance

The exact same noise and tracking signal is used for the threshold comparison between the MSD and MM adaptive threshold. The signals is filtered by the CMF with one pulse period in the filter, whereas the filter output therefore stays the same for the comparison between the two different adaptive thresholds. The fixed threshold is the same in both simulations, hence the same probability in the Table 5.2 and Table 5.3. By using the same tracking and noise signal in the simulation of MSD and MM adaptive threshold, these two thresholds are also comparable in performance.

Adaptive Filter - Moving Standard Deviation

The difference between a fixed threshold and an adaptive threshold is illustrated in Fig. 5.14. The moving standard deviation (MSD) technique is used as the adaptive

threshold for this simulation. The MSD adaptive threshold performs at its best when the tracking signals has a low duty cycle, see Table 4.2, then the combined signal characteristics is determined by the noise signal as illustrated in Fig. 4.10. The adaptive filter appears to walk up an down along the tracking signal, thus increasing the detection of the incoming tracking signal in comparison to the fixed threshold. The result after the detection stage is displayed in the two upcoming figures in Fig. 5.15 and Fig. 5.16.

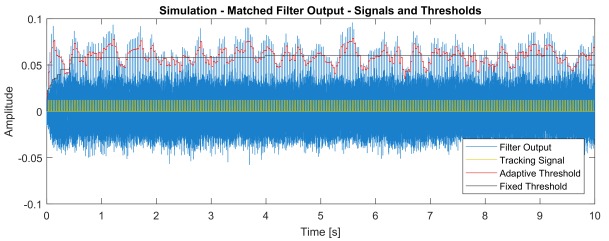

Figure 5.14 The tracking signal (yellow) is amplified and plotted on-top of the filtered output signal from the CMF (blue), the ”fixed” threshold is plotted in black color and the adaptive filter is plotted in red.

Non-Adaptive Threshold

With a non adaptive threshold, the threshold stays fixed in relative to the signal characteristics over time. This is apparent in Fig. 5.14, but also that this is true to a fixed signal characteristics. In the beginning a ramp-up occurs, which also changes the fixed value of the threshold, this is not considered to be adaptive in this thesis, since the signal characteristics changes during this part.

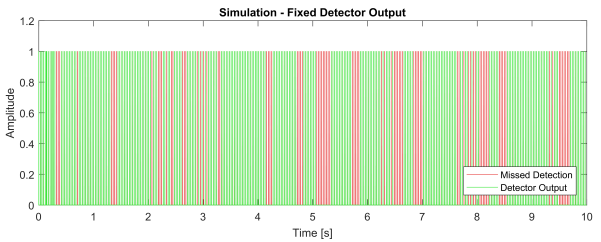

Figure 5.15 Detection stage output, where the output is either a high or a low signal. The green line is the detection of the tracking signal, the red line indicating missed detection. By studying the plot up-close the different detection probabilities can be determined. Where the green line appears a bit thicker, it has occurred a false alarm, and the lack of this signal at some ”gaps” indicates that the signal was missed, plotted in red. By counting these two factors, all probabilities can be determined.

Adaptive Threshold MSD

The movement of the moving standard deviation adaptive threshold is illustrated in Fig.5.14. The adaptive threshold (red line) changes even if the signal characteristics is not changed over time. The algorithm tries to stay away from the noise floor as much as possible, to avoid false alarm detection when sudden peaks in the noise signal occurs. The MSD adaptive filter also tries to not miss detection of the tracking signal when sudden drops in the tracking signal occurs.

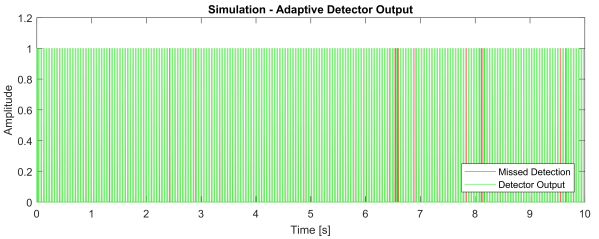

Figure 5.16 As in Fig. 5.15, the green line is the detection stage output of detected tracking signal and the red line is indicating missed detection. A thicker green line indicate a false alarm and a signal "gap" indicates missed signal detection, plotted in red. The probability in tracking signal detection has increased from Fig. 5.15.

Probability

A visual comparison from Fig. 5.15 and Fig. 5.16 gives an indication that the detectivity of the tracking signal has improved. It is however difficult to determine the exact probabilities from an visual inspection. In Table 5.2 these probabilities are displayed.

Table 5.2 Probabilities from simulations, calculated by counting the amount of missed detection and false alarm. By Calculating these values, Detection for signal an non-signal could be determined. In focus is the P_{FA} and P_{D_1}.

	Adaptive Threshold MSD	**Fixed Threshold**
P_M	4.72%	25.96%
P_{D_1}	95.28%	74.04%
P_{FA}	6.19%	9.44%
P_{D_0}	93.81%	90.56%

Adaptive Filter - Moving Maximum

The difference between a fixed threshold and an moving maximum (MM) adaptive threshold is illustrated in Fig. 5.17. By comparing Fig. 5.17 and 5.14, similarities in the adaptive threshold appearance could be made. The adaptive threshold posi-

tion looks identical in an holistic perspective, but when zoomed in view, they appear to be positioned differently. In comparison to the MSD adaptive threshold, the MM adaptive threshold is dependent on another sampling channel, including another photodetector for the threshold calculation illustrated in Fig 4.18. This filter performs good even when the tracking signal has a higher duty cycle than for the tracking signal characteristics in this thesis, this is the advantage of this adaptive threshold over the MSD adaptive threshold. The MM adaptive threshold detection stage result is displayed in the two upcoming figures in Fig. 5.18 & 5.19

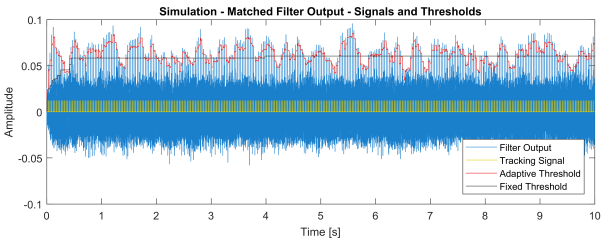

Figure 5.17 The moving maximum technique is used as the adaptive threshold for this simulation. The result after the detection stage is displayed in the two upcoming figures in Fig. 5.18 & 5.19.

Non-Adaptive Threshold

The identical fixed threshold performance between the simulations is displayed by comparing Fig. 5.15 with Fig. 5.18.

Figure 5.18 The green line is the detection stage output of detected tracking signal and the red line is indicating missed detection. A thicker green line indicate a false alarm and a signal ”gap” indicates missed signal detection, plotted in red

Adaptive Threshold MM

Detection stage controlled by an adaptable threshold, where the moving maximum is used. The moving maximum ”scans” a neighboring channel with no incoming

signal, to detect the noise level, and adapt the threshold after this altering value. The output is either a high or a low signal.

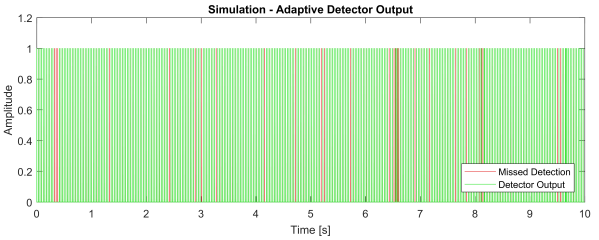

Figure 5.19 The green line is the detection stage output of detected tracking signal and the red line is indicating missed detection. A thicker green line indicate a false alarm and a signal ”gap” indicates missed signal detection, plotted in red

Probability

The probability of detection seems to be improved from the fixed threshold. In fact, all of the probabilities are better than for the fixed threshold, the same holds for the MSD adaptive filter.

Table 5.3 Probabilities from simulations, calculated by counting the amount of missed detection and false alarm. By Calculating these values, Detection for signal an non-signal could be determined. In focus is the P_{FA} and P_{D_1}.

	Adaptive Threshold MM	**Fixed Threshold**
P_M	7.96%	25.96%
P_{D_1}	92.04%	74.04%
P_{FA}	4.13%	9.44%
P_{D_0}	95.87%	90.56%

The MSD adaptive filter has a $P_{D_1} = 95.28\%$ and $P_{D_0} = 93.81\%$ in comparison of $P_{D_1} = 92.04\%$ and $P_{D_0} = 95.87\%$. This means that the MSD adaptive threshold has better detectivity of active pulses than the MM adaptive threshold, but the opposite holds for detectivity of non-active tracking pulses. Therefore the false alarm probability of the MM adaptive threshold is less than for the MSD adaptive threshold.

6

Real-Time Operation

The vast difference between simulation and real-time operating system is the sampling procedure. In the simulation the sampling process always registers a data sample at the given discrete time, however, in reality this could be more difficult than expected. It was discovered that the system needed to have an complete DMA data handling process, in order to never miss a data sample, with a high sampling frequency. A semi-complete DMA solution, where the sampling was performed with DMA and the CPU handles the sampled buffer array, frequently missed to register data samples. This eventually affected and limited the performance of the DSP.

Instead of increasing the SNR gain by extending the number of pulses utilized in the pulse train, the results deteriorate with the number of extensions. This limits the SNR improvements to the extent that it only worked with a single pulse, instead of a pulse train for the DSP. This contradicts the purpose of using several pulse periods to improve the DSP performance. Therefore this needed to be solved before further development and analysis could be made, the problem is illustrated in Fig. 6.1.

A triangular wave was generated and sampled in order to detect if sampling failure occurred. To ease the detection of failure, a difference calculation was made and plotted under ”Error detection sequence” in Fig. 6.1. On the left hand side is an zoomed out version where data is sampled and transmitted from the Cortex M4 to MATLAB. The zoomed-in version of the error is displayed on the right-hand side, where several samples are missing from the triangular sequence. The plot is somehow misleading, since it appears that the triangular wave amplitude change for that instance. However, an interrupt in sampling is not displayed, when the DMA transfer values from the AD to the memory array.

The angular estimation was never analyses in depth. However, an IR-camera was used in studies of detecting if the right transmitting diodes was turned on, since the three IR-LED facing the tracking signal should be turned on and the other IR-LEDs should be turned off. This was studied with IR-cameras, and confirmed to be working. The tracking detection system was put in a corner, whilst receiving a tracking signal transmitting towards the corner, and the system managed to respond with a replica signal without detecting itself, only the received tracking system. This means

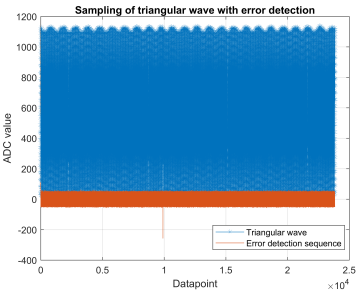

Figure 6.1.a

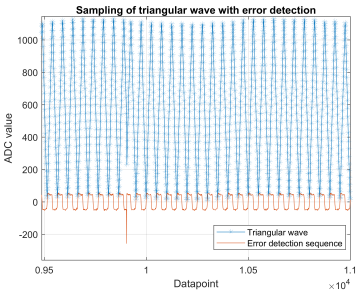

Figure 6.1.b

Figure 6.1 A triangular wave of 1 kHz was sampled with a semi-complete DMA solution. The sampling is interrupted at around 10 000 sample instances.

that the tracking detection system was not transmitting anything in the opposite direction of the incoming tracking signal. The tracking system, confirmed that a signal connection was established during the test.

The accuracy in estimating the tracking distance was found to be inaccurate, but sufficient for controlling the signal output, see Fig. 6.2. The main goal was to be able to control the output of the IR-LED in relation to the tracking signal, such that saturation was avoided, and that the power was increased for longer tracking distances.

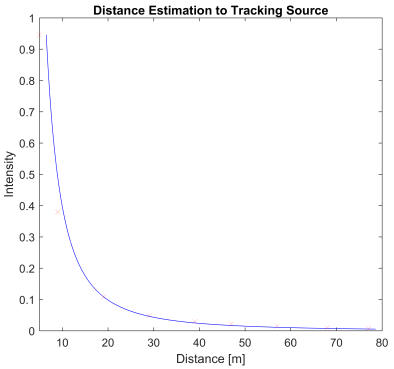

Figure 6.2 Distance estimate of the tracking systems distance from the tracking detection system. The red crosses define the real position, and the blue curve defines the estimated position, in relation to received tracking signal intensity.

6.1 Cascade Matched Filter

The cascade matched filter performance is illustrated in this section, with a first filter length up to 10 pulse periods displayed. Then an extension up to 300 pulses is presented. We introduce 3-dimensional graphs in an attempt to compress more information into fewer graphs. Figure 6.3.a displays the SNR before any filtering is performed, thus the pulse period along the y-axis does not affect the SNR, but the SNR changes with distance from the tracking source along the x-axis. The SNR after filtering is changed on both x- and y-axis, where the improvement is still difficult to observe.

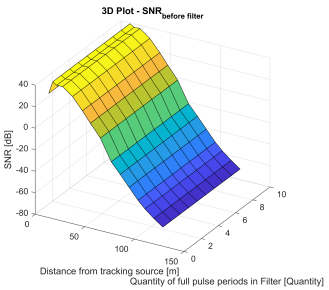

Figure 6.3.a

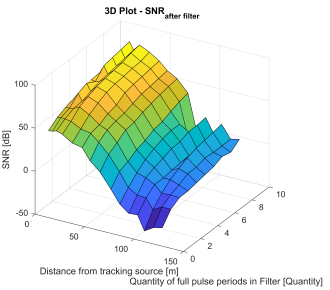

Figure 6.3.b

Figure 6.3 Figure 6.3.a illustrates the SNR at the CMF input and Figure 6.3.b the SNR at the CMF output. The tracking distance is illustrated along the x-axis, the y-axis indicates the filter length, and the SNR is plotted in the z-axis

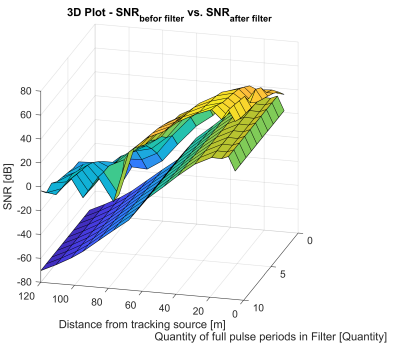

Figure 6.4 The improvement in performance is more visible in this comparison plot between before and after filtering, where the difference between the two planes is the SNR gain. This plot also illustrates how much greater energy the noise signal contains in comparison to the desired tracking signal, when the tracing distance is increased.

Since pure SNR is not a good indicator on improved performance, the SNR gain is used instead. By combining the two plots, as in Fig. 6.4, the improvement is illustrated as the gaps in-between the surfaces. This graph is still not evidential enough for illustrating the filtering performance.

Figure 6.5 plots the SNR gain solely, displaying only the improvement in performance. Ideally, the surface plot should not contain any descending parts along the x- and y-axis, meaning, with changing distance and filter length. Greater descending along the filter length axis (y-axis) for a fixed distance, depends on tracking signal interruptions as in Fig. 6.10. The smaller spikes generated along the filter length axis is due to the normal distribution of the tracking signal. The tracking signal intensity and characteristics is considered to be stable, but the noise signal is altering the tracking signal amplitude. Thus, by increasing the filter length, the SNR gain is not increased homogeneously. Inconsistencies in improved performance with extending the filter length, visible at longer tracking distances, is due to errors in signal and noise separation in the energy calculation of both signals.

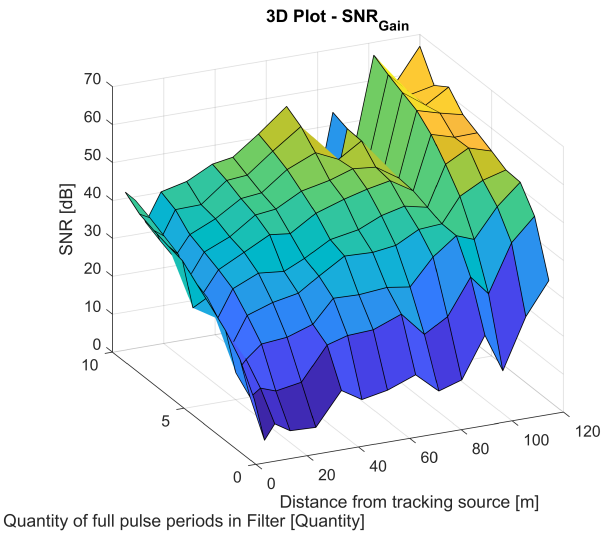

Figure 6.5 Different SNR gain is illustrated along the z-axis in the 3-dimensional graph. The tracking distance varies along the x-axis and the CMF filter length with full pulse periods along the y-axis.

Figure 6.6 and 6.11, display the CMF output at each discrete time instance for an entire pulse period. Even if several pulse periods have been used for the CMF template, the output duration will only be for one pulse period. By applying a threshold to this output, the exact time of signal occurrence can be determined, hence the tracking detection system can synchronize with the incoming tracking signal.

Simulation implies that the SNR Gain curve should be the same for all input power of the tracking signal, indicating that the curve projection from a registered tracking signal at 102 meters, fulfill the estimated SNR gain performance. Therefore, this tracking signal was used in the real-time operation section for comparison in performance.

In Figure 6.6, the compressed CMF output of one pulse period with extending the filter length to include several pulse periods at 102 meters tracking distance is illustrated. The linear appearance of the CMF is observable in this plot and also what the SNR gain improvement describes in actual filtered signal output. Along the x-axis is the the sample instance in time, where on pulse period at the assigned sampling frequency is roughly 1301 samples. The rising edge of the pulse, occurs almost in the beginning of the pulse period.

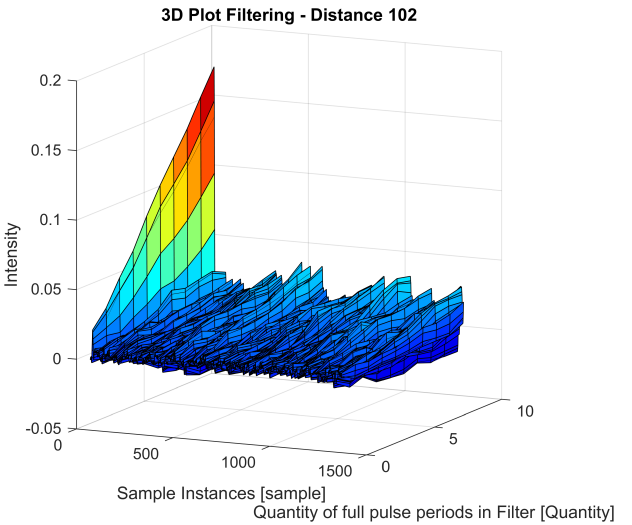

Figure 6.6 Along x-axis is the 1301 registered sample instances, with changing filter output intensity visualized in z-axis respective to filter length transitions in y-axis.

Thus far, the filter length extension was limited to 10 pulse periods with an increment of 1 pulse period between the filter lengths. The following graphs contains filter lengths up to 300 pulse periods, starting from 10 pulse periods, the filter lengths is then incremented with 10 pulse periods at the time. Similar figures to Fig. 6.4 - 6.6 is illustrated in Fig. 6.7, 6.8 & 6.11, where the tracking distance is the same but the filter length is changed as well as the outputs in z-axis.

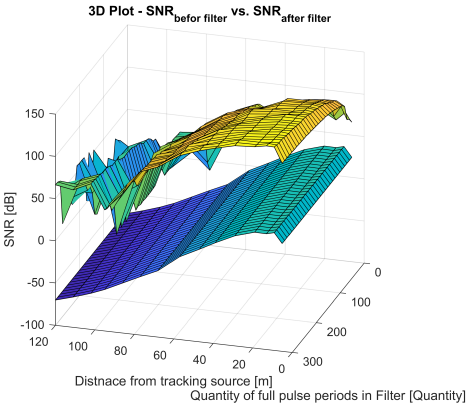

Figure 6.7 Two surface plots of SNR before and after filtering an incoming tracking signal at 102 meters distance. The above surface plot is from after filtering, whereas the lower surface plot is from before filtering.

Figure 6.8 with extended filter length of Fig. 6.5. At lower filter lengths at roughly 40 meters, the surface descends, this is due to signal interruption displayed in Fig. 6.10. By increasing the filter length even further, the effect of the signal interruption is surpassed and the SNR gain is incremented.

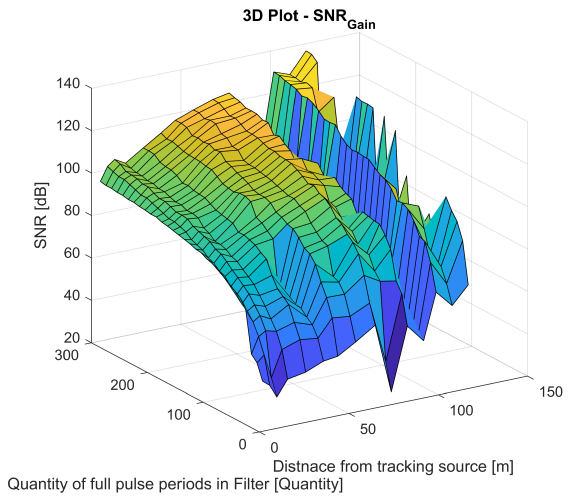

Figure 6.8 Different SNR gain is illustrated along the z-axis, the tracking distance varies along the x-axis and the CMF filter length with full pulse periods along the y-axis. The filter vary from 10 pulse periods up to 300 pulse periods in the CMF filter template.

When extending the CMF length beyond the 10 pulse periods, the signal interruption appear more distinctively. The generated SNR gain values at 95 meters is hidden below the SNR gain surface. In order to display the impact from the interference in Fig. 6.8, the 3-dimensional graph is rotated to Fig. 6.9.

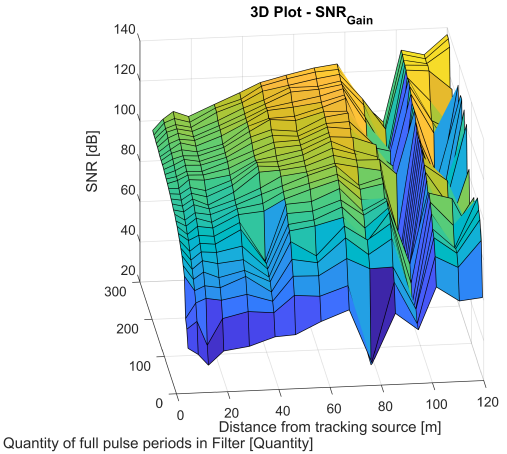

Figure 6.9 Same plot as in Figure 6.8, however the surface plot is rotated to display the surface descend at 95 meter, created by signal interruption.

From observing Fig. 6.8 one can suspect that a signal interruption has occurred at the tracking distance of 95 meters. This is even more evidential when observing the raw sampled signal in Fig. 6.10.b.

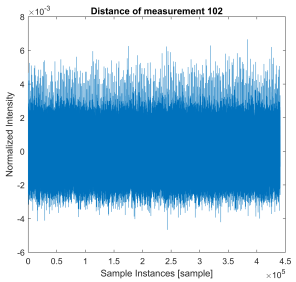

Figure 6.10.a

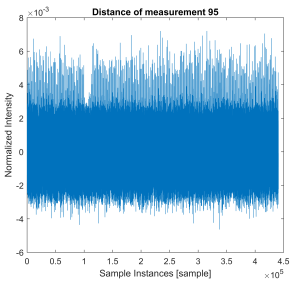

Figure 6.10.b

Figure 6.10 Signal interruption of the desired tracking signal displayed in Fig. 6.10.b at a distance of 95 meters from the tracking source. The received tracking signal from 102 meters is displayed in Fig. 6.10.a. The thick blue plotted area is the noise signal and the many small spikes are the received tracking signal.

Figure 6.11 visualizes the compressed CMF output filter extension between the filter length of 10 - 300 pulse periods, with an increment of 10 pulse periods, at 102 meters tracking distance. The tracking signal appears to be amplified more than the noise signal, making the noise signal weaker in comparison to the tracking signal for each filter extension.

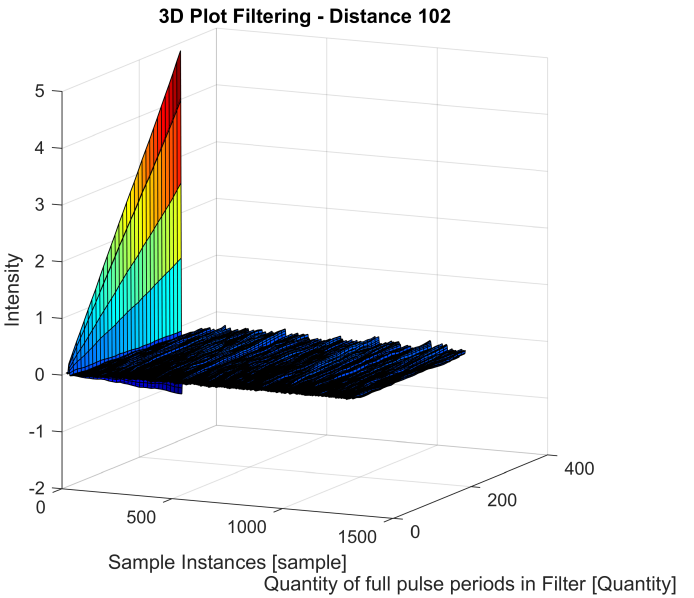

Figure 6.11 Along the x-axis is the the sample instance in time, and the pulse period at the assigned sampling frequency is roughly 1301 samples. The tracking signal detectivity is increasing linearly with every extended filter length of including one more pulse period in y-axis.

6.2 Threshold Performance - ROC

The three dimensional receiver operating characteristics (ROC) plot, visualizes the different threshold performances during different filtering conditions and tracking distances. As usual the probability of detection and probability of false alarm are plotted with changing threshold values, for the third dimension, different signal intensities/tracking distances are set to calculate the individual possibilities. Worst and best case additive noise are applied for fixed filtering lengths, or fixed additive noise is applied with changes in the filter length are plotted in this section. Every fifth intensity step on the surface curves are plotted in black color.

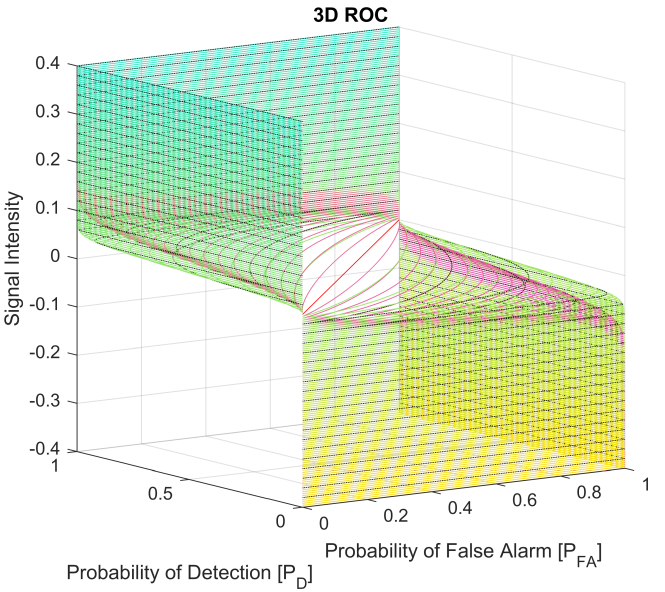

Figure 6.12 Step size of the tracking signal intensity in this plot is 0.0016. The surface of the green/yellow plot belongs to detection of signals with maximum noise illuminance of 120 kLux, at the detection overlap area. The second surface plot in light orange/violet is the detection overlap area after CMF with 10 pulse periods.

Figure 6.12 - 6.15 are 3-dimensional ROC representations, were the P_D and P_{FA} is plotted as a function of received tracking signal intensity, which is normalized to the raw sampling saturation limit. As in standard ROC curves, different threshold values is applied to the sampled data, such as in Fig. 6.6, in order to plot the P_D and P_{FA} at different thresholds at varying signal intensities. In Fig. 6.12 one can observe a slight change of ROC characteristics, when cascade matched filtering the incoming signal with 10 pulse periods. This makes the detection much better for the filtered version (orange/violet surface), but it is not obvious from only studying the ROC curve in Fig. 6.12. This graph actually only takes the noise signal amplification into account, since the noise amplitude is amplified after the CMF, the ROC gets worse after filtering. From studying Table 6.1, the same probability between the non-filtered and filtered ROC curves, is registered at almost double the intensity value for the filtered ROC. However, by filtering the tracking signal, the intensity/amplitude is increased from its original value. From observing Fig. 6.6, the tracking amplitude is registered as more than 10 times greater than the original amplitude. So if the ROC is degraded with double the amount, then the tracking signal amplification by 10 times still improves the detectivity. The SNR gain improvement from filtering the signal, testifies the separation between the tracking and noise signal.

Filtering with more pulse periods increases the standard deviation of the distribution, thus the overlap areas is greater for each added pulse period in the CMF, however the SNR gain increased for each added pulse period and so does the detectability. In Figure 6.13, the maximum noise level without filtering the incoming signal is now compared with a CMF containing 100 pulse periods.

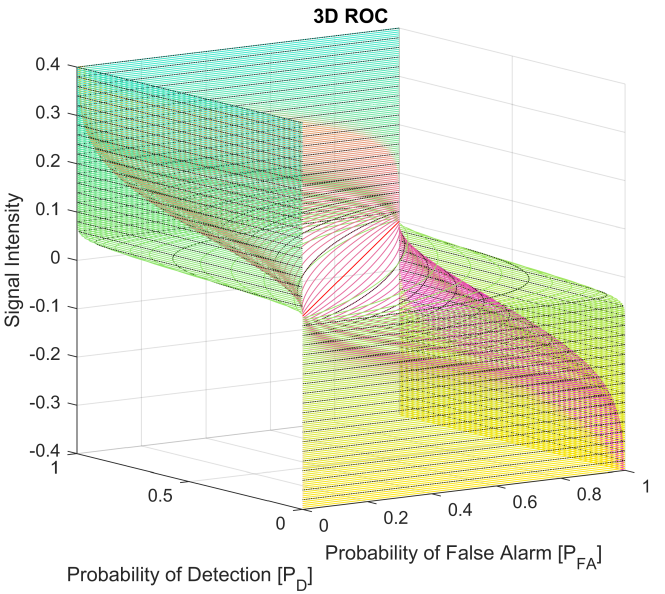

Figure 6.13 Step size of tracking signal intensity is 0.004. The surface of the green/yellow plot belongs to detection of signals with maximum noise illuminance of 120 kLux without filtering, at the detection overlap area. The second surface plot in light orange/violet is the detection overlap area after CMF with 100 pulses periods.

From Fig. 6.15, the difference in detection overlap area is quite huge. A detection made with an adaptive filter without filtering would be able to detect a tracking signal at 100 meters in low background noise radiation with $P_D = 99.24\%$ and $P_{FA} = 0.3165\%$. The fixed threshold would not stand a chance at this distance with a $P_D = 0.9872\%$ and a $P_{FA} = 0.33\%$. Similar detectability characteristics for the fixed threshold, as the adaptive threshold at 100 meters, was found at 30 meters distance. This is a huge deterioration in performance and an implication that fixed threshold could decrease the detection performance massively. Since comparing the ROC from different filter lengths, is a bit more difficult, a simple example is set by studying Fig. 6.14. In this graph, non-filtered data with different noise levels are compare to each other, clarifying how the ROC graph work. An increment of the noise level at the

detection stage input, will degrades the ROC. In this case the same tracking distance will generate the same tracking signal amplitude at the tracking detection stage input, then the ROC performance is in direct relation to the detection probability of the two cases. This notes the importance of the adaptive threshold, since otherwise the ROC of the orange/violet surface will never be achieved.

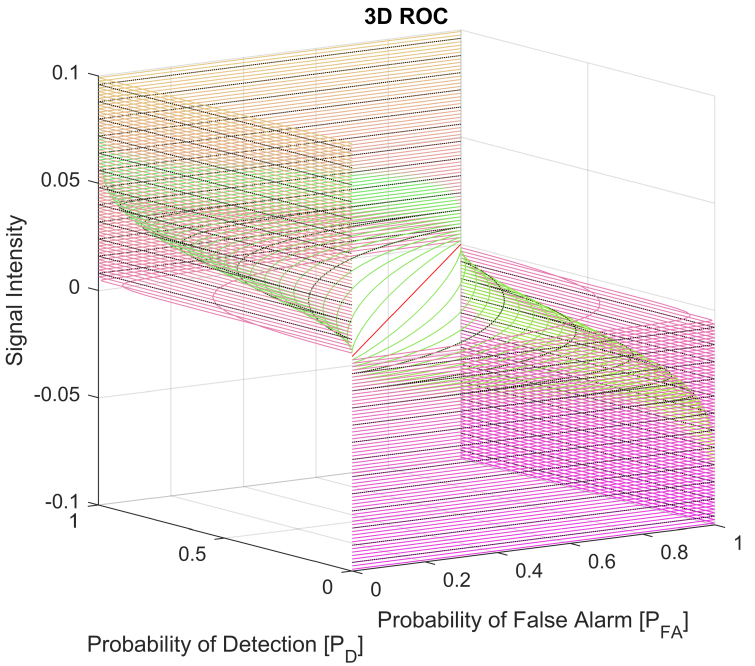

Figure 6.14 A 3-dimensional ROC comparison for best and worst noise characteristics, with a step size of 0.0016 in signal intensity. The black dotted line occurs after 5 steps, hence after 0.008 in signal intensity. The green/yellow surface plot is the same as in Fig. 6.13, which illustrates the worst case noise illuminance of 120 klux. The light orange/violet surface plot represents the best case noise illuminance of 500 lux.

By determine a P_{FA} and a P_D, a point on each ROC surface is returned. This point is set to a value precisely before ROC saturation. By comparing the ROCs between each other, this point lets us know at what signal intensity the same probability occur. This comparison is visible in the Table 6.1.

Table 6.1 The same detection probability for different incoming signal to the detection stage. First column is the best case noise illuminance, second column the worst case noise illuminance. The two last columns are filtered signals with filter lengths of 10 pulse periods and 100 pulse periods, these filtered signals contains noise illuminance of 120 klux.

500 lux	**120 klux**	**10 pulse periods**	**100 pulse periods**
$5 \cdot 10^{-3}$	$8 \cdot 10^{-2}$	$14 \cdot 10^{-2}$	$4 \cdot 10^{-1}$

Same ROC surface plot as in Fig. 6.14 is illustrated in Figure 6.15, with a signal intensity step size of 0.0001. There is a great difference in ROC performance between worst case noise values and lowest values, this is apparent from above Fig. 6.15.

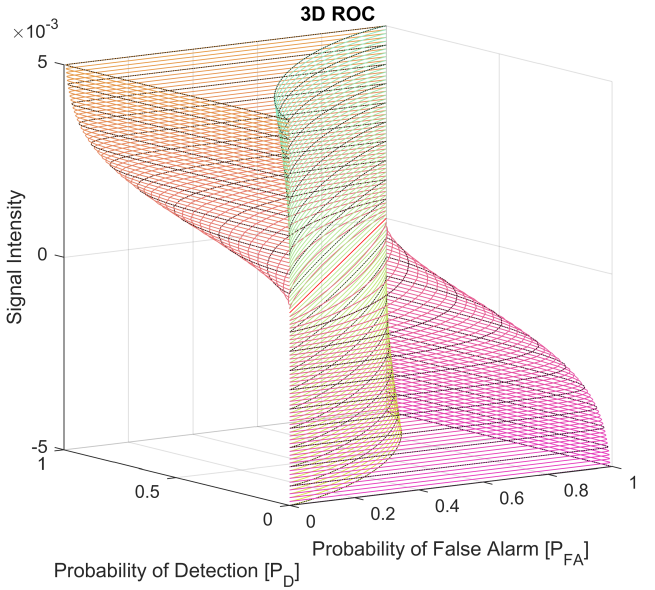

Figure 6.15 The lowest measured outdoor noise illuminance of 500 lux, is plotted in light orange/violet, and maximum noise illuminance of 120 klux is plotted in green/yellow.

Same 3D ROC plot as in Fig. 6.15, is illustrated in Figure 6.16, where only observing the x- and y-axis makes the ROC curves more visible. The solely the intensity level before ROC saturation occurs, at the lowest noise level, is plotted in Fig. 6.16.

This illustrate how poor the ROC for a fixed threshold would be in comparison to an adaptive threshold.

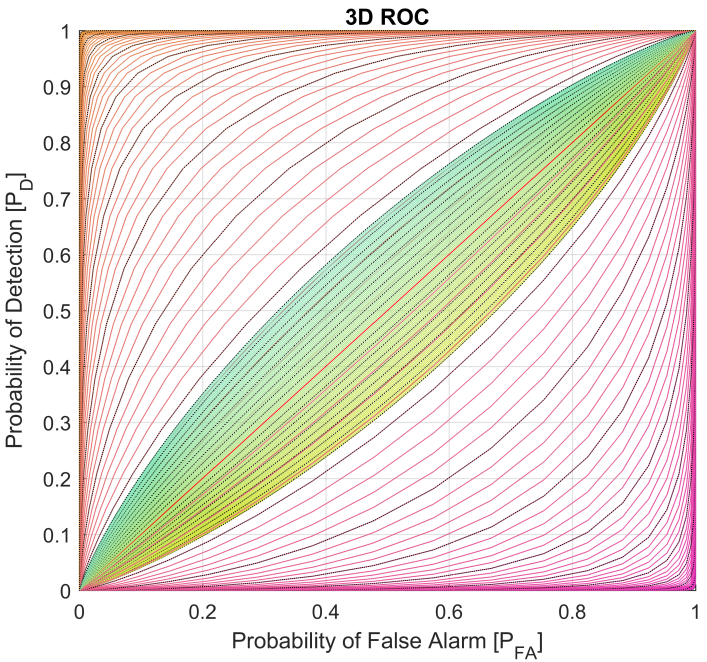

Figure 6.16 Step size of tracking signal intensity is 0.0001. The good performance of the adaptive threshold is observed, as is the poor performance of the fixed threshold, at the same distance. The adaptive threshold improves quickly with increased tracking signal intensity, whilst the fixed threshold never really has the time to improve to a satisfying state.

6.3 Guaranteed Detection By Filter Length Extension

Since the ROC comparison between different filter length was quite difficult to make, another approach was made to visualize the filter length performance in relation to detectivty. By combining above results from the linearity of the CMF and the inverse square law of the tracking signal intensity, a 3-dimensional diagram describing required filter length for specific tracking distance and noise levels can be derived and plotted. Each filter length having the capability of sustaining detection up to a certain distance, after this point the noise level will be greater than the tracking signal level.

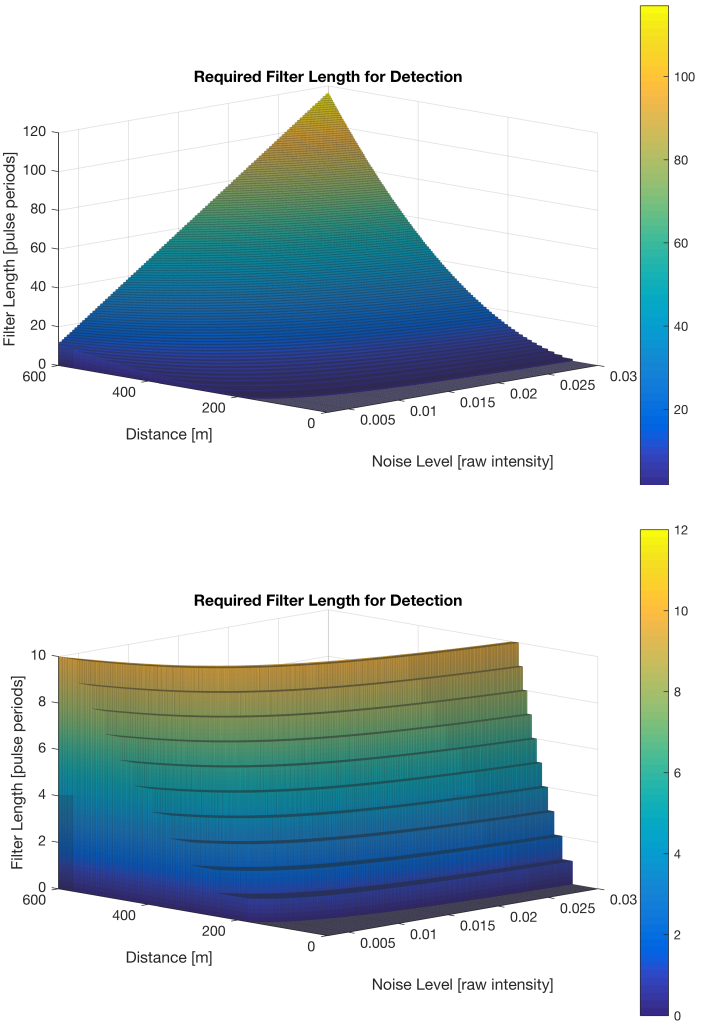

Figure 6.17 On the z-axis, the required filter length for tracking signal detection is plotted. Along the x-axis, denoting noise level, the linear increment of the filter is detectable, specially in the atop plot at longer tracking distances. The required filter length increment, for compensating the inverse square law signal power decrement by increased distance, is visible in the y-axis, denoting the tracking distance. By observing one filter length at the time, the appearing curve shape also describes the decrement of the signal power, allowing signals to be tracked for longer distances with lower noise level than in higher noise level environment. Atop and bottom diagrams illustrates filter length extensions up to 120 and 10 pulse periods respectively.

7

Discussion

There are a lot of advantages that could be utilized from the pulse characteristics. The fact that the duty cycle is low, implies that the distribution fit of the data matches the noise distribution quite well, as seen in Fig. 4.10. From this characteristic, some adaptive detection thresholds were effectively designed, displayed on Section 4.5. The MF of a rectangular pulse with low duty cycle, has quite an impact on the computational complexity, more on this in Section 4.3. The key utilization of the low duty cycle, was the ability to extend several filter length while allocating small portions of the memory, and fast filter operation. By introducing the cascade matched filter (CMF) and excluding the perfect replica template, fast filter extensions and operations was achieved.

By cascade matched filtering the tracking signal the amplitude is increased linearly, however, the SNR-gain in dB, is increased logarithmically. This is apparent by comparing Figs. 6.6 & 6.11 with Figs. 6.5 & 6.8. Without converting the SNR into dB, the SNR increment linearly with filter length extension. By analogizing Fig. 6.11 with Fig. 5.1, one can see the similarities from simulation to real time filtering when it comes to filtering output. Hence, the strong similarities between Fig. 6.8 and Fig. 5.2. The signal separation, between noise and tracking signal for real data, is the main issue behind the jagged behavior in the surface plot. Otherwise, we can see continuous filter output improvements, by adding pulse periods to the CMF.

There is a low-level limit called NEP, of which the filter can not under any circumstances amplify the incoming tracking signal. Until this low level of power is reached, improvements could be made, so that the amplitude of the tracking signal is greater than the amplitude of the noise signal. At a certain stage, an extension of the filter length will saturate the detection probabilities. This is observable in, e.g., Fig. 6.13, where the green surface plot does not change for greater values than 0.08 in signal intensity. When this improvement is achieved, an extension of operational distances is exclusively made by utilizing better detection algorithms. This implies following, instead of setting a fixed threshold for each filter length, and then amplify the tracking signal such that the signal amplitude surpasses the threshold limit and noise signal with the exact probability of false alarm. The tracking signal amplitude is already amplified above the noise amplitude, so an adaptive filter with predefined

detection probability should be able to detect the tracking signal from the noise signal and also adapt to changes in the noise data.

In Fig. 6.17, a graph of the detection performance is illustrated with different noise environments. This figure indicates that with an adaptable detection algorithm, the signal detection could be improved greatly. Deterioration of the ROC, by increasing the length of the filter, is compensated by the greater increment of signal amplification by the filter. In other words, the ROC does not stabilize until greater signal intensities, meanwhile the filter compensates for this by amplifying the tracking signal intensity even greater so the desired tracking signal intensity threshold could be set by the adaptable detection stage.

For example, Fig. 6.13 illustrates great detection probabilities at a signal intensity of 0.4, when a filter length of 100 pulse periods are used. However, by observing the signal intensity after filtering a tracking signal at 102 meters distance, the tracking signal registers an amplitude of 1.64, which is 4 times greater than the threshold guaranteeing great detection probabilities. With 10 pulses of filter length, perfect signal detection probabilities are found at a signal intensity of 0.14, this is a better match with the registered signal of 0.17, therefore it would be enough with 10 pulse periods in the filter. This will guarantee almost no false alarm detection with a fixed threshold in all environmental conditions. However, with an adaptive detection stage, a filter length with one pulse period would be enough for the same tracking distances in many cases.

From Fig. 6.17, a quandary appears. The signal intensity is decremented by distance with the inverse square law, while the filter increments linearly with filter length extension. Hence, the filter will not be able to draw level with the fast decrement of the signal power. Thus, there is a defined limit by filter execution time and memory allocation space, which will define the maximum tracking distance. Heavy noise level alternations, when tracking signals at longer distances, will therefore create rapid changes in required filter lengths for signal detection. Although, The CMF method allows quick extensions and contraction of the filter length, it is recommended to keep the filter length greater than needed during heavy noise level alterations.

7.1 Simulation

From simulating different noise and tracking signal amplitude relationships, the SNR gain curve seen in Fig. 5.2, always stays the same. This means that from simulation, the tracking signal amplification is consistent and also unrelated to the tracking distance. The SNR gain increases with increased sampling frequencies, but so does also the memory allocation. The ratio in sampling is proportional to the ratio of filter length needed to achieve the same SNR gain, so an increment in sampling frequency of a factor 2 means that half of the filter length is needed to achieve the same

SNR gain, this is illustrated in Fig. 5.6.

$$SNR_{gain} \propto f_{sample} \cdot L_{filter}$$

A twofold increment in sampling rate, results in a constant SNR gain increment of 14 dB, for each applied filter length extension, illustrated in Fig. 5.7. In conclusion, by incrementing the sampling rate from 100 kHz to 800 kHz, the SNR gain for each filter length extension has roughly increased by 42 dB and the filter length only needs to be 1/8 of the length to achieve the same SNR gain results. Hence, a system sampling with 800 kHz, requires a filter length of 30 pulse periods to fulfill the same filter output as a system sampling with 100 kHz operating with a filter length of 240 pulse periods, comparison seen in Fig. 5.5.

The faster the sampling rate of the system, the faster the system can adapt to noise alternations or changes in tracking distances. The longer the tracking distance or the greater the noise signal gets, the more pulse periods are needed in the filter to amplify the tracking signal above the noise level. By allocating memory for several consecutive pulse periods, matching the upper limit of selected filter length, the tracking detection system will not be effected in filter performance. If no preallocation is performed, then there will be a delay before the filter can achieve the right filtering performance, such as adapting to noise signal alterations. If the sampling frequency can not be incremented, then the pulse period repetition frequency can be incremented to obtain the same result. From Eq. 2.53, 2.19 & 2.20 the following interrelation could be stated

$$SNR_{gain} = 20 \cdot log_{10}\left(\frac{y_{xx}}{e_{xx}}\frac{e_{nn}}{y_{nn}}\right)$$

This relationship says that we can either improve the detectivity by amplifying the desired tracking signal y_{xx} or suppressing the noise signal y_{nn} by filtering. Neither MF nor the CMF does any noise suppression, but aims to only amplify the desired tracking signal. Since the noise signal is zero mean value Gaussian distributed, the noise signal will not be suppressed on a longer time average, but for small time instances it will be suppressed and sometimes amplified. The noise suppression is fully up to the stochastically behavior of the noise signal, illustrated in Fig. 5.7, where the different sampling incidents will lead to different SNR gain results.

Fig. 5.8 illustrates the incoming noise and tracking signal with amplitudes of $3 \cdot 10^{-2}$ and $8 \cdot 10^{-3}$ respectively, the output of the CMF after filtering the combined signal in Fig. 5.9, is demonstrated in Fig. 5.10. The filter length is of one pulse period, capable of amplifying the tracking signal above the noise level, which is a factor 3 3/4 greater in amplitude than the tracking signal. This amplification correlates to a SNR gain of 25 dB, observable in Fig. 5.11, resulting in an average output amplitude of $6 \cdot 10^{-2}$, which is a factor of 7.5 in amplitude improvement. Even if the average is more than beyond the output noise amplitude of $3 \cdot 10^{-2}$, the tracking signal is still influenced by the stochastically noise signal, such that the output of the tracking

signal sometimes disappear into the noise signal. For this reason, the tracking signal amplitude average needs to be amplified, to more than double of the noise amplitude, to be able to guarantee tracking signal detection in the system at all time, as long as the tracking signal power and noise power remain constant over time.

Since this is not very likely to happen in a real time operation, it is important to have the ability to change the detection algorithm parameters during operation. Without having a detection system adjustable to noise changes, system information will be disregarded, this is illustrated by the detection stage output for the fixed threshold in Fig. 5.15. The Gaussian distributed noise signal, characterized from real sampled data, is enough to change the fixed threshold in such a way, that 20% of the information will be overlooked. These detection statistics is represented in Table 5.2 and the runtime changes in the fixed threshold is illustrated in Fig. 5.14, as well as the MSD adaptive threshold. From solely observing Fig. 5.14, one can observe how the threshold is continuously updated and tries to catch every change in tracking signal behavior.

Fig. 5.16 illustrates how the signal detection has increased for the detection stage and Table 5.2 confirms this statement. When transitioning to a MSD adaptive threshold from a fixed threshold, the probability of miss was lowered with a factor of 5 representing 15.68%, although the probability of false detection increased with a factor of 5 representing 1.18%. There are some parameters that one can tune for the adaptive threshold such as update rate, sample window and parameter weighting. This will change the detection output and its statistics. The selected parameters, was selected in such a way that the signal detection was improved on the cost of increasing the probability of false alarm, to a still satisfying extent.

Since probability of false alarm will not be zero, there must be an algorithm in the detection system which will catch these occurrences. If the probability of false alarm is high, then the system will be occupied on correcting errors instead of the designed function. Therefore, with this threshold transition, the system will deal with false alarm detection every 2nd second in lieu of every 10th second. As long as the system can handle these deadlines, neither is better then the other. However, for pulse synchronization perspective, it is better to have a greater probability of signal detection, because more data is used to estimate the frequency drifting of the tracking pulse synchronization. If the signal detectivity is to low, then the time synchronization of the signals might get unsynchronized, an the tracking communication is lost.

When signal interruption occurs for longer time instances, e.g., blocked signal path, unsynchronized tracking communication or bad signal detectivity of the detection stage, a search mode will be initiated to find a tracking signal. This is expected to only occur when the signal gets blocked by an obstacle for a longer time period, thus if the detectivity is to low of the system, then the system will enter the tracking search stage while in normal operation mode. Causing the system to operate in an unstable manner, switching between operational mode and search mode. Hence, if the deadline of handling false alarm could be sustained, then a better detectivity is preferred.

The MM adaptive threshold, performs great even if the duty cycle of the tracking pulse is substantially longer than for the used tracking signal model. From simulations analyzing the difference between a fixed threshold compared to a MM adaptive threshold, seen in Fig. 5.17 and Table 5.3, almost the same great improvement in detectivity is noted, with an increment of false alarm. The false alarm probability is still relatively low in comparison to the signal detection probability, implying that an false alarm rejection algorithm should still be stable, and the false alarm is estimated to occur every second. If this falls into the deadline limit, the system is still performing better than with a fixed threshold. One thing which is not simulated is the effect of moving the tracking detection system between several narrow obstacles on a sunny day, which would drastically change the noise level.

7.2 Real-Time Operation

By studying Fig. 6.7 and Fig. 6.4, one can notice that the lower surface representing the SNR before filtering is increasing with shorter tracking distances, since the tracking signal intensity increases. The SNR surface after filtering varies a lot in shape, this is due to small details i.e. as Gaussian noise distribution impacting the tracking signal and blocked tracking signal. The shape of the SNR gain surface after filter will be projected on the SNR gain surface illustrated in Fig. 6.5, 6.8 & 6.9. A tracking signal blockage is illustrated in Fig. 6.10.b for the tracking distance of 95 meters. This blockage of 10 tracking pulses, from a passing object, gets projected in the SNR gain surface plots. SNR gain surface plot is expected to sustain the same SNR gain along the x-axis, representing the tracking distance, this is apparent in these plots. This is due to the pulse shape alteration, from low-pass anti-aliasing filtering. This filter transforms the output from rectangular pulses to slightly more triangular appearing pulse. This is why the closer distances related to greater registered signal intensity, will loose more signal power in relation to the signals from greater tracking distances, resulting in lower SNR gain performance for the closer tracking distances.

As mentioned in Section 6.1, simulation implies that the SNR gain curve will be the same for varying tracking signal inputs. By observing Fig. 5.2 and Fig. 6.8, the best fit from simulation to real-time operation is registered at 102 meters tracking distance. By studying the incoming signal of a tracking process at 102 meter distance, in Fig. 6.6 and 6.11, some interesting observations can be noted. The amplification of noise is less than the amplification of the tracking signal, for all filter lengths. This means that if the registered tracking signal power is greater than NEP, it will be possible to amplify the tracking amplitude above the noise amplitude by filter the tracking signal. The mapping from several pulse periods into a single filter output period, proves to give great results in determine the pulse discrete-time occurrence in sampling instances. By studying Fig. 6.15 and Fig. 6.16, the importance of adaptive threshold performance is displayed. By not using an adaptive threshold,

the tracking distance can reduce massively from 100 meters to 30 meters in worst case reality. Since the signal intensity reduces with the inverse square law, this indicates massive deterioration in performance by not selecting an adaptive threshold.

Table 7.1 Value statements from an incoming tracking signal at 102 meters distance.

Filter length	10 pulse periods	100 pulse periods	Ratio
ROC [Max noise]	0.14	0.4	2.86
CMF [Noise signal amplitude]	0.036	0.088	2.45
CMF [Tracking signal amplitude]	0.17	1.64	9.61
CMF [SNR gain]	65.6	107	1.63

Observing the difference in above Table 7.1, indicates that filter length extensions will not increase the noise floor as much as the signal amplitude. This yields better detection probabilities by extending the filter length, which is presented by the moderate incremental of signal intensity threshold of the ROC by extending the filter length, and strong increment of signal amplitude and increasing SNR gain. The SNR gain increases logarithmically, indicating that the amplification in signal energy of the tracking signal in contrast to the noise signal will increment slowly with high values of filter length extension. The linear improvement on noise and tracking signal after filtering, does not increase equally, according to Table 7.1 the desired tracking signal is increased almost 4 times more than the noise signal. This implies that even if the SNR gain, is increasing logarithmically in dB and slowly for greater filter lengths, the detection will still be increased linearly as long as the registered signal power is greater than the NEP.

Without converting the SNR gain into dB, the linear increment the SNR gain is perceptible. From simulation, the slope is estimated to be 4.9 times linear growth, in comparison to the 4 times linear growth from the real time data. By mapping real-time sampled data from post processing into a 3 dimensional graph as in Fig. 6.17, plotting required filter length for each tracking distance for different noise levels, we can extract the required parameters to achieve a tracking signal connection for the system. This figure illustrates the linearity of the filter, by observing the trend on the x-axis, interpreted as more filter lengths are needed to get the same tracking distance performance when the noise level is increased. This impacts the computational time increasing the time complexity, reducing the reaction time of the tracking system output signal level control. The last mentioned point, is due to the fact that the tracking position speed is indirectly low-pass filtered, by a resemblance with a moving average filter.

Extending the filter length means decreasing the cutoff frequency and lowering the bandwidth of the reaction time for adjusting the tracking signal amplitude output. This is not a problem since the received tracking signal intensity changes very little at

longer distances, where greater filter lengths might be needed, illustrated in Fig. 6.2. The inverse is applicable on the output of the tracking signal of the tracking detection system. The output has a maximum power and this is applied after a certain tracking distance, which already occurs before the distance gets above 10 meters in this thesis, therefore there will be no output intensity changes at e.g. 200 meters. If this needs to be regulated for longer distances as well, then one needs to perform speed estimates of the moving tracking position, to set the right filter length at the right time and not incrementing/decrementing the filter length by one for each filter length adjustment.

In the same graph Fig. 6.17, the inverse square law of the signal power for changing tracking distances is observable along the y-axis, where the curve gets drastically steeper for higher noise levels. It is at this level, the minimal performance limit in tracking distance, for the system is defined. With a non-adaptive detection stage, the systems maximum and minimum performance limits will be equal and defined after the minimum performance limit, lastly the computational complexity will always be equal to the worst case. As a fixed threshold is adapted to stay stationary the whole time throughout operation, the right probabilities must be met. Thus, the system must be adjusted to match the ROC of the heavy noise environment. The ROC for the detection stage will therefore have the same poor performance even for tracking signals within a low noise level environment. To achieve the great detection stage performance, the filter needs to increase its filter length more than needed, to meet the detection probabilities. In this case, the system detection will not be adaptive, and therefore the filter length is needed to be established in beforehand, and stay fixed. Moreover, if the filter length is extended the noise level will also be amplified, then the systems false alarm probability will increase drastically, creating a non-functional system.

The filter length needs to be established, so that the longest tracking distance specified will be able to be detected. This means that shorter tracking distances will generate an output amplitude of the tracking signal which will be much greater in relation to the noise signal amplitude, comparable to the CMF output in Fig. 6.11 for greater filter lengths. The ROC will stay the same at all time for a fixed threshold system. As an example, in Fig. 6.13 the orange/violet surface illustrates the ROC for a detection system with a filter length of 100 pulses in the CMF, a fixed threshold would be equal to setting a fixed signal intensity at this surface. This will map to specific detection probabilities, and these will stay the same as well as the signal intensity limit. As the noise level varies so does the ROC of an adaptive detection stage as well, entailing that the space in-between the orange/violet ROC surfaces in Fig. 6.14 and Fig. 6.13 will be used, as from a non-adaptive detection system. By utilizing the space in-between these surfaces, some fixed detection probabilities could be achieved even with changing tracking signal intensities, allowing the filter length for the CMF to adjust adaptively, decreasing the time complexity of the filter section, subsequently extending the operational time of the system when operating on a battery power supply.

8

Conclusions

The general outcome from the results are that some trade-offs are needed to be made. With increased filter length, such as adding several pulse periods, will increase the computational time, allocation of memory storage but also improve the detectability. At some point, the memory gets full or the filter length extension will not be able to perform within the operating time frame, thus misses its deadline. A recommendation is to first determine the important details, such as deadlines, memory storage, noise variations.

The deadline for this thesis was the internal clock synchronization of both systems, where the tracking detection system sets its internal clock after the tracking system. The clock of the embedded tracking detection systems was of lower quality than of the tracking system, so the system was found to be able to hold a rectangular pulse train synchronized for 6 seconds before the pulses was not overlapping each other any longer. The system was working good with 80% overlap, so this resulted in $6 \cdot (1 - 0.8) = 1.2$ seconds of deadline before the performance was visibly deteriorating. The system needed to be re-synchronized, before the deadline was missed. The deadline was never an issue, however, the memory space only allowed 8 consecutive pulse periods with a sampling frequency of 100 kHz, hence resulting in a system limitation. Thus a more advanced threshold could be developed to increase the detectability, since the deadline was always met. From the results and discussion sections, the applied threshold has a large impact on the detectability, and could thus extend the operating distance further. Another compressed algorithm was also developed to increase the amount of stored pulse periods in the same memory space as before, allowing more pulse periods for the cascade matched filtering. This algorithm is illustrated below and it enabled an increment from 8 pulse periods to 536 pulse periods.

8.1 Further Investigation

Extending detectability in other interferences

The system is not only effected by outdoor illumination from the sun. Other light sources such as spotlights on a construction site or an indoor light source, produces other interferences. These light sources is very common and have sinusoidal disturbances from the AC frequency. A CMF template was designed in order to prevent the system from these interferences, where instead of a designing symmetric impulse response coefficients and tracking pulses, antisymmetric characteristics was designed. The antisymmetric design retains the linear phase property and improves the disturbance rejection from these sinusoidal interferences. By adjusting the AAF, an inverted duplicate of the signal could be constructed, such that the pulse first has a positive peak, and directly afterwards a negative peak. A sinusoidal interference should cancel out when the area underneath these two pulses are equally great. This system was not investigated in detail, but an embedded version and offline code was developed. The results will be displayed and commented upon, however no more example will be displayed, since limited tests on this solution under these interferences was performed.

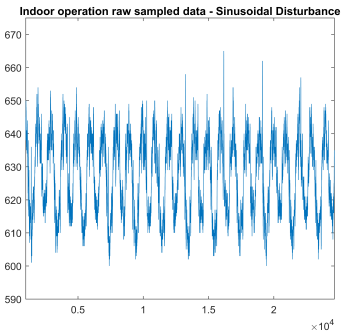

Figure 8.1.a

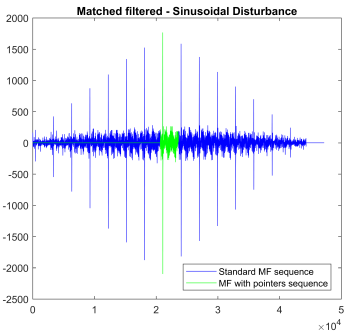

Figure 8.1.b

Figure 8.1 Antisymmetric characteristics by the AAF, MF and CMF is observable by notating the positive and negative peak illustrated in the right diagram. In the left diagram, the desired signal is buried inside of the sinusoidal noise, except from three consecutive visible signals. Cascade matched filtering of this signal with antisymmetric impulse response coefficients, suppresses the noise and increases the desired signal. Fig. 8.1.b displays a traditional convolution between a template and the raw signal in blue and in green is the CMF output.

From Fig. 8.1.b, the CMF includes 8 pulse periods in the filter template, but maps the result only into one pulse period. The result is the same as with the traditional template convolution, but the memory storage and time complexity is much

smaller. The traditional convolution will have a ramp-up and ramp-down phase of the filtered desired signal, this is due to the fact that the template is shifted in to the raw signal for convolution, and the maximum peak aligning with the green plot, represents the position where the 8 template pulses is fully shifted in and convoluted with the underlying signal from the raw data.

Fig. 8.1 illustrates how the sinusoidal noise is reduced but the same great detectivity as before is retained, implying a noise reduction as well as an amplification of the desired signal. For further investigation, different signal frequencies could be examined to cover several alternating current interferences in multiple countries. Different designs options on the antisymmetric characteristic could be explored a bit further and the impact of different noise levels could examined.

Extending filter length and pulse periods

In the embedded code, a mode was implemented which sampled in smaller portions around the pulse location when a signal was detected and synchronized to. By narrowing the sampling time to the estimated location of the tracking pulse, the memory storage was able to contain several pulse periods, in fact 536 pulse periods was then able to be stored, in stead of the 8 pulses. Moreover, by sampling in a narrowed area around the estimated tracking pulse location, the pulse repetition frequency appears to increase enormously but the other characteristics stays the same. This has the advantage of not storing unnecessary data, not containing the signal, and increasing the filter SNR gain further by extending the filter length with more pulse periods.

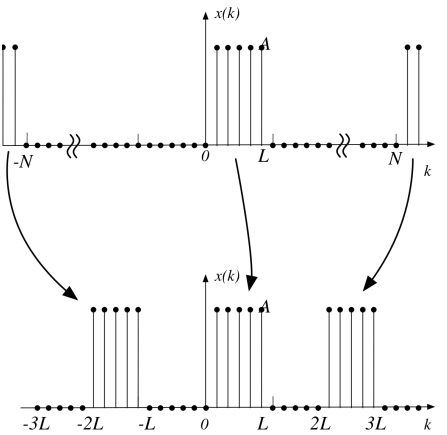

Figure 8.2 Instead of ceaseless sampling, a special sampling process could be entered where sampling is performed in narrowed discrete time frames, so that several pulse periods could fit into the memory for DSP. The pulse repetition frequency increases put the other pulse characteristics stays intact.

Practically, this means that the timer triggering the ADC sampling is configure in N-pulse mode, and the DMA is operating when an EOC is performed by the ADC. This means that the N-pulse mode is sampling N repeatedly instances before waiting to next pulse period. The N instances is set to 4 times the pulse length of L, so that a phase change of the tracking signal could be detected. Figure 8.2 shows the principle, where N in the image denotes the pulse period and not the N in the N-pulse mode sampling. The memory array stored by the DMA will then be able to contain 536 pulses instead of 8. The CMF time complexity is of order $O(L \cdot M)$, so for example a pulse length of 10 using all the pulse periods will have a time complexity of 5360. This still makes the system fast and being able to run within the deadline, but the adaptive threshold could not use the moving standard deviation approach, instead the second alternative could be used were only moving maximum to determine the threshold. Further investigation on operational distances, robustness to noise etc is needed for this algorithm approach.

8.2 Future development

For future investigation and development it would be interesting to implement an extending Kalman filter (EKF), for the tracking process. If a system model could be clearly identified, the signal characteristics would be strongly connected to selected variables of interest.

For further development, more advanced programming of the Simulink simulations could be made, so the CMF could be used with different thresholds. The current simulation filters with one full pulse period, since increasing with more pulse periods is not possible due to allocating to much memory space. The traditional approach by creating a matched filter template could not be used to create the CMF. Further simulation features could be added such as plotting a 3D ROC curve of the detection performance and 3D SNR curves of the filter performance. More analysis on the accuracy of distance and angle estimation, could be made. In simulation mode, the performance between different *sampling frequencies* was observed, although in real-time operation it was never analyzed in depth. With increased sampling frequency, comes better resolution and more sampled data for each active pulse. Since the MF and CMF output is the sum of convoluted data, the more of the pulse that is summed together, the greater the filtered output gets. A study could be made on the trade-off between increased SNR filter output in relation to memory allocation, when increasing the sampling frequency. Moreover, investigate if the Gaussian distribution of the noise signal remains the same for increased sampling frequencies, proving that the simulated estimation of SNR gain improvement remains true.

A

Calculus

A.1 Neglectable doppler effect

The doppler effect occurs when either or both receiver and transmitter moves during a transmission. The equation below describes how the received frequency is changed when the transmitter is stationary and the receiver is moved during transmission in a transmission medium.

$$f_R = \frac{c}{n\lambda}\frac{\frac{c}{n} \pm v_R}{\frac{c}{n}} = f_T \frac{\frac{c}{n} \pm v_R}{\frac{c}{n}} = f_T K|_{K=\frac{\frac{c}{n} \pm v_R}{\frac{c}{n}}} \tag{A.1}$$

where f_R is the received frequency [Hz], f_T is the transmitted frequency [Hz], v_R is the speed the receiver is moving in [m/s], c is the speed of light in vacuum [m/s], n is the density of the transmission medium [kg/m^3], λ is the wave length of the transmitted signal. Equation A.1 implies that the frequency is changed with a factor of K. For normal walking speed and hectic walking speed this factor will not change the received frequency outside the receiving characteristics tolerances for the system. The factor K for a typical walking speed of 1.4 m/s in air [Mohler et al., 2006][Levine and Ara, 1999] would result in a difference of $6.03818 \cdot 10^{-9}$ [Hz] in frequency. For fast walking speed of 2.2 m/s [Mohler et al., 2006] the factor K result in a difference of $9.48856 \cdot 10^{-9}$ [Hz]. These changes have no significant changes in the receiving characteristics inside of this thesis scope.

A.2 Neglectable incoming angle

The incoming signal strength does not change noticeably for a half angle of 20°, as seen in Fig. 4.6. So how about the actual distance miscalculation? It turns out that the incoming angle does not affect the system noticeably if the device is much smaller in comparison to the measured distance.

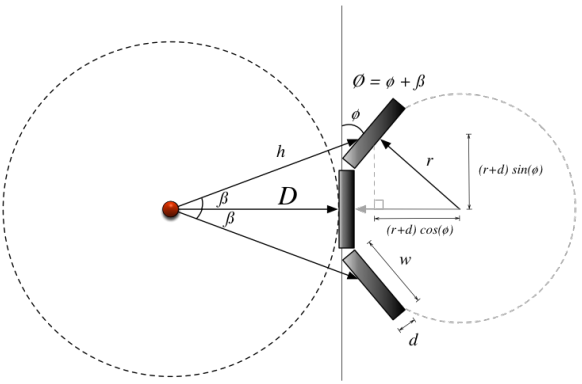

Figure A.1 Reduced model of Fig. 4.5, which illustrates the calculation steps for the equation A.2.

$$
\begin{aligned}
h^2 &= ((r+d)sin(\phi))^2 + (D+(r+d)(1-cos(\phi)))^2 \\
h &= \sqrt{((r+d)sin(\phi))^2 + (D+(r+d)(1-cos(\phi)))^2} \\
h &= \sqrt{a^2+D^2+2Db+b^2}
\end{aligned}
\tag{A.2}
$$

$$
\lim_{D\to\infty} h = \lim_{D\to\infty} \sqrt{a^2+D^2+2Db+b^2} = D,\ if\ a \ll D\ \&\ b \ll D \tag{A.3}
$$

In our case a & $b << 1$ and D is typically much greater than 1. When the ratio of true and "false" distance is plotted as a change of distance and also incoming angle ϕ, the following result in Fig. A.2 is observed.

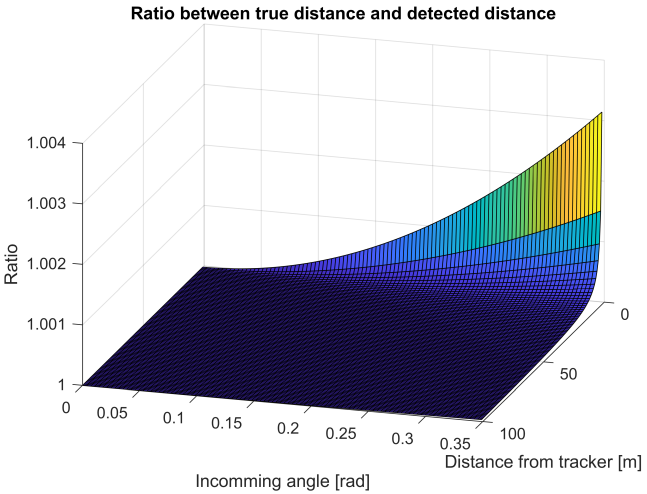

Figure A.2 Ratio between true distance and detected distance. The maximum angle of 20° of misplacement at a close distance, will result in the greatest error. The percentage of error is very small, and decreases to zero as the distance increases. The error will be roughly 1.2% at 1 meter distance and decreasing exponentially to 0.4% at 3 meters. The error ratio is displayed in Fig. A.3

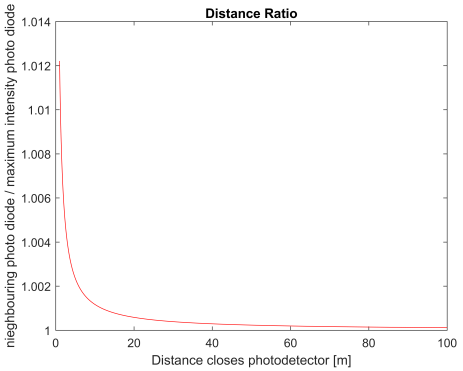

Figure A.3 The graph illustrates how the error of distance detection decreases exponentially with increased tracking distance, when the incoming angle of the tracking signal is fixed at 20°. The first plotted value is at 1 meters distance, corresponding to an error of 1.2%. If the tracking module would have less than 9 photodetectors, then the worst case incoming angle is greater then 20°, subsequently the errors will also be greater than in the plotted graph.

B

Matlab

B.1 Simulation Model - Simulink

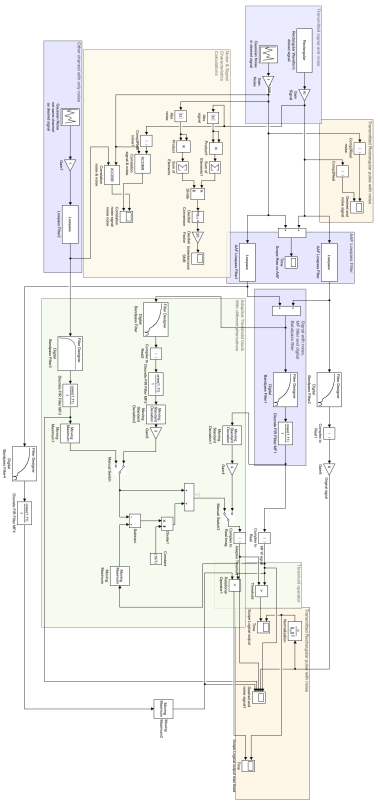

Figure B.1 The Simulink model used for simulations. The *blue areas* defines the used signals and their filters, the *green areas* defines different thresholds that could be switched manually between simulations and the *yellow areas* is for observations and calculations.

B.2 Rectangular Pulse Train

Continuous-time

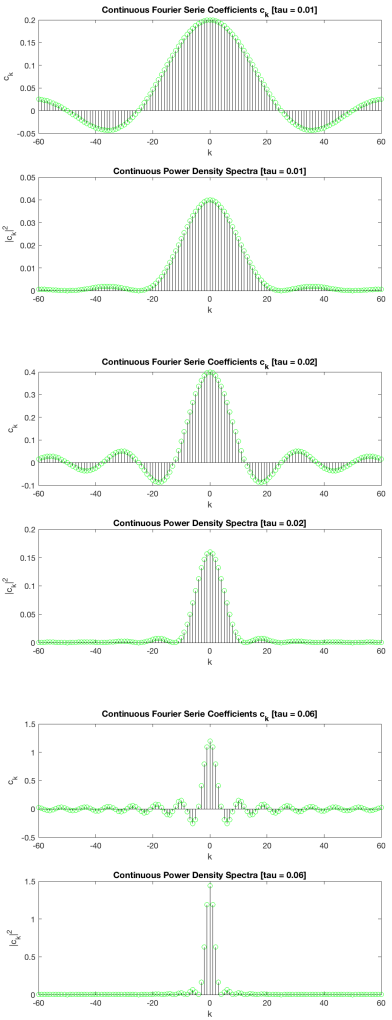

Figure B.2 This is the continuous-time plot of the Fourier coefficients and power density spectrum. The following values are inserted in Eq. 2.29, where $A = 5,\ F_0 = 4,\ T_p = 1/F_0,\ k = 0, \pm 1, ..., \pm 60$. The following τ values are used starting from the top to the bottom, $\tau = 0.01, 0.02, 0.06$

Discrete-time

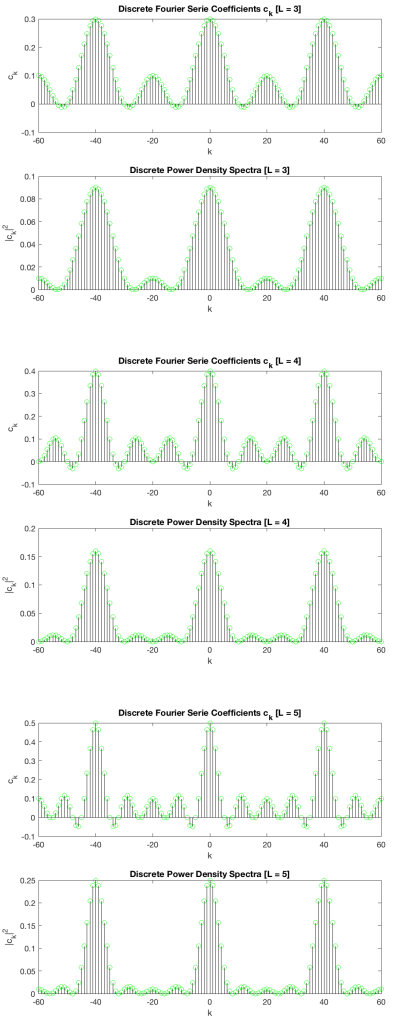

Figure B.3 This is the discrete-time plot of the Fourier coefficients and power density spectrum. The following values are inserted in Eq. 2.29, where $A = 4$, $N = 40$, $k = 0, \pm 1, ..., \pm 60$. The following L values are used starting from the top to the bottom, $L = 3, 4, 5$

Since the discrete-time Fourier transform is periodic the repetition will occur every $N * m = 40 * m$ sample, where $m = \mathbb{Z}$. The period of a discrete-time Fourier transform $X(\omega)$ is 2π

$$X(\omega + 2\pi n) = \sum_{k=-\infty}^{\infty} x(k)e^{-i(\omega+2\pi n)k} = \sum_{k=-\infty}^{\infty} x(k)e^{-i\omega k}e^{-i2\pi nk}$$
$$= \sum_{k=-\infty}^{\infty} x(k)e^{-i\omega k} = X(\omega)$$

Bibliography

ARM (2018). *Overview arm cortex m4*. "https://developer.arm.com/products/processors/cortex-m/cortex-m4". Accessed: 2018-03-20.

Casini, A., G. Castellini, and P. Emiliani (1975). "Sampling rate selection for a digital matched filter". *Proceedings of the IEEE.* **63**:5, pp. 830–831.

D'Amico, A., C. D. Natale, F. L. Castro, S. Iarossi, A. Catini, and EugenioMartinelli (2009). *Unexploded ordnance detection and mitigation, volatile compounds detection by IR acousto-optic detectors*. Editor: James Byrnes. Springer Netherlands, Dordrecht, Netherlands.

Hamamatsu (2011). *Technical information - characteristics and use of infrared detectors*. "http://www.hamamatsu.com/resources/pdf/ssd/infrared_kird9001e.pdf?_ga=2.72066899.1565294432.1515394284-810185650.1515394284". Accessed: 2018-01-08.

Integrated, M. (2001). *Understanding sar adcs: their architecture and comparison with other adcs*. "https://www.maximintegrated.com/en/app-notes/index.mvp/id/1080". Accessed: 2018-02-04.

ISO. *Optics and photonics - Spectral bands*. Tech. rep. ISO:20473-2007.

Johansson, R. (1993). *System modeling & identification*. Prentice Hall, Englewood Cliffs, N.J., United States.

Kester, W. (2005). *The data conversion handbook*. Newnes, Burlington, MA, USA.

Kester, W. (2009). *Understand sinad, enob, snr, thd, thd + n, and sfdr so you don't get lost in the noise floor*. Analog Devices - "http://www.analog.com/media/en/training-seminars/tutorials/MT-003.pdf". Accessed: 2018-01-14.

Kitchin, C. (2013). *Astrophysical techniques, sixth edition*. Taylor & Francis, London, England.

Krzanowski, W. J. and D. J. Hand (2009). *ROC curves for continuous data*. Chapman and Hall/CRC, Boca Raton, FL, United States.

Kuo, S. M., B. H. Lee, and W. Tian (2013). *Real-time digital signal processing - fundamentals, Implementations and Applications*. John Wiley & Sons, Incorporated, Chichester, England.

Levine, R. V. and N. Ara (1999). "The pace of life in 31 countries". *Journal of cross-cultural psychology* **30**:2, pp. 178–205.

Liu, J., B. Zhou, and Y. Zhang (2017). "The method of multiple observations adaptive matched filter for target detection". In: *IEEE Radar Conference (RadarConf17), Seattle, WA, USA, 8-12 May 2017*. IEEE, pp. 1292–1296.

Liu, W. and W. Lin (2013). "Additive white gaussian noise level estimation in svd domain for images". *IEEE* **22**:3, pp. 872–883.

Mahafza, B. R. (2013). *Radar systems analysis and design using MATLAB, third edition*. CRC Press, Boca Raton, FL, United States.

Martinez, A. B. and J. B. Thomas (1986). "Finite length discrete matched filters". *Journal of the Franklin Institute* **321**:5, pp. 251–260.

Mohler, B. j., W. B. Thompson, S. H. Creem-Regehr, H. L. P. Jr, and W. H. W. Jr (2006). "Visual flow influences gait transition speed and preferred walk speed". *Springer, Berlin, Germany*.

Nölling, K., K.-S. Schober, and P. Hoff (2016). *Digitalization in the construction industry - building Europe's road to "construction 4.0"*. Act Magazine, Roland Berger GMBH.

Olsson, G. and C. Rosen (2005). *Industrial automation*. Media-Tryck, Lund University, Lund, Sweden.

Oppenheim, A. V. and G. C. Verghese (2010). *Signals, systems and inference*. Pearson, Harlow, England.

Optoelectronics, O. (2013). *Photodiode characteristics and applications*. "http://www.osioptoelectronics.com/application-notes/AN-Photodiode-Parameters-and-Characteristics.pdf". Accessed: 2018-01-13.

Orton, J. W. (2008). *The story of semiconductors*. Oxford Scholarship, Oxford, Oxford University Press, England.

Proakis, J. G. and D. K. Manolakis (2014). *Digital signal processing, 4th edition*. Pearson New International Edition, Harlow, England.

Salminen, D. (2013). *Adaptive filters applied on radar signals*. MA thesis. UPTEC-F-13042, Uppsala University, Teknisk- naturvetenskaplig fakultet, UTH-enheten, Ångströmslaboratoriet, Uppsala, Sweden.

STM (2018). *Overview stm32f407vg - arm cortex m4 core*. "http://www.st.com/en/microcontrollers/stm32f407vg.html". Accessed: 2018-03-20.

Wittenmark, B., K. J. Åström, and K.-E. Årzén (2016). *Real-time control systems*. Department of Automatic Control, Lund University, Lund, Sweden.

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER'S THESIS

Date of issue
March 2019

Document Number
TFRT-6076

Author(s)
Sebastian Elm

Supervisor
Andreas Winter, Hilti
Rolf Johansson, Dept. of Automatic Control, Lund University, Sweden
Karl-Erik Årzén, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle

Cascade Matched Filtering & Adaptive Threshold Techniques on Optical Tracking Signals under AWGN

Abstract

A cascade matched filtering (CMF) approach is purposed for filtering rectangular pulse train tracking signals, with known signal characteristics, in an embedded signal detection unit. Two adaptive detection algorithms are suggested, adapting to alterations in noise signal characteristics, due to environmental changes. The end product was an embedded tracking detection system, enabled to identify distance and direction to a tracking source, and transmit a well synchronized replica signal back to the tracking source. The suggested filtering and detection method, enhanced the tracking distance of the system significantly.

An introduction to the topic is given in the first section of the thesis, followed by underlying theory on the broad topic. The thesis methodology is explained and tracking detection system with its subcomponents are modeled and described. Then the results from both simulation and embedded real-time data is presented, proceeding with discussions and ultimately conclusions.

Both simulation and embedded real-time data confirms improvements by implementing purposed filtering and detection techniques, by studying the SNR gain from the filter segment, and the ROC of the detection segment. The SNR gain in dB increases logarithmically with each added pulse period to the matched filter, and the linearity of the matched filter is clearly visualized in the filtered output signal. The 3-dimensional ROC curves, illustrates saturations in detection probabilities at a certain threshold, implying a correlation between filter extension and achieving a certain detection probability.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-106

Recipient's notes

Security classification