Code Optimisation Framework for Multicode Sonar Systems

by

M. S. Rashed

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the
requirements for the degree of
Doctor of Philosophy
in Engineering.

Victoria University of Wellington
2020

Abstract

The ocean is a temporally and spatially varying environment, the characteristics of which pose significant challenges to the development of effective underwater wireless communications and sensing systems.

An underwater sensing system such as a sonar detects the presence of a known signal through correlation. It is advantageous to use multiple transducers to increase surveying area with reduced surveying costs and time. Each transducers is assigned a dedicated code. When using multiple codes, the sidelobes of auto- and crosscorrelations are restricted to theoretical limits known as bounds. Sets of codes must be optimised in order to achieve optimal correlation properties, and, achieve Sidelobe Level (SLL)s as low as possible.

In this thesis, we present a novel code-optimisation method to optimise code-sets with any number of codes and up to any length of each code. We optimise code-sets for a matched filter for application in a multi-code sonar system. We first present our gradient-descent based algorithm to optimise sets of codes for flat and low crosscorrelations and autocorrelation sidelobes, including conformance of the magnitude of the samples of the codes to a target power profile. We incorporate the transducer frequency response and the channel effects into the optimisation algorithm. We compare the correlations of our optimised codes with the well-known Welch bound. We then present a method to widen the autocorrelation mainlobe and impose monotonicity. In many cases, we are able to achieve SLLs beyond the Welch bound.

We study the Signal to Noise Ratio (SNR) improvement of the optimised codes for an Underwater Acoustic (UWA) channel. During its propagation, the acoustic wave suffers non-constant transmission loss which is compensated by the application of an appropriate Time Variable Gain (TVG). The effect of the TVG modifies the noise received with the signal. We show that in most cases, the matched filter is still the optimum filter. We also show that the accuracy in timing is very important in the application of the TVG to the received signal.

We then incorporate Doppler tolerance into the existing optimisation algorithm. Our proposed method is able to optimise sets of codes for multiple Doppler scaling factors and non-integer delays in the arrival of the reflection, while still conforming to other constraints.

We suggest designing mismatched filters to further reduce the SLLs, firstly using an existing Quadratically Constrained Qaudratic Program (QCQP) formulation and secondly, as a local optimisation problem, modifying our basic optimisation algorithm.

Acknowledgments

Firstly, I would like to thank Almighty "*Allah*" for giving me strength, consistency and wisdom throughout my life.

Then, I would like to take this opportunity to thank all the people who have contributed to this thesis. I would like to express my sincere appreciation to my supervisors Paul D. Teal and Remmer C. (Marco) Meijer. I am highly indebted to Paul for his patience while I struggled initially, and his consistent and continuous support throughout the conduct of research. I am extremely grateful for his invaluable ideas and thoughts; for helping me develop a mathematical understanding of the design problems, subsequently leading to the optimisation iterations. I have also been truly fortunate to have Marco as my co-supervisor, whose keen, sharp, and analytical mind helped me to carryout this research in a meaningful way. Our one-on-one discussions were really a source of great motivation and thought provoking, giving insights on different angles on the findings of the research. The discussions guided me towards the right direction and more than anything, Marco taught me how to critically analyse my own writing, and then, write succinctly.

I would also like to appreciate the support rendered from the Faculty of Science and the School of Engineering and Computer Science. These include but are not limited to, Patricia Steins, Dr Sara Cotterall, Dr Craig Watterson, Dr Diana Siwiak, Suzan Hall, Tony Mc Loughlin, Monoa Taepa, Varsha Narasimhan, Jasmin Shoukri and Dr Lizzie Towl.

I am very grateful to have had a chance to share my PhD journey with extraordinary friends and colleagues. I would like to thank my long old friend Dr Asim D. Bakhshi for being always available for our discussions on various aspects of the research, despite being thousand of miles away. I would also like to acknowledge the support of my friends here at Victoria University of Wellington, namely Dr M Ali Hashmi, Miss Ayesha , Dr Adeel, and Dr Umar. I would also like to mention my fellow colleagues in the Signal Processing Research Group, namely Ali, Jasmine, Ganlong, Guang Yang, Steven and of course, the chattering Miniruwani.

And last but not the least, I would like to express my deepest gratitude to all my family members, for their unconditional love, support and sacrifice which helped me complete this research in a manner I am proud of.

iv

Dedication

To *my respectable late father **Rashed***, who will always be *my role model*,

my *beloved mother **Musawwer***, in who's heart I always saw ***Allah***,

my *truest friend **Masooma***, who will always be the greatest source of *inspiration and motivation*

and my *lovely daughters **Sahal*** and ***Mishaal***

Abbreviations

ADC   Analog to Digital Converter

AF   Ambiguity Function

AM   Amplitude Modulation

BER   Bit Error Rate

CA   Cyclic Algorithm

CDMA   Code Division Multiple Access

CW   Continuous Wave

dB   decibel

DFT   Discrete Fourier Transform

ECDF   Empirical Cumulative Density Function

FM   Frequency Modulation

GO   Generalised Orthogonal

HCZ   High Correlation Zone

ISI   Inter Symbol Interference

ISL   Integrated Sidelobe Level

LCZ   Low Correlation Zone

LPG   Loss in Processing Gain

MBES   Multibeam Echo Sounder

MLS   Maximum Length Sequence

MMF   Mismatched Filter

MWBE   Maximum Welch Bound Equality

PA   Power Amplifier

PAPR   Peak to Average Power Ratio

PG   Processing Gain

PM	Phase Modulation
PSL	Peak to Sidelobe Level
QCQP	Quadratically Constrained Qaudratic Program
RMS	Root Mean Square
SAS	Synthetic Aperture Sonar
SLL	Sidelobe Level
SMPA	Switched-Mode Power Amplifier
SNR	Signal to Noise Ratio
SVP	Sound Velocity Profile
TVG	Time Variable Gain
UWA	Underwater Acoustic
WBE	Welch Bound Equality
ZCZ	Zero Correlation Zone

Contents

Chapter 1

Introduction

Never say more than is necessary.
— Richard Brinsley Sheridan.

This chapter gives a brief introduction to the thesis. Section 1.1 presents the background problems pertaining to the UWA channel leading to the motivation for this research. This is followed by the aim of this research in Section 1.2. Section 1.3 introduces the important contributions and results from this research. Section 1.4 lists the publications arising from this research. Section 1.5 describes the organisation of this thesis.

1.1 Motivation

More than 70 % of the earth is covered with water, and yet we know very little regarding the bottom of the oceans due to the extremely challenging nature of the underwater environment. Despite these challenges, underwater sensing is important for various applications including military [1], oil and gas exploration [2], oceanography [3], fish-finding [4] and seismic monitoring [5] to name a few. There has been a rapid growth in the maturity and reliability of underwater sensing and communication technologies in the last few decades [6].

Underwater sensing system designs prefer the use of acoustic waves over radio and optical waves to transmit and receive data [6]. Radio waves are subject to severe attenuation due to the conducting nature of the medium, especially in the case of seawater. The frequency-dependent nature of underwater absorption also limits the use of radio waves. Electromagnetic waves have the advantage of the speed of propagation which could favour their use [7]. However, the severe attenuation negates this advantage, limiting the distance these waves can travel. Optical waves

suffer heavily from scattering in the water channel due to suspended particles as well as heavy attenuation in the optical frequency band [8], restricting their use to shorter distances as well.

The use of acoustic waves, although the preferred choice, is also challenging in its own way. Despite considerable research over the years, UWA channel modelling is a challenging problem [9]. The underwater acoustic channel has limited capacity and slow propagation speed [10]. The slow speed of acoustic waves results in propagation delays for data to be transmitted across the medium [11]. Then there is multipath propagation [12] due to the transmitted signal traversing the medium through more than one path. One of the causes of multipath propagation is refraction since the speed of sound changes with the salinity, the temperature and with the depth. Another factor is reflection from the surface and seafloor. Multipath signals lead to inter-symbol interference at the receiver.

Acoustic waves suffer from attenuation due to both geometrical spreading loss and absorption [13, 14]. For reasons of simplicity, the spreading loss is often modelled as either spherical or cylindrical depending on the system geometry. Absorption is frequency-dependent; higher frequencies suffer a greater loss than lower frequencies.

Then, the underwater channel is extremely incoherent; the channel properties vary with time. The coherence time varies from a few tens of milliseconds to a few seconds [15–18]. Even within the coherence time, underwater channels still exhibit random variations that cannot be predicted or modelled.

Noise is another factor that affects underwater communication. This may be the ambient noise or site-specific noise [17]. Sonar systems are usually designed to operate within the frequency range of 1 kHz to 1 MHz. Within this bandwidth, the dominant sources of noise in shallow water coastal areas are ship transit noise, sea-surface agitation [19] and noise due to snapping shrimp [20, 21]. These create a situation where the classical assumption of a Gaussian noise distribution often does not hold [22].

Movement of the source and/or the target results in a Doppler shift of the transmitted signal [23]. The Doppler effect in sonar is more severe than in radars, due to the slow speed of acoustic waves as well as the flow of the water. The underwater acoustic channel is wideband in the sense that the fractional bandwidth, defined as the ratio between the bandwidth to the carrier frequency, is much higher than a comparable RF communication network [23]. This bandwidth contributes to the delay spread of the signal [24].

The amplitude of the received signal is considerably attenuated, and there are changes in the frequency as well as the phase. This leads to poor detection at the receiver end. In the case of a matched filter based receiver system, this causes an increase in the correlation sidelobe levels (SLLs)[1], limiting the range and the Bit Error Rate (BER) of the system. In the case of sonar, this

---

[1]The matched filter and correlation sidelobe levels are introduced in detail in Section 2.2.3.1

implies reduced ability of the sonar to detect targets that have weak reflections.

Use of multiple transducers adds another dimension to the complexity of the system. There is the obvious advantage of increased surveying area, with reduced surveying costs and surveying time. However, the multiple codes in the code-sets are required to be mutually orthogonal to minimise interference in the regions of overlap.

All the above mentioned factors become more pronounced in the scope of a multicode sensing system, where the codes are not only required to have only good autocorrelations but also low crosscorrelations with respect to other codes. The correlation SLLs are limited by a theoretical bound presented by Welch in [25]. The bound lays a limit to the minimum value of the highest SLLs. Signals from a sonar reflected back from the seabed and from a fish near the seabed can have an energy difference of about 40 dB [26–28]. This lays down a limit on the number of codes in the set of codes and the (minimum) length of each code that can be used in order to detect a fish near the seabed for a multicode system.

The above factors invite considerable interest in the field of UWA sensing systems and forms the motivation for research on optimal codes for a multicode sonar system in this thesis.

1.2 Aim

The aim of this thesis is to propose a general framework for optimising sets of signals (codes) for use in the UWA channel. The research focuses on optimisation sets of codes, for lower crosscorrleations and autocorrelation sidelobes. A specific aim of this research is to suggest a framework for code optimisation in a multicode sonar system, conforming to a number of constraints. It also leads to suggesting ways to achieve SLLs beyond the Welch bound [25] in some respects.

1.3 Contributions

The research in this thesis develops further the novel work commenced in [29] to optimise sets of codes having desired correlation properties under various constraints. Major contributions of the thesis are listed below:-

a. A gradient-descent based algorithm for optimising correlations of sets of codes is presented in Chapter 4. These optimised codes from the algorithm have a narrow, sharp peak at zero-offset with flat and low SLLs across the entire range of offsets. The optimisation algorithm is flexible to include a number of design constraints on the set of codes.

b. A compromise is suggested and the algorithm is modified for optimising the set of codes to have SLLs that are in some cases beyond the Welch bound [25]. The findings of this chapter have been submitted to IET – Radar, Sonar & Navigation and are under review.

c. TVG causes the stationary white noise to transform into non-stationary white noise. In Chapter 5, we compare the SNR improvement at the output of a matched filter with a filter designed for the non-stationary noise. We show that although there is very slight gain in SNR with the designed filter, the degradation of the correlation sidelobes is considerable. Findings from this chapter have been accepted for publication in IET - Radar, Sonar & Navigation.

d. A mathematical model to simulate Doppler shift is presented in Chapter 6. This model is typical to an UWA channel and used to optimise sets of codes to be robust against the Doppler shift while still maintaining low correlation SLLs. Comparisons are drawn with other known Doppler tolerant codes such as logarithmic chirps. The ambiguity functions of the Doppler tolerant optimised sets of codes are also compared. Findings from this chapter have been accepted at IEEE/MTS Oceans 2019 Marseille and published as part of the technical proceedings.

e. The SLLs from a matched filter can still be high while maximising SNR in white noise. The SLLs can be reduced by using a "mismatched" filter. In Chapter 7, we suggest designing a mismatched filter for the Doppler tolerant codes using two different approaches. The optimisation can be formulated and solved using convex optimisation tools. We suggest an alternate method by adopting an gradient-descent optimisation algorithm. Findings based on the design of mismatched filter from this chapter have been accepted at IEEE/MTS Oceans 2019 Seattle and published as part of the technical proceedings.

1.4   Publications from this Research

- Rashed, M. S., Meijer, M., & Teal, P. D. (2019, October). Mismatched Filtering of Doppler Tolerant Codes for Multicode Sonar Systems. In OCEANS 2019 MTS/IEEE Seattle (pp. 1-10). IEEE.

- Rashed, M. S., Meijer, M., & Teal, P. D. (2019, June). Doppler Tolerant Code Optimisation Scheme for Multicode Sonar Systems. In OCEANS 2019 MTS/IEEE Marseille (pp. 1-9). IEEE.

- Rashed, M. S., Meijer, M., & Teal, P. D. (2019). Performance of the Matched Filter in Sonar Systems having Time Variable Gain. IET Radar, Sonar & Navigation, (In Press).

- Rashed, M. S., Pratap R., Meijer, M., & Teal, P. D. A Novel Optimisation Scheme for Multicode Sonar Systems. IET Radar, Sonar & Navigation, (Under review).

1.5   Organisation of Thesis

The rest of the thesis is organised as follows. Chapter 2 familiarises the reader with the fundamentals of UWA channel characteristics, and its effects on a sonar system. Chapter 3 presents a review literature on various aspects of code opt, incl correlation bounds, code opt methods, opt codes, and methods to reduce SLLs with with reference to a multicode sensing system. Chapter 4 presents the novel gradient-based algorithm to optimise sets of codes to have flat and low SLLs across the entire range of offsets for various optimisation constraints. The algorithm is modified to achieve SLLs below the Welch bound [25].

Chapter 5 covers the SNR gain of the optimised codes of this research in a TVG scenario. We also investigate the sensitivity of the correlations SLLs due to application of the TVG. Chapter 6 introduces robustness of the optimised codes of this research to the Doppler shift. The performance of the optimised codes is discussed with reference to correlations SLLs and Ambiguity Function (AF). In Chapter 7, two methods to design a mismatched filter to further reduce SLLs are investigated and results are presented. Important conclusion from this research and suggested future work are presented in Chapter 8.

Chapter 2

Underwater Acoustics and Sonar Systems

'If you cause your ship to stop, and place the head of a long tube in the water and place the outer extremity to your ear, you will hear ships at a great distance from you'.

Leonardo da Vinci

Since the year 1490 when the concept was first presented by Leonardo da Vinci [30], sonar systems have evolved to become the advanced underwater acoustic systems of today. Their initial utility was exclusively military until after World War II. Sonars are now used in diverse applications such as fish-finding and fish-herding [31], fish population estimation [32], oil and mineral exploration [33, 34], oceanography [35], bathymetry [36], environmental monitoring [37], seismic measurement [38], biological signal [39] and noise measurement [40].

The development of sonar systems is, however, affected by a range of complexities. These include the saline nature of water and its peculiar response to pressure waves [41], channel incoherence [42], specific radiation patterns of electroacoustic transducers [43], range and resolution limitations, non-linearity of high power transmitters [44] as well as the Doppler effect [45] due to motion of the source and / or target.

In this chapter, we discuss the fundamentals of acoustics in an underwater environment. We also discuss the basics of sonar systems and how different aspects of underwater acoustics affect a sonar system, culminating into the Sonar Problem before concluding with a summary.

2.1 Fundamentals of Underwater Acoustics

To develop an understanding of UWA communication systems, it is necessary to study the nature of sound and the various factors affecting the propagation of sound in the water. These factors are quite disparate and may be categorised as either regular factors or random factors. Regular factors are those that can be estimated and have a deterministic relationship, such as the speed of sound or the transmission loss. Random factors are those factors that are difficult to model and occur randomly. These include the Doppler effect, multipath propagation, ambient noise, reverberation, scattering and other such phenomenon. While most of the factors are pertinent to all kinds of UWA communications, some of these are pertinent to Synthetic Aperture Sonar (SAS) only. Both regular and random factors will be discussed briefly in the next subsections with reference to the utility in sonar systems.

2.1.1 Speed of Sound

Sound propagates four times faster in water than in air but is still five orders of magnitude slower than the speed of electromagnetic waves [46]. The speed of 29 sound increases with an increase in the water temperature, the salinity and with the pressure. Near the surface of the sea and far from rivers, the effects of temperature are predominant and salinity does not vary much. In deeper channels, the effects of salinity and temperature on the speed of sound are quite minimal but increasing pressure with an increase in depth has the largest effect. An appropriate formula for estimating the speed of sound as a function of these factors is given by Mackenzie [47, 48]. The effect of all these factors results in the sound of speed varying from 1450 m/s to 1560 m/s [49]. Figure 2.1 shows the Sound Velocity Profile (SVP) through the depth of an example underwater channel.

2.1.2 Transmission Loss

The acoustic signal suffers a loss in amplitude during its course of propagation in water due to scattering, spreading and absorption within the environment. The acoustic waves may be scattered away from the intended path due to the existence of debris in the water channel, salt, air-bubbles in the water, SVP and non-uniformity of the seafloor. Scattering causes some of the scattered paths to be directed away from the target, resulting in loss of energy. Loss due to scattering is frequency-dependent.

Spreading loss is caused due to the effect of geometrical spreading when an acoustic wave propagates away from the source. This can be seen as a reduction in the intensity (pressure squared) of the acoustic wave as it travels through the water channel. The geometrical spreading
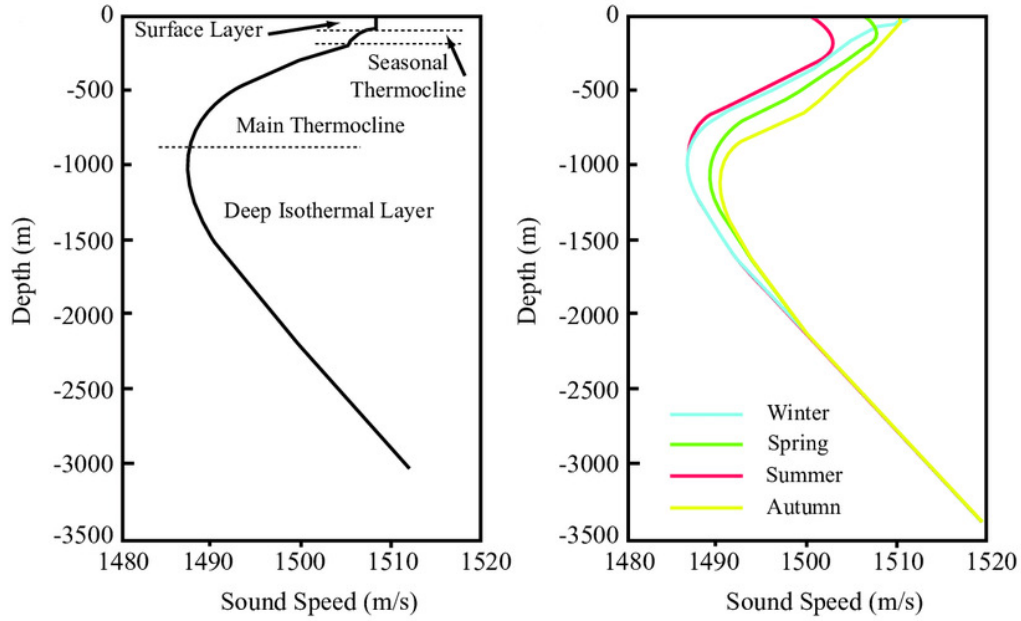
Figure 2.1: Sound velocity profile (SVP) in the underwater channel [50].

is a complicated phenomenon but for simplicity, may be categorised as cylindrical or spherical spreading. A typical shallow underwater acoustic channel predominantly exhibits cylindrical spreading. The wavefront spreads horizontally and the loss is directly proportional to $\frac{1}{R}$, where R is the distance between the source and the target. A deep underwater channel is usually characterised by spherical spreading where the spread is omni-directional [8] and the loss is proportional to $\frac{1}{R^2}$. The spreading loss is a frequency-independent loss.

The acoustic waves are also subjected to absorption when propagating through the underwater channel. This absorption, like scattering, is also frequency-dependent. Low-frequency acoustic waves (1-10 Hz) can propagate up to thousands of kilometres before suffering considerable attenuation due to absorption. In contrast, high frequency acoustic waves are attenuated more quickly, limiting their propagation distances to a few hundred metres [30]. For high frequencies, the absorption constant α_{water} increases monotonically by about four orders of magnitude from 30 Hz to 300 kHz and quadratically with frequency thereafter [35].

The attenuation loss due to absorption is attributed to the fact that the acoustic energy is transformed into heat, primarily due to complex chemical processes and minerals in the underwater channel. The absorption coefficient α_{water} increases with both distance and frequency [51] and is expressed in dB/km. [52] gives a simplified formula for calculating the absorption loss for a given frequency, range, temperature, salinity and depth of an underwater channel.

Loss due to scattering cannot be modelled easily. An empirical model that can predict scattering in all scenarios is difficult and may not be accurate. Spreading and absorption loss, then, account for the majority of the transmission loss. Mathematically, the one-way transmission loss for a shallow water channel (cylindrical expansion) can be described as

$$L_{\mathrm{T}} = 10\log_{10}r + \alpha_{\mathrm{water}}r \qquad\qquad (2.1)$$

where r is the transmission range in km, α_{water} is attenuation coefficient expressed in dB/km, and L_{T} is in dB.

For short ranges or close targets, the terms r in (2.1) are replaced by r^2 to account for spherical spreading. At the receiver end, a suitable Time Variable Gain (TVG) based on (2.1) is often used to compensate for the transmission loss in the received signal[1].

2.1.3   The Doppler Effect

Among the intricacies of the complex underwater channel is another phenomenon that affects the waves travelling from the source to the target and received back after reflection. A moving source and/or target along with the movement of the water in the channel induces a Doppler shift [41, 53, 54] to the signal. A transmitted signal from a radar or a sonar in such a scenario undergoes changes in both the frequency and time domains: a change in the carrier frequency and bandwidth of the signal, which causes a corresponding time-scale dilation or contraction[2] The Doppler effect changes the frequency and phase so that the received signal is different from the transmitted signal in its carrier frequency as well as bandwidth and its length in the time domain. This results in a degraded correlation between the received and transmitted signals meaning higher sidelobes.

For a signal being transmitted with frequency f_{c}, the relative frequency change due to Doppler shift is simply

$$f_{\mathrm{d}} = \frac{v}{c} \times f_{\mathrm{c}} \qquad\qquad (2.2)$$

where v is the relative velocity of the source and/or target, c is the speed of the acoustic wave in the UWA channel, and f_{d} is the resultant relative frequency change.

The slow speed of the acoustic waves makes them vulnerable to relatively large Doppler shifts because of the fact that the ratio $\frac{v}{c}$ is large as compared to electromagnetic waves. The Doppler effect also causes a relative change in bandwidth. The transmitted signal was centred at f_{c} but the received signal is distributed between $f_{\mathrm{c}} - f_{\mathrm{m}}$ to $f_{\mathrm{c}} + f_{\mathrm{m}}$, where f_{m} is the largest

---

[1]TVG and effects of TVG on the processing of the received signal is discussed in detail in Chapter 5.

[2]Robustness to Doppler induced stretching is investigated in Chapter 6.

value of f_d due to fastest velocity v. The change in the carrier frequency of the modulated signal may be compensated for in the receiver, but only if there is only one Doppler shift present. Optimised codes with low autocorrelation SLLs provide good range resolution. The improved range resolution reduces the signal-to-reverberation ratio in the presence of the Doppler effect [55, 56], improving the signal detection and supporting use of optimised sets of codes.

2.1.4 Multipath Propagation

Multipath propagation occurs when a signal propagates from a source to a receiver along several different paths. In an underwater channel, the direct path is complemented by several other propagation paths. In shallow water channels, these paths are results of reflections at the surface of the sea and the seabed. In deeper channels, reflections from the surface and the bottom have a negligible effect. The acoustic waves travel through the water channel, with temperature, salinity and depth varying along its path of propagation. The acoustic waves are affected by varying speed of sound in each part of the channel resulting in refraction at different angles and increase of multipaths.

Multipath propagation results in a received signal that is dispersed in time. Each path has its own delay and the time dispersion leads to a form of intersymbol interference. Delay spread is a measure of the multipath profile of a wireless communications channel. Delay spread is a random variable, and the standard deviation is a common metric to measure it [57]. Due to the slow speed of sound, the channel delay spread is quite significant for even a short range. Typical underwater channels have a delay spread of a few ms and but this can be as large as 50–100 ms [58]. At the receiver end, arrivals from these different paths may result in constructive or destructive interference and cause Inter Symbol Interference (ISI).

2.1.5 Reverberation

During propagation, an acoustic wave scatters due to various inhomogeneities within the channel and at the peripheries of the channel, that is, the surface and the bottom. These inhomogeneities form discontinuities in the physical properties of the channel and thus, re-radiate a portion of the acoustic energy incident upon them [30]. This results in the transmitted wave being reflected back by many other sources besides the prospective target. Any reflection other than the target is of no interest in the sonar when received.

The re-radiation of acoustic waves is called scattering and the total scattering from all scatterers accounts for reverberation. The scattering due to scatterers within the volume of the channel such as the marine life and particles in the water is termed volume reverberation [59]. Scattering due to scatterers located on or near the surface of the UWA channel is termed surface

reverberation. Similarly, the bottom reverberation is due to scatterers on or near the seafloor. Together, the total sum of all reverberation results is signal related interference which is a major concern for high-power active sonars.

The reverberation is assumed to be frequency independent for a rocky sea-floor. Soft-bottom surfaces like sand and silt bottoms result in reverberation which is frequency-dependent. There is an increase in volume reverberation at the rate of 3 dB/octave [60, 61].

2.1.6 Noise

Noise is another factor affecting signal sensing and is characterised as an unwanted signal. In an underwater channel, the noise is classified as ambient noise and site-specific noise [17]. Ambient noise is present everywhere, whereas localised noise is site-specific. Ambient noise includes thermal noise due to movement of water particles, noise due to shipping, noise due to waves and winds at the surface of the sea, turbulence, as well as flow noise [62] due to water flow around the hydrophone surface. Cavitation [63] is another source of ambient noise, resulting from the movement of the propellers of a boat. Cavitation is frequency dependent and an important factor as it necessitates use of long sonar signals for increased energy.

The ambient noise in an underwater channel is often modelled with Knudsen curves [64, 65] which shows a decrease in the ambient noise of about 18 dB per decade of frequency [66, 67]. Similarly, the wind-sea noise is related to the sea states, so that a higher sea state with vigorous breaking waves results in an increase in ambient noise. Thermal noise starts becoming prominent at a frequency of 10 kHz, increasing by about 6 dB per decade of frequency and becomes the dominant source of ambient noise at frequencies greater than 160 kHz. A detailed account of ambient noise and how it affects underwater acoustics is given in [68].

Site-specific sources of noise include noise due to factors local to the UWA channel. These may include breaking waves, noise due to flow of water under the boat, engine noise, self-noise of transducer and electronics, noise due to the water temperature, rain, seismic activity, wind speed as well as marine biological effects.

2.1.7 Dispersion

Acoustic dispersion is the separating of a sound wave into its constituent frequencies while propagating through a medium. The phase velocity of the sound wave is a function of the frequency, i.e., the sound waves of different frequencies propagate at different phase velocities. The separation of component frequencies is then measured by the rate of change in phase velocities as the sound waves propagate through a given medium [69]. The frequency-dependent dispersion is known to follow the Kramers-Kronig relation [70]. The effect of acoustic dispersion in an UWA channel is relatively insignificant for narrowband signals, but it can become a factor for broadband signals or widely-separated narrowband signals [71]. Moreover, the acoustic dispersion in an UWA channel is independent of the range of travel of the transmitted signal [72].

In this thesis, we will be ignoring the effect of dispersion. However, future work may focus on incorporating the effect of dispersion on code-design for multicode sonar systems.

2.1.8 Speckle

Speckle is the name given to the mottled or "speckled" appearance often evident in the displayed output of a coherent imaging system. The seafloor, along with the vast majority of natural and synthetic surfaces, is often rough on the scale of underwater acoustic wavelengths [73]. When an acoustic wave is incident on a rough surface such as the seafloor, the incident waves are scattered in a complex manner. The returning reflections to the hydrophone consist of many coherent wavelets, each reflected from a different microscopic element of the rough surface [74]. These waves add coherently, and so introduce interference. The resultant image formed is marred due the interference, as granular spots; bright at points of constructive interference, dark at points of destructive interference, and varying in between these two extremes. The granular pattern in the image is what is termed as speckle.

Speckle was initially associated with optics and images formed when the target was illuminated by light or laser. However, the acoustic speckle inherits similar characteristics. Whereas speckle may be used for learning about target characteristics [75], most sonar applications tend to reduce the speckle, incorporating various speckle reduction techniques, see [76–79].

Speckle remains an undesirable effect in UWA communications including synthetic aperture sonar (SAS), regardless whether they are narrow-band or broad-band. However, speckle and its effect on the optimal sets of codes designed in this research have not been investigated.

2.1.9   Scatterers

In an imaging sonar such as an SAS, every image is composed of an array of images received from multiple swathes. An image may be composed of reflections from one dominant (or point) scatterer, which is termed a *Persistent Scatterer (PS)*. Reflections from other smaller targets introduce additional noise called *clutter*. At the same time, an image may be composed of several reflections from a number of targets with similar target strengths. There is no dominant scatterer in such an image, and the image is said to be composed from *Distributed Scatterers (DS)*. Most man-made features are PS, and there is a high density of PS in urban areas. PS can also be found in mountainous terrain, in the form of large rocks and boulders.

The optimised sets of codes proposed in this thesis have been designed targeting PS. In the case of DS, it is anticipated that there will be reduced interference between the sub-images through the use of optimised codes. The lower crosscorrelations would ensure that reflections from other codes will not contribute to constructive coherent interference, and so there will be reduced speckle. However, this aspect has not been investigated in this research, but recommended as a future work subsequent to this research.

2.2   Sonar Systems

Systems which employ underwater acoustic energy for observation are called sonar systems [80]. The word SONAR stands for SOund, NAvigation, and Ranging, coined for the first time in World War II. Sonar is defined as a technique for detecting and determining the distance and direction of underwater objects by acoustic means [81]. A simple sonar system may be divided into four elementary parts. The first is a source that transmits an acoustic wave through the medium which is water. It may be a transmitter and a projector that converts electrical energy into acoustic energy, or the signal could be created by the object of interest. The second part is the hydrophone that receives the acoustic energy and converts it to electrical signals for processing. The third part is a signal processing system that transforms the electrical signal to allow meaningful interpretation. The fourth part is the display where the information is presented.

As alluded to above, sonars systems may be classified as *active* or *passive* sonar systems. Figure 2.2 shows an active and passive sonar system in their simplest form. In a passive sonar, the receiver hydrophone relies only on the acoustic energy radiated by the target itself, and thus, only "listens"; transmission through water, from the source to the receiving sensor is one way. In *active* sonar, the acoustic energy is transmitted by a projector, which is part of the sonar itself; the energy travels through the water until it comes across an object which causes it to be reflected, with some dissipation of energy. The reflection is received back at the hydrophone and converted
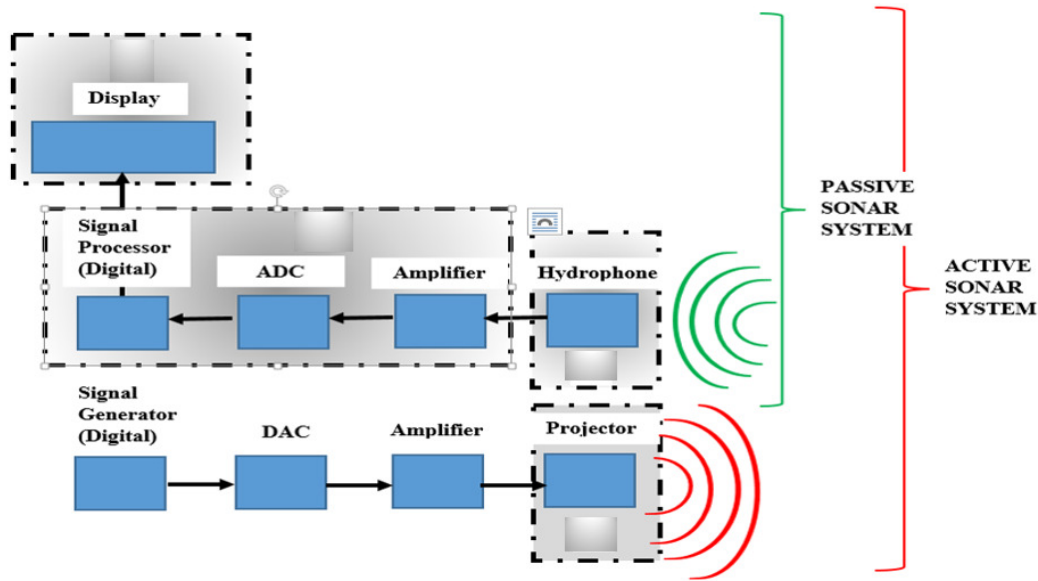
Figure 2.2: A simplified active and passive sonar system.

into an electrical signal. Detection in the receiver is performed by correlating the received signal with a copy of the transmitted signal. The output of the detector indicates the presence or absence of a reflection. This thesis is concerned only with active sonar systems.

2.2.1 History of the Sonar

The first step towards developing an active sonar system was the ability to generate sound in water. Water has an acoustic impedance almost 3500 times higher than air. This initially prohibited sound generation in an underwater environment. But the real breakthrough was the discovery of piezoelectricity by Pierre and Jacques Curie in 1880 [82]. Certain crystals such as quartz have the ability to generate electricity under physical stress and conversely are deformed when an electrical charge is applied. This led to the use of piezoelectric transducers for underwater acoustic wave generation.

Detecting objects in an underwater environment using acoustic waves was a very popular topic especially after the sinking of Titanic in 1912. At the time, theoretical principles of sonar were being widely researched but no one could make a physically workable device to be used on a ship. The first patent for a passive sonar was filed by Lewis Richardson [83], a year after the sinking of the Titanic. Paul Langevin developed the first prototype of an active sonar, but his equipment was unpractical being very heavy and bulky and never used on any ship.

Soon after the WWI broke out, the British and French governments sent a high-level delegation to Washington led by Ernest Rutherford. Rutherford returned and initiated the research on submarine defences. He concluded that acoustic detection was the way forward. Throughout September 1915 – December 1916, Rutherford continued as a member of a joint British-US anti-submarine committee, before returning back to his nuclear research. Around the same time, in 1916, Canadian physicist Robert Boyle and his team started working on a practical realisation of a sonar under the British Board of Invention and Research. Working with French physicist Paul Langevin, they were able to produce a prototype for testing just before the First World War ended.

Boyle used quartz piezoelectric crystals, which made the design compact and lightweight compared to Langevin's earlier prototype. The practical design also had a better resolution. This was the first sonar to be mounted on a warship [84]. To maintain secrecy, the acronym used to describe the early work was "ASD"ic, "ASD" for "Anti-Submarine Division", suffixed by "ic" from supersonic; hence the British acronym ASDIC. Boyle could not publish his findings due to the secret nature of the research. Paul Langevin did file a patent for an active sonar and invention of modern-day active sonar is thus attributed to Langevin [85].

Sonars generally utilise frequencies ranging from infrasonic for seismic [86] and navigational purposes to ultrasonic for short and medium-range sonars used for mine-hunting [87, 88].

2.2.2 Sonar Transmitter System

The information carried by the electric signal in the baseband frequency is not feasible for direct transmission. The information must be translated to a higher carrier frequency. The modulation is done by using a mixer that accomplishes the modulation of the baseband signal onto a carrier frequency. The modulated signal is then amplified through a power amplifier. In a sonar system, the modulated and amplified electric signal is converted into pressure waves by using an electroacoustic transducer, or a projector. The sonar transmitter completes the transformation of the information into a meaningful acoustic signal for transmission in an UWA channel. The following subsections elaborate on the pertinent details of a transmitter for a sonar system.

2.2.2.1 Electroacoustic Transducers

A transducer forms the central part of any sonar system. Transducers are used both for transmitting and receiving the information in an underwater acoustic channel. The basic principle of transduction is the conversion of energy from one form to another. Electroacoustic transducers convert acoustic energy into an electrical signal and this transduction is reversible [89].

Loudspeakers and microphones are common examples of acoustic transducers.

An electroacoustic transducer in transmitting mode vibrates its surface due to an applied electrical signal. The vibrating surface, in turn, transfers the movement to the molecules in the medium, generating acoustic waves. In receiving mode, the surface of the transducer is made to vibrate due to incident acoustic waves. The vibrating surface of the transducer, in turn, produces an electric signal corresponding to the pressure of the incident acoustic waves.

The beam pattern or radiation pattern of a transducer is a characteristic function of spatial angle. Factors affecting the beam pattern of a transducer include the operating frequency, size, shape and the acoustic properties of the vibrating surface. The beam patterns are reciprocal, which implies that the beam pattern does not depend on whether the transducer is used as a projector (transmit transducer) or hydrophone (receive transducer). Transducers may be designed to radiate as omnidirectional, radiating in all directions or as a narrow beam, radiating in a single direction, by exploiting the transducer radiation patterns.

2.2.2.2 Power Considerations

Sonar systems often use high power transmitters for transmission of signals over long distances and to counter the attenuation properties of seawater. There are two ways these high power transmitters affect the signal.

First, the power supplies associated with high power transmitters have power disturbances which are prominent whenever a high power-system changes its state from one stable steady-state to another stable steady-state. This is predominantly evident in the form of a transient as a non-constant voltage, at the start of a transmission [44], when the power-supply changes from zero-power state to full-power state.

The transient affects the initial few samples of the signal (depending on the length of the transient) by applying a non-constant voltage. For the duration of the transient, the affected samples of the signal are not transmitted with the same magnitude as the rest of the samples. Depending on the length of the transient, once the power supply has stabilised and is providing a constant power, the samples of the signal are transmitted with a constant magnitude. This entails a special requirement for the codes to have a specific power shaping. The initial few samples (depending on the length of the transient) are ramped up from zero amplitude so that the magnitude of the voltage transient is minimised.

Second, the transducer is a resonant device. At the end of the transmission when the power is switched off, there is an oscillatory ringing effect. These oscillations are directly dependent on the Q rating of the transducer. A higher Q rating implies longer ringing at the end of the transmission. Transducers with low Q have reduced ringing with fast rise and fall times, and

(a)   (b)
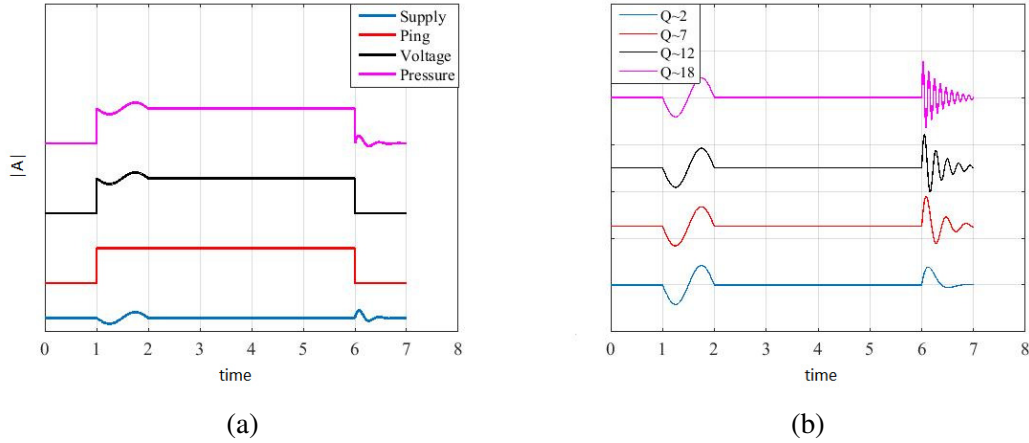
Figure 2.3: Power considerations associated with high power transmitters. (a) Transient at the start of the transmission. (b) Ringing effect at the end of the transmission; for various values of Q.

common transducers typically have a $Q > 2$. To minimise the effect of ringdown, the last few samples of the code can be ramped down at the end of the transmission. The transient at the start of the transmission and the ringing effect at the end of transmission are shown in Figure 2.3. A typical target profile for the magnitude ramps up initially to minimise the transients, remains stable for the rest of the duration of signal before ramping down to minimise the ringing effect.

Peak to Average Power Ratio (PAPR) is another consideration for power constraints of a signal. PAPR is the ratio of the peak power of the signal to the average power of the signal. The efficiency of a power amplifier is maximised when PAPR is 1 or close to 1. A high PAPR requires an amplifier having a large dynamic range which typically leads to high-power consumption with low efficiency. A high PAPR also pronounces the effects of non-linearity of an economical amplifier, thus forcing a requirement to use a high-power amplifier which is not only less efficient but also more expensive.

Processing Gain (PG) is another reason for keeping PAPR close to 1. If the transmitted power is kept fixed, but the signal duration is increased, the signal energy is increased and so an increase in the SNR is generated. This increase in SNR is called PG. Large PG allows greater sensing range with the same transmit power. The PG can only be obtained in the receiver if the signals are transmitted and received accurately, and this is best achieved using a signal having a PAPR close to 1.

2.2.2.3 Power Amplifiers

The Power Amplifier (PA) is another key element of transmitter systems. The purpose of the PA is to increase the signal power before transmission. Power amplifiers are classified based on their circuit configuration, method of operation, the efficiency of converting the input to output power and the linearity of operation. Sonars system transmitters generally use a Switched-Mode Power Amplifier (SMPA). These power amplifiers are very efficient as they dissipate minimal power during switching. Being either fully-on or fully-off, SMPAs significantly reduce power losses so that the efficiency is theoretically rated at $90 - 95$ %.

The two main classes of SMPAs include Class D and Class E PAs [90]. Both of these power amplifiers are highly efficient. Class D power amplifiers suffer from a higher switching loss and hence are not suited for high-frequency applications sonar applications. Class E power amplifiers are more suited to high-frequency applications due to their soft-switching property (switching on when the voltage is zero and switching off when the current is zero) [91], and have minimal switching loss. However, class E power amplifiers have the disadvantage of being constrained to the frequency and bandwidth for which they are designed. The choice of PA thus depends on the intended type of application.

2.2.2.4 Modulation

The modulator is responsible for mixing the baseband signal carrying information, as either an amplitude (Amplitude Modulation (AM)) or phase (Phase Modulation (PM)) property of a sinusoidal wave at a much higher frequency [92]. It may be noted here that Frequency Modulation (FM) is a special case of PM. Each type of modulation has its integral characteristics and may be chosen for optimal performance of a system. The baseband signal can be a voltage or current varying analog signal or a digital signal consisting of a sequence of binary bits or digital words. An analog modulated signal is generally represented as

$$v(t) = a(t) \, \cos(2\pi f_c t + \phi(t)) \tag{2.3}$$

where a is the amplitude, f_c is the carrier frequency, ϕ is the phase of the modulated signal. All of these may or may not be a function of time, depending on the type of the modulation scheme.

2.2.3 Sonar Receiver System

The receiver of a sonar consists of a hydrophone that receives the reflected signal. The hydrophone is an electroacoustic transducer which is a reversible transduction device as already discussed in Section 2.2.2.1. The received acoustic wave is converted to an electrical signal by the hydrophone,

amplified by a pre-amplifier, subjected to an appropriate TVG, and mixed down (and lowpass filtered) to baseband signal before being processed. The processing in the receiver is done through a correlator, which matches the received signal with a copy of the transmitted signal. The output is then used for further processing such as target detection or classification or range estimation.

2.2.3.1   Matched Filtering

In a modern sonar system, the received signal may be subjected to pulse compression with a matched filter [93–95]. A matched filter maximises the peak SNR of the received signal in the presence of white, stationary noise. The received signal consists of the reflection of the transmitted signal as well as noise. The received signal is scaled in amplitude, delayed and possibly Doppler shifted. Matched filtering is a process for amplifying a known signal from a received signal which has been corrupted by substantial noise. This is achieved by maximising the SNR of the signal being detected [54, 96, 97].

Correlation is a measure of how closely (or otherwise) signals are related [98]. The impulse response of the matched filter **h** is simply the time-reversed, complex conjugate of the transmitted signal. A matched filter correlates the received signal with a clean copy of the signal to produce an output which is then subjected to a detection threshold.

In case of a sonar operating with multiple signals, the output of the matched filter is an autocorrelation with the required signal and crosscorrelations with all the other signals. Based on the correlation levels, a decision is made and the signal strength may be displayed based on whether the filter output exceeds a threshold. For non-interfering signals, good correlation properties are characterised by an autocorrelation with a narrow mainlobe at zero-offset, and all SLLs as low as possible. To minimise interference, the signals in a set of code are required to be mutually orthogonal. This implies the crosscorrelations for all offsets are also as low as possible.

2.2.4   Sonar Equation

The SNR in a sonar system is determined by the sonar equation, which in its simplest form is

$$D_T \lessgtr \left(\frac{P_s}{P_v} \right) \tag{2.4}$$

where D_T is the decision threshold, P_s stands for signal power, and P_v stands for noise power.

The passive sonar action in decibel form is

$$I = L_S - L_T - (L_N - I_D) \lessgtr D_T \tag{2.5}$$

where I represents the echo Intensity level in dB.

For an active sonar, the sonar equation takes the form

$$I = L_S - 2L_T + S_T - (L_N - I_D) \lessgtr D_T \tag{2.6}$$

The passive terms $L_S - L_T$ are replaced by the terms $L_S - 2L_T + S_T$, where S_T is called Target Strength and represents the strength of the target to reflect energy. The term L_T is doubled to account for twice the range that the signal travels from the source to the receiver.

A detailed discussion of the different terms in equation (2.5) and (2.6) is given in [35] and a brief explanation of the terms is given below:

- L_S: Source Level is the signal pressure at the source. For an active sonar, it is at the projector, and for a passive sonar, it is the target source level.

- I_D: Directivity Index captures the directionality of both a projector and hydrophone. In the case of a passive sonar, it represents the directional of the hydrophone only.

- L_T: Transmission Loss due to propagation, as already mentioned, is a combination of scattering, spreading and absorption loss.

- L_N: Noise Level includes all sources of the noise, including receiver noise.

- S_T: Target Strength is the ratio between the incident pressure and the return pressure as an echo due to the incident pressure.

All the above factors are measured in decibels (decibel (dB)). L_S, L_T, and S_T are measured relative to the standard reference intensity of a 1 μPa plane wave at 1 m distance.

2.3 The Sonar Problem

Unlike free space radio-frequency transmission which is relatively straight forward, transmission in an underwater acoustic (UWA) channel is complicated by a number of factors. Brief details and implications of these factors for the sonar system have already been discussed in Section 2.1 and its subsections.

An important aspect of designing a sonar for any application requires it to have a long range as well as a better range resolution. Having a long range implies having a high SNR and having a better range resolution means an increased ability to differentiate between two targets in close vicinity to each other. However, using the traditional Continuous Wave (CW) sonars, it is sometimes difficult to achieve both the merits simultaneously.

In pulsed CW sonars with single frequency bursts, the range resolution is given by

$$R' = \frac{l\,c}{2}$$

where R' stands for range resolution, l denotes the input pulse length and c denotes the speed of sound in water.

The range resolution R' is directly proportional to the pulse length l. A longer pulse will result in a higher value of range resolution and lesser ability of the sonar to differentiate. As an example, a pulse length of 0.5 ms with a speed of sound in the water of 1500 m/s gives a range resolution of 0.375 m or 37.5 cm. A longer pulse of 1 msec increases the range resolution to 75 cm. We need to reduce the pulse length in order to achieve a good resolution.

Range resolution also depends on the frequency of the signal. The target must be larger than the wavelength of the signal in order to have the transmitted wave reflected back, where the wavelength λ is inversely proportional to the frequency f of the signal

$$\lambda = \frac{c}{f}$$

For a pulsed CW sonar with good range resolution, we need to have a smaller λ, which implies using higher frequencies. However, the transmission loss increases as we increase the frequency.

The pulse length also accounts for the energy of the signal. The SNR at the output of the correlator may be improved by increasing the length of the pulse so as to inject more energy in the water. This will cause the transmitted signal to have more energy and have a corresponding strong reflected signal. However, without careful design, a longer pulse degrades the range resolution.

There is always a compromise between the range and resolution for constant power when we are using a pulsed CW sonar. The solution that does not require compromise lies in pulse compression [99] where we transmit a long modulated pulse that has a bandwidth corresponding to a short unmodulated pulse.

In pulse compression, the autocorrelation mainlobe width and the sidelobes of the transmitted signal define the ability of the sonar system to distinguish between different targets. A narrower mainlobe helps in identifying two targets having the same target strength but a small distance apart. In contrast, a wider mainlobe would make it difficult to distinguish between two closely-spaced echoes, due to the mainlobes of the two echoes overlapping each other. The target strength of fish may vary coincidentally on their size and attitude with respect to the sonar. Reflections from the same type of fish are known to vary by as much as 3 dB [100]. To resolve two reflections from the same type of targets, the Rayleigh Criterion [101] must be met. The Rayleigh Criterion was presented for the optical resolution of objects using non-coherent signals but is generally applicable to any signal resolution in general [102]. An example of this resolution is shown in
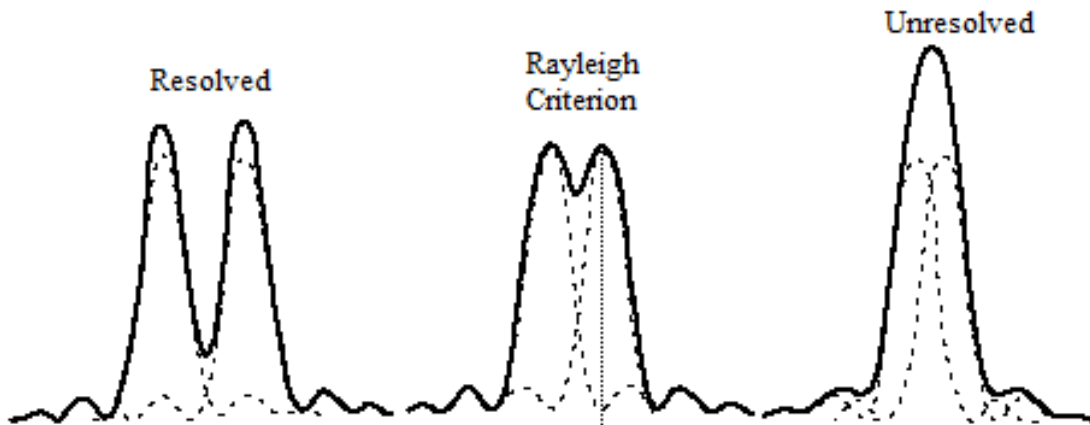
Figure 2.4: The Rayleigh Criterion for target resolution [103].

Figure 2.4. For signals of similar strength to be clearly resolved as separate signals, the points of overlap must not exceed half the strength of the peak of the signal. This is equivalent to the -3 dB point of each signal. In sonar signal processing, this is known as the -3 dB range resolution. This implies that the reflections from the two fish must meet the Rayleigh Criterion to be able to be identified as two separate targets.

Similarly, low autocorrelation SLLs help in differentiating between echoes from two targets with varying intensities. Typical reflections from the seafloor and any fish near the seafloor are known to differ in strength by 40 dB or more [26–28]. To distinguish a fish near the seafloor implies resolving a weaker signal from the fish in the presence of a stronger signal from the seafloor. To correctly resolve the two reflections, the autocorrelation mainlobe must be narrow and the sidelobes must be 40 dB less than the autocorrelation mainlobe peak. This ensures that if a strong signal from the seafloor is received with a weaker signal from a fish, the weaker signal is not masked by the mainlobe or sidelobes of the strong signal. These requirements suggest using signals that have autocorrelation mainlobe satisfying the Rayleigh Criterion and sidelobes 40 dB lower than the mainlobe peak.

A simple CW pulse sonar may be beneficial if it fulfils all the requirements as outlined in Section 2.3. The solution falls short for a scenario where the scanning area is very wide. In terms of angular coverage, transducers have a limited beam-width which limits their spatial coverage. A single pulse CW sonar used for ocean floor mapping, for example, would take considerable time with multiple swathes to cover the required area.

In such a scenario, one approach is to use a Multibeam Echo Sounder (MBES) with a single projector and multiple hydrophones (for echos received from different directions) to illuminate the wider area. Another approach is to use multiple projectors and receiver pairs, with each

transducer placed so as to cover the blind-spots of other transducers. This results in the overlap and causes interference of received signals from different directions when operating in the same frequency band.

To overcome this problem and ensure there is minimal interference in shared coverage of two neighbouring transducers, we must use independent minimally interfering signals for each projector. These signals when reflected and received can then be uniquely identified in the receiver through respective matched filters.

Another scenario for multiple codes could be when we use a single transducer for multiple pings so that each ping would require a specific transmission code. The codes may be transmitted from the same transducer using Code Division Multiplexing (CDM), Frequency Division Multiplexing (FDM) or Time Division Multiplexing (TDM). The multiple codes used in such a scenario must possess the same characteristics as outlined in Section 2.3.

A multibeam sonar system enables an increased volume coverage for surveying and detection of prospective targets such as fish while maintaining spatial resolution [104]. Multibeam sonars were originally designed for sea-floor bathymetry, but are now employed for a range of applications [105]. Use of multiple codes in a synthetic aperture sonar (SAS) may also help increase the area rate coverage and and improved resolution.

For a given set of codes, the autocorrelation mainlobe is ideally a delta function at zero-offset while the sidelobes are zero at all other offsets. Similarly, the crosscorrelation is ideally zero for all offsets. It has been proved that it is not possible to have a set of codes with ideal auto- and crosscorrelations simultaneously. The sidelobes of auto- and crosscorrelations are restricted by the Welch bound [25] which describes a limit on the minimum value of the highest of the sidelobes. The Welch Bound [25] states that for a set of codes of m signals with each signal having length n, the maximum SLL of autocorrelation and all regions of crosscorrelation cannot be less than

$$\mathbf{c}_{\max max} \geq \sqrt{\frac{1}{m-1}\left(\frac{m}{n}-1\right)} \tag{2.7}$$

For a set of two codes with length 10, 100, and 1000 samples, this limits the SLLs to -16 dB, -26 dB and -36 dB respectively. As discussed in Section 2.3, for resolving a fish near the seafloor with a difference in target strength of up to 40 dB, a code of length more than 1000 samples must be used. However, there may be requirements to use a code with a length shorter than 1000 samples and yet be able to detect all targets with their respective varying strengths. For example, in shallow waters, there would be instances where reflections from a potential target are received back in the receiver hydrophone even before the transmission of a longer code is completed. Such transmissions need sets of codes that have properties beyond the Welch bound [25].

For optimal performance of the system, a given code must also comply with specific power constraints as already mentioned in Section 2.2.2.2. These include the power profile for transient and ringing effect as well as maintaining the PAPR. The codes must also be tuned to the channel and transducer frequency responses as well as robust to the Doppler effect. In a sonar system employing multiple codes, the effect of all factors as outlined in Section 2.1 is that the transmitted acoustic waves suffer considerable degradation during propagation. The received signals are no longer a true copy of the transmitted signals. This makes it difficult to differentiate between them or retrieve the intended information from the received echoes.

One solution to this problem lies in optimising codes while maintaining their orthogonality so that they are robust to all these factors. The codes are optimised for desired correlation properties while subject to different constraints such as attenuation, power considerations, channel variations, Doppler effects, etc. The optimised codes then promise optimal performance in a sensing system such as a multicode sonar system.

2.4 Complex Signal Notation

Sets of codes designed for optimal correlation properties in this thesis are represented by complex vectors at bare hand. It is therefore necessary to recall the relationship between the real signal and the complex signal.

The pass-band signals can be described in terms of a real signal $s(t)$, having an amplitude function $a(t)$ and a phase function $\phi(t)$, represented as

$$s(t) = a(t) \cos \phi(t) \tag{2.8}$$

where $a(t)$ is a slowly varying function and is referred to as the signal envelope. The instantaneous frequency $f_i(t)$ is the first differential of the phase function as

$$f_i(t) = \frac{1}{2\pi} \frac{\mathrm{d}\phi(t))}{\mathrm{d}t} \tag{2.9}$$

An important characteristic of real signals is their frequency spectrum being even or real part and odd for imaginary part, i.e., $S(-f) = S^*(f)$. So only one half of the spectrum is required to fully specify a signal waveform. This facilitates a simplified signal notation of the real signal $s(t)$ in terms of a complex signal $\mathbf{s}(t)$ [106] as

$$\mathbf{s}(t) = s(t) + j\hat{s}(t) \tag{2.10}$$

where the imaginary part $\hat{s}(t)$ is chosen to be the Hilbert transform [107, 108] of the real signal $s(t)$.

2.5   Summary

This chapter familiarised the reader with the fundamentals of underwater acoustics wave propagation. A brief explanation of various factors affecting acoustic wave propagation in an underwater environment with implications for a sonar system was presented. These included the varying speed of sound, transmission loss, Doppler effect, multipath propagation, reverberation, and noise.   Other factors associated with and intrinsic to sonar systems such as power considerations and transducer imitations were also discussed. All these factors reveal the complexity associated with underwater acoustic communication.

All these limitations culminated into defining the sonar problem. The implications were extended to a multicode sonar system and its advantages. It is evident that specialised (sets of) codes are to be preferred for UWA communications. These codes are required to be optimal in sensing as well as tolerant of the various factors explained in the chapter. We investigate some of these factors affecting UWA communications in the subsequent chapters of this thesis.

Chapter 3

The State of the Art in Code Optimisation

The only good is knowledge and the only evil is ignorance.

Socrates.

Designing sets of codes for a multicode sonar system requires optimising them for a number of factors affecting the signal during the course of its propagation in the UWA channel. These include the effects of the transmitter, the channel, the target, and the receiver.

An underwater acoustic channel is similar to a wireless channel in that it is frequency-dependent, time-dependent and position-dependent. The channel characteristics affect an acoustic wave propagating through an underwater channel by frequency-dependent attenuation, time delay, Doppler spread and ambient noise [109].

At the receiver end of a sonar system, the sensing performance improves if the signal has good correlation properties for the given channel environment. The optimal correlation of signals is bounded by certain theoretical thresholds on correlation SLLs. Signals may be optimised to possess "optimal" correlation properties. However, each of the above-mentioned channel characteristics has a degrading effect on the correlation properties, and considerable literature has been devoted to code optimisation for various constraints and applications.

In this chapter, a review of various aspects of code optimisation, including correlation bounds, code optimisation methods, optimised codes, and methods to reduce SLLs with reference to a multicode sensing system. Section 3.1 presents an important aspect of the channel itself, that is, the channel coherency, and different aspects of the channel coherency. This is followed by a review of bounds on optimality of correlations for designed codes. A review of code optimisation methods which are generally used to design codes with optimal correlations for use in radars

	Range (km)	Bandwidth (kHz)
Long	$10-100$	$2-5$
Medium	$1-10$	≈ 10
Short	$0.1-1$	$20-50$
Very Short	< 0.1	> 100

Table 3.1: Available bandwidth for different ranges in an UWA channel [8, 110, 111].

and sonars is presented next. When codes are designed for optimal correlations and subjected to propagation in an UWA channel, there are factors that are external to code design but still affect the performance of the designed codes. Examples are TVG and the Doppler effect. A review of TVG and how it could affect the designed codes in an UWA channel, followed by a review of code optimisation for Doppler tolerance is briefly presented. The effect of these factors is invariably a deterioration of correlation properties. A review of literature on mismatched filtering as a technique for improving the correlations of the optimised sets of codes beyond the limits of a matched filter is presented before concluding the chapter with a summary.

3.1   Channel Coherency

The ocean is a temporally and spatially varying propagation environment, the characteristics of which pose significant challenges to the development of effective underwater wireless communications and sensing systems [16].   Despite being a favourable choice over electromagnetic and optical waves, the dynamics of acoustic propagation severely limit underwater communication systems.   Communication techniques typical of wired and RF channels have to be modified significantly to adapt them for UWA channels. A review of major developments in this field is presented in [9] and references therein. Communication in an UWA channel is best supported at low frequencies due to high transmission loss for higher frequencies. However, using low frequency limits the available bandwidth. Table 3.1 shows the available bandwidths for different ranges in the UWA channel.

The ocean is a dynamic time-varying/random medium [112]. Designing codes for such a channel requires knowledge of the channel characteristics, in terms of an impulse response. Channel sounding is a technique used to ascertain the channel impulse response $h(t, \tau)$ as a function of time t and time delay τ [113, 114]. This may be done by transmitting dedicated probing signals through the UWA channel. Channel sounding is important as it helps us understand the UWA channel characteristics. This, in turn, facilitates the design of the codes for UWA sensing systems to match with the real marine environment and achieve better performance [115, 116].

The inconsistency of the UWA channel characteristics with time decreases the time for which the channel may be considered coherent. The time during which the UWA channel characteristics remain unchanged is described using the term *channel coherence time*. Coherence time is the duration over which the impulse response of the channel can be considered to not change. It is important to understand the channel coherency as a code designed for an UWA channel is only optimal as long as the channel is coherent. If the time interval between the two signal transmissions is much greater than the coherence time, the channel will likely affect the two signal transmissions differently; otherwise, they will be affected similarly. A channel is said to be coherent if the signal coherence or the autocorrelation peak between the transmitted and received signal drops to $\tau = 1/e \approx 0.38$ of the actual, however, $\tau = 0.8$ is a practical measure also used in the literature [112]. Assuming the propagation time for the signal to travel from the transmitter to the target and back to the receiver is Δt. Then the channel characteristics during the signal propagation do not change if the coherence time is larger than the time taken for the signal to travel from the transmitter to the receiver, that is

$$t_c > \Delta t$$

or

$$R < R_c$$

where R is the distance between the source and the target and, R_c is the coherence length of the UWA channel.

Doppler spread and delay spread are two measures associated with the channel coherency. Coherence time t_c is the time domain dual of Doppler spread. The Doppler spread and coherence time are inversely proportional to each other so that we have

$$t_c \propto \frac{1}{f_m}$$

where f_m is the maximum Doppler shift. Doppler spread and coherence time are parameters that describe the frequency-dispersion nature of the channel.

The time-varying chemical and physical characteristics of the UWA channel may cause the arrivals from multipaths to change in length and time. This, in turn, gives rise to a varying delay spread over the time that the channel characteristics change. The Doppler spread and the varying delay spread are two of the common effects causing a non-coherent channel.

3.2   Correlation Bounds

Correlation SLLs for a set of codes are bounded by certain theoretical limits. A multicode sonar system may use one transducer with many codes or many transducers, each assigned a distinct code. Identification and detection of the received signal, then, is accomplished through correlation with a copy of the transmitted codes. For any set of codes[1] with 2 codes $\mathbf{a} = \{a_1, a_2, ..., a_n\}$ and $\mathbf{b} = \{b_1, b_2, ..., b_n\}$ each with length n, the aperiodic crosscorrelation function $\mathbf{C}(j)$ of codes $\mathbf{a}$ and $\mathbf{b}$ is defined as

$$\mathbf{C}(j) = \sum_{i=1}^{n} a_i \, b_{i-j}^* \qquad (3.1)$$

where $(^*)$ denotes the complex conjugate. When $\mathbf{a} = \mathbf{b}$ in the above expression, the function $\mathbf{C}(j)$ represents the autocorrelation function of the code.

Low autocorrelation sidelobes increase the ability of the system to resolve two targets that are close in range but have different strengths. The autocorrelation mainlobe width around the zero offset determines the resolution for range measurements. Crosscorrelations determine the interference level between the two signals. A high value of crosscorrelation will increase the interference and hence cause detection errors.

The ideal autocorrelations of codes $\mathbf{a}$ and $\mathbf{b}$ have the characteristic of a delta function, which has a single sharp peak at zero-offset and is zero at all other offsets. All crosscorrelations are also zero for all offsets. However, theoretical constraints do not allow codes to simultaneously have ideal autocorrelation and crosscorrelation. The aim is then to achieve a solution approximating the ideal. The limits on what is theoretically achievable are described by various bounds. The bounds lay down a limit as to the lowest of the maximum level of sidelobes of autocorrelations and crosscorrelations. These bounds, hence, have implications on the sensing applications that work on the principle of correlations.

Welch [25] studied the correlations of two or more codes as the inner product of two or more vectors. He published his findings in a seminal paper [25] in 1974. He derived a lower limit on the maximum value of zero-lag correlations of two or more signals, and extended the result to autocorrelation sidelobes and crosscorrelations at all other lags. A simplified derivation of the bound is reproduced in Appendix A for reference. Corresponding expressions are derived for bounds on both aperiodic and periodic correlations. There have been subsequent adaptations of the Welch bound [25]. Most are related to binary sequences, some to complex roots-of-unity (which include binary sequences as a special case) as well as bounds for codes designed to have

---

[1]Throughout this thesis, a set of codes represented with two vectors would mean vectors $\mathbf{a}$ and $\mathbf{b}$, whereas a vector set representing more than two vectors, say m, will denoted with $A = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$.

lower SLLs in a specific region of interest. The Welch bound [25] and the modifications to the bound will be discussed in the next subsections.

3.2.1 Aperiodic Correlation Bounds

Aperiodic correlations are important where the communication is characterised by a period of transmission followed by a longer wait time. Sonar systems are usually characterised by aperiodic signals. During this wait time, the sonar receiver waits for a reflection of the transmitted signal from the target. There is considerable literature on periodic correlation bounds [117–120] and some optimised codes that asymptotically meet these bounds. However, aperiodic correlation bounds are comparatively difficult to meet, especially as we increase the length of the codes n [121, 122] and literature on the subject is limited [123] as compared to periodic bounds.

Considering a set of vectors $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m\}$ with m unit vectors representing number of codes, and each having length n representing the length of each code, i.e. $\mathbf{a_1} = \{a_{11}, a_{12}, ..., a_{1n}\}$ in $\mathbb{C}^n$ with $m \leq n$, Welch presented a set of lower bounds on maximum periodic crosscorrelation $c_{\max} = \max_{i \neq j} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$ given by

$$c_{\max}^{2k} \geq \frac{1}{m-1} \left[\frac{m}{\binom{n+k-1}{k}} - 1 \right] \tag{3.2}$$

for all $k \geq 1$. The most common bound for $k = 1$ is

$$c_{\max}^2 \geq \frac{1}{m-1} \left(\frac{m}{n} - 1 \right) \tag{3.3}$$

Welch showed that for aperiodic correlations, we replace m by mn and then $n = 2n - 1$ to obtain

$$c_{\max,\text{ap}}^2 \geq \frac{m-1}{m(2n-1)-1} \tag{3.4}$$

(3.4) gives a limit on how low the value of $c_{\max}$ can get for aperiodic correlations of a given set of m codes with length n. Derivation of the Welch bound [25] can be found abundantly in the literature [124–130]. The tightness of the bound and conditions for equality of (3.2) have been investigated in the above-mentioned references. An alternative proof for higher-order Welch bound for $k > 1$ has been presented in [131]. It has been noted in the same literature that finding sets of codes achieving equality in the bound $c_{\max}$ for $k > 1$ is not only challenging but is impossible in many cases.

Welch constrained the vectors $\mathbf{a}_i$ to have unit-norm, that is

$$\sum_{j=1}^{n} |a_{ij}|^2 = 1 \quad \text{for i = 1, 2, ..., m}$$

However, relaxing the unit-norm constraint on the vector set **A**, the modified expression for the bound was presented by Waldron in [129]. Waldron recalled the main expression of the Welch bound [25] as

$$\sum_{i=1}^{m} \sum_{k=1}^{m} |\langle \mathbf{a}_i, \mathbf{a}_k \rangle|^2 \geq \frac{m^2}{n} \tag{3.5}$$

Waldron then presented that for any vector set $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$ with $m \leq n$ and with all $\mathbf{a_i}$ not unit vectors, an expression corresponding to the Welch bound [25] in (3.5) is

$$\frac{\sum_{i=1}^{m} \sum_{j=1}^{n} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|^2}{(\sum_{i=1}^{m} \langle \mathbf{a}_i, \mathbf{a}_i \rangle)^2} \geq \frac{1}{n} \tag{3.6}$$

The expression in (3.6) is a generalisation of the Welch bound [25] expression in (3.5) where $\mathbf{a_i}$ are not necessarily unit vectors, and so was a new result.

Sarwate in [132] has extended the work on the periodic and aperiodic bounds on correlations. He presented an alternative approach by deriving the bounds as the sum of autocorrelations and crosscorrelation terms. He assumes a vector set $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$ with m unit vectors, each having length/period n, so that

$$\sum_{j=1}^{n} |a_{ij}|^2 = n \quad \text{for i = 1, 2, ..., m} \tag{3.7}$$

where a_{ij} is the j-th element of i-th vector.

For aperiodic correlations, assuming $\mathbf{C_{aa}}$ is the maximum value of the aperiodic autocorrelation between any two codes $\langle \mathbf{a_i}, \mathbf{a_j} \rangle_{i=j}$ and $\mathbf{C_{ab}}$ is the maximum value of aperiodic crosscorrelation between any two codes $\langle \mathbf{a_i}, \mathbf{a_j} \rangle_{i \neq j}$ then

$$\frac{(2n-1)}{n} \left(\frac{\mathbf{C_{ab}^2}}{n} \right) + \frac{2(n-1)}{n(m-1)} \left(\frac{\mathbf{C_{aa}^2}}{n} \right) \geq 1 \tag{3.8}$$

The expression of (3.8) may provide a lower limit than the Welch bound [25], on either $\mathbf{C_{aa}}$ or $\mathbf{C_{ab}}$, provided the value of the other is known. However, it requires either one of the values of $\mathbf{C_{aa}}$ or $\mathbf{C_{ab}}$ to be known, hence reducing its usefulness. Moreover, it does not ensure a common lower value for both the $\mathbf{C_{aa}}$ and $\mathbf{C_{ab}}$ simultaneously. Simplifying the expression in terms of c_{max} yields

$$\frac{c_{\text{max}2}}{n} = \max \left\{ \frac{\mathbf{C_{aa}^2}}{n}, \frac{\mathbf{C_{ab}^2}}{n} \right\} \geq \frac{n(m-1)}{2nm - m - 1} \tag{3.9}$$

The Sarwate bounds include the Welch bound [25] as a special case for aperiodic correlations.

In 1999, Levenshtein [133] derived bounds on the aperiodic correlations of binary sequences. Assuming m binary codes $\{1, -1\}$ of length n, he proved that for all $m \geq 4$ and $n \geq 2$, the maximum value of $c_{\max}^2$ is given by

$$c_{\max}^2 \geq n - \frac{2n}{\sqrt{3m}} \quad \text{for } m \geq 3 \tag{3.10}$$

and

$$c_{\max}^2 \geq n - \frac{\pi n}{\sqrt{8m}} \quad \text{for } 5 \leq m \leq n^2 \tag{3.11}$$

While deriving the new bound, Levenshtein observed that the Welch bound [25] assumes equal weights for all shifts of the codes of the set. For a given binary code with the number of codes $m \geq 4$, the new bound can be considerably tightened over the Welch bound [25]. This is achieved by suitably choosing weights w_i for shifts of codes for calculating correlations $w_i \geq 0$, and $\sum_{i=}^{2n} w_i = 1$. The method proposed by Levenshtein does not ensure an equal lower value of all SLLs of auto- and crosscorrelations.

Extending the approach of weighting the correlations as proposed by Levenshtein, low correlation spreading sequences have been investigated in [134–137] for mobile communications. However, a limitation of the method is that it does not show any improvement for codes $m \leq 2$. New theoretical bounds on aperiodic correlations of binary codes have been presented by Peng & Fan in [138]. A number of new but complicated expressions for various numbers of codes m and different code lengths n have been presented. These bounds are lower than the previously known Welch bound [25], Sarwate bound [132] and Levenshtein bound [133], but are only applicable to binary codes.

Some research has also been carried out on correlation bounds for complex root-of-unity sequences. These codes include binary codes as a special case. For aperiodic correlations of complex roots-of-unity codes, Welch bound [25], Sarwate bound [132] and Levenshtein bound [133] bound are generally applicable. In 1998, Boztas [139] has generalised the Levenshtein bound to be applicable for complex roots-of-unity sequences. These bounds are specific to the complex roots-of-unity sequences and hence, cannot be generalised to be applicable for all types of codes.

In [123], the authors have used another metric Integrated Sidelobe Level (ISL) for sidelobes of aperiodic correlations. For any set of vectors $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$ with m unit vectors, each having length n, with

$$\sum_{j=1}^{n} \left| a_{i_j} \right|^2 = n \quad \text{for i} = 1, 2, ..., \text{ m}$$

the ISL is defined as

$$\text{ISL} = \sum_{i=1}^{m} \sum_{j=-n+1, j\neq 0}^{n-1} |\mathbf{C}_{ii}(j)|^2 + \sum_{i=1}^{m} \sum_{k=1, k\neq i}^{m} \sum_{j=-n-1}^{n-1} |\mathbf{C}_{ik}(j)|^2 \qquad (3.12)$$

The first term on the right-hand side of equality is the sum of all terms of autocorrelation except for the zero-offset. The second term on the right-hand side of equality is the sum of all crosscorrelations. Using the ISL metric, the authors have then defined new bounds on the Peak to Sidelobe Level (PSL) which is the same as the Welch bound [25] $c_{\max}^2$. Optimising codes for the ISL metric promises a lower sum of all the crosscorrelations and sidelobes of autocorrelations. However, it does not guarantee an overall minimum value of sidelobe with flat crosscorrelations and autocorrelation sidelobes, across the entire range of offsets.

3.2.2 Aperiodic Bounds for Low/Zero Correlation Zone

In the absence of sets of codes that possess the desired ideal correlation properties, some researchers have used alternate design criteria. For conventional codes with delta-like correlation, the bound on the maximum value of crosscorrelations and sidelobes of autocorrelations permits all the remaining correlation SLLs to be less than this maximum. So there could be a selective region around the zero-offset where SLLs are comparatively very low (or even zero) when compared to the SLLs in the remaining range of correlations.

Instead of optimising the SLLs across the entire range of correlations, a small region of interest is chosen and codes are optimised for very low (or zero) SLLs within this region. The SLLs in the remaining region are also low but have a value greater than the levels in this region. If this region has lower SLLs than the remaining SLLs, the region is called a Low Correlation Zone (LCZ). If the SLLs in this region are zero, the region is called a Zero Correlation Zone (ZCZ).

Recalling (3.1), this concept can be mathematically represented as

$$\left| \mathbf{C}_j \right| = \left| \sum_{i=1}^{n} a_i b_{i-j}^* \right| = \begin{cases} 1 & \text{for } j = 0, \ a = b \\ \varepsilon & \text{for } j = 0, \ a \neq b \\ \varepsilon & \text{for } 1 \leq |j| \leq p - 1 \\ \mathbf{C} & \text{for } p \leq |j| \leq n - 1 \end{cases}$$

where ε takes a very small value, $p < n$, and c has an arbitrary value with $c > \varepsilon$. When the value of ε is equal to zero, the LCZ becomes a ZCZ.

A detailed account of the concept has been presented in [140, 141]. Codes that are optimised for characteristics of LCZ or ZCZ are referred to as Generalised Orthogonal (GO) codes or ZCZ codes [134]. GO codes have been preferred over normal orthogonal codes for multipath propagation channels as they are more robust [134]. These codes find use in Quasi-Synchronous Code Division Multiple Access (CDMA) or Approximately-Synchronous CDMA (AS-CDMA) systems.

Based on the concept of LCZ / ZCZ, lower bounds on aperiodic and odd periodic correlations [142–144] of spreading sequences with LCZ and ZCZ have been proposed in [145] and referred to as Tang-Fan bounds in some literature. Tang has used the same approach followed by Welch to derive these bounds as the maximum value on the inner products of vectors. These bound include the Welch bound [25] and the Sarwate bound [132] as special cases.

An alternate bound on aperiodic correlations for LCZ / ZCZ binary sequences has been presented by Peng et al. in [146] which is lower than the previously known Tang-Fan bounds. Peng has presented a lower bound in [141] which includes all previous bounds on LCZ / ZCZ and GO codes, as well as the Welch bound [25] and Sarwate bound [132] as special cases.

In [147], Peng and Fan have derived and presented a lower bound for aperiodic correlation of complex roots-of-unity codes for LCZ / ZCZ. These bound are the lowest that exist for such codes. Peng has given a detailed comparison of the Welch bound [25], Sarwate bound [132] and Levenshtein bound [133] and Boztas bound [139] for codes with different numbers m of codes and lengths n of each code.

The correlation bounds in LCZ sequences are only applicable to the LCZ zone and SLLs in remaining region are allowed to take arbitrary values. These bounds are only applicable to sets of codes where correlations SLLs are required to be lower in a specific region of interest. Beyond this region, the SLLs are higher and hence, not applicable to the general case. The Welch bound [25] in its original form remains the standard for all other applications where the auto- and crosscorrelation SLLs are required to have the same value equal to or above a certain maximum value, across the entire range of correlation offsets.

3.2.3   Periodic Correlation Bound

The Welch bound for periodic correlation is given by (3.3)

$$c_{\text{max,p}}^2 = \frac{m - n}{n(m - 1)}$$

where $c_{\text{max,p}}^2$ denotes the periodic Welch bound [25] for a given set of codes with m codes each having length n.

An alternative to the Welch bound [25] for binary periodic correlations has been reported in [132] and is known as the Sarwate bound. He assumes a vector set $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$ with m unit vectors, each having length/period n, so that

$$\sum_{j=1}^{n} \left| a_{i_j} \right|^2 = n \quad \text{for i} = 1, 2, ..., \text{ m} \tag{3.13}$$

where a_{i_j} is the j-th element of i-th vector.

For periodic correlations, assuming $\mathbf{C'_{aa}}$ is the maximum value of the aperiodic autocorrelation between any two codes $(\mathbf{a_i}, \mathbf{a_j})_{i=j}$ and $\mathbf{C'_{ab}}$ is the maximum value of aperiodic crosscorrelation between any two codes $(\mathbf{a_i}, \mathbf{a_j})_{i \neq j}$ then

$$\left(\frac{\mathbf{C'^2_{ab}}}{n} \right) + \frac{n-1}{n(m-1)} \left(\frac{\mathbf{C'^2_{aa}}}{n} \right) \geq 1 \tag{3.14}$$

The expression of (3.14) provides a lower bound on either $\mathbf{C'_{aa}}$ or $\mathbf{C'_{ab}}$ provided the value of the other is known. The bound is only applicable for periodic correlations of binary codes. The simplified form of this inequality relating to the expression of the Welch bound [25] is

$$\frac{c'^2_{\max}}{n} = \max \left\{ \frac{\mathbf{C'^2_{aa}}}{n}, \frac{\mathbf{C'^2_{ab}}}{n} \right\} \geq \frac{n(m-1)}{nm-1} \tag{3.15}$$

Sarwate has further discussed in [127] that codes satisfying the Welch bound [25] with equality were initially meant to imply codes that had the maximum magnitude of all autocorrelation sidelobes and all crosscorrelations equal to the bound given by (3.3). However, Sarwate referred to [126, 148] where the Root Mean Square (RMS) value of the inner products is taken to be equal to the Welch bound [25]. As a consequence of this, codes are classified on the basis of these two distinct properties. Welch Bound Equality (WBE) codes are those codes where the RMS value of the inner products of vectors satisfies the Welch bound whereas Maximum Welch Bound Equality (MWBE) codes are those codes that satisfy the criteria of the maximum magnitude of the inner product meeting the Welch bound [25].

The maximum inner product magnitude $c_{\max}$ of a set code $\mathbf{A}$ is defined by

$$c_{\max}(\mathbf{A}) = \max_{i \neq j} \left| \langle \mathbf{a}_i, \mathbf{a}_j \rangle \right| \tag{3.16}$$

The root-mean-square (RMS) inner product magnitude c_{rms} of a set of codes $\mathbf{A}$ is defined by

$$c_{rms}(\mathbf{A}) = \left[\frac{1}{m(m-1)} \sum_{i=1}^{m} \sum_{i \neq j\, j=1}^{m} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|^2 \right]^{\frac{1}{2}} \tag{3.17}$$

From [126], we have for any set of codes $\mathbf{A}$,

$$\sum_{k=1}^{m} \sum_{l=1}^{m} |\langle \mathbf{a}_k, \mathbf{a}_l \rangle|^2 \geq m^2 n \tag{3.18}$$

Recalling the original expression of the Welch bound [25] from (3.2). If the equality holds in (3.18), the set code is said to meet the Welch bound [25] on c_{rms} and can be called a WBE set. Similarly, sets of codes that meet the Welch bound [25] on c_{max} with equality can be called MBWE sets.

Further discussing MBWE and WBE codes, Ding [125] has observed that an MBWE code must be a WBE code. However, a WBE code might not be an MBWE code. Also, WBE codes can be easily found but it is difficult to find MBWE codes. A brief compilation of known classes of MBWE codes is given in [125].

Peng and Fan in [149] have derived a set of periodic correlation bounds for the LCZ / ZCZ codes, known as the Generalised Sarwate bounds for binary codes. Peng has shown the Welch bound [25], the Sarwate bound [132] and the Tang-Fan bound [145] are special cases of the periodic bound presented in [149]. These specialised codes have application for cases where the correlation levels for a particular region of interest around the zero-offset. Beyond this region, the SLLs are higher and hence are not suitable for all sensing applications.

For periodic complex roots-of-unity sequences, Sidelnikov presented the earliest periodic bound in 1971 in [150, 151]. A slightly lower bound was published by Kumar and Liu in [152]. They have shown the Welch bound [25] can be modified to provide an improved bound for this class of sequences. These bounds are an improvement over the previous Sidelnikov bound but require a more complicated approach for deriving the expression for the bound.

Summarising, binary periodic codes are mostly applicable to mobile communication systems. Similarly, GO codes with correlation properties of Low or Zero Correlation Zone (LCZ / ZCZ) only find applications where the low SLLs are required in a particular region of interest around the zero-offset. Similarly, complex roots-of-unity sequences are specific in their application. Outside the limited scope just described, these codes may not find many practical applications, especially when considering a multicode sonar system which is being researched in this thesis.

3.3   Optimised Codes

It has been noted in [127, 130, 153] that finding optimal codes with minimal sidelobes is very difficult, both "analytically and numerically". There has been continuing research to find codes as close to the ideal as possible. Codes are regarded as optimal when having a single delta-like peak in autocorrelation at zero-offset with all other sidelobes of autocorrelation and all crosscorrelations being minimum. Accordingly, the research has focused on ways to find or develop ideal codes or codes with properties near to ideal codes.

A brief account of optimised codes for various communication applications is presented next. Section 3.3.1 gives an account of optimal codes designed prior to the Welch bound [25] whereas section 3.3.2 gives an account of such-like codes developed after the Welch bound [25] were presented.

3.3.1   Optimal Code Design Prior to the Welch bound

Before the Welch bound [25] was published in 1974, communication codes were classified as "optimum" based on the ratio of the main peak to side peaks ratio [154]. There is extensive literature on code-design methods to achieve sets of codes with optimal correlation properties and some of the codes or design methods proposed to achieve optimal codes are covered in the following paragraphs.

Barker codes [155] were categorised as one of these optimum binary codes with main-peak-to-side-peak ratio equal to the length of the sequence. However, these codes were restricted to a maximum length of 13. An alternate scheme of polyphase codes known as Frank codes were published in [154]. These codes have n phases and n^2 length. Frank codes are found to have better SLLs than Barker codes and were found to be more tolerant to Doppler shift as well.

Gold codes [156] are a family of distinct binary periodic codes generated using a particular combination of Maximum Length Sequence (MLS) from a Linear Feedback Shift Register (LFSR). These MLS have very good autocorrelation properties, but the crosscorrelation properties are usually not as good. However, the exclusive-ORs of 2 MLS generate a Gold code. These codes have good autocorrelation and good crosscorrelation properties. For any odd number $r = 1$, 3, 5, 7,..., the resulting Gold codes have length $2^r - 1$ and a period of $2^r - 1$, and with SLLs equal to $\sqrt{2^{r+1}} + 1$. These codes were considered superior with the lowest SLLs for a given number and length of the codes [157]. Because of their characteristic properties, Gold codes find wide use in various CDMA applications including satellite navigation. Although Gold codes are excellent for certain applications, they are bound in their length which is restricted to $2^r - 1$.

Kasami codes [158] are another set of binary periodic codes that are constructed using

the same two *m*-sequences that formed a Gold code and adding a decimated version of one of the *m*-sequence that was used to construct the Gold code. Kasami codes also inherit good autocorrelation and crosscorrelation properties from the constituent *m*-sequences.

Heimiller in [159]and Frank in [160] have proposed phase shift pulse codes with good periodic correlation properties. The codes resulting from the proposed method achieve low and in some cases zero autocorrelation sidelobes, but are only applicable to periodic correlations. Schroeder's proposed method in [161] addresses a method to achieve good PAPR with low autocorrelation sidelobes, but is restricted to periodic correlations of binary sequences only. Chu's proposed technique in [162] also addresses periodic correlations with low sidelobes for autocorrelations only. Lewis has proposed a new class of polyphase codes [163] as an improvement over Frank codes [154], but considering only lower autocorrelation sidelobes. Suehiro, in [164] also suggested N-shift orthogonal sequences, which are applicable to autocorrelations of binary codes. Similarly, Golomb's proposed code-design method [165] targets binary periodic autocorrelation sidelobes only.

A comprehensive comparison of the above-mentioned codes and some other binary periodic codes for CDMA applications has been carried out in [166]. Lang used various correlation parameters such as maximal correlation, mean correlation and square-sum values of correlation to analyse these codes for lengths $15 \leq n \leq 2047$.

All the proposed methods fall short for a multicode sonar system, where there is a requirement of not only lower autocorrelations, but correspondingly low crosscorrelations as well.

3.3.2 Post Welch bound Optimal Code Designs

Prior to the Welch bound [25], design of sets of codes for optimal correlation properties was considered with SLLs lower than an arbitrary number. Welch [25] presented the first standard for codes to be regarded as optimal based on maximum magnitude of inner products. Designing special codes satisfying the correlation properties has been mostly confined to two approaches. The first approach exploits different properties including phase and amplitude of the binary and ternary sequences to design optimum sets of codes. These may be termed Non-Algorithmic methods. The second approach designs codes using algorithms to reduce the sidelobes of autocorrelation and all crosscorrelations. The following subsections provide an overview of both approaches.

3.3.2.1   Non-Algorithmic Optimised Codes

The literature on the design of codes and sets of codes against the Welch bound [25] is extensive. A brief account of codes optimised on number theory and computer search will be presented here. These codes vary from binary periodic to aperiodic and random complex codes. The utility of these codes varies from application to application, however, most of the codes have been designed to have optimal correlation properties for accurate detection and identification.

Subsequent to publication of the Welch bound [25], Altop discussed bounds on periodic correlations of sets of codes with m codes each having length n [167]. He discussed three types of sets of periodic codes which possess optimal correlation properties. These are called quadric phase, cubic phase, and power residue. For a period n, all elements of normalised codes have magnitude $|x_i| = n^{-1/2}$. Altop has shown that the quadric and cubic phase sequences meet the Welch bound [25]. He has also shown that for n being an odd number, there is a family of $p-1$ quadric phase sequences, p being the smallest divisor of n. These quadric phase sequences have an autocorrelation peak of $n^{-1/2}$ and zero autocorrelation sidelobes. Cubic phase sequences meet the Welch bound [25] only when n is prime, but unlike quadric phase sequences, do not have zero autocorrelation sidelobes. For power residue sequences, Alltop has extended the number theory to derive a set of m sequences, each having magnitude $(2n)^{-\frac{1}{2}}$, and peak autocorrelation equal to $(2n)^{-\frac{1}{2}}$. The set of derived sequences meet the Welch bound [25] for $mn + 1 = p$, where p is a prime number. Altop has focused on meeting the Welch bound [25] for small sets of codes, that is, $m \leq n$.

Luke [168] has classified good codes as ones having low sidelobes as well as high energy efficiency. He has used the concept of high energy efficiency η [169] or alternately, a low power factor $\frac{1}{\eta}$. For any given code $\mathbf{a_i}$ of length n, η_i is defined as

$$\eta_i = \sum_{j=1}^{n} \frac{a_{i_j}^2}{\left(n \max \left(a_{i_j}^2\right)\right)}$$

where a_{i_j} represents element j of the code $\mathbf{a_i}$. Sets of binary and ternary sequences have been presented with length $3 \leq n \leq 60$ and energy efficiency greater than 60% that have perfect periodic correlations. Using the relationship between the autocorrelation in time domain and power spectrum in frequency domain, he has extended the search in the frequency domain to construct perfect periodic sequences with high energy efficiency.

In [157], an improvement to Gold Codes [156] has been presented. For every odd integer $r > 3$, these binary periodic codes have length $n = r(r-1)/2$. The codes are non-linear in nature and are not so susceptible to jamming, which is another improvement over Gold codes.

Subsequent to the classification of codes meeting the Welch bound [25] on RMS magnitude

and maximum magnitude of SLLs, Xia et al [130] and Ding [170] have suggested construction of MBWE codes from difference sets. In [125], a generic construction of MBWE codes has been presented which includes the earlier constructions as a special case.

3.3.2.2 Algorithmically Optimised Codes

Finding optimal codes based on number theory or computer searches is demanding and gives few results. The difficulty in finding codes close to ideal with desired autocorrelations and crosscorrelations has been tackled by employing various optimisation algorithms. These algorithms start from some initialisation and minimise the correlation sidelobes. However, computational complexity of these algorithms can become prohibitive as the length n of the codes increases, limiting their performance for lengths $n > 1000$ [171].

Using evolutionary algorithms, sets of optimised binary codes with low aperiodic correlations have been presented in [172]. These binary codes have length $49 \leq n \leq 100$ and can be used, for example, in frame-preambles for synchronisation of Direct Spreading(DS)/CDMA systems. Kocabas used combinations of preamble types (i.e., random or predefined) and whether phase ambiguity exists or not, to define three types of sidelobe measures. These are the normal Peak SLL (PSL), Relaxed-PSL (PSL^r) and Modified PSL (MPSL). Three sets of codes have been presented, optimised for each of these SLL types.

Evolutionary algorithms have their limitations. These algorithms take a long time to converge. Besides, these algorithms depend on many factors with inter-dependencies, and these factors limit code optimisation for desired correlation properties.

Another approach for optimising codes using a Cyclic Algorithm (CA) for a MIMO radar was first presented in [173]. In [171], several new CAs as an extension of the original algorithm were introduced for finding local minima of the Integrated Side Lobe (ISL). These include the CA-Pruned (CAP) algorithm which deals with a weighted ISL (WISL) metric given by

$$\text{WISL} = \sum_{j=1}^{n-1} w_j \left| \mathbf{C}_j \right|^2$$

where W is the weight, $w_j \geq 0$ for $j = 1, 2, ..., n-1$ and $\mathbf{C}_j$ is the correlation value at offset j for any set of code.

Because of memory constraints, the CAP algorithm is limited to code lengths up to $n \approx 1000$ as it requires Singular Value Decomposition (SVD) of a matrix of order n. Another variant of CA proposed in the same literature [171] is the CA-New (CAN) algorithm which works using Fast Fourier Transform (FFT) operations and hence can be used for length of codes up to

$n \sim 1{,}000{,}000$. The CAN algorithm can be further modified to use arbitrary non-negative weights w_k. This algorithm is referred to as Weighted-CAN (WeCAN). The arbitrary weights increase the computational complexity of the algorithm, but, it can still manage to optimise the same codes lengths as CAN.

A further modified algorithm presented in [171] is a hybrid of WeCAN and CAP. This algorithm alternates between CAP and WeCAN to use arbitrary weights for ISL whilst minimising the correlation SLLs. Using this algorithm, some interesting results as regards ISL and SLLs have been presented for codes with lengths as large as $n = 10{,}000$. The CAN algorithm has been extended to periodic codes as Periodic-CAN algorithm (PeCAN) in [174]. This algorithm can optimise random initialisations so that autocorrelation SLLs are as low as -260 dB for a code of length $n = 256$, without affecting the crosscorrelation levels.

The limitation of these algorithms is that they tend to optimise codes for very low sidelobes in an LCZ and ZCZ. For a code of length $n = 100$, the SLLs are lower than -270 dB for a particular range of offsets which have been weighted. However, the SLLs in the remaining region are not optimised to the same level, hence a flat profile for SLLs across all the offsets is not ensured.

Besides the above-mentioned algorithms, an alternate approach is to use an economical local optimisation method such as the gradient-descent method. Gradient-based methods are efficient and commonly used techniques to optimise a function with respect to a variable [175]. A global minimum may not always be guaranteed, but gradient-descent based optimisation methods are very simple and with very low inter-dependencies. However, literature on optimisation sets of codes using this approach for any desired characteristics such as correlation properties is not abundant.

3.4   Time Variable Gain

An important factor that affects the performance of optimised codes during propagation in an UWA channel is the non-constant transmission loss, as already discussed in 2.1.2. The correlation of the optimised codes is degraded if the propagation loss is not appropriately compensated. The propagation loss of acoustic waves in an UWA channel is not only dependent on the range but is also dependent on the varying acoustic velocity due to unpredictable and varying channel characteristics. This results in a non-constant propagation loss due to scattering, spreading and absorption [176] which affects the performance of the sonar. To compensate for the loss and remove range dependence of the echo strength from similar targets, an appropriate Time Variable Gain (TVG) [177] based on (2.1) is used.

The use of TVG has two advantages. First, echoes from the same type of target at different ranges are received with similar target strengths. Second, the TVG reduces the required dynamic
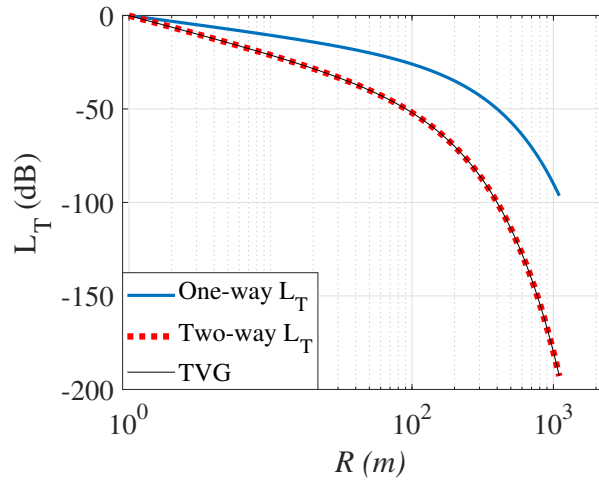
Figure 3.1: 1-way and 2-way transmission loss L_T and TVG versus range. The TVG curve exactly overlaps and compensates for the two-way L_T. A shallow channel is considered having cylindrical spreading and the TVG is calculated for $\alpha_{water} = 60\,\text{dB/km}$.

range of the Analog to Digital Converter of (ADC) in the sonar receiver. High transmission losses can cause the dynamic range of the echo signal to exceed 200 dB [178], and use of a TVG reduces this dynamic range.

Figure 3.1 shows the one-way and two-way transmission losses L_T versus range R, assuming a cylindrical propagation model, for a distance up to 1100 m, with an example absorption coefficient $\alpha_{water} = 60\,\text{dB/km}$. At short ranges, the time gradient of the TVG is large. It is generally assumed that the TVG function will exactly compensate for the transmission loss of the acoustic wave. However, empirical results have shown that the compensation from TVG is accurate only for a very short sounding pulse or at very large range [179]. The errors are prominent at shorter ranges where the slope of the TVG is high and the gain of the receiver changes sharply within the pulse length. A slight mismatch in applying TVG to the signal results in non-constant amplification of the signal to magnitudes lower or higher than is actually required. This results in errors in target strength estimation, range estimation as well degradation of the correlations. It is therefore suggested that an accurate TVG function must be used based on the range of the targets from the transducer [177, 180].

Based on (2.1), three models of acoustic wave propagation have been discussed in [178]. Marszal has suggested a $20 \log R$ TVG where a cylindrical wave propagation model is more appropriate, i.e., for very long distances ($16 - 32\,\text{km}$) or for shorter distances for low depths. Similarly, $40 \log R$ TVG is suggested for relatively small distances (4km maximum) for significant depths and a $30 \log R$ for for medium distances ($4 - 8\,\text{km}$) for medium depths. A detailed

discussion on exact, approximate and the theoretical $20/30/40 \log R$ TVG function is presented in [181] along with the merits of each type of TVG and applicability. In [105], a piece-wise linear TVG to compensate for the transmission loss has been suggested. A weighted TVG instead of the standard $20/30/40 \log R$ models for enhanced target detection has been suggested in [179] as the application of a TVG in its simplest form based on (2.1) has limitations.

In this thesis, we use the standard TVG function based on (2.1), and investigate its effects on correlation sidelobes of the optimised sets of codes from this research.

3.5   Doppler Tolerant Code Optimisation

The design of sets of codes having good correlation properties is a traditional problem in radar and sonar systems [182–184]. The codes in such a set must have good correlation properties, defined by a narrow mainlobe autocorrelation peak, flat and minimal autocorrelation sidelobes as well as all crosscorrelation sidelobes. Another factor of merit is the Doppler tolerance which enhances the robustness of these codes to movement of the source and / or the target.

An overview of Doppler modelling divided into 4 different stages from the mid-1980s until the most recent models has been presented in [23]. Methods to compensate for Doppler dispersion effects for enhanced detection and resolution have been proposed as early as 1965 in [185]. Pulse compression codes for radars and sonars have been proposed in [163, 186–189]. However, all these methods create codes that have a limitation in their length. Performance assessment of these codes has been carried out in [190]. A novel technique has been proposed in [191] for passively locating targets which are moving at subsonic and supersonic speeds to produce a significant Doppler shift.

The Doppler effect tends to degrade the correlation properties of signals if the matched filter is not Doppler compensated, or the codes are not robust to the Doppler effect. Designing sets of codes using gradient-descent methods for minimal correlation SLLs is a difficult problem and achieving Doppler tolerance adds another constraint to this complex optimisation problem. Some compromise is often required to achieve a solution that meets some aspects of the design specification. Designing such a set of codes is a motivation for research and a number of design methods have been contributed toward this problem.

Deng [192], Liu [193] and Babur et al. [194] have suggested approaches for optimising short-length codes for good correlation properties in the presence of Doppler. Different approaches to improve Doppler tolerance of polyphase codes have been suggested in [195–197]. However, these methods only target polyphase codes which inherit the disadvantage of limited lengths. In the case of sonar, signals must be sufficiently long to inject a substantial amount of energy. In [198], the suggested method targets autocorrelation optimisation for different Doppler shifts.

Similarly, another proposed algorithm in [199] optimises weighted integrated sidelobe levels (WISL) of autocorrelations for different Doppler shifts, but does not consider crosscorrelation.

For Low Correlation Zone (LCZ) [141], several cyclic algorithms (CA) have been proposed in [200]. The CA optimise the autocorrelation SLLs for a LCZ while ignoring the crosscorrelations. Wang [201] and Palomer [202] have also proposed algorithms to reduce autocorrelation SLLs in an LCZ. Stoica and He in [171, 203] have proposed CAs to reduce SLLs of both the auto- and crosscorrelations, however, in an LCZ. The most recent algorithm proposed in [201] also optimises codes for low auto- and crosscorrelation SLLs for Doppler tolerance in an LCZ.

All of these methods do not completely fulfil the requirements for an optimal set of codes for a multicode sonar system; flexibility in length n of codes [201], optimality in both the auto- and crosscorrelations, optimising SLLs across the entire range of offsets for both auto- and crosscorrelations, optimal power profile and optimality for a range of Doppler shifts [204]. Some algorithms have only targeted code optimisation for Doppler shifts for periodic codes, which find limited application for a sensing application like a multicode sonar system.

3.6 Mismatched Filtering

Improving range resolution and SNR in radar or sonar requires decreasing the pulse width and increasing peak power respectively. Where circumstances do not allow an increase in peak power, this is accomplished by using pulse compression techniques using a matched filter. The impulse response of a matched filter is simply a scaled, time-reversed complex conjugate of the transmitted signal. Higher SLLs from the matched filter may mask important yet weaker reflections from other targets in the vicinity at non-zero offsets. As discussed previously, for systems employing a pulse compression technique, it becomes imperative to minimise the SLLs while maintaining a relatively narrow peak mainlobe.

A partial solution exists in the form of mismatched filtering which reduces the SLLs [205] at the cost of peak SNR [206, 207]. A Mismatched Filter (MMF) can be considered as a combination of an inverse filter and a sidelobe reduction filter. The mismatched filter differs from the matched filter in terms of the transfer function and filter length [208]. Since the "mismatched" filter is now no longer matched to the transmitted signal, there is a slight loss in the mainlobe peak, hence a loss in the SNR as compared to a matched filter. The length of a mismatched filter is usually longer than the transmitted signal.

Nuthalapati has given a brief history of sidelobe reduction methods by using various mismatched filtering techniques [209]. Existing optimisation techniques such as Minimax approach, Recursive Least Squares (RLS), Least Mean Square (LMS), etc., have all been considered but usually found only efficient for shorter code lengths, or effective for minimising

the integrated sidelobes. For longer code lengths such as $n > 1000$, all these approaches become computationally intensive requiring more processing power and memory [210]. Unlike the code design problem, designing a mismatched filter for required properties is a relatively easy and it is simple to find a solution.

Mismatched filtering of sonar signals was discussed as early as 1973 by Ackroyd and Ghani in [206]. It has been suggested to extend the method of geophysicists for sidelobe suppression using a mismatched filter because it was equally applicable to the radar signal processing domain. It has also been suggested that these methods could be extended to codes other than Barker sequences [155].

Kesler and Haykin [211] have shown that modifying the matched filter by weighting using a Kaiser window can reduce the sidelobes effectively. There is a corresponding increase in mainlobe width, for which it has been proposed to increase the length of the modified filter by as much as 50%. This results in sidelobe reductions to as low as 40 dB below the peak value. The mainlobe width is increased by a mere 5% and the output SNR is affected by 1.5 dB. Baden and Cohen [212] have used weighted mismatched filtering with a filter length different than the code length to suppress sidelobes in an LCZ / ZCZ. Kesler and Haykin argued that in the case of orthogonal codes, it is not possible to suppress the sidelobes across the range of all auto- and crosscorrelations. Hence, they have used a weighting function that minimises sidelobe in a region around the zero-offset only. The outputs of these mismatched filters give results similar to matched filtering of GO codes [134].

Luke in [213] has discussed construction methods for quadriphase and 8-phase "perfect" periodic sequences which have almost zero SLLs across the entire range by using mismatched filtering. There is some loss of SNR level, however, the loss compared to the matched filter is very slight. The sequences can be of any arbitrary length and are an improvement over previous existing sequences of short lengths. Another method suggested in the publication is using periodic sequences for aperiodic applications. This is achieved by transmitting three periods of the sequence and mismatched filtering with a single period of the sequence to yield optimal correlation behaviour.

Nuthalapati has suggested an LMS based iterative algorithm coupled with minimax optimisation for designing mismatched filters [209]. The method effectively reduces SLLs for any code lengths. Nuthalapati suggests the option of using a mismatched filter coupled with a matched filter as well as using a mismatched filter directly. Using a mismatched filter derived from this technique, with the filter length equal to the code, the SLLs are lower than when using a matched filter of the same length. There is however a slight loss in the autocorrelation mainlobe peak. Increasing the length of the mismatched filter results in a further reduction of SLLs, with a corresponding, slightly increased loss in mainlobe peak. The improvement, however, is only

viable for a filter length of 3 times the length of the code.

Adly et. al. have proposed a mismatched filtering technique in [214]. This technique produces zero sidelobes for aperiodic codes with no widening of the mainlobe. The technique has been applied to various classes of aperiodic codes. These include aperiodic codes obtained from one period of perfect periodic codes and maximum length sequences (MLS), bi-phase complementary pairs and odd length Barker codes to name a few. These classes of codes have been specifically optimised for various communication applications, radars and as preamble codes for synchronisation. However, the codes are all short in length.

An optimal sidelobe suppression method has been suggested in [212] using weighted mismatched filters to create regions of LCZ / ZCZ around the zero-offset of autocorrelation. Baden has shown that this method promises a higher range resolution provided the LCZ /ZCZ is wide enough. The length of the weighted mismatched filter need not be the same as that of the transmitted codes.

Rabaste [215] has suggested a method to design an optimal mismatched filter to reduce SLLs for autocorrelations only. Rabaste models the problem finding the mismatched filter that will reduce the SLLs as a QCQP. The method is modified to include additional constraints such as a specific Loss in Processing Gain (LPG), specific mainlobe and sidelobe shapes as well as the threshold on SLLs.

3.7 Summary

This chapter presented an overview of the state of the art in code optimisation and different factors affecting the optimality of designed codes, with reference to application in multicode sonar systems. The chapter started with introducing the concept of channel coherence and its effects on a signal while propagating through UWA channel. This was followed by a detailed review of literature on correlation bounds for set of optimal codes, that restrict the SLLs for any given set of codes. The state of the art in the design of optimised sets of codes prior and post Welch bound [25] was then presented, with reference to achieving SLLs against the bounds. A review of the TVG and its effects on the optimised codes and their correlation is briefly discussed. This is followed by a brief review of designing codes that are Doppler tolerant. The effect of the channel, the TVG and the Doppler shift all contribute to degrading of SLLs of the designed sets of codes especially when a matched filter is used. The state of the art in reducing the SLLs using a mismatched filter is presented thereafter,

Chapter 4

Code Optimisation for Multi-code Sonar Systems

In the middle of difficulty lies opportunity.
Albert Einstein.

Much research has been done to find orthogonal sets of codes with properties of delta-like autocorrelations and zero crosscorrelations. Such codes are not obvious and must be found by optimisation for applications like a multicode sonar system.

In underwater sensing applications, one of the factors limiting the survey area is the limited beam-width of electroacoustic transducers. This implies that a transducer can only cover a certain area effectively. In order to increase the survey area and speed, one idea is to use multiple transducers to cover a wider area. However, the overlap and interference of transmitted signals from various transducers, when operating in the same frequency band, reduce the performance of the system. Another option is to use different frequency bands for each transducer. However, reducing the bandwidth of the system reduces the range resolution that can be obtained in each of the sub-bands. Yet another viable option is to use individual codes dedicated to each of the multiple transducers. This option addresses the shortfalls of the previous two options, yet requires the set of codes to be mutually orthogonal to achieve the desired aims for such a multicode sonar system.

This chapter covers details of a novel code optimisation method for a multicode sonar system. The optimisation algorithm proposed in this thesis aims to design codes that have good correlation properties; a single peak at zero-offset with flat and minimal autocorrelation sidelobes and flat and minimal crosscorrelations. The optimisation algorithm is flexible so that it can work for any number of codes in the codes set, comprising codes of any length. It can also include a

number of constraints such as the projector and hydrophone frequency response, channel effects, conformance to a target power profile, as well as Doppler tolerance. In this chapter, we discuss detailed results pertaining to all the above constraints except for Doppler tolerance, which will be discussed in Chapter 6. We also suggest a novel compromise to achieve SLLs beyond the Welch bound [25].

4.1   Motivation for Optimising Two or More Codes

Underwater surveying over a wide swathe of the seafloor can be time-consuming and costly if the transducer has limited spatial coverage for transmitting and receiving probing signals. The area that a transducer can effectively cover depends on the angle-dependent frequency response of the transducer, that is, the beam pattern. The limitations of the projector and hydrophone can be overcome by dividing the coverage area into two or more zones. Each of these zones may then be covered by a dedicated transducer with its own code. Multiple transducers with dedicated codes will enable a wider coverage in a single swathe. Figure 4.1 shows the two cases for a single transducer (with single or multiple codes) and two transducers with their own dedicated codes.

The codes in the set of codes for such a scenario are required to be mutually orthogonal, so as to reduce mutual interference. The optimised codes may then be transmitted from the same projector, or from multiple dedicated transducers simultaneously. One way to achieve this is to transmit codes in disjoint frequency bands. This has the advantage of fairly simply ensuring minimal interference between the two codes. One the other hand, if we allowed the code to occupy the full available bandwidth, but algorithmically ensure low mutual interference, we can achieve greater range resolution. It is this latter approach that is explained in detail in the remainder of this thesis.

Minimising mutual interference between two or more codes using algorithms is usually implemented based on one of two metrics. The first is the peak sidelobe level (PSL), the largest value of the sidelobes of either auto- or crosscorrelation after the mainlobe. Optimising the set of codes for PSL implies to minimise the overall SLLs as much as possible, thereby ensuring a flat profile across the entire range of SLLs. A solution to this is to achieve a threshold with all SLLs equal to or less than this threshold. PSL optimisation is favoured for applications which require detection of targets in the presence of large discrete signals from clutterers [216].

The second metric is the Integrated Sidelobe Level (ISL), which is the sum of all the individual SLLs of auto- and crosscorrelations of the codes being optimised from the set of codes. Optimising the ISL means to minimise all the SLLs so as to achieve a lower overall sum. Optimising sets of codes for ISL may result in very low SLLs for some of the regions of offsets, and possibly zero
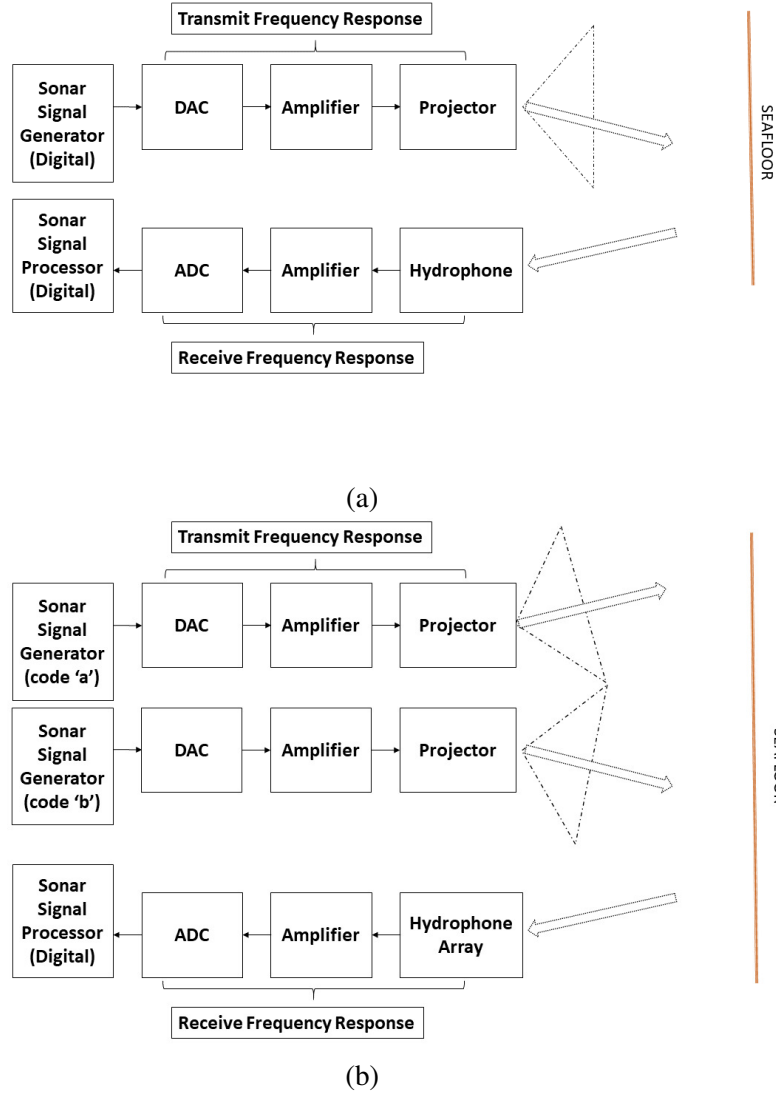
Figure 4.1: Simplified multicode sonar systems for surveying. (a) Single Projector. (b) Two Projectors.

SLLs as well. However, this does not ensure an overall low and flat profile of correlations. This method is usually favoured when the radar or the sonar is designed for a scenario dominated by distributed clutter and the total power from all the sidelobes is important [216].

In this thesis, we choose to optimise the first metric, that is, the PSL of the correlations of the optimised codes. We achieve mutual orthogonality and minimal interference between the codes of the optimised set, and are able to achieve flat and low profile of auto- and crosscorrelations for

all offsets[1].

4.2   Code Optimisation

Optimising a set of codes for some desired properties or constraints may be achieved in a number of ways. Sets of codes have been optimised for various properties, using different optimisation approaches and algorithms. A brief introduction to code optimisation has already been covered in Section 3.3.2. An overview of different optimisation methods with their advantages and limitations are covered next.

Optimisation methods help to solve problems by finding variable values that optimise a multi-variate objective function under a set of constraints. The constraints define a search space within which the optimisation must be solved and the solution must be enclosed. An optimisation algorithm is a procedure which is executed iteratively by comparing various solutions until an optimum or a satisfactory solution is found.

A standard form for many optimisation problems is finding a solution to the vector set $\mathbf{A}$ with m vectors each of length n, that is, $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_m}\}$

$$
\begin{aligned}
\text{Min} : \quad & f(\mathbf{a}) \\
\text{Subject to} : \quad & g_j(\mathbf{a}) \quad \leq \quad 0 \quad j = 0, 1, 2, ..., p \\
& h_k(\mathbf{a}) \quad = \quad 0 \quad k = 0, 1, 2, ..., q \\
& a_{iL} \leq a_i \leq \quad a_{iU} \qquad i = 0, 1, 2, ..., r
\end{aligned}
$$

where $f(\mathbf{a})$ represents the objective function, $g_j(\mathbf{a})$ are inequality constraints, $h_k(\mathbf{a})$ are equality constraints. a_i is the design variable of size n, bounded by the lower limit a_{iL} and the upper limit a_{iU}. Optimisation methods attempt to find the combination of design variables that satisfy all the equality, inequality and side constraints and achieve the objective function with minimum cost. The optimisation problem may have a number of equality and inequality constraints, or it may have none.

Optimisation algorithms may be broadly classified into local and global algorithms. Most local optimisation methods are gradient-based iterations [217]. Some of the gradient-based algorithms may be found in [218] and references therein. Local optimisation methods also include non-gradient based methods such as Powell's Method [219] and Nelder-Mead simplex algorithm [220].

Local optimisation algorithms do not always guarantee convergence of the optimisation problem to a global minimum. The optimisation iterations mostly end up in a local minimum. The probability of finding a global or near to global minimum are higher with global optimisation

---

[1]Section 4.3 elaborates more on the results of optimised codes.

methods. Global optimisation methods may be further classified as evolutionary algorithms or deterministic algorithms. Within the evolutionary algorithms, the two most common algorithms are Genetic Algorithm [221] and Particle Swarm Optimisation (PSO) [222].

4.2.1 Local vs Global Optimisation Methods

Each type of optimisation method briefly described in the above subsection may be suitable for an application, given the objective function, the constraints, and the circumstances. The local optimisation methods are efficient in terms of the information required and the number of iterations required to reach the local optimum, while being simple to implement for large design variables, and requiring a little parameter tuning. They often lead to very simple and economical solutions. Local algorithms tend to solve the optimisation problem by finding one of the local minima, which may not be the global minimum. For local optimisation methods such as the gradient-based methods, the stopping criterion for optimisation is usually satisfying the Karush-Kuhn-Tucker (KKT) conditions [218, 223]. The conditions imply there is no change in the gradient or the change is less than a predefined threshold. Depending on the problem, this local minimum may or may not be the global minimum. The drawbacks associated with the local optimisation methods include that they find a local minimum which is not necessarily a global minimum, and difficulty with discrete problems [218].

Global methods, on the other hand, have the advantage of being robust, having an increased chance of finding the global or near-global minimum, easy to implement and well-suited to discrete problems. However, the major disadvantages are high computational cost, poor constraint-handling properties, problem-specific parameter tuning and limited problem size. Based on the advantages and disadvantages of each, it is often recommended to use global optimisation methods unless they are not viable and so a local optimisation method must be used.

This thesis suggests a novel gradient-descent based optimisation method, that has sufficient flexibility to optimise sets of codes for the desired correlation, the power profile, the channel response, the projector, and the hydrophone response, and the Doppler effect.

4.2.2 Gradient-Descent Method

The gradient-descent algorithm [224, 225] is the most commonly used local optimisation method. It uses a two-step iteration to find the optimum. Consider a function $f(\mathbf{a})$ where a_n is the n-th element from the vector $\mathbf{a} = [a_1, a_2, ..., a_n]^\mathrm{T}$. The first step is computing the derivative $g(\mathbf{a})$ from $f(\mathbf{a})$ for finding the direction of next move. The gradient vector $g(\mathbf{a})$ of this function is given by

partial derivatives of $f(\mathbf{a})$ with respect to each of the independent variables x_n as follows

$$f'(\mathbf{a}) = \bigtriangledown f(\mathbf{a}) \equiv \begin{bmatrix} \frac{\partial f}{\partial a_1} \\ \frac{\partial f}{\partial a_2} \\ . \\ . \\ . \\ \frac{\partial f}{\partial a_n} \end{bmatrix} \tag{4.1}$$

The derivative $f'(\mathbf{a})$ is then scaled by a small step size μ to move in the direction of optimum. This process is applied iteratively as follows

$$f(\mathbf{a})_{t+1} = f(\mathbf{a})_t - \mu \, f'(\mathbf{a}) \tag{4.2}$$

The step size μ is also known as the learning rate. The learning rate is a positive scalar value that determines the size of each step in the optimisation process. The step size μ is chosen so that the approach towards finding the optimal may be controlled. If the learning rate is too small, the optimisation process can be slow. If the learning rate is too large, optimisation can overshoot the minimum and may fail to converge, or even diverge.

Gradient-descent based optimisation can converge to a local minimum even with a fixed learning rate. As we approach a local minimum, the gradient becomes smaller and the optimisation process will automatically take smaller steps. An adaptive learning rate may also be used so that the step size is adapted according to certain conditions.

The process is repeated until a local minimum is achieved. The stopping criterion may be a fixed number of iterations or

$$\left\| f'(\mathbf{a}) \right\| < \varepsilon \tag{4.3}$$

where ε is some predefined threshold.

4.3   Investigations – Aperiodic Correlations

Sonar and radar generally use aperiodic codes for sensing and hence the aperiodic Welch bound [25] from (3.4) is applicable for all such cases. An aperiodic transmission is characterised by a single transmission followed by a wait for the received echo. The basis of the Welch bound [25] is considering an ideal autocorrelation with a delta-like feature at zero-offset. All other offset values are very low (ideally zero), and the largest of these can be no less than the value given by the Welch bound [25] for that set of codes. These bounds have an important implication in code

n	m	$20\log_{10} c_{\max}$(dB)
10	2	-15.7
100	2	-26
1000	2	-36
10000	2	-46
10	3	-14.5
100	3	-24.7
1000	3	-34.7
10000	2	-44.8

Table 4.1: Bounds for aperiodic codes consisting of *m* codes of length *n*.

and waveform designs for multicode communication systems. Lower SLLs in autocorrelations increase the resolution for differentiating between two likely targets. The crosscorrelation SLLs measure the interdependence of the codes in the set of codes. Lower SLLs in crosscorrelations indicate a lower interdependence of the codes and better identification.

The Welch bound [25] for different length *n* and number *m* of aperiodic codes using (3.4) are given in Table 4.1. The values of $c_{\max}$ in Table 4.1 are the limits on how low the highest of the sidelobe can be for a particular set of codes.

4.3.1 Code Optimisation Algorithm

In this research, we used a gradient-descent based algorithm that was developed in [29] to work iteratively on randomly initialised unit-norm complex vector-sets. The algorithm is able to optimise sets of codes for $m \geq 2$ codes each of length $n \geq$ m. This algorithm allows only the zero-offset of autocorrelation to have a maximum value whereas all sidelobes of the autocorrelations and all crosscorrelations are treated alike and pushed down using a gradient-descent optimisation method. As already mentioned in Section 4.1, we aim to optimise the PSL.

The algorithm starts with initialisation of a vector set with randomly initialised numbers, e.g., two codes **a** and **b**, $\in \mathbb{C}^n$, denoting a set of codes with number of codes $m = 2$ each with length *n*. The set of codes is normalised to have unity norm i.e., $\|\mathbf{a}\| = \|\mathbf{b}\| = 1$. Next, a target power profile is defined. The receiver and transmitter effects and the underwater acoustic channel are modelled as an FIR filter, and the vector-set representing a set of codes is convolved with this filter before being subjected to the algorithm for optimisation.

The iteration starts with the magnitude of the codes being forced to conform to the designated power profile with PAPR ≈ 1. This is done by finding the maximum and minimum values of the magnitude of the samples as compared to the target profile. The samples with a magnitude higher

than the target profile are forced to be reduced while the samples with magnitude lower than the target profile are forced to increase. The magnitude of the code is optimised in conformance to the target power profile using a variable learning rate μ_1, while the phase is retained. The code is then normalised again in every iteration for magnitude constraint $\|\mathbf{a}\| = \|\mathbf{b}\| = 1$.

All possible auto- and crosscorrelations are then computed between the individual codes of the set of codes. Ignoring the autocorrelation value at the zero-offset, the maximum value of the autocorrelation sidelobe or crosscorrelation is identified, with reference to its offset and location in an auto- or crosscorrelation. Based on the offset and reference to the auto-or crosscorrelation, the first derivative is computed[2] The derivative is then scaled using a second learning rate μ_2 and then subtracted from the respective codes, based on whether it was an auto- or crosscorrelation. The updated code is then subjected to the entire iteration again until the stopping criterion is applied.

While enforcing the PAPR constraint, the adaptive learning rate μ_1 was chosen as follows

$\mu_1 = 10^{-1}$ if the PAPR > 10,
$\mu_1 = 10^{-2}$ if the PAPR > 1.001,   and
$\mu_1 = 10^{-3}$ if $1 < $ PAPR ≤ 1.001.

The learning rate μ_2 was chosen to be fixed at 10^{-3} which was found optimal for the algorithm. Choosing a learning rate much lower or higher than this resulted in the failure of the optimisation method to converge to a local minimum.

A single execution of the iteration reduces a single maximum value in the sidelobes of autocorrelations and all crosscorrelations. The stopping criterion was that the difference between maximum SLLs in successive stages was lower than a threshold or a fixed number of iterations is completed. Successive iterations result in converging of the optimisation problem to a local minimum. The result is flat and low SLLs across the entire range of crosscorrelations and all autocorrelation sidelobes. The maximum number of iterations K was one million, although we modified this for some results as will be discussed in Section 4.3.2. The algorithm implementing this optimisation is shown in Algorithm 1.

The algorithm is flexible and so allows for a number of constraints to be added to optimise sets of codes. Modifications to the algorithm will be introduced later. Detailed mathematical derivations to calculate the gradients for various offsets of auto- and crosscorrelations are given in Appendix B. The PAPR constraint has not been included in the optimisation criteria of previous works. All references to SLLs in previous literature have been quoted with reference to optimised sets of codes but do not include the PAPR constraint. This is a novel aspect of this research.

---

[2]See Appendix F for derivatives of complex numbers.

1: ***define PAPR profile;***
2: *choose learning rates μ_1 and μ_2;*
3: *initialise set of codes (**a** and **b**) from pseudo-random sequences;*
4: ***define filter for transmitter, receiver and channel effects;***
5: *convolve **a** and **b** with the channel filter;*
6: **for** iter=1 **to** *K* **do**
7:   ****find code samples with max and min magnitude deviation from profile;***
8:   ****multiply max magnitude code sample by 1 - μ_1;***
9:   ****multiply min magnitude code sample by 1 + μ_1;***
10:   *normalise code sets for unit norm;*
11:   *calculate aperiodic correlations;*
12:   *ignoring the autocorrelation zero-offset, find offset with maximum correlation value;*
13:   *compute the derivative of maximum value of correlation wrt **a** and / or **b***
14:   *subtract μ_2 scaled derivative from code **a** and / or **b***
15:   *check for stop criteria; maximum iterations or difference between successive maximum SLLs less than some threshold;*
16: **end for**

Algorithm 1: Pseudo-code for codes optimisation using the gradient-descent method.

4.3.2   Aperiodic Code Optimisation

Using the optimisation algorithm described in the previous subsection, a set of experiments was run to optimise sets of codes. The algorithm was run 40 times, each with a different random initialisation for the set with $m = \{2,3,4,5\}$ and lengths $n = \{8,10,20,...1000\}$. The codes were optimised without any additional constraints except for the desired correlation properties. This means the codes are not forced to follow the target profile with PAPR ≈ 1, and neither is the channel and transducer impulse response incorporated in Algorithm 1 (steps highlighted in bold). Figure 4.2 shows the best SLLs achieved, and also the Welch bound [25]. Only results for $m = 2$ and $m = 5$ have been presented for brevity. The SLLs are pushed down close to the Welch bound [25], however, they do not quite achieve the bound.

Another set of experiments was run forcing the magnitude of the set of codes to follow the target power profile, as well incorporate the channel and transducer impulse response. The constraints from Algorithm 1 (highlighted in bold) were incorporated in the iterations. Figure 4.3 show that the magnitude of the set of codes samples closely following the target power profile. These additional constraints slightly affect the performance of the optimisation algorithm so that SLLs are not pushed down so effectively. Results are shown in Figure 4.4 with the best SLLs for sets of codes with $n = \{8,10,20,...1000\}$. Only results for $m = 2$ and $m = 5$ are presented for brevity. We notice slightly higher SLLs compared to when the codes are not subjected to power constraints in Figure 4.2.
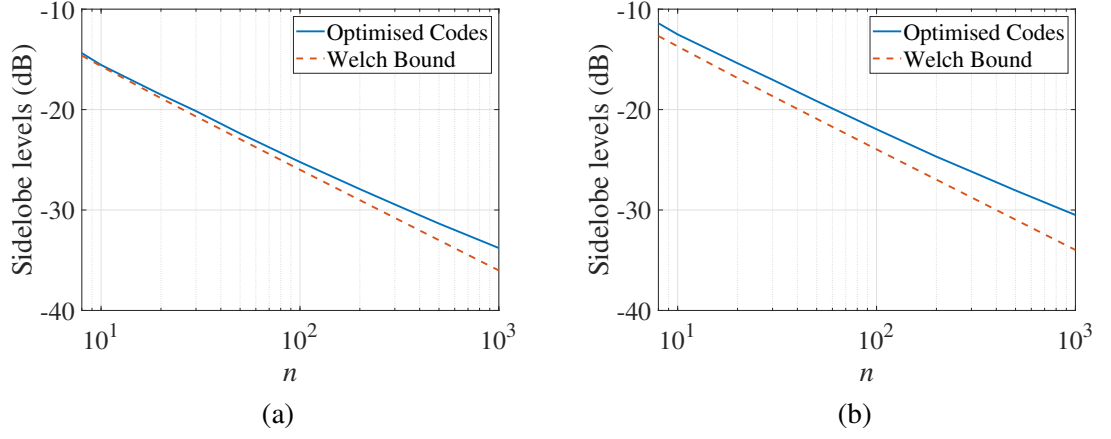
Figure 4.2: Aperiodic correlation SLLs. Codes have length $n = \{8, 10, 20, ...1000\}$ and are optimised without power constraints. Number of codes (a) $m = 2$ (b) $m = 5$.
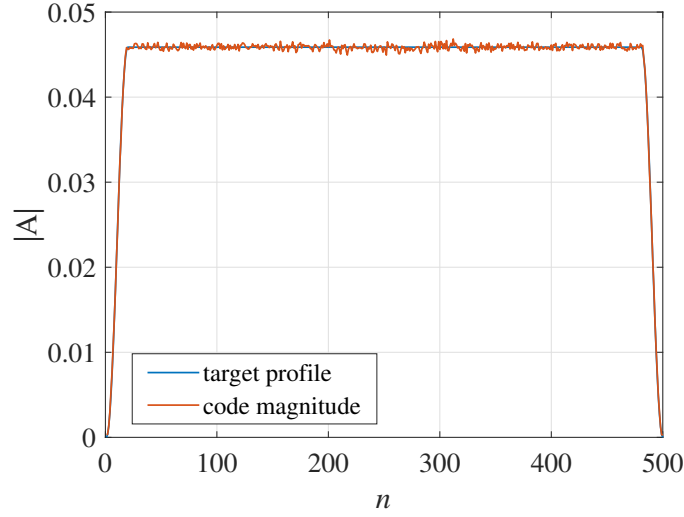

Figure 4.3: Set of codes with $m = 2$ and length of codes $n = 500$ optimised for a target power profile as shown. The magnitude of the code samples closely follow the target power profile.

The above-mentioned optimisations comprised iterations which were run for one million times or until there was no significant change in SLLs. A single optimisation for a set of codes with $m = 2$ and lengths $n = 1000$ took ≈ 1240 secs on a commercial 7th generation Core $i7$ Desktop PC, having a clock speed of 3.2 GHz and 16 GB RAM. Each optimisation terminated on a local minimum and this local minimum is particular to each initialisation.

In order to explore the nature of the local minimum, we repeated the experiments by setting
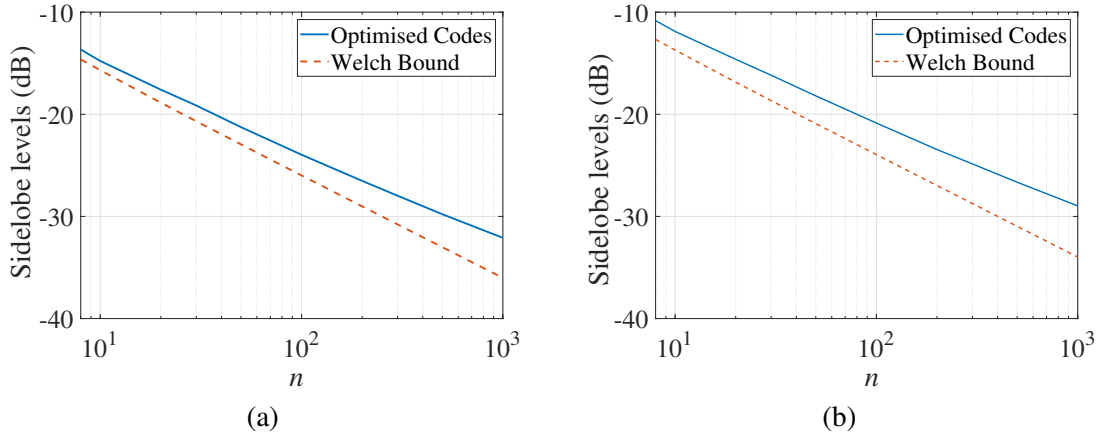
Figure 4.4: Aperiodic correlation SLLs. Codes have length $n = \{8, 10, 20, ... 1000\}$ and are optimised with power constraints. Number of codes (a) $m = 2$. (b) $m = 5$.
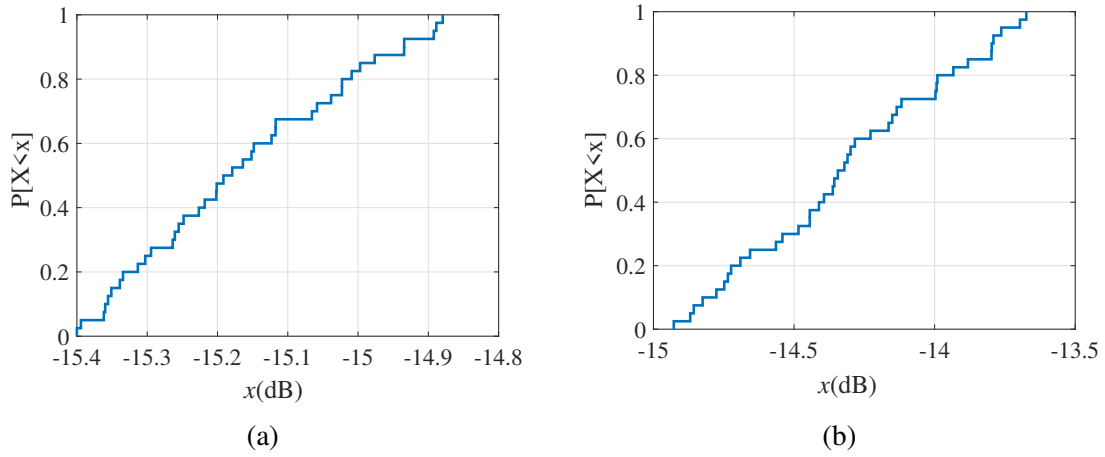

Figure 4.5: Empirical Cumulative Density Function (ECDF) plots for codes $m = 2$ and length $n = 10$ optimised for a very long time. (a) Without power constraints. (b) with power constraints.

the number of iterations $100 \times$ higher, that is, set to 100 million iterations. Another set of 40 experiments was run for some very short-codes, with $m = \{2, 3\}$ and length $n = 10$. Figure 4.5 shows the ECDF for the resulting SLLs when codes with $m = 2$ are optimised for a very long time. Figure 4.5(a) shows the ECDF of SLLs when codes were optimised without without any constraints. Figure 4.5(b) shows the ECDF of SLLs when codes are optimised with the additional constraints of conformance to the power profile and incorporating channel and transducer impulse response. The SLLs reached for the various initialisations are close to each other but not equal.

Results from this set of experiments can be compared with the results shown in Figure 4.2(a)

and Figure 4.4(a). The best SLLs for a given code in both cases are almost equal. For example, a set of code with $m = 2$ and length $n = 10$ has its best SLLs as low as 15.4 dB (without power constraints) as depicted in Figure 4.2. This is equal to the best sidelobe in ECDF plotted in Figure 4.5(a). Results for other sets of codes and constraints may be compared in the same way. This suggests but does not prove that codes probably cannot be optimised to achieve sidelobes as low as the Welch bound [25].

In the experiments above, the codes were optimised allowing only the zero-offset of the autocorrelation to have a maximum value. All other values are treated alike and pushed down, resulting in flat and low SLLs. Relaxing this constraint and allowing some of the values around the zero-offset of autocorrelations to be weighted, we expect some different results. It is revealed that the remaining SLLs can be pushed down lower than existing bounds using this approach. Section 4.3.3 elaborates on this concept.

4.3.3   Wider Mainlobe

In the previous sections, a typical set of codes with good correlation properties was optimised for delta-like autocorrelations and flat and low crosscorrelations. The SLLs of the optimised set of codes from Section 4.3.2 were higher than the Welch bound [25]. However, if some of the values around the zero-offset of autocorrelations are "exempted" from the optimisation criterion, it is observed that SLLs beyond the Welch bound [25] are possible. The concept is analogous to pushing energy from sidelobes into the centre; the sidelobes samples will contribute their energy to the mainlobe samples so that the mainlobe grows wider and the sidelobes go down.

This can be implemented in many ways. One is by defining a region comprising a few values of autocorrelations around the zero-offset, and weighting these values monotonically between 0 and 1 before subjecting them to optimisation. Weighting by 0 implies multiplying the autocorrelation sample by 0, and hence, being completely ignored by the optimisation algorithm. Weighting by 1 implies subjecting the sample value as is to the optimisation algorithm, alongside the remaining sidelobe values.

In the algorithm, this is implemented by defining a mask and passing the mask to the optimiser function. The aperiodic correlations are computed as before, however, the autocorrelations are multiplied by the mask before being subjected to the search for the maximum value. Multiplying by the mask ensures that the wider mainlobe is exempted from the search for the maximum value and only sidelobes in the region excluded by the mask are pushed down. The resulting autocorrelation shape is similar to the type of the mask used, with a wider mainlobe centred around the zero-offset. The modified algorithm is shown in Algorithm 2.

1: *define PAPR profile;*
2: *choose learning rates μ_1 and μ_2;*
3: *initialise set of codes (**a** and **b**) from pseudo-random sequences;*
4: ***generate and replicate desired mask***
5: *define filter for transmitter, receiver and channel effects;*
6: *convolve **a** and **b** with the channel filter;*
7: **for** iter=1 **to** K **do**
8:     *find code samples with max and min magnitude deviation from profile;*
9:     *multiply max magnitude code sample by $1 - \mu_1$;*
10:    *multiply min magnitude code sample by $1 + \mu_1$;*
11:    *normalise code sets for unit norm;*
12:    *calculate aperiodic correlations;*
13:    ***multiply autocorrelations by desired mask***
14:    *find offset with maximum value of correlation;*
15:    *compute derivative of maximum value of correlation wrt **a** and / or **b***
16:    *subtract μ_2 scaled derivative from code **a** and / or **b***
17:    *check for stop criteria; maximum iterations or difference between successive maximum SLLs less than some threshold;*
18: **end for**

Algorithm 2: Pseudo-code for codes optimisation with wider autocorrelation mainlobe, the changes from Algorithm 1 highlighted in ***bold**.

4.3.3.1   Masks for a Wider Mainlobe

Using a mask to weight the autocorrelation sample values results in a wider mainlobe with sidelobes lower than the non-weighted case. These masks have two parameters that control the shape and width of the resulting mainlobe. First is the number of samples in a *"Don't Care Region"* denoted by D and second is the number of samples in a *"Partial Care Region"* denoted by P. The samples in the don't care region are treated similar to the zero-offset sample. They are ignored and allowed to have an arbitrary maximum value. The samples in the partial care region are weighted and their maximum value is controlled by the optimisation algorithm and pushed down according to the weight. The remaining correlation samples are treated normally by the optimisation algorithm.

Figure 4.6(a) shows variants of a mask referred to as a Triangular Mask. This mask has only one sample in the don't care condition (at the zero-offset, so that $D = 0$). There are one or more samples in the partial care region ($P \geq 1$). The width of this mask can be controlled by increasing the number of samples in the partial care region. Similarly, Figure 4.6(b) shows different parallelogram masks, with more than one samples in the don't care region ($D \geq 1$), and one or more samples in the partial care condition ($P \geq 1$). In addition to these geometrical

shapes, any arbitrary shape may be used for the masks. Figure 4.6(c) shows a mask generated by a Discrete Fourier Transform (DFT) of the Kaiser window. It is similar to a triangular window with the difference in the number of samples in the partial care.

To implement these masks and see the resulting SLLs, a set of optimisations was run starting with random initialisations. Using different masks as mentioned above, the resulting SLLs for sets of codes with $m = 2$ and lengths $n = \{8, 10, 20, ..., 1000\}$ are compared with the Welch bound [25] as shown in Figure 4.7(a). The legend indicates D as the number of samples in the don't care region and P as the number of samples in the partial care region for each of the mask. The sets of codes have been optimised without any power constraints.

There are some very interesting SLLs which are much lower than the Welch bound [25] using this modified criterion. This is true, especially for the shorter codes. However, the results reveal that if a mask with fixed mainlobe width is used for increasing code length, the resulting SLLs are not as consistently low. The ratio of mainlobe width to the code length must increase for consistent improvement of SLLs.

Another set of optimisations was run by imposing the power constraints with a target power profile. The resulting SLLs are shown in of Figure 4.7(b). Adding the power constraint reduces the ability of the algorithm to optimise the set of codes with SLLs as low as without the power constraint. Comparing Figure 4.7(a) and Figure 4.7(b), a set of two codes with length $n = 100$ samples optimised using a DFT of Kaiser $\beta = 40$ mask has SLLs are as low as -39 dB without power constraint. In the case of power constraint, the SLLs are around -32 dB. Other results may be compared along similar lines.

The modified criterion of widening the mainlobe width yields SLLs lower than the Welch bound [25]. For consistently good SLLs with the increasing code length, the ratio of a number of samples in mainlobe to the length of code must remain approximately consistent. The SLLs are constrained by the number of samples in the mainlobe.

We now present the resulting autocorrelation shapes using a parallelogram mask. A set of optimisations was run for sets of codes with $m = 2$ and lengths $n = \{10, 100, 1000\}$ with various values of mainlobe width d. For the shorter length of codes ($n = 10$), the mainlobe is a well-defined dome. Figure 4.8 shows the results pertaining to a random initialisation of code with $m = 2$ and length $n = 10$. Figure 4.8(a) shows various masks, whereas the remaining sub-figures show some of the resulting correlations. The dome maintains its monotonic shape with an increase in width. With $d = \pm 2$ to $d = \pm 6$, SLLs range from -21 dB to -47.4 dB against the Welch bound [25] of -16 dB. These short-codes may not find practical applications for sonar, but they may be useful in other applications such as frame preambles for synchronisation [226–228].
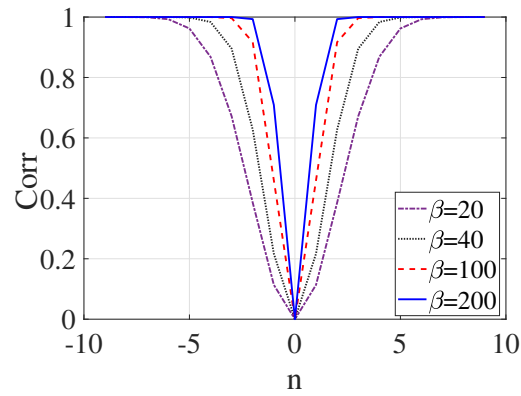
Figure 4.9 shows the masks and resulting correlation plots for a code with $n = 100$. The SLLs are still lower than the Welch bound [25], but the autocorrelation mainlobe starts losing its
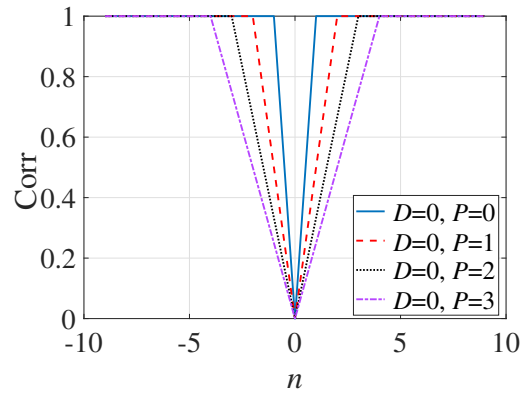
(a)
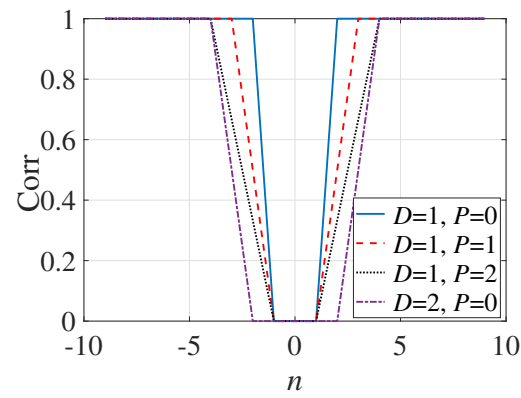
(b)

(c)

Figure 4.6: Masks for various weighting autocorrelations. (a) Triangular Masks with $D = 0$ and different values of P. (b) Parallelogram Masks for various values of D and P. (c) Mask based on DFT of the Kaiser window for various values of β.

(a)

(b)
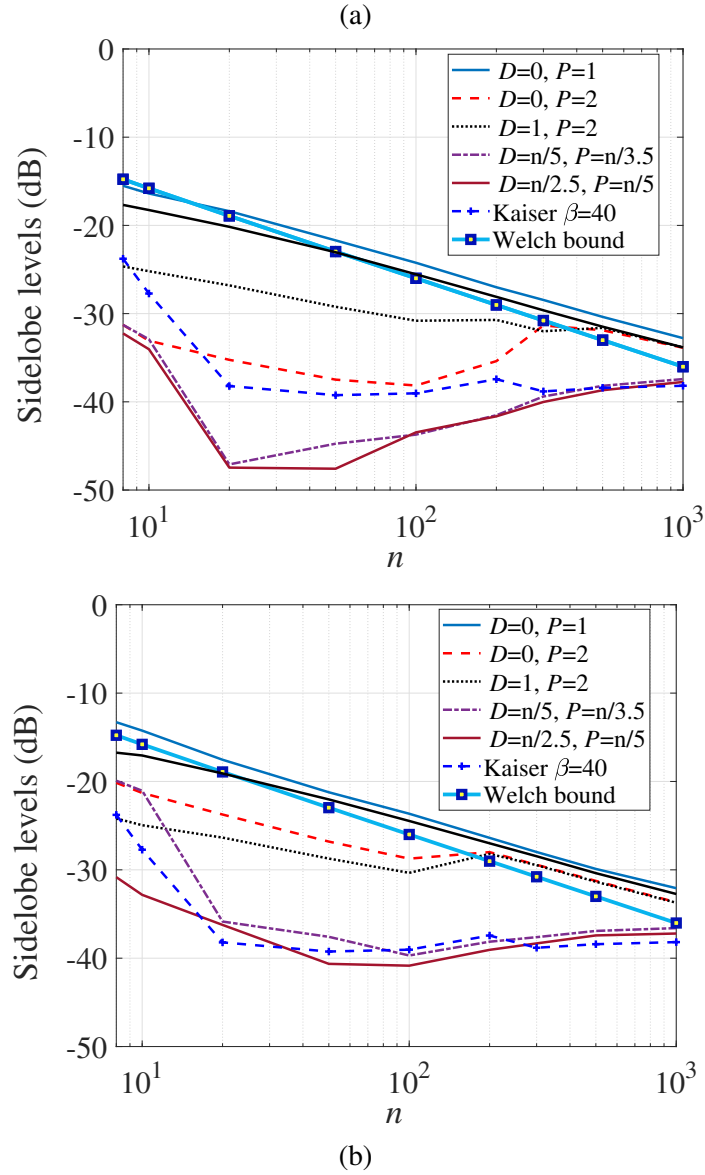
Figure 4.7: SLLs for optimised codes against the Welch bound for various masks. The legend shows corresponding mask used. (a) Without power constraints. (b) With power constraints.

monotonic shape. With mainlobe width $d = \pm 20$ to $d = \pm 60$, the best SLLs range from -45 dB to -47.5 dB respectively, against the Welch bound [25] of -26 dB. Figure 4.10 shows the masks and resulting correlations for code with $n = 1000$. The autocorrelation mainlobe is no longer monotonic.

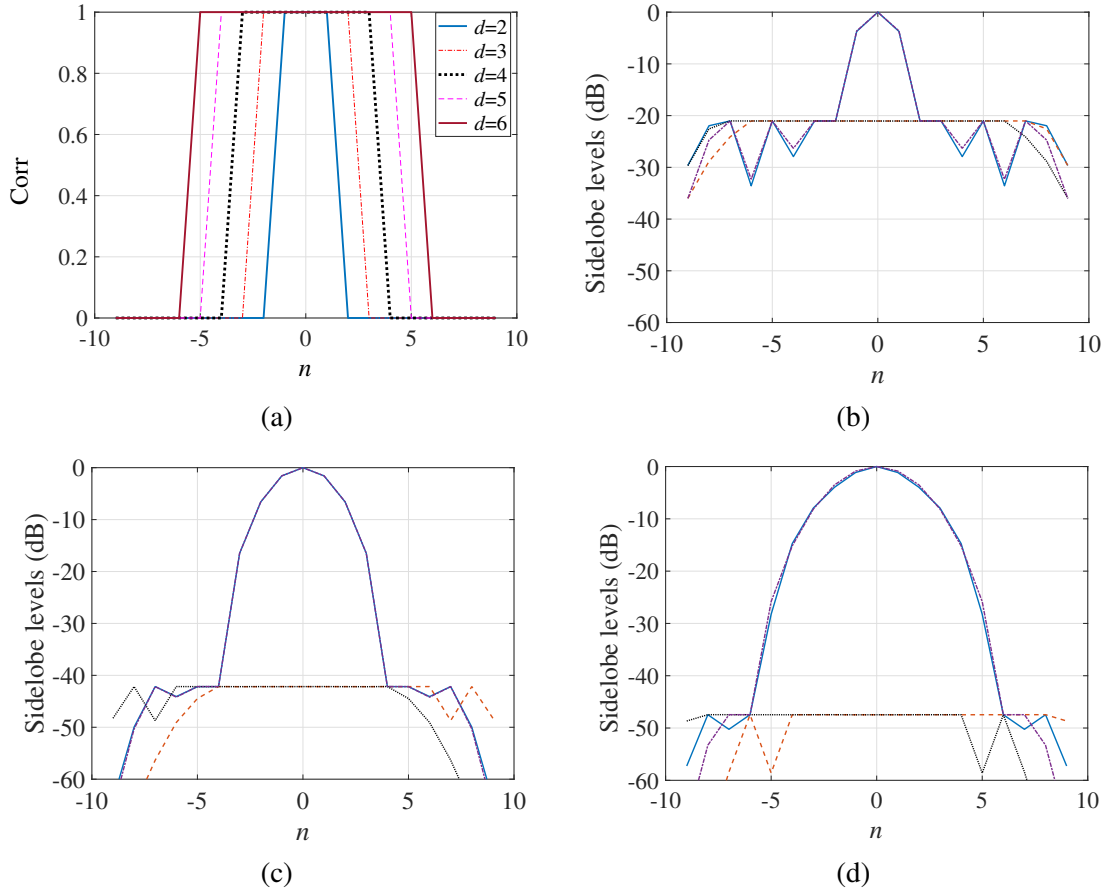Figure 4.8: Aperiodic correlations for code length $n = 10$ with various don't care masks. (a) Target masks. (b) Mask $d = \pm 2$ (c) Mask $d = \pm 4$ (d) Mask $d = \pm 6$.

4.3.3.2 Monotonic Mainlobe

For accurate detection, it is required to minimise the probability of false detections that are far from the correct detection delays. For this, it is required that the autocorrelation mainlobe is monotonic. Monotonic shaping of the mainlobe can be achieved in a number of ways and these techniques are elaborated in the following. This new criterion is introduced by defining two masks, one each for the mainlobe and the sidelobes of autocorrelation and all crosscorrelations. The masks are generated at the beginning of the algorithm. The mainlobe mask is the target mask for mainlobe shaping with a weight of one for mainlobe samples, and only optimises the mainlobe of autocorrelation while ignoring the sidelobes of autocorrelation and all crosscorrelations. The sidelobe mask has a weight of one for all sidelobes of autocorrelation and all crosscorrelations while ignoring the mainlobe of autocorrelation so that sidelobe values are optimised as before.
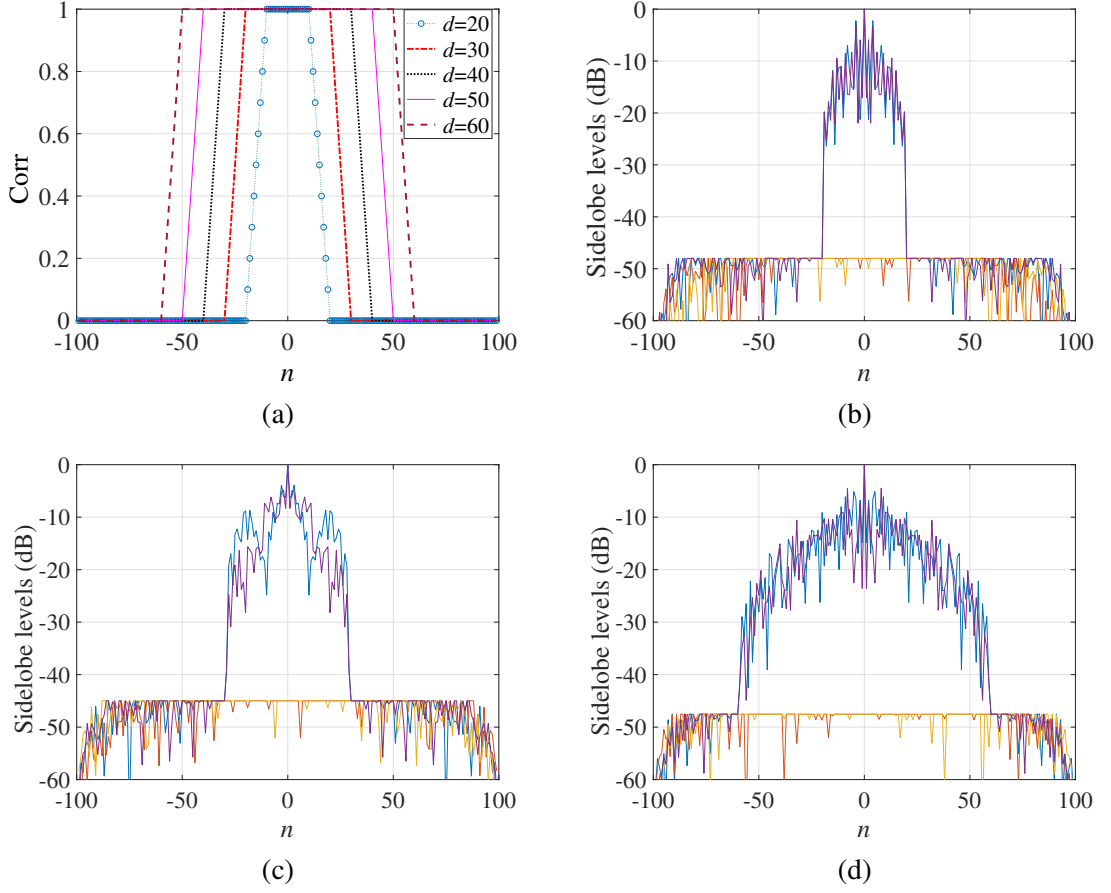
Figure 4.9: Aperiodic correlations for code length $n = 100$ with various don't care masks. (a) Target masks. (b) Mask $d = \pm 20$. (c) Mask $d = \pm 40$. (d) Mask $d = \pm 60$.

The optimisation steps are then made to run alternately between mainlobe and sidelobe optimisations. In each iteration, the correlations are computed as before and then multiplied with the respective mask. While optimising for the mainlobe, the algorithm finds the largest deviation and reduces it if it is larger than the target mask, or increases it if it is smaller than the target reference sample. The sidelobes in the optimisation iterations are continuously pushed down, as there is no reference for sidelobes, in the same way as being done in all previous cases. This is shown in Algorithm 3. It is anticipated that using this method, the mainlobe will have a monotonic shape and the SLLs will also be pushed down simultaneously.

As regards to the choice of target mask, the DFT of some of the windows such as Kaiser, Blackman-Harris, Chebyshev, have a narrow mainlobe with sharp roll-off to very low sidelobes. These windows were used in the time domain as a target mask for the mainlobe so that, the
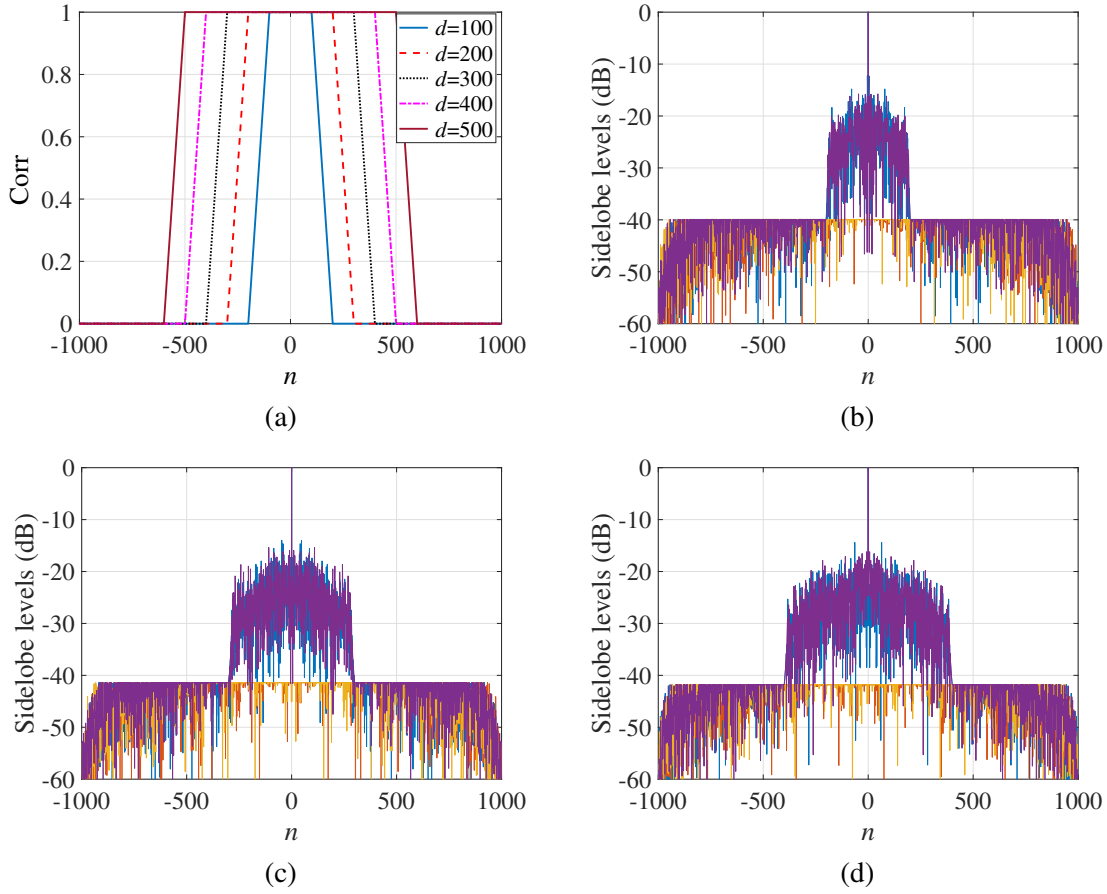
Figure 4.10: Aperiodic correlations for code length $n = 1000$ with various don't care masks. (a) Target masks. (b) Mask $d = \pm 100$. (c) Mask $d = \pm 200$. (d) Mask $d = \pm 300$.

autocorrelations are monotonically shaped and SLLs may be maximally suppressed. Results for codes optimised using the Fourier transform of digital windows as target masks are summarised in Table 4.2 - 4.4 for $n = \{10, 100, 1000\}$ respectively. These include SLLs when codes are optimised without power constraints denoted as $c_{\text{max, NP}}$ and when codes are optimised with power constraints denoted as $c_{\text{max,P}}$. The mainlobe width indicated in Tables 4.2- 4.4 is for the -3 dB width measured in samples. The Welch bound [25] for the given code length is denoted as WB. The Chebyshev-50 indicates a Chebyshev window with SLL suppression of -50 dB and so on. Similarly, Kaiser–10 window indicates a Kaiser window with $\beta = 10$.

This new technique helps to achieve the desired aims of a monotonic mainlobe and flat and lower sidelobes to some extent. The mainlobe is monotonic and perfectly follows the mask with SLLs beyond the Welch bound [25] for codes with lengths up to $n = 100$.

1: *define PAPR profile;*
2: *choose learning rates μ_1 and μ_2;*
3: *initialise set of codes (**a** and **b**) from pseudo-random sequences;*
4: *generate mainlobe and sidelobe masks*
5: *define filter for transmitter, receiver and channel effects;*
6: *convolve **a** and **b** with the channel filter;*
7: **for** iter=1 **to** *K* **do**
8:     *find code samples with maximum and minimum magnitude deviation from profile;*
9:     *multiply maximum magnitude code sample by 1 - μ_1;*
10:     *multiply minimum magnitude code sample by 1 + μ_1;*
11:     *normalise code sets for unit norm;*
12:     *calculate aperiodic correlations;*
13:     **if iter = an even number then**
14:         ***multiply autocorrelations by mainlobe mask***
15:         *find offset with maximum deviation of the correlation from mask;*
16:         *compute derivative of maximum deviation of correlation wrt **a** or **b***
17:         *add or subtract μ_2 scaled derivative from code **a** or **b**, depending if the deviation is less than or greater than the mask sample*
18:         *check for stopping criterion; maximum iterations or difference between successive maximum SLLs less than some threshold;*
19:     **else**
20:         ***multiply correlations by sidelobe mask***
21:         *find offset with maximum value of correlation;*
22:         *compute derivative of maximum value of correlation wrt **a** and / or **b***
23:         *subtract μ_2 scaled derivative from code **a** and / or **b***
24:     **end if**
25:     *check for stopping criterion; maximum iterations or difference between successive maximum SLLs less than some threshold;*
26: **end for**

Algorithm 3: Pseudo-code for codes optimisation with monotonic wider autocorrelation mainlobe, with changes from Algorithm 1 and 2 highlighted in ***bold**.

The parallelogram masks gave us the best SLLs for any length of the codes, but at the cost of non-monotonic mainlobe. However, we sought alternative ways of imposing monotonic mainlobe shaping in addition to optimising codes for low SLLs and power constraints. These alternative approaches for this are discussed in Section 4.3.3.3.

n	Window	Mainlobe Width	WB (dB)	$c_{\text{max, NP}}$(dB)	$c_{\text{max,P}}$(dB)
10	Chebyshev-100	3	-15.7	-21.0	-17.0
10	Chebyshev-200	4	-15.7	-29.8	-24.0
10	Kaiser-20	4	-15.7	-37.1	-24.5
10	Kaiser-40	5	-15.7	41.1	-24.3
10	Hanning	2	-15.7	-17.0	-15.8
10	Blackmanharris	3	-15.7	-24.6	-18.8
10	Flattop	4	-15.7	-42.8	-18.9

Table 4.2: Sidelobe levels and mainlobe width for aperiodic correlations of codes with length $n = 10$ against various Windows used as target mask.

n	Window	Mainlobe Width	WB (dB)	$c_{\text{max, NP}}$(dB)	$c_{\text{max,P}}$(dB)
100	Chebyshev-100	3	-26.0	-30.0	-24.4
100	Chebyshev-200	4	-26.0	-33.1	-24.6
100	Kaiser-20	4	-26.0	-31.8	-24.5
100	Kaiser-40	5	-26.0	-29.2	-24.8
100	Hanning	2	-26.0	-27.1	-24.8
100	Blackmanharris	3	-26.0	-29.7	-24.1
100	Flattop	4	-26.0	-33.8	-25.2

Table 4.3: Sidelobe levels and mainlobe width for aperiodic correlations of codes with length $n = 100$ against the different Windows used as target mask.

n	Window	Mainlobe Width	WB (dB)	$c_{\text{max, NP}}$(dB)	$c_{\text{max,P}}$(dB)
1000	Chebyshev-100	3	-36	-33	-32.4
1000	Chebyshev-200	4	-36	-33	-31.7
1000	Kaiser-20	4	-36	-33	-31.8
1000	Kaiser-40	5	-36	-33	-31.8
1000	Hanning	2	-36	-33	-31.6
1000	Blackmanharris	3	-36	-33	-31.8
1000	Flattop	4	-36	-33	-31.8

Table 4.4: Sidelobe levels and mainlobe width for aperiodic correlations of codes with length $n = 1000$ against various Digital Windows used as target mask.

4.3.3.3 Monotonic Mainlobe Shaping

For code lengths with $n > 100$ and a wider mainlobe, the resulting autocorrelations develop a jagged mainlobe. This is undesirable because it may result in errors in target detection. To address this, three methods were tried. The first method used is a monotonic mask formed using
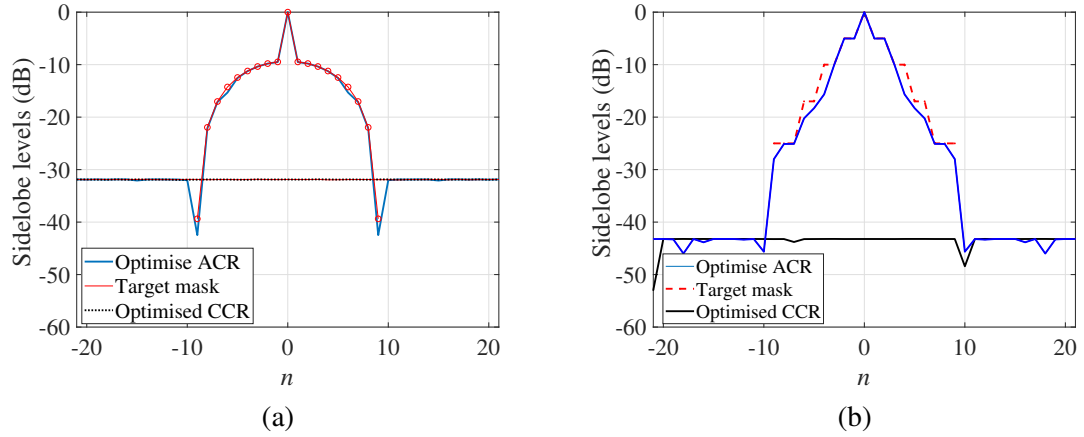
Figure 4.11: Monotonic mainlobe autocorrelation plots for code lengths $n = 100$ plotted against target mask (zoomed-in). (a) First method with target mask averaged over 25 experiments, $d = \pm 10$. (b) Second method with step-like target mask, $d = \pm 10$.

the first stage of a two-stage optimisation. A set of 25 random initialisations were optimised for a code with $m = 2$ with length $n = 100$, and $d = \pm 10$. The resulting autocorrelation values in the mainlobe were saved. The average of the mainlobe sample values from the 25 results was computed, for each offset. A generalised monotonic mask was modelled using a 2nd–order quadratic equation for curve-fitting over the average values of the correlation samples for each offset. This creates a monotonic mask that has a sharp peak at zero–offset and gradually tapers off to each side. The resulting mask was then used as a target mask.

Another set of random initialisations were optimised using the above mentioned mask as a mainlobe mask, and the sidelobe mask is the same as previously mentioned. The algorithm is the same as Algorithm 3. Figure 4.11(a) shows (zoomed-in) plots of the mask along with the resulting autocorrelations. The mainlobe has a sharp peak at zero-offset which spreads out symmetrically towards the sidelobes on either side. The sidelobes are -30.4 dB against the Welch bound [25] of -26 dB, and the mainlobe is monotonic.

The second method uses a target mask with a step-like profile and a different number of steps. We used a mainlobe target mask with three distinct steps. The first step is at -15 dB and last step at -25 dB, each step 3 samples long. With a few exceptions, the mainlobe closely follows the target mask and is monotonic. The resulting SLLs are -26 dB which just manages to meet the Welch bound [25]. Another mainlobe target mask with 4 steps reveals sidelobes of -42.7 dB. Figure 4.11(b) shows (zoomed-in) plots of the mask along with the autocorrelations. This technique has been found very effective for mainlobe shaping and low SLLs. The SLLs depend on the number of steps and the levels of each of the steps of the target mask. In all cases, the mainlobe
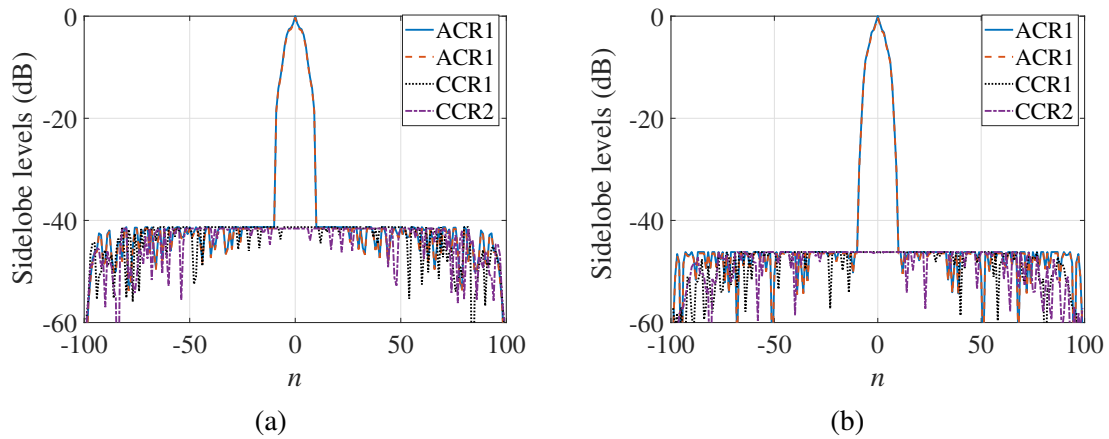
Figure 4.12: Correlation plots for code length $n = 100$ using third method for monotonic mainlobe shaping, mainlobe width $d = \pm\, 10$. (a) Worst SLL -41.3 dB.(b) Best SLL -46.2 dB.

is monotonic with SLLs below the Welch bound [25].

A third method used requires no target mask. The monotonic profile is imposed in the mainlobe by checking the values of two subsequent samples of mainlobe while moving from the sides towards zero-offset. There can be two ways to impose monotonicity; if the second sample has magnitude smaller than the first, we can either push up the second sample or push down the first sample. This ensures that the sample violating the monotonicity is optimised. In either case, a monotonic profile of mainlobe is ensured. Figure 4.12 shows the correlation plots for the best and worst cases when the first approach is used, by pushing up the second sample of the two samples being compared. As expected, the mainlobe is wide, monotonic and all samples in the mainlobe have high magnitude. All sidelobes of autocorrelations and all crosscorrelations are low and flat. The results are very encouraging, however, every initialisation ends up with a local minimum individual to the initialisation. For a set of 25 optimisations starting with random initialisations, optimised SLLs range from -41.3 dB in the worst case to -46.2 dB for the best case.

4.3.4 Peak Sidelobe Level versus Integrated Sidelobe Level

Optimising PSL for applications like sonars ensures a flat and low profile of sidelobes. Reflections from weaker targets have to be higher than a threshold for the purpose of detection, irrespective of the distance of the weaker target from a stronger target. In case of the ISL optimisation method, there is no guarantee that the sidelobes would be uniformly below a certain threshold. There would be regions where the sidelobes would be higher than an average and regions where the

sidelobes would be lower than an average, with a non-uniform, non-flat profile. Reflections from weaker targets in such a scenario could possibly be masked by SLLs caused by a stronger reflection, hence, decreasing the chances of detection.

We now compare our results of optimising the PSL versus optimising the ISL. This is done by studying the correlation sidelobes when a region of offsets is allowed to have SLLs higher than the remaining offsets. We use a mask so that correlation samples in this region are weighted and so are ignored from the optimisation iterations. The autocorrelation value at the zero-offset is maximised and sidelobes at all other offset are minimised as before. The mask from Algorithm 2 is modified so that the region of weighting the autocorrelations is symmetrical about the zero-offset. The region of high sidelobes is termed as High Correlation Zone (HCZ). In the first iteration, the HCZ is centred at the zero-offset, represented by a shift of $s' = 0$. In the next iteration, the HCZ is moved an equal number of samples away on either side of the zero-offset. This done merely by adjusting the location of the weights in the mask. The experiment is repeated until the HCZ reaches the ends of the correlation offsets, and the optimised correlation sidelobe values are noted for each shift.

For test results, the first optimisation is done on a set of two codes with length $n = 100$ and an HCZ of $d = \pm 10$ samples. In the first optimisation, $s' = 0$, and in the next optimisation, the HCZ is shifted by $s' = 10$ samples away from the zero-offset, so that the HCZ now spans correlations offset 81–90 and 111–120. In the next experiment, the same methodology was repeated for a set of two codes each with length $n = 500$, and mainlobe width $d = \pm 10$.

The SLLs are best when the HCZ is around the zero-offset. For the set of two codes with length $n = 100$, this is around -47 dB for $s' = 0$. The SLLs increase in height as the HCZ is moved away from zero-offset towards the sides, reaching -25 dB when the HCZ is at the extreme ends of the correlations. Figure 4.13(a) and (b) show the SLLs versus the shift of HCZ from the zero-offset, for the set of two codes with length $n = 100$, on a linear and logarithmic scale. Similarly, Figure 4.13(c) and (d) shows the same plots for the set of two codes with length $n = 500$.

Results from this experiment support our choice of optimising PSL versus the ISL metric of the SLLs. The lowest SLLs are achieved when the HCZ is around the zero-offset, and the SLLs start getting higher and degraded as the HCZ is moved away from the zero-offset. To ensure an optimal code with minimum mutual interference, the codes should be optimised for the uniform lower SLLs, and if the SLLs are allowed to higher in some region, then the most preferred region is centred at zero-offset, resulting in a wider mainlobe.
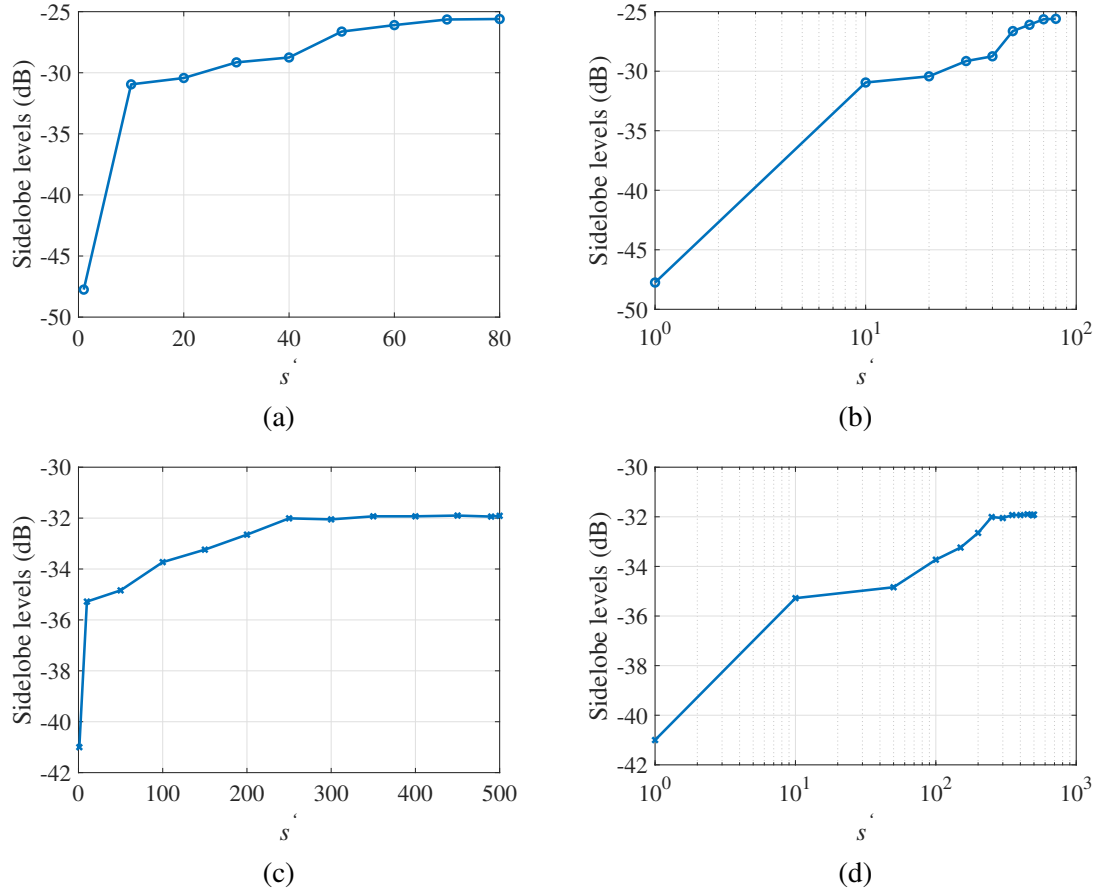
Figure 4.13: Sidelobe levels versus the shift s' of HCZ from zero-offset for set of optimised codes. (a) A set of two codes with length $n = 100$, mainlobe width $d = \pm 10$, and presented on linear scale. (b) A set of two codes with length $n = 100$, mainlobe width $d = \pm 10$, and presented on logarithmic scale. (c) A set of two codes with length $n = 500$, mainlobe width $d = \pm 10$, and presented on linear scale. (d) A set of two codes with length $n = 500$, mainlobe width $d = \pm 10$, and presented on logarithmic scale.

4.4 Summary

This chapter covered details of code optimisation methods and elaborated the gradient-descent method proposed and used in this thesis. Detailed results pertaining to optimising sets of codes for aperiodic correlations, conformance to a target power profile while incorporating the transducer frequency response and the channel response are discussed. It is also shown that while the SLLs are restricted by the Welch bound [25], widening the mainlobe is a viable compromise to achieve SLLs beyond the Welch bound [25]. Corresponding results for optimisation of code sets

with a wider mainlobe and weighted correlation sidelobes are presented. Different methods are suggested to achieve a monotonic wider autocorrelation mainlobe and flat and low sidelobes. It has also been shown that optimising a set of codes for optimal properties is best when the PSL metric is used for uniformly low and flat sidelobes. If the sidelobes are allowed to be higher in one region than the other, the region must be around the zero-offset for lowest SLLs.

Chapter 5

Signal to Noise Ratio and Time Variable Gain in the Underwater Acoustic Channel

There is nothing new to be discovered in physics now,
All that remains is more and more precise measurement.

Lord Kelvin (1900)

The Signal to Noise Ratio (SNR) at a sonar receiver is directly related to the strength of the received signal and the strength of the noise received with the signal. The noise includes the ambient noise of the UWA channel as well as the receiver noise and is often modelled to be white and wide-sense stationary [229].

In an UWA channel, loss in signal strength is high and consequently, the SNR is low due to two major factors. The first factor is attenuation or absorption that increases with the carrier frequency of the signal. The second factor is destructive interference from multipath propagation due to the channel acting as an acoustical waveguide, constrained by the seafloor and the sea-surface. The reflections from these can be of varying strength depending on the distance travelled and the time taken to traverse the distance. The third factor is the spreading loss which may be assumed (for simplicity) to be spherical or cylindrical as already mentioned in Section 2.1.2.

The acoustic wave is also affected by the slow speed of sound in the water. The slow propagation speed of acoustic waves makes it vulnerable to the time-varying characteristics of the UWA channel. The channel coherence time is frequency-dependent and, for any given UWA channel, it varies from a few tens of milliseconds to a few seconds [15–18]. The inevitable time-
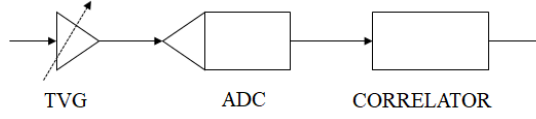
Figure 5.1: A simple TVG model to compensate for transmission loss in a sonar receiver system.

varying nature of the channel is extremely difficult to model. The channel tends to affect the signal, and these effects may be classified as small-scale variations and large-scale variations. [230]. Small-scale variations are caused by displacements of the order of one or a few wavelengths and occur over comparatively short distances. They influence the instantaneous channel response, and consequently, the instantaneous SNR. In contrast, large-scale variations are caused by displacements spanning many wavelengths, and take into account the random variations resulting from environmental effects such as the weather and uncertainty in geometry. They influence the SNR causing it to vary over longer periods of time. These factors and the uncertainties associated with them mean that underwater channel modelling is still open to improvements.

Reflections from a given target vary in strength depending on the distance of the target from the source. The wide range of reflection strength from similar targets at different ranges results in a large dynamic range of the input signal in a sonar system. A Time Variable Gain (TVG) is used to compensate for the transmission loss so that reflections from identical targets at different ranges are presented with the same strength to the digitiser of the receiver [178, 179]. The TVG function also helps reduce the large dynamic range of the received signal thereby reducing the requirements on the Analog to Digital Converter (ADC) of the receiver. We consider a simple TVG model as shown in Figure 5.1. The received signal is subjected to a TVG for compensating the transmission loss before being quantised and sampled by ADC. The ADC converts the received signal, restored in magnitude, into discrete samples, and passes it on to the correlator.

In this chapter, we investigate the SNR improvement for a signal in an UWA channel assuming the noise is white and stationary. We specifically use the optimised sets of codes from the previous chapter for this investigation. We derive the filter that will maximise the SNR of a signal when the noise is white and stationary. We then study the effects of TVG on the signal and the noise and derive the equivalent filter to maximise the SNR of the signal subjected to TVG. We compare the performance of both the filters in terms of SNR and the correlation sidelobes.

5.1   Channel Simulation Models

The UWA channel is an extremely challenging medium for the transfer of information. Channel coherence and the variability of different characteristics of the channel are important in modelling

the response of the UWA channel to the propagating signal. In addition to the physical aspects as already mentioned in the introductory paragraphs of this chapter, UWA communication is affected by the Doppler and delay spread, and ambient noise. These factors vary from channel to channel, and so, the effect of each of these factors on the propagating acoustic wave is different. Each of these factors poses constraints to communications and sensing systems through loss in SNR and ISI.

The channel can be modelled by its impulse response $h(t, \tau)$, which can then be used to design communication and sensing systems. In some literature [231], $h(t, \tau)$ is the response of a system at time t to an impulse occurring at time $t - \tau$, whereas in other literature [232], $h(t, \tau)$ is the response of a system at time t to an impulse occurring at time τ. In this thesis, the latter notation is used to represent channel impulse response.

Signals and modulation techniques can be appropriately chosen according to the channel impulse response for reliable and effective communications. An UWA channel may be modelled as a waveguide. Within this waveguide, the acoustic wave travels with a spatially variant sound speed. The reflections from the surface and the bottom, and the refractions within the water, due to the varying speed of sound result in multipath propagation. These multipaths give rise to delay spread which varies from channel to channel and vary with time. The refractions can also cause the channel impulse response to give rise to arrivals that are non-minimum phase, that is, the first arrival is not the strongest [16]. The rough sea-surface also causes a spreading delay in each of the surface reflections, so that the arrivals from multiple-paths have a widely fluctuating intensity. The bubbles generated by breaking waves, both in the open ocean or near-shore regions, scatter as well as absorb acoustic energy of the propagating acoustic wave. These bubbles may last for a fraction of a second to a few seconds after formation, depending on the channel conditions. Lastly, the ambient noise, mainly due to breaking waves, bubbles, biological sources and weather effects, varies from channel to channel and time to time within the same channel. The channel conditions and its effects on the acoustic wave propagation, therefore, cannot be predicted accurately.

A system designed to work effectively in one environment (for example, a deep water channel) may fail completely in another environment (for example, a shallow water channel) due to varying channel characteristics. A generalised design that is sufficiently flexible and reliable to work effectively across a broad range of environments remains challenging. While the literature suggests that there is no "typical" acoustic channel [16], an UWA channel simulator for empirical modelling has been presented in [230]. Another design of an underwater acoustic channel model has been proposed in [233]. Some experiment-based models have also been derived, see [15] and references therein. These models are applicable to specific locations where the experiments were conducted and so, are not applicable to a general case. It is extremely difficult to generalise a channel impulse response and predict the performance of a signal or a sensing system that

has been designed for some specific purpose. In the absence of a standard channel model, we investigate just one aspect of the performance of optimised codes in an UWA channel in this thesis, by determining the SNR and the spatial correlation of the signal.

5.2   Signal to Noise Ratio

The Signal to Noise Ratio (SNR) is defined as the ratio of the power (P_S) of a received signal *s* to the power (P_V) of the noise *v* in the received signal [94, 234, 235]

$$\text{SNR} = \frac{P_S}{P_V} \tag{5.1}$$

where P is the average power. Both signal and noise power must be measured within the same system bandwidth. If the noise is random with zero mean and variance σ, then the instantaneous SNR becomes

$$\text{SNR} = \frac{s^2}{\sigma_V^2} \tag{5.2}$$

and

$$\text{SNR (dB)} = 10 \log_{10}\left(\frac{P_S}{P_V}\right) = 10 \log_{10}\left(\frac{s^2}{\sigma_V^2}\right) \tag{5.3}$$

A higher SNR, or a high ratio of P_S to P_V makes it easier to detect the desired signal or extract useful information from the received signal in the presence of noise.

5.2.1   SNR in White Stationary Noise

Recall that the matched filter [93–95] is the optimal filter to maximise the peak SNR of a received signal in white stationary noise. The impulse response of the matched filter can be derived in a number of ways, see for example [236, 237].

We review one such derivation of this optimal filter in Appendix C, which gives us the time-domain matched filter as

$$h_{\text{opt}} = k s_1^*(-t) \tag{5.4}$$

This shows that the optimal filter to maximise the SNR of the received signal in white stationary noise $h_{\text{opt}}(t)$ is simply the scaled, time-reversed conjugate of the input signal $s_1(t)$.

A matched filter obtained from (C.15) is the optimal filter. Correlation of the received signal with the time-reversed, complex conjugated, impulse response of the matched filter will result in the maximum output SNR for the signal that was transmitted. For a multicode sonar system,

a matched filter bank is used in the receiver with one matched filter for each transmitted code. Identification of each of the received signal is achieved at the output of the respective matched filter.

5.2.2 SNR in White Non-Stationary Noise

A matched filter serves to maximise the peak SNR in the presence of white and stationary noise. The spectrum of white stationary noise is flat across all frequencies, and all the samples of the noise are uncorrelated. If the noise is still white, but no longer stationary, that is, the spectrum is flat over all range of frequencies but no longer constant, and the samples in the noise are still uncorrelated, then the matched filter ceases to be the optimal filter. We consider circumstances where the noise is still white but no longer stationary and also derive the filter that will be optimal for maximising the SNR for such a case.

We now move to a discrete-time model of the system and derive the expression. We derive an alternate expression for the matched filter that will maximise the output SNR of the received signal in white and stationary noise. We assume that all signals and noise components are vectors. From (C.2), we have

$$\mathbf{y} = \mathbf{s}_o + \mathbf{v}_o = \mathbf{s} * \mathbf{h} + \mathbf{v} * \mathbf{h} = \mathbf{y}_s + \mathbf{y}_v \tag{5.5}$$

where $\mathbf{h}$ is the receiver filter, so that convolving it with either $\mathbf{s}$ or $\mathbf{v}$ implies time-reversed $\mathbf{h}$. We now define the Hermitian covariance matrix $R_{\mathbf{v}}$ of the noise $\mathbf{v}$ [235] as

$$R_{\mathbf{v}} = E\left\{\mathbf{v}\mathbf{v}^H\right\} \tag{5.6}$$

where $\mathbf{v}^H$ denotes the conjugate transpose of the noise vector $\mathbf{v}$ and E denotes the expectation operator. Note here that the covariance matrix of the white, non-stationary noise will still be a diagonal matrix, as the noise samples are uncorrleated. However, the covariances along the diagonal will no longer be the same.

The output SNR_o from (5.5) and (5.6) can then be defined as

$$\text{SNR}_o = \frac{|\mathbf{y}_s|^2}{E\left\{|\mathbf{y}_v|^2\right\}} \tag{5.7}$$

From (5.5), this becomes

$$\text{SNR}_o = \frac{\left|\mathbf{h}^H\mathbf{s}\right|^2}{E\left\{\left|\mathbf{h}^H\mathbf{v}\right|^2\right\}} \tag{5.8}$$

The numerator term in (5.8) can be simplified as

$$\left|\mathbf{h}^{\mathrm{H}}\mathbf{s}\right|^2 = \mathbf{h}^{\mathrm{H}}\tilde{\mathbf{s}}\,\tilde{\mathbf{s}}^{\mathrm{H}}\mathbf{h} \tag{5.9}$$

where $\tilde{\mathbf{s}}$ denotes the vector reversal of $\mathbf{s}$. Similarly, the denominator of (5.8) can be simplified to

$$E\left\{\left|\mathbf{h}^{\mathrm{H}}\mathbf{v}\right|^2\right\} = E\left\{\left(\mathbf{h}^{\mathrm{H}}\mathbf{v}\right)\left(\mathbf{h}^{\mathrm{H}}\mathbf{v}\right)^{\mathrm{H}}\right\} = \mathbf{h}^{\mathrm{H}}E\left\{\mathbf{v}\mathbf{v}^{\mathrm{H}}\right\}\mathbf{h} = \mathbf{h}^{\mathrm{H}}R_{\mathbf{v}}\mathbf{h} \tag{5.10}$$

Substituting (5.9) and (5.10) in (5.8), the SNR becomes

$$\mathrm{SNR_o} = \frac{\mathbf{h}^{\mathrm{H}}\tilde{\mathbf{s}}\,\tilde{\mathbf{s}}^{\mathrm{H}}\mathbf{h}}{\mathbf{h}^{\mathrm{H}}R_{\mathbf{v}}\mathbf{h}} \tag{5.11}$$

Following Therrien pp 241–243 [238], the filter that maximises the SNR will be

$$\mathbf{h} = \frac{1}{\sqrt{\mathbf{s}^H R_{\mathbf{v}}^{-1}\mathbf{s}}}R_{\mathbf{v}}^{-1}\tilde{\mathbf{s}}^* \tag{5.12}$$

where $\tilde{\mathbf{s}}^*$ represents the reversed, complex conjugate of $\mathbf{s}$.

In the case of the white and stationary noise, the covariance matrix $R_{\mathbf{v}}$ is

$$R_{\mathbf{v}} = E\left\{\mathbf{v}\mathbf{v}^{\mathrm{H}}\right\} = \sigma_{\mathbf{v}}^2\mathbf{I} \tag{5.13}$$

where $\sigma_{\mathbf{v}}^2$ is a scalar and denotes the noise variance, and $\mathbf{I}$ is the identity matrix, so the normalised filter is $h = \frac{s^*}{\|s\|}$.

We now assume that the transmitted signal suffers non-constant transmission loss during propagation. The loss is subsequently compensated by applying a similar non-constant gain at the receiver (discussed in more detail in Section 5.3). The signal power is restored, but the non-constant gain affects the white and stationary input noise $\mathbf{v}$. At the output of the gain block, it is still white but no longer stationary. We denote this noise by $\mathbf{w}$.

The effect of the non-constant gain will be more pronounced when the change in the gain is rapid, so that the noise has a large variation. However, if the non-constant gain does not vary much over the duration of the signal and range of propagation, the noise variance may not change much and the conventional matched filter remains a good solution. We consider the case when the matched filter $\mathbf{h}$ from (5.12) will no longer be the optimal filter for maximising the SNR of this signal in the presence of noise $\mathbf{w}$. To ensure optimal performance of the receiver filter, we derive the expression for the receiver filter $\mathbf{g}$ to optimise the SNR of the received signal in presence of $\mathbf{w}$. The covariance matrix $R_{\mathbf{w}}$ for the noise $\mathbf{w}$ is still diagonal but the covariances along the diagonal are no longer the same. The covariances increase proportional to the change in the non-constant gain over the duration of the signal, and the depending on the distance travelled

by the signal. The optimum filter **g** to maximise the SNR can thus, still be derived from (5.12), simply by replacing $R_{\mathbf{v}}^{-1}$ with $R_{\mathbf{w}}^{-1}$ so that

$$\mathbf{g} = \frac{1}{\sqrt{\mathbf{s}^H R_{\mathbf{w}}^{-1} \mathbf{s}}} R_{\mathbf{w}}^{-1} \tilde{\mathbf{s}}^* \qquad (5.14)$$

and the normalised optimal filter is then

$$\mathbf{g} = \frac{R_{\mathbf{w}}^{-1} \tilde{\mathbf{s}}^*}{\left\| R_{\mathbf{w}}^{-1} \tilde{\mathbf{s}}^* \right\|} \qquad (5.15)$$

5.3 Time Variable Gain

Time Variable Gain (TVG) in radars and sonars, is a technique used to compensate for transmission loss and reduce the dynamic range of the received signal. The delay between the transmitted pulse and the received echo increases with the range R considerably due to the slow speed of sound. The amplitude of the echo also decreases with the range R, due to transmission loss.

The concept of TVG is also used by bats and dolphins, two animal species that use acoustic waves and TVG for sensing [181, 239, 240]. In the sonar receiver model assumed in this research and shown in figure 5.1, the TVG is applied to the received echo signal, followed by an (Analog to Digital Converter (ADC)) before being input to the correlator. The use of a TVG has two advantages. First, echoes from the same target at different ranges are received with similar echo strengths. Second, the varying echo levels result in a large dynamic range, often exceeding 200 dB [178] at the ADC of the receiver. Use of TVG reduces this dynamic range for the ADC.

Figure 5.2a shows the one-way and two-way transmission loss L_{T} versus range R for a distance up to 1000 m. A shallow channel is considered for this plot with cylindrical spreading and the TVG is calculated for $\alpha_{\text{water}} = 60$ dB/km. A baseband frequency of 200 kHz has been chosen and speed of sound 1500 m/sec has been assumed. TVG is calculated using

$$\text{TVG} = 20 \log_{10} R + 2 \, \alpha_{\text{water}} R \qquad (5.16)$$

A well defined TVG curve matches the two-way L_{T} curve. At short ranges, the time gradient of the TVG is large. This renders the signal and noise vulnerable to high errors if the gain is not properly applied. It is generally assumed that the TVG function would compensate sufficiently for the transmission loss of the acoustic wave, as can be seen from Figure 5.2a. However, empirical results have proved that the compensation from TVG is accurate only for a very short sounding pulse or at very large range [179]. The errors are prominent at shorter ranges where the slope
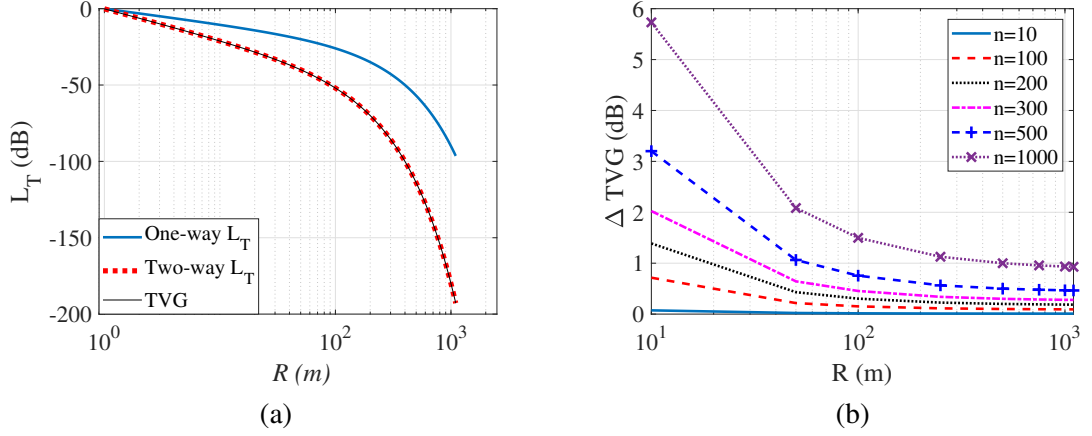
Figure 5.2: TVG and ΔTVG versus plotted for reflections at various ranges R (m) from the transducer. (a) 1-way and 2-way Transmission Loss L_T and TVG versus range. The TVG curve exactly overlaps and compensates for the Two-way L_T. (b) The difference in TVG between the first and last samples of a code for different code lengths against Range R.

of the TVG is high and the gain of the receiver changes sharply within the pulse length. It has been suggested that a more accurate TVG function must be used when targets are closer to the transducer [177, 180]. A detailed discussion on the exact and the approximate TVG and the theoretical $20\log_{10}R$ TVG function is presented in [179] along with the merits and applicability of various types of TVG.

We now study ΔTVG: the difference between the TVG values for the first and the last sample of the code with length $10 \leq n \leq 1000$ over the range $10 \leq R \leq 1000$ m. Assuming, as before, a baseband sampling frequency of 200 kHz and speed of sound 1500 m/sec, these codes vary from 0.075 to 7.5 m in length. Figure 5.2b shows how ΔTVG changes for the codes over the range R. The gradient of ΔTVG is highest for longer codes at shorter ranges and it decreases as either the code length n decreases or the range R increases. This implies that the accuracy of applying the TVG in terms of offset in gain or time is always important, but is more so for longer codes at shorter ranges.

5.4   Experimental Settings

It is anticipated that the designed filter **g** from (5.15) would maximise the SNR for a signal received with the transformed noise **w**. We now investigate the SNR improvement at the output of the receiver due to the filters **h** and **g**. For the purpose of investigating the SNR improvement,

we use a set of two codes optimised in Chapter 4. The optimised codes have length n with $10 \leq n \leq 1000$. As already mentioned, using a baseband sampling frequency of 200 kHz and speed of sound 1500 m/sec, these codes vary from 0.075 to 7.5 m in length. For reasons of simplicity, we assume the noise input level at the input of the TVG is 0 dB. For TVG calculations, we assume cylindrical spreading, with a carrier frequency of 200 kHz and $\alpha_{\text{water}} = 60$ dB/km[1]. We then use this value of α_{water} in (2.1) to obtain the required TVG function over the range $10 \leq R \leq 1000$ m from the transducer.

5.5 Results and Discussion

In the following subsections, we investigate the effect of TVG on the SNR improvement (or PG) at the output of the receiver filter, in two different scenarios. We also investigate the effect of TVG on the correlation sidelobes of the optimised codes and then evaluate the importance of accuracy in application of the TVG to the received signal.

5.5.1 SNR Improvement

We investigate the SNR improvement (or PG) at the output of the receiver for two different scenarios. First, we assume that the received echo signal is subjected to the TVG function, and the matched filter $\mathbf{h}$ is used at the receiver end. The effect of the TVG is to transform the noise $\mathbf{v}$ to $\mathbf{w}$, which is still white but no longer stationary. In the second case, we use the filter $\mathbf{g}$ instead of the filter $\mathbf{h}$ for correlating the received echo signal, after applying the TVG function. The noise, in this case, is also $\mathbf{w}$. The SNR for each case is then calculated using (5.15), for both $\mathbf{h}$ and $\mathbf{g}$. The filter $\mathbf{h}$ is the same for all cases, where the filter $\mathbf{g}$ is calculated for each case separately, due to the difference in the length of the codes as well as the range for which the the signal propagates.

Figure 5.3(a) shows the SNR at the output of the matched filter $\mathbf{h}$ against the length n of the code on a logarithmic scale for reflections at different ranges R from the transducer. The transmitted codes suffer increased transmission loss with an increase in range, and so the SNR decreases with increase in the range. When we calculate the SNR at the output of the receiver having filter $\mathbf{g}$ to match the noise $\mathbf{w}$ in the received signal, the result is similar to Figure 5.3(a), and so has been omitted. Figure 5.3(b) shows an alternate presentation of SNR at the output of a matched filter $\mathbf{h}$, for the noise $\mathbf{w}$, the SNR being presented against the range R for different lengths n of the code.

The difference between the outputs of $\mathbf{h}$ and $\mathbf{g}$ becomes more visible if we remove the effect of the range over the total noise level. We normalise the covariance matrix $R_{\mathbf{w}}$, so that the sum

---

[1]For calculating the α_{water}, we assume temperature of 12 °C, depth of 10 m, salinity of 35 ppm and a pH of 8 to obtain $\alpha_{\text{water}} = 60$ dB/km
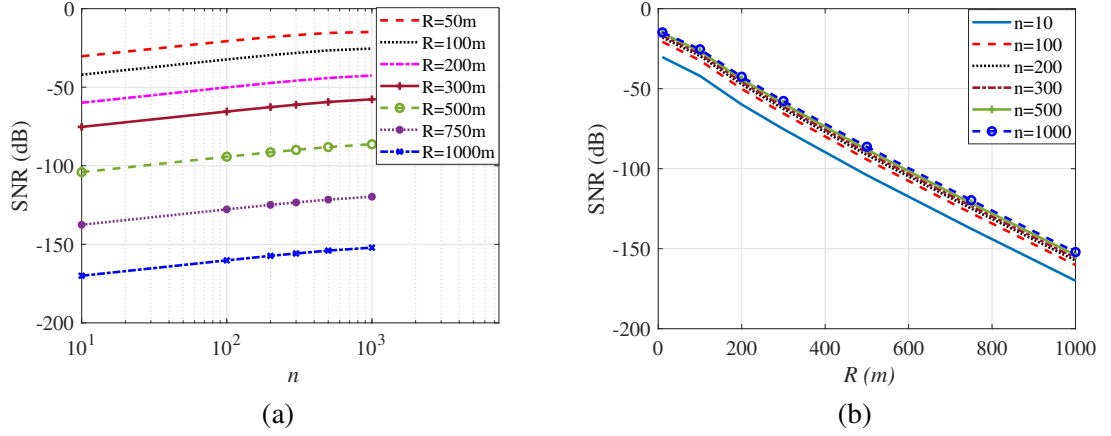
(a)

(b)

Figure 5.3: SNR for different lengths of code n with unit norm, reflected from target at various ranges R. Cylindrical propagation with $\alpha_{\text{water}} = 60$, and TVG matching 2–way transmission loss is assumed. (a) SNR at the output of the matched filter **h** versus length of code n. (b) SNR at the output of the matched filter **h** versus range R.
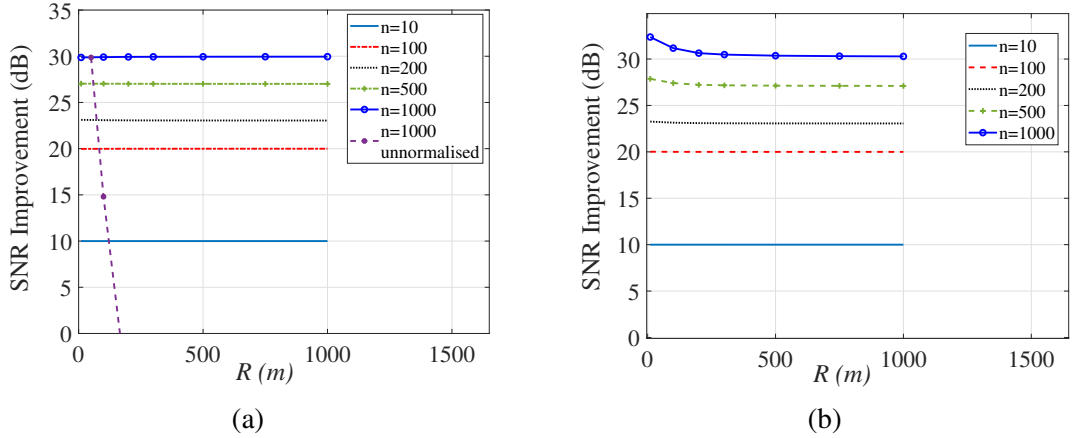

(a)

(b)

Figure 5.4: SNR Improvement against range R for different length of codes due to: (a) Matched filter **h** with non-stationary white noise **v**. (b) Designed filter **g** for non-stationary white noise **w**. Except for the "unnormalised" curve, the effect of range on the absolute noise level has been removed by normalisation.

of the noise energy for any given code is equal to the length of the code n, or $\text{Tr}(R_{\mathbf{w}}) = n$. This ensures that all codes will have the same mean SNR. Figure 5.4 shows the SNR improvement versus the range R for filter **h** and filter **g** when the noise in the received signal is **w**. Figure 5.4(a) shows the SNR improvement level at the output of the receiver due to **h**, and Figure 5.4(b) shows the SNR improvement level at the output of the receiver due to **g**. Comparison of the two plots
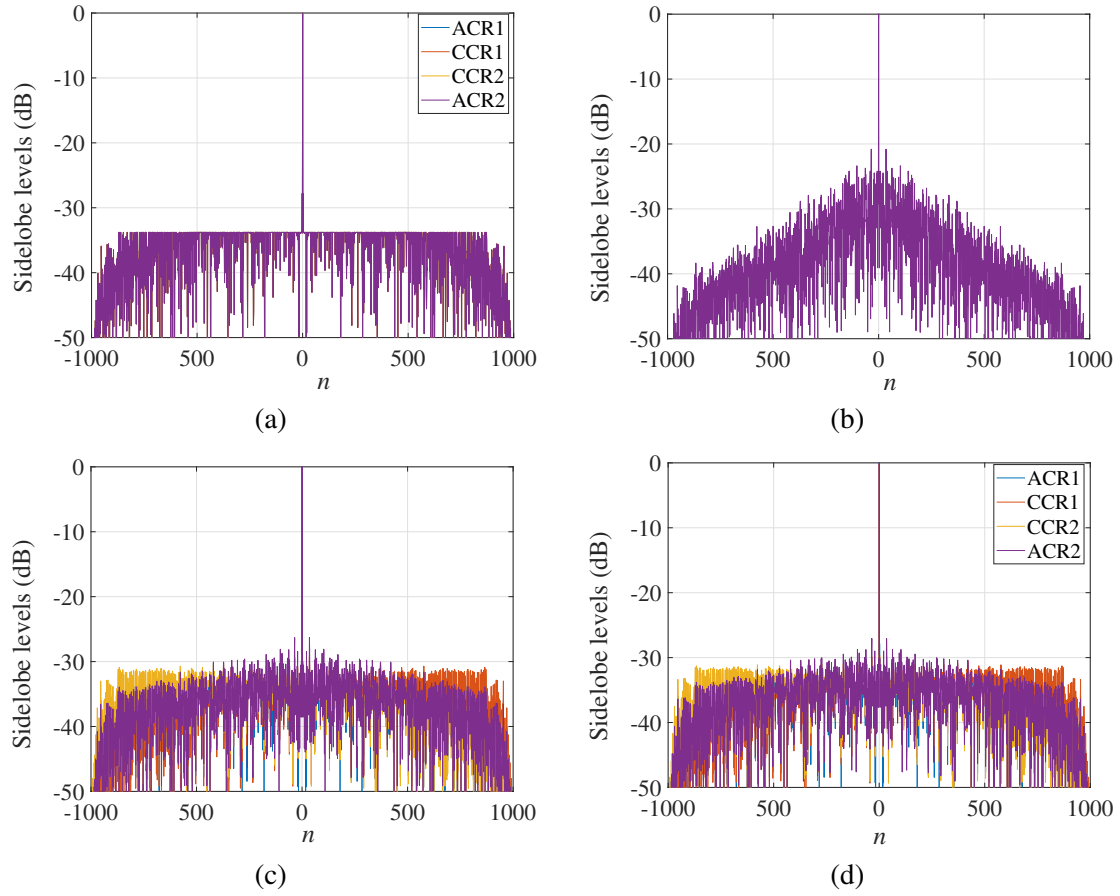
Figure 5.5: Auto- and crosscorrelation of the two received codes with matched filter **h** and filter **g**, when TVG is applied. Correlations are with (a) Matched filter **h** with no noise and echo from 50 m. (b) Filter **g** and echo from 50 m. (c) Filter **g** and echo from at 500 m. (d) Filter **g** and echo from 1000 m.

reveal that there is only a noticeable difference in SNR improvement by using a filter **g** designed to match the non-stationary noise as compared to the matched filter **h**, for codes with length $n > 500$ at ranges $R < 100$ m.

5.5.2 Correlations Sidelobes due to TVG

We investigate the effect of filter **g** on the auto- and crosscorrelations of the optimised codes used in this analysis when TVG is applied. Figure 5.5(a) shows the output due to matched filter **h** without applying the TVG, for a signal echo returned from a range of 50 m, for a set of code with length $n = 1000$ samples, without noise. The sidelobes in the case of filter **h** are low and flat at

-33 dB. Figure 5.5(b) shows the autocorrelation of the filter **g** with an echo returned from a range of 50 m. The sidelobes show considerable deterioration and are no longer low and flat. Figure 5.5(c) shows the output of the filter **g** for an echo returned from a range of 500 m. The sidelobes are slightly flatter but still not comparable with Figure 5.5(a). Figure 5.5(d) shows the output of the filter **g** for an echo signal returned from a range of 1000 m. The sidelobes are similar to the case when the echo was assumed to be received from 500 m.

We had observed from Figure 5.4 that in the presence of non-stationary noise, the designed filter **g** does give a slight advantage in terms of SNR improvement, for longer codes and at shorter ranges only. However, results from Figure 5.5 reveal that the SNR improvement is at the cost of considerable deterioration of SLLs. The matched filter **h** is still a good approximation for the received signal in the presence of both stationary and non-stationary noise.

5.5.3   TVG Accuracy for Correlation Sidelobes

Lastly, we investigate the accuracy of applying the TVG in terms of offset in time, and its implications on the auto- and crosscorrelations of the received code with both filters **h** and filter **g**. We do this by applying a slight offset in range R when applying the TVG to the set of codes. This results in a mismatch in the application of the TVG. Figure 5.6(a) shows the auto- and crosscorrelation of the received signal with the filter **h**. A slight offset of 2 % of TVG in the range R at a range of $R = 50$ m is applied to the received echo. The deterioration in SLLs is prominent. Figure 5.6(b) shows the auto- and crosscorrelations of the received code with the designed filter **g** at the same range with the same TVG mismatch. The sidelobes are similar. Figure 5.6(c) shows the auto- and crosscorrelations of the received code with the filter **h** when a TVG with inaccuracy of 5 % at a range $R = 1000$ m is applied. Due to the increase in range and the reduced value of ΔTVG, the sidelobes are slightly better than Figure 5.6(a) and (b) but still, neither flat nor low. Figure 5.6(d) show the auto- and crosscorrelations of the received code with the filter **g** when a TVG with the inaccuracy of 5 % at a range $R = 1000$ m is applied.

5.5.4   Discussion

The findings from our experiments lead us to conclude that in an UWA channel, the matched filter **h** is a good approximation for the optimal filter either in stationary or non-stationary noise. The filter designed to match the non-stationarity of the noise shows a very small SNR improvement for longer codes at very short ranges. This small improvement in SNR comes at the compromise of the SLLs being considerably degraded. The accuracy of the TVG is critical for both filter **h** and filter **g**. The SLLs are also considerably degraded even if there is a slight mismatch in the
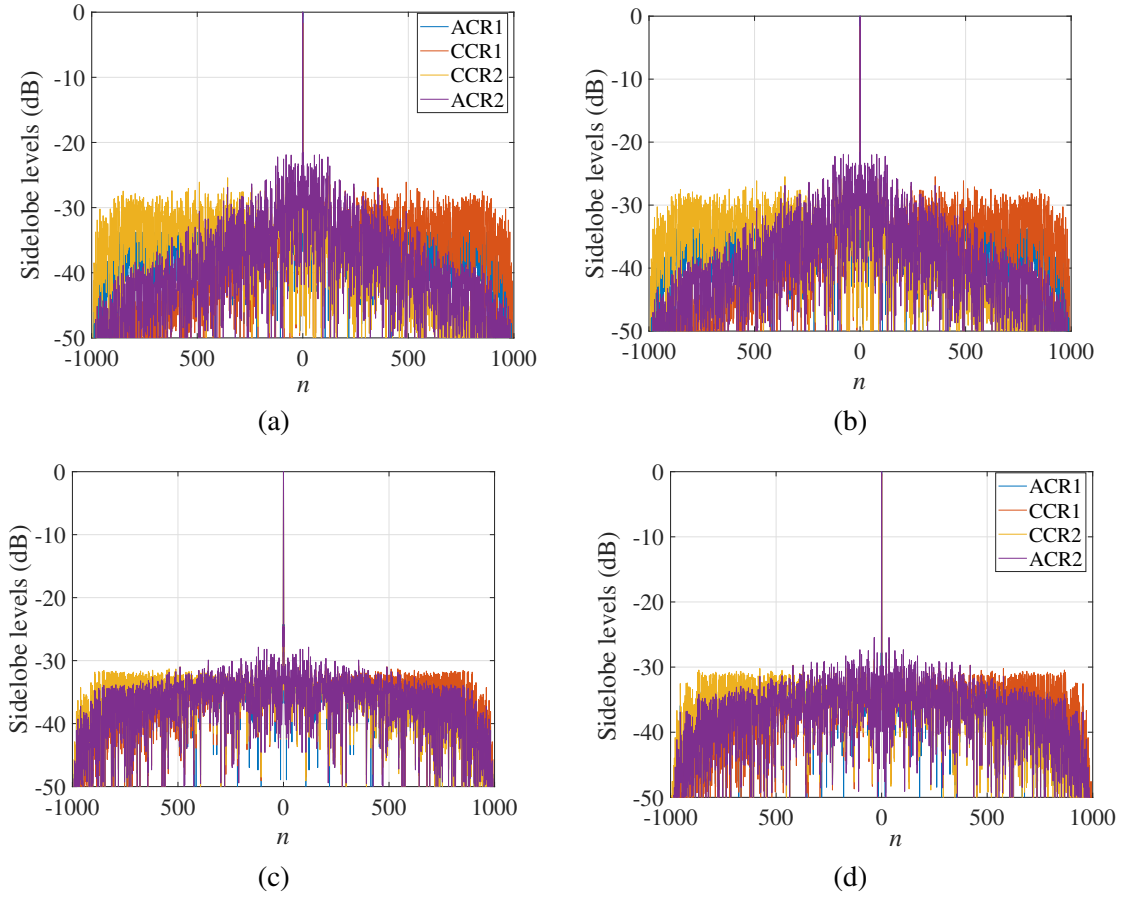
Figure 5.6: Correlations of the two received codes with filters **h** and **g** when TVG is applied with a slight mismatch. (a) Matched filter **h** with TVG mismatch of 2 % at $R = 50$ m. (b) Designed filter **g** with TVG mismatch of 2 % at $R = 50$ m. (c) Matched filter **h** with TVG mismatch of 5 % at $R = 1000$ m. (d) Designed filter **g** with TVG mismatch of 5 % at $R = 1000$ m.

accuracy of the TVG being applied to the received echo. The deterioration in the SLLs is highest for codes at shorter ranges. At longer ranges, the deterioration is present but not very significant. The findings are consistent with the TVG model as shown in Figure 5.1 only, and the optimised codes from Chapter 4. However, it could be interesting to see the results for other models of TVG and other codes that may be used in sonars.

5.6 Summary

In this chapter, we have investigated the Signal to Noise Ratio (SNR) for codes in an UWA channel. We have defined the SNR and the corresponding expression for the matched filter that

will optimise the SNR of the received signal, in the presence of white, stationary noise. We have then introduced the Time Variable Gain (TVG) to compensate for the transmission loss that an acoustic signal undergoes in an UWA channel. We then studied the effects of the TVG on the received signal, and on the noise of the received signal, and derived an expression for the filter that will optimise the SNR of the received signal in the TVG scenario. We then investigated the theoretical SNR improvement at the output of the receiver due to the matched filter and the filter designed to maximise the SNR in the TVG affected noise. We also studied how the filter designed to maximise the SNR of received signal in non-stationary noise affects the correlation properties of the optimised codes.

We have shown that in most cases, modifying the design of the matched filter to account for non-stationarity of the noise caused by the TVG is not warranted. The classical matched filter in itself is a good approximation regardless of whether the noise is stationary or otherwise. The slight advantage of the designed filter is at the cost of higher and deteriorated SLLs.

We have also found that accuracy in applying the TVG is very important, as a slight mismatch in timing in terms of the range R can prove detrimental. This is true for both the matched filter $\mathbf{h}$ and the designed filter $\mathbf{g}$.

Chapter 6

Code Optimisation for Doppler Tolerance

The pessimist sees difficulty in every opportunity.
The optimist sees the opportunity in every difficulty.
Winston Churchill

Underwater acoustic sensing (i.e., sonar) requires probing signals to have a narrow autocorrelation peak with minimal autocorrelation SLLs and minimal crosscorrelations. Another requirement is that the signals should be robust to Doppler effects. Signal reflections from moving targets are subjected to the Doppler effect [41, 53]. The Doppler effect causes the signal to be frequency shifted down or up and to either stretch or shrink, depending on whether the target is moving away from the source or towards it, respectively.

The Doppler effect [41, 53] induced on the signals due to movement of the target or the source or both causes mismatch between the actual received signal and the expected received signal. At the correlator, this results in higher SLLs as well as a decrease in the SNR (or PG) and mainlobe magnitude. Doppler tolerance enhances the robustness of the codes to this movement. The design of codes with good autocorrelation, crosscorrelation and Doppler tolerance properties make the optimisation problem more difficult.

For a sonar scenario, movement of the boat, or moving fish motivate consideration of the Doppler effect. In this chapter, the transformation of the signal from the time it is transmitted to the time it is received back incorporating the shrinking or dilation of code length due to the Doppler effect is modelled. This model is then used to optimise the codes. The aim is to maximise the autocorrelation mainlobe peak resulting from matched filter output while reducing
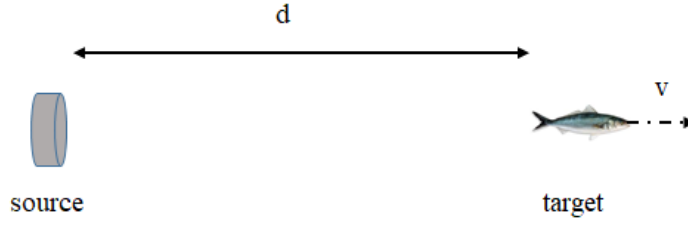
Figure 6.1: A simple model to explain the Doppler effect due to static source and moving target.

the sidelobes of autocorrelation and all crosscorrelations. The optimisation results are compared with logarithmic chirps which are known to be Doppler tolerant.

Previous methods quoted in the state of the art of Doppler tolerant code optimisation in Chapter 3 fall short of addressing all three requirements, that is, to have good auto- and crosscorrelations, a good PAPR and Doppler tolerance, simultaneously. Some methods are limited in the length of the optimal sets of codes, require the use of an LCZ for optimising correlations within a given range of offsets only, or are tolerant to a fixed number of predictable Doppler offsets. Some of the algorithms target optimisation of periodic correlations instead of the aperiodic correlations necessary for sonar systems.

In this chapter, we extend our previously proposed algorithm of Chapter 4 to optimise sets of codes so that they have flat and low auto- and crosscorrelation SLLs across the entire range of offsets, with a good PAPR (≈ 1) and for a range of Doppler scaling factors. We do not model the frequency shift due to the Doppler shift but the resultant shrinking and dilation in the length of the codes in the time domain. We also optimise sets of codes so that they are robust to delays which are a non-integer multiple of the symbol period.

6.1   Doppler Model for a Typical Sonar System

Radar or sonar signals are bandpass signals which are affected by the movement of the transmitter or the target or both. The first effect is the change in carrier frequency. The second effect is a corresponding variation in the time domain, as a time-scale dilation or contraction. This causes the reflected signal to have a length different from that of the transmitted signal. These effects cause a mismatch in the receiver, resulting in higher correlation sidelobes, as well as a reduction in and shift of the autocorrelation mainlobe peak.

To model the Doppler effect [41] for a sonar system, we start with a simple case, assuming the source (acting both as a transmitter and receiver) is stationary and only the target is moving with velocity v (which is positive if the target is moving away from the source, and vice versa) as

shown in Figure 6.1. We assume the target is at a distance R from the source. The transmitted pulse starts at time $t = 0$ and propagates in the UWA channel towards the target at velocity c. Let the pulse duration be δt. Let the time at which the start of the pulse reaches the moving target be t_1, so that

$$t_1 = \frac{R}{c} \tag{6.1}$$

The end of the pulse reaches the moving target at the time

$$t_2 = t_1 + \Delta t \tag{6.2}$$

The target has travelled a distance of $R_2 = v\Delta t$ as compared to when the start of the pulse hit at t_1. Here, it must be understood that Δt is different from δt due to the movement of the target. We have from (6.1) and (6.2)

$$t_2 = \frac{R + v\Delta t}{c} + \delta t = \frac{R}{c} + \delta t \tag{6.3}$$

which simplifies to

$$\Delta t = \frac{c}{c - v}\delta t \tag{6.4}$$

The echo from the start of the pulse reaches back the source at time t_3. The distance does not change during the time the echo travels back to the source as the source is assumed to be stationary, and remains R, so that

$$t_3 = \frac{R}{c} + t_1 \tag{6.5}$$

The echo from the end of the pulse returns to the source at time t_4. The distance of the target to the source remains unchanged as the source is not moving. So, the distance to travel is the same as $R + R_2 = R + v \times \Delta t$, so that

$$t_4 = \frac{R + v\Delta t}{c} + t_2 \tag{6.6}$$

The length of the echo received from the target is $\delta t' = t_4 - t_3$ which is

$$\delta t' = \frac{v \times \Delta t}{c} + t_2 - \left(\frac{R}{c} + t_1\right) = \frac{R + v\Delta t}{c} + t_2 - t_1 \tag{6.7}$$

n	Doppler dilation or contraction g		
$v =$	0.7 m/s	1.4 m/s	2.4 m/s
500	0.467	0.934	1.603
1000	0.934	1.868	3.205
5000	4.669	9.342	16.026

Table 6.1: Doppler scaled dilation or contraction g (in samples) for sets of code with length n and target speeds v.

From (6.1) and (6.3), we get

$$\delta t' = \frac{R + v\Delta t}{c} + t_2 - \left(\frac{R}{c} + t_1\right) = \Delta t \left(\frac{v}{c} + 1\right) \tag{6.8}$$

Substituting (6.4) into (6.8)

$$\delta t' = \frac{c}{c - v}\left(\frac{v}{c} + 1\right)\delta t = \left(1 + \frac{2v}{c - v}\right)\delta t \tag{6.9}$$

we get the Doppler Scaling factor as

$$\eta = \left(1 + \frac{2v}{c - v}\right) \tag{6.10}$$

Movement of the target away from the source will result in positive v, and so the signal would be dilated; movement of the target towards the source will cause a negative v, resulting in the signal being contracted. The dilation is slightly different if the source is moving as well as the target, but if $v \ll c$, only the relative speed is significant.

For a sonar system, fish are often the prospective targets. Fish are known to have varying swimming speeds. Dab, butter-fish and lemonsole are considered slow whereas mackerel and sea-trout are fast [241]. The swimming speeds of these fish vary from $0.2 - 2.4$ m/s [241]. For the purpose of brevity, we have chosen three swimming speeds. The value of Doppler scaling factor η is calculated for the three swimming speeds using the speed of sound $c = 1500$ m/s. The actual addition or subtraction g in number of samples to the length of the code n can then be calculated using the Doppler scaling factor η and the length of the code n by $g = (\eta - 1)n$.

Table 6.1 shows the Doppler dilation or contraction g in number of samples against sets of codes with lengths n for different speed of fish v. For any fish with a given velocity, movement *towards* the transmitter will cause the code to contract by g samples, movement *away* from the transmitter will cause the code to dilate by the same number of samples.

The Doppler model for multicode optimisation is developed as follows. Let the two codes of a code-set be represented by two complex vectors **a** and **b**, each of length n. Before transmitting, the two codes are up-sampled, mixed with a carrier at a higher frequency, and bandpass-filtered. It is assumed that a prospective target is moving with a certain speed v and the transmitted code is reflected by the target and undergoes a Doppler shift. At the receiver end, the Doppler shifted reflected signal from the target is received, mixed down to baseband, lowpass filtered, down-sampled and lowpass filtered again before being passed to the receiver filter. In the receiver filter, the Doppler shifted reflected code is then correlated with the copy of the baseband transmitted code.

For the Doppler model, we denote all the above mentioned transformations by a single matrix **T**. The **T** matrix is the linear shift varying operator, that simulates the scaling of the transmitted code of length n to a Doppler shifted received code of length $\hat{n}$, where $\hat{n} > n$ if the target is moving away from the source and $\hat{n} < n$ if the target is moving towards the source. The number of rows of the matrix **T** is equal to the length of the Doppler shifted codes $\hat{n}$, and the number of columns of the T matrix is equal to length of the original transmitted codes n. This **T** matrix incorporates a particular carrier frequency, as well as cut-offs of the lowpass and the bandpass filters.

The received signal deteriorates under channel effects and the relative movement of the target causes Doppler shift, which is seen as either a dilation or contraction in the length of codes in the time domain. The transmitted code **a** and **b** of length n are received as code **a**$'$ and **b**$'$ of length n where, $\hat{n} = n \pm g$, so that

$$\mathbf{a}' = \mathbf{T}\mathbf{a} \tag{6.11}$$

$$\mathbf{b}' = \mathbf{T}\mathbf{b} \tag{6.12}$$

where the linear shift varying operator **T** denotes all the transformation processes described above.

The Doppler shift in the time domain is introduced by the ratio of the up-sampling to down-sampling and will be referred to as Doppler scaling from here on in this thesis. We have optimised sets of codes for multiple Doppler scaling factors by using matrices $\mathbf{T}^l$ for $l = \{1, 2, ..., L\}$ where L is the number of Doppler scaling factors.

6.2 Doppler Tolerant Code Optimisation

In the receiver, there are four possible outputs due to the transmitted codes **a** and **b**, received as Doppler scaled codes **a′** and **b′**. These are the autocorrelations $C_{\mathbf{aa'}}$ and $C_{\mathbf{bb'}}$ and the crosscorrelations $C_{\mathbf{ab'}}$ and $C_{\mathbf{ba'}}$, defined as

$$
\mathbf{C_{aa'}}(j) = \sum_{i=1}^{\hat{n}-1} \mathrm{a}_i (\mathbf{a}'_{i-j})^* = \sum_{i=1}^{\hat{n}-1} \mathrm{a}_i (\mathbf{T}_{i-j,:}\mathbf{a})^*
$$

$$
= \sum_{i=1}^{\hat{n}-1} \mathrm{a}_i \mathbf{a}^* \left(\mathbf{T}_{i-j,:} \right)^* \tag{6.13}
$$

$$
\mathbf{C_{bb'}}(j) = \sum_{i=1}^{\hat{n}-1} \mathrm{b}_i \mathbf{b}^* \left(\mathbf{T}_{i-j,:} \right)^* \tag{6.14}
$$

$$
\mathbf{C_{ab'}}(j) = \sum_{i=1}^{\hat{n}-1} \mathrm{a}_i \mathbf{b}^* \left(\mathbf{T}_{i-j,:} \right)^* \tag{6.15}
$$

$$
\mathbf{C_{ba'}}(j) = \sum_{i=1}^{\hat{n}-1} \mathrm{b}_i \mathbf{a}^* \left(\mathbf{T}_{i-j,:} \right)^* \tag{6.16}
$$

where $\mathbf{T}_{i,:}$ is the i-th row of the transformation matrix of (6.11) and (6.12).

We extend our gradient-descent based optimisation algorithm proposed in Chapter 4 to include the Doppler scaling model as described in (6.13) – (6.16). We modify the optimisation algorithm to maximise the autocorrelation values of $\mathbf{C_{aa'}}$ and $\mathbf{C_{bb'}}$ at zero offset, while minimising these at all other offsets. Similarly, the algorithm minimises the crosscorrelation values of $\mathbf{C_{ab'}}$ and $\mathbf{C_{ba'}}$ at all offsets. Detailed calculations of derivatives for $\mathbf{C_{aa'}}$, $\mathbf{C_{bb'}}$, $\mathbf{C_{ab'}}$ and $\mathbf{C_{ba'}}$ are shown in Appendix D. The algorithm can define multiple matrices $\mathbf{T}^l$ for multiple Doppler scaling factors, that is, $l = 1, 2, 3, ...$ The iterations in the algorithm are then made to alternate between these matrices $\mathbf{T}^l$ so that the set of codes are optimised for multiple Doppler scaling factors. The algorithm is shown in Algorithm 4. The iterations are repeated until a local minimum is found.

The zero-offset of all autocorrelations $\mathbf{C_{aa'}}$ and $\mathbf{C_{bb'}}$ is expected to be at index $(\hat{n} + n)/2$, but is observed to be shifted from this. This is due to the length of the Doppler affected codes **a′** and **b′** not being exactly equal to $\hat{n} = n + g$, but longer than this. The filtering processes after up-sampling and down-sampling as well as mixing increase the length of the signal from $\hat{n} = n + g$. The length of vector **a′** and **b′** is longer than $\hat{n}$ and now equal to the number of rows of **T**.

1: *initialise set of codes (**a** and **b**) from pseudo-random sequences;*

2: *choose learning rates μ_1 and μ_2;*

3: *calculate $\left\{\mathbf{T}^1, \mathbf{T}^2, ..., \mathbf{T}^l\right\}$ matrices for*
   the Doppler scaled lengths of interest

4: *define PAPR profile*

5: *define filter for transmitter, receiver and channel effects;*

6: *convolve **a** and **b** with the channel filter;*

7: **for** iter=1 **to** *K* **do**

8:     *find samples with max and min magnitude deviation from profile*

9:     *multiply max magnitude sample by 1 - μ_2*

10:     *multiply min magnitude sample by 1 + μ_2*

11:     *normalise set of codes to unit norm*

12:     *calculate **a**′ and **b**′ using (6.11) and (6.12)*

13:     *rotate between $\mathbf{T}^l$ in subsequent iterations*

14:     *calculate aperiodic correlations $\mathbf{C_{aa′}}$, $\mathbf{C_{bb′}}$, $\mathbf{C_{ab′}}$, $\mathbf{C_{ba′}}$ due to respective $\mathbf{T}^l$*

15:     *ignoring the $\mathbf{C_{aa′}}$ and $\mathbf{C_{bb′}}$ at zero offset, find offset k having maximum correlation value*
   and identify whether it belongs to autocorrelation or crosscorrelation;

16:     *compute derivative of maximum correlation value wrt **a** and / or **b***

17:     *subtract μ_1 scaled derivative from **a** and / or **b***

18:     *check stopping criteria, maximum iterations or difference between successive maximum*
   SLLs less than some threshold;

19: **end for**

Algorithm 4: Pseudo-code for Doppler tolerant code optimisation.

Figure 6.2 shows the shift of the zero offset for different Doppler scaled lengths $\{-5 \leq g \leq 5\}$ for a code of length $n = 500$. For a Doppler scaling of $g = 0$, the zero-offset is at index 523 instead of 500. For every Doppler scaling of $g = \pm 1$ samples, the mainlobe peak is observed to be offset by ± 0.5 so that for some values of g, the zero offset spans two samples. The figure also suggests that there are values of g for which the zero offset or the mainlobe peak is not at an integer value. The shift in the zero-offset and the mainlobe peak not being constrained to a single sample is not associated with the Doppler effect, but because of the change in length of the Doppler scaled codes. This is something that was not considered until now. This fact reasserts our motivation to perhaps, increases the mainlobe width of the autocorrelations when optimising codes for Doppler tolerance.
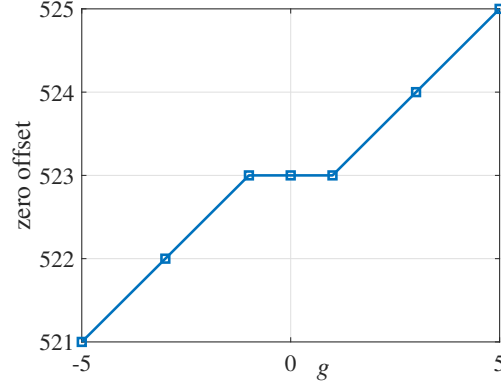
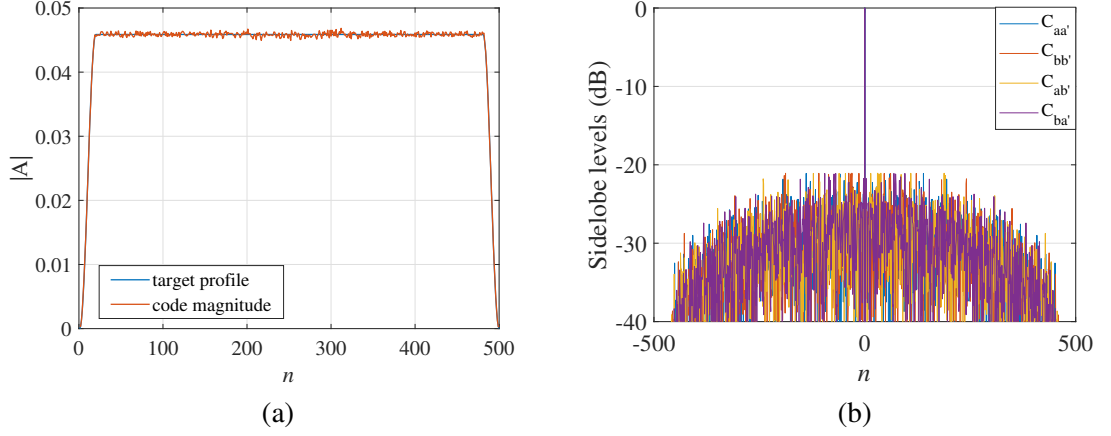Figure 6.2: Shift in zero offset of correlations for different lengths of Doppler scaled sets of codes.

Figure 6.3: set of codes with two codes optimised for good correlation properties as well as tolerant to Doppler scaling of $g = 1$ sample. (a) Plot of the optimised set of codes magnitude against power profile with PAPR ≈ 1. (b) Auto- and crosscorrelations of the optimised code with a Doppler stretched version.

6.3   Results and Discussion

We initialise a set of two codes **a** and **b** each of length $n = 500$, with random complex values. We select a Doppler scaling factor $g = \pm 1$. We then implement Algorithm 4 for the given values of n and $\hat{n}$.

Figure 6.3(a) shows the magnitude of the optimised set of codes showing a PAPR ≈ 1 while matching the desired power profile. Figure 6.3(b) shows the auto- and crosscorrelations of the optimised set of codes, when correlated with Doppler shifted versions of the codes by $\hat{n} = 501$
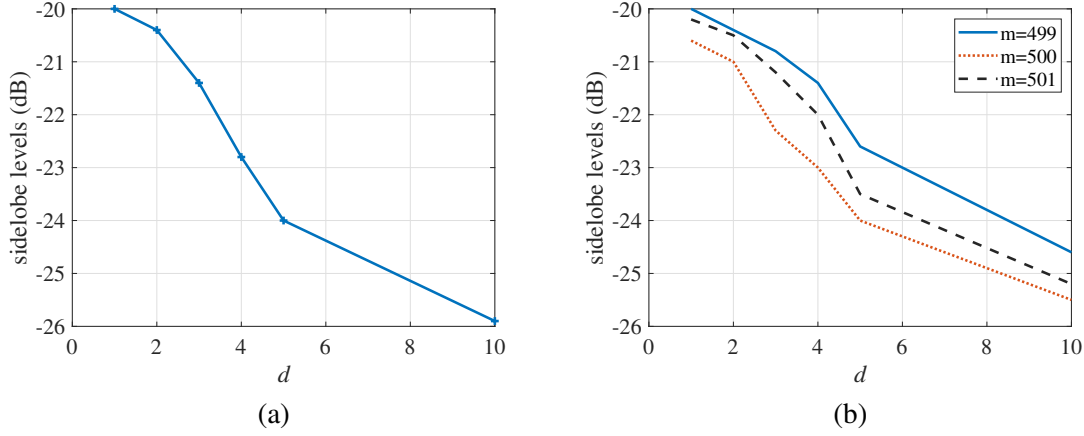
Figure 6.4: SLLs plotted against mainlobe width for optimised set of codes with length $n = 500$, Doppler scaling factor $g = 1$ and mainlobe width $\{1 \leq d \leq 10\}$. (a) set of codes optimised and correlated with Doppler scaled version. Increasing mainlobe width results in lower SLLs. (b) set of codes optimised for multiple non-integer delay in arrival $499.5 \leq \hat{n} \leq 500.5$, and evaluated at Doppler scaled lengths $\hat{n} = n \pm g$, i.e. $\hat{n} = 499$ and $\hat{n} = 501$.

samples. The maximum SLLs are at -20.8 dB.

The shift in the mainlobe peak as observed in Figure 6.2 at non-integer values for some values of g implies that the mainlobe in Algorithm 4 should not be constrained to a single point. We optimise codes **a** and **b** but increase the mainlobe widths for $\mathbf{C_{aa'}}$ and $\mathbf{C_{bb'}}$, with $1 \leq d \leq 10$. This widens the mainlobe across the zero-offset symmetrically by d samples on each side of zero-offset. As the mainlobe width increases, lower SLLs are achieved. Figure 6.4(a) shows the SLLs versus the mainlobe width for the set of codes when optimised for Doppler scaling $g = 1$. For a mainlobe width of $d = 10$, SLLs as low as -25.9 dB can be achieved with these optimised codes.

The transmission delay in a sensing system is not guaranteed to be an integer number of sampling periods and the delay is also unpredictable. This implies that each code must also be optimised for multiple non-integer delays in time of arrival. It is desirable for the designed code to be robust against this unknown. To achieve this, we generate multiple Doppler affected codes, by defining multiple T matrices, denoted by $\mathbf{T}^l$, where $l = 1, 2, 3, \ldots$. Algorithm 4 is made to alternate between multiple $\mathbf{T}^l$ matrices created during the process of down-sampling for different non-integer timing offsets. This implies generating multiple T^l matrices for non-integer values of g in the interval $-0.5 \leq g \leq 0.5$. Figure 6.4(b) shows the correlations SLLs of the optimised set of code when optimised for multiple (eleven) non-integer delays in the interval $499.5 \leq \hat{n} \leq 500.5$. The mainlobe width is $1 \leq d \leq 10$ and the correlations of the optimised set of codes are evaluated
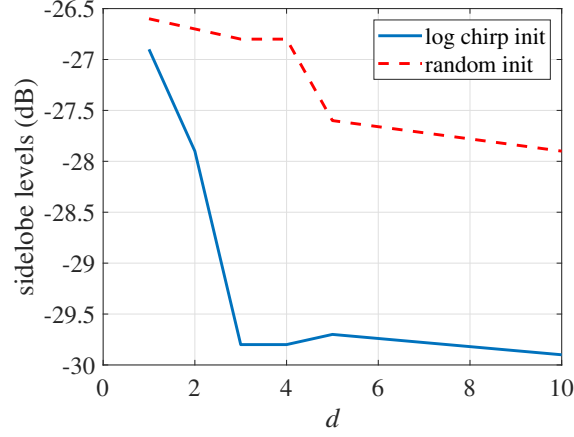
Figure 6.5: SLLs for code optimised from random and Log chirp initialisations plotted against mainlobe width d. Codes have lengths $n = 500$ and optimised for Doppler tolerance to scaled length $\hat{n} = 501$.

at $\hat{n} = 499, 500$ and 501. The best SLLs are obtained for a case of $\hat{n} = 500$ for any mainlobe width and the lowest SLLs are -25.7 dB for $d = \pm 10$. For the same mainlobe width, the lowest SLLs are -25.3 dB for $\hat{n} = 499$ and -24.6 dB for $\hat{n} = 501$. As expected, the SLLs are generally higher when correlations are evaluated at Doppler scaled lengths which were not chosen for optimisation compared to lengths which are chosen for optimisation.

(a)

(b)
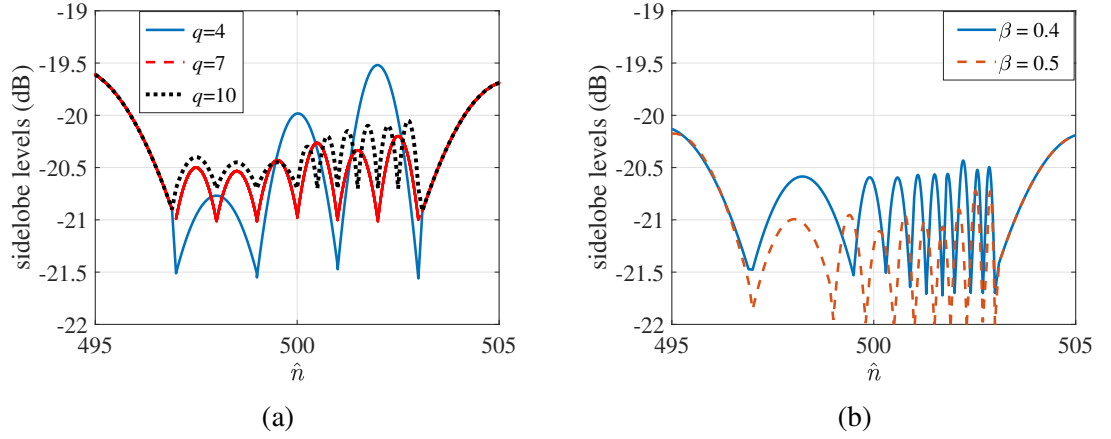
Figure 6.6: SLLs for optimised set of two codes of length $n = 500$. Codes have been optimised for Doppler tolerance for different lengths of dilation and contraction in the interval $-495 \leq \hat{n} \leq 505$. (a) Doppler scaled lengths k linearly chosen within the given interval. (b) 10–scaled lengths chosen with $\beta = 0.4$. and $\beta = 0.5$ according to Table 6.2.

Nonlinear chirps, following the quadratic or logarithmic laws, are known to be tolerant to ranging inaccuracies due to Doppler's effect [45, 242], with lower loss in the magnitude of autocorrelation peak as compared to other codes. Their SLLs are also not affected when the code is Doppler scaled. However, their autocorrelation is a sinc function with first peak sidelobe as high as -13 dB unless an amplitude shaping is applied. Using a set of codes based on logarithmic chirps, with a combination of an up-chirp and a down chirp, can be an option to try to achieve orthogonality of the individual codes. The crosscorrelation sidelobe levels can be reduced by increasing the time-bandwidth product of the two codes. If both chirp-codes have a constant time duration, increasing the bandwidth can reduce the sidelobe levels. [243] gives a detailed account of investigations based on this idea. However, the sonar system in such a case would be limited to using only two codes at the most, restricting the area coverage and so increasing the time of surveys. Moreover, increasing the bandwidth of the system to achieve lower SLLs is another cost associated with this.

In order to exploit the Doppler tolerance of the logarithmic chirps, a set of two codes comprising of an up-chirp with frequency sweep $10 - 200$ kHz and a down-chirp with frequency sweep $200 - 10$ kHz was used as the initialisation of the optimisation. Although our optimisation algorithm allows for large numbers of codes, the results are restricted to two codes for consistency and comparison of results. Interestingly, optimised sets of codes resulting from Logarithmic chirps as initialisations exhibit not only better correlation characteristics but also better Doppler tolerance than the original Logarithmic chirps that were used as initialisation. Figure 6.5 shows a comparison of the SLLs for optimised sets of codes when initialised with random initialisation and when initialised with logarithmic chirps. The logarithmic chirp initialisation offers lower SLLs with an improvement of 2 dB over random initialisations for wider mainlobe widths. Codes initialised with logarithmic chirps retain their Doppler tolerance properties and the optimisation algorithm is able to improve the SLL properties of the set of codes.

For a rapidly moving target with speed of 5 m/s, a set of codes of length $n = 500$ may be dilated or shrinked by as many $g = 3$ samples. For this set of codes when optimised for multiple Doppler scaled lengths in the interval $497 \geq \hat{n} \geq 503$, the best SLLs are found at the lengths chosen for optimisation as shown in Figure 6.6(a). When the same code is evaluated for correlations at any other Doppler scaled length, the SLLs are higher. However, the SLLs are generally lower for Doppler scaled lengths $\hat{n} < 500$ when compared to SLLs at lengths $\hat{n} > 500$. Increasing the number of lengths to be chosen for optimisation within the interval results in smaller variations of the absolute values of SLLs. This can be seen from the plot of Figure 6.6(a).

Results from Figure 6.6(a) show that the Doppler tolerance property of the optimised set of codes exhibits certain characteristics of continuity around the range of lengths for which they are optimised. The SLLs are slightly higher at lengths other than those for which the code has been

optimised, but does not vary discontinuously. This motivates the following method for choosing optimisation points.

For optimal performance of the codes as regards to SLLs, we need to minimise the worst SLLs at lengths other than the ones chosen for optimisation. This can be done by choosing Doppler scaling lengths for optimisation within a given interval such that these lengths are non-linearly distributed across the interval. The Doppler scaled lengths chosen for optimisation must be such that they are widely spaced in the interval $\hat{n} \leq n$ but thinly spaced in the interval $\hat{n} > n$. This helps the optimiser put the bias of optimisation towards the higher SLLs in the region $\hat{n} > n$ as compared to slightly lower SLLs in the region $\hat{n} < n$.

We use a simple formula to choose Doppler scaled lengths that meet the above-mentioned criteria. First, we choose the number of intermediate points q within the interval $n \pm g$. The spacing between the intermediate points on the normalised scale is calculated as follows

$$l = \frac{1}{q-1} \tag{6.17}$$

and the normalised scaling factor $\mathbf{x}$ is calculated as

$$\mathbf{x} = (0 : l : 1)^{\beta} \tag{6.18}$$

where $\beta < 1$.

Then, we use $\mathbf{x}$, n and g to find $\mathbf{y}$, the code lengths for optimisation

$$\mathbf{y} = (n - g) + (2g\mathbf{x}) \tag{6.19}$$

The vector $\mathbf{y}$ gives us the points or scaling lengths to be chosen for optimisation so that the code is robust to Doppler scaled lengths in the given interval, with the minimum difference between the best and worst SLLs when evaluated at any length within the interval. The value of β controls the spacing of the chosen points across the range $n \pm g$ so that there are fewer samples with lengths less than n and more samples with lengths greater than n. Table 6.2 shows how the value of β controls the selection of the $l = 10$ optimisation lengths within the range of $\hat{n} = n \pm g$ for any code with length $n = 500$ and Doppler scaling length $g = 3$.

Figure 6.6(a) shows the SLLs when the chosen lengths were selected to be linearly spaced in the given interval. As the number of lengths chosen for optimisation increases, the difference between the best and worst SLLs decreases. The difference between the best and worst SLLs is almost 1.8 dB for $q = 10$. Figure 6.6(b) shows the SLLs when the bias of selected lengths is more towards $\hat{n} > n$. The lowest SLLs are at -21.9dB and the difference between the best and worst SLLs is as low as 1.2 dB for $q = 10$ and $\beta = 0.5$.

	$\hat{n}$			
$\beta =$	0.4	0.5	0.6	0.7
	497	497	497	497
	499.5	499	498.6	498.3
	500.3	499.8	499.4	499.1
	500.9	500.5	500.1	499.8
	501.3	501	500.7	500.4
	501.7	501.5	501.2	501
	502	501.9	501.7	501.5
	502.4	502.3	502.2	502
	502.7	502.7	502.6	502.5
	503	503	503	503

Table 6.2: Non-linear lengths according to different values of β for multiple Doppler scaled optimisation lengths in the interval $497 \leq \hat{n} \leq 503$ and $q = 10$.

To evaluate the performance of the optimised sets of codes at various Doppler scaled lengths, we use the Ambiguity Function (AF) and plot the ambiguity diagrams for the codes. Figure 6.7 shows the ambiguity diagrams [244, 245] for multiple codes, with length $n = 500$ and correlations evaluated for a range of Doppler scaled lengths $497 \leq \hat{n} \leq 503$. The diagrams have been zoomed-in to the sidelobes around the zero-offset. Figure 6.7(a) shows the ambiguity diagrams for a set of codes optimised from random initialisations. The code has been optimised only for desired correlation properties and a good PAPR. When correlated with Doppler scaled codes, the SLLs are quite high and not flat at all. Figure 6.7(b) shows the ambiguity diagrams for a set of codes optimised from random initialisations using the Algorithm 4 and correlated with multiple Doppler scaled versions of the code. The mainlobe width has been kept at $d = 1$. The SLLs are still not low and flat. Figure 6.7(c) shows the ambiguity diagrams for a random initialisation optimised for various constraints as above, but with the mainlobe width increased to $d = \pm 10$. Figure 6.7(d) is for an logarithmic chirp. The sinc-like autocorrelation is prominently visible. Figure 6.7(e) is for a single-logarithmic chirp code using a Dolph-Chebyshev window in the time domain. Figure 6.7(f) is for a set of codes optimised using an logarithmic chirp as initialisation with same objectives as for 6.7(c). It appears that Figure 6.7(c) and (f) provide the best set of codes which are robust to Doppler scaling. Figure 6.7(e) also provides some very low SLLs, the lowest being -35 dB, but at the cost of mainlobe width or range resolution.

6.4   Summary

In this chapter, we have extended our code optimisation algorithm to include robustness of the optimised codes to Doppler shift. A simple Doppler model is presented to formulate the code optimisation problem for Doppler effect as a change in length of the received signals. We have presented results for sets of codes optimised for a number of Doppler scaled lengths, as well as for multiple non-integer delays. We have also shown that our algorithm is able to improve on the SLLs of logarithmic chirps while retaining their integral Doppler robustness. We have also suggested a way to reduce the difference between the best and worst SLLs when optimising set of codes for a number of Doppler scaled lengths, and evaluating their correlation at other Doppler scaled lengths. Lastly, we have presented the performance of the different sets of codes optimised using the Doppler model presented in this chapter plotting their ambiguity functions (AFs).
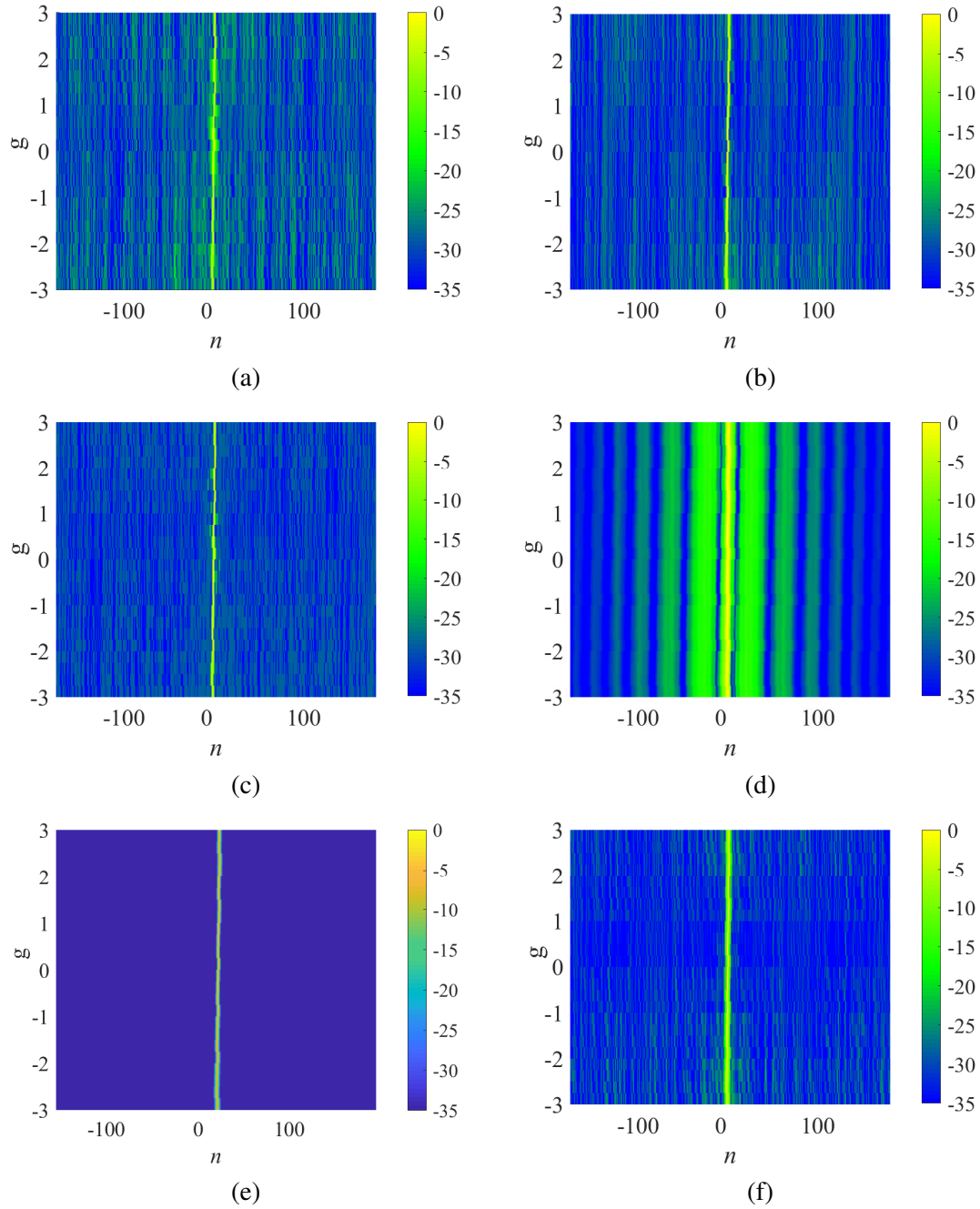
Figure 6.7: AF plots of optimised codes with $n = 500$ versus Doppler scaled lengths $-3 \leq g \leq 3$. AF plots are of: (a) Codes optimised for correlation and a good PAPR, with mainlobe width $d = \pm 1$. (b) Codes optimised for correlations, a good PAPR and Doppler scaling with mainlobe width $d = \pm 1$. (c) Codes optimised for correlations, a good PAPR and Doppler scaling, with autocorrelation mainlobe $d = \pm 10$. (d) Correlation of a logarithmic chirp with no window shaping. (e) Correlation of a logarithmic code with Dolph-Chebyshev window. (f) Codes optimised from log chirp initialisations, for optimal correlations, a good PAPR and Doppler scaling with autocorrelation mainlobe width $d = \pm 10$.

Chapter 7

Mismatched Filter Design for Doppler Tolerant Sets of Codes

Difficulty is the excuse history never accepts.

Edward R. Murrow

A matched filter is usually the preferred sonar signal pre-processor because it is known to maximise SNR. However, the sidelobes resulting from matched filtering are limited by the Welch bound [25]. The high autocorrelation sidelobes may mask weaker reflections from targets in the vicinity of stronger reflections, as well as induce false alarms [246]. In a multicode sonar system, higher crosscorrelation between codes of the set of codes means increased interference of the signals and reduced ability of the system to differentiate between two or more signals. To achieve SLLs lower than the matched filter, a possible solution lies in using an optimisation method to design a mismatched filter. Mismatched filters, no longer being constrained to match the incoming signal, are known to provide optimal Peak to Sidelobe Ratio (PSLR) [247].

From the results of the previous chapter, it is evident that there is a compromise between the Doppler tolerance of the codes and the correlation SLLs that can be achieved for any given optimised set of codes. Use of a mismatched filter to reduce SLLs beyond a matched filter is established and mismatched filters may permit improved detection of received signal in the presence of Doppler shift. Design of mismatched filters for radars and sonars has been open research as already covered earlier in Section 3.1, with designs, suggested as early as 1971 [246].

Some of the advantages of the mismatched filter arise from the fact that the length of the mismatched filter can be longer than the code, hence providing greater flexibility in terms of the coefficients that can be tuned. The filter is longer in length and has coefficients different from the transmitted (and subsequently received) code. Hence, the name mismatched filter. The length of

the mismatched filter directly affects the SLLs. The achievable SLLs reduce with increase in the length of the filter, up to three times the code length. Thereafter, there is no further reduction. The improvement in SLLs is, however, at the cost of a non-negligible loss in autocorrelation peak, or LPG, and hence a loss in SNR. The design of the mismatched filter for minimising the peak sidelobe levels (PSL) though, is relatively complicated as compared to minimising the integrated sidelobes (ISL) [248, 249].

In this chapter, we apply mismatched filter designs to the Doppler tolerant codes from the previous chapter. Unlike the original code design problem, we find that the mismatched filter design is a convex problem. We use two different methods for the design. The first method involves designing the mismatched filter as a convex optimisation problem, and using the CVX Toolbox [250–252]. The second method we propose is to optimise the problem using a gradient-descent method. We propose a novel algorithm along the lines of the previously proposed Algorithms 1 to 4. We present the results from both methods along with the pros and cons of each method. We then carry out a comparison of the methods before concluding the chapter.

7.1   Mismatched Filter Design

The design of an optimal mismatched filter for any application involves a trade-off between different factors, the most important being an acceptable compromise between the loss in processing gain versus the gain in SLLs.

Some designs proposed in the literature [201, 249] have suggested mismatched filters to suppress higher SLLs due to the Doppler effect. These methods attempt to compensate for the degradation in the correlations due to the Doppler effect, but may not be effective for scenarios where the Doppler shift is neither predictable nor fixed. In this thesis, we made our set of codes tolerant to a wide range of Doppler spreads, as well as to multiple non-symbol period delays. We now suggest designing mismatched filters for the optimised sets of codes, hence, promising a wider range in terms of flexibility of application and robustness of the set of optimised codes. Our suggested method also promises fewer ambiguities arising from the Doppler effect, as our codes are already tolerant to a wider range of Doppler shifts. In the following subsections, we suggest designing the mismatched filters for optimised codes from Chapter 6 in two different ways.

7.1.1   Convex Optimisation Method

Convex optimisation is the study of methods to find the global minimum of problems which are convex, that is minimisations of convex functions over convex sets [250]. Convex problems contain no local minima; equivalently, any local minimum of a convex problem is also the global

minimum. Many such problems can be efficiently solved by using interior-point methods [251]. Some examples and properties [250, 253] of convex functions and convex sets are listed below:

   a. The Euclidean norm is convex.

   b. The absolute value of a linear function is convex.

   c. The square of a non-negative convex function is convex.

   d. The point-wise maximum of convex functions is convex.

   e. A non-negative weighted convex function is convex.

   f. An upper bound on a convex function yields a convex set.

Much more on convex optimisation can be found in [250].

   Convex optimisation tools have been used to design mismatched filters for multiple input, multiple output radar applications [253]. In this thesis, we first model the design of a mismatched filter for the Doppler tolerant optimised codes of this thesis, as a QCQP, originally suggested for radar [215, 254] though the approach is equally applicable to sonar. Based on the above-mentioned properties, the problem is convex in nature so that it can be easily solved by various convex optimisation methods. An advantage of this method is that one can add additional constraints of defining an upper limit to the optimised SLLs, a limit of the loss of PG, the mainlobe shaping, and the sidelobe shaping.

   The design of the mismatched filter is implemented as follows. We consider two codes **a** and **b** each of length n and corresponding mismatched filters **c** and **d** of length f such that $f > n$. For a matched filter, $f = n$ and $\mathbf{a} = \mathbf{c}^*$, with $\mathbf{a}^* = [a_1, a_2, ..., a_n]^H$, where $.^H$ denotes the Hermitian transposition. At the receiver end, the output due to the received code **a** and the mismatched filter **c** will be

$$\mathbf{C_{ac}} = \Gamma_f(\mathbf{a})\,\mathbf{c} \tag{7.1}$$

where $\Gamma_f(\mathbf{a})$ is the convolution matrix of size $(n + f - 1) \times f$ associated with code **a** and defined by

$$
\Gamma_f(\mathbf{a}) =
\begin{bmatrix}
a_n & 0 & 0 & \cdots & \cdots & \cdots & 0 \\
a_{n-1} & a_n & 0 & \cdots & \cdots & \cdots & 0 \\
\vdots & a_{n-1} & 0 & \cdots & \cdots & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_1 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\
0 & \cdots & \cdots & \cdots & \cdots & \cdots & a_n \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & \cdots & \cdots & \cdots & a_1
\end{bmatrix}
\tag{7.2}
$$

To maximise the mainlobe at zero-offset and minimise all sidelobes, the model for solving the optimisation problem is

$$
\min_{\mathbf{c}} \quad \|\mathbf{F}\mathbf{C_{ac}}\|_{\infty} \\
\text{s.t. } \mathbf{a}^H\mathbf{c} = \mathbf{a}^H\mathbf{a}
\tag{7.3}
$$

where the term $\|\mathbf{x}\|_{\infty} = \max_i |x_i|$ represents the infinity norm of any vector $\mathbf{x}$, and $\mathbf{F}$ is a diagonal matrix with 1's along the diagonal except for a 0 at index $m+n$ for the zero-offset. This ignores the value of $\mathbf{C_{ac}}$ at the zero-offset (the centre of the main lobe) while minimising sidelobes at all other offsets. The constraint $\mathbf{a}^H\mathbf{c} = \mathbf{a}^H\mathbf{a}$ ensures discarding the trivial solution of $\mathbf{c} = 0$. This problem in its present formulation is a non-convex problem, and hence cannot be solved for an exact solution or a global minimum.

To convert this non-convex problem formulation to a convex problem, we denote by v_i the $i-$th row of the matrix $\Gamma_f(a)$. The problem is expressed as a Quadratically Constrained Quadratic Program (QCQP) [250, 255–257]

$$
\min_c \quad e \\
\text{s.t. } \mathbf{a}^H\mathbf{c} = \mathbf{a}^H\mathbf{a}, \\
\mathbf{c}^H v_{m+n+i}^H v_{m+n+i} \mathbf{c} \leq t \quad \text{for} \quad |i| \geq 1
\tag{7.4}
$$

where e is the highest SLL. Each of the constraints in (7.4) $\mathbf{c}^H v_{m+n+i}^H v_{m+n+i} \mathbf{c} \leq t$ ensures an upper limit t to all SLLs at an offset other than the zero-offset, so that sidelobes are optimised to a value less than t. The optimisation problem then is to maximise the autocorrelation mainlobe of $\mathbf{C_{ac}}$ and $\mathbf{C_{bd}}$ at the zero-offset, while minimising the autocorrelation sidelobes at all other offsets, and minimise all the values of cross-correlations $\mathbf{C_{ad}}$ and $\mathbf{C_{bc}}$.

The only limitation of this method is that it is restricted to shorter code lengths. Codes with lengths $n > 200$ result in the creation of very large correlation matrices, which become memory constrained.

7.1.2 Gradient-Descent Method

It was discussed in Chapter 4 that local optimisation methods may be resorted to solve optimisation problems when it is not feasible to use a global optimisation method. Due to the obvious shortfalls of the QCQP for being memory constrained for sets of codes with longer lengths, we resort to an alternate method of designing the mismatched filter. We propose a gradient-descent method, similar to the previously proposed Algorithms 1 to 4 in this thesis. The algorithm designs the mismatched filter by updating the coefficients of the filter in every iteration so as to minimise the SLLs resulting from correlations of the received Doppler tolerant signal and the mismatched filter.

Even though this method does not make explicit use of the fact that the problem is convex, we know from the QCQP formulation that the problem is convex, and hence the minimum on which the algorithm converges must be the global minimum.

We start with two optimised codes $\mathbf{a}$ and $\mathbf{b}$, of length n that are Doppler tolerant (from Chapter 6), having a wider monotonic mainlobe with low autocorrelation sidelobes and low crosscorrelations. We then take two randomly initialised filters $\mathbf{c}$ and $\mathbf{d}$ of length f, where $f > n$ for each of the codes $\mathbf{a}$ and $\mathbf{b}$, respectively.

For $n = 4$, we have codes $\mathbf{a}$ and $\mathbf{b}$:

$$[\mathbf{ab}] = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ a_4 & b_4 \end{bmatrix} \tag{7.5}$$

Assuming $f > n$, we suppose $f = 6$, so that we have mismatched filters $\mathbf{c}$ and $\mathbf{d}$:

$$[\mathbf{cd}] = \begin{bmatrix} c_1 & d_1 \\ c_2 & d_2 \\ c_3 & d_3 \\ c_4 & d_4 \\ c_5 & d_5 \\ c_6 & d_6 \end{bmatrix} \tag{7.6}$$

Before computing the correlations, we make the size of code set $[\mathbf{ab}]$ equal to the size of filters $[\mathbf{cd}]$, by adding zeros at the start and the end of the code samples, so that the set of codes

[**ab**] is

$$[\mathbf{ab}] = \begin{bmatrix} a_0 & b_0 \\ a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ a_4 & b_4 \\ a_5 & b_5 \end{bmatrix} \tag{7.7}$$

where $a_0 = b_0 = a_5 = b_5 = 0$.

The initial steps of the algorithm are similar to the previous algorithms proposed in this thesis. We calculate the multiple T^l matrices to create the desired Doppler effect. We then define a mask to weight the mainlobe of the autocorrelations so that the mainlobe characteristics of the Doppler tolerant codes are preserved. In each iteration, we first Doppler scale the codes **a** and **b** to calculate codes **a**′ and **b**′ from (6.11) and (6.12). We then compute the correlations of codes **a**′ and **b**′ with the mismatched filters **c** and **d**. Let all possible correlations be denoted by $\mathbf{C_{a'c}}$, $\mathbf{C_{b'c}}$, $\mathbf{C_{a'd}}$, and $\mathbf{C_{b'd}}$. The length of each of the above mentioned correlations is $2f - 1$, with offset j varying over $\{-f + 1 \le j \le f - 1\}$.

At the receiver end, we want to maximise the autocorrelation mainlobe of $\mathbf{C_{a'c}}$ and $\mathbf{C_{b'd}}$, while minimising all sidelobes. We also want to minimise all components of $\mathbf{C_{a'd}}$ and $\mathbf{C_{b'c}}$. We multiply the autocorrelations $\mathbf{C_{a'c}}$ and $\mathbf{C_{b'd}}$ with the mask to preserve the mainlobe width of the autocorrelations of the codes, and ensure they are exempted from the optimisation. We then find the highest sidelobe from the remaining auto- and cross correlations, and its offset j. We then compute the derivative of this highest value sidelobe with respect to the mismatched filter coefficients. The derivatives are calculated below, detailed calculations of derivatives are attached as Appendix E to this thesis.

For all offsets of correlations $j \ge 0$,

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C_{a'c}}(j)|^2 = \begin{bmatrix} a'_{j+1} \\ \vdots \\ a'_n \\ j \text{ zeros} \end{bmatrix} \mathbf{C}^*_{a'c}(j) \tag{7.8}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C_{b'd}}(j)|^2 = \begin{bmatrix} b'_{j+1} \\ \vdots \\ b'_n \\ j \text{ zeros} \end{bmatrix} \mathbf{C}^*_{b'd}(j) \tag{7.9}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{a'd}(j)|^2 = \begin{bmatrix} a'_{j+1} \\ \vdots \\ a'_n \\ j \text{ zeros} \end{bmatrix} \mathbf{C}^*_{a'd}(j) \tag{7.10}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{b'c}(j)|^2 = \begin{bmatrix} b'_{j+1} \\ \vdots \\ b'_n \\ j \text{ zeros} \end{bmatrix} \mathbf{C}^*_{b'c}(j) \tag{7.11}$$

Similarly, for all correlations offsets $j < 0$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{a'c}(j)|^2 = \begin{bmatrix} |j| \text{ zeros} \\ a'_1 \\ a'_2 \\ \vdots \\ a'_{n+j} \end{bmatrix} \mathbf{C}^*_{a'c}(j) \tag{7.12}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{b'd}(j)|^2 = \begin{bmatrix} |j| \text{ zeros} \\ b'_1 \\ b'_2 \\ \vdots \\ b'_{n+j} \end{bmatrix} \mathbf{C}^*_{b'd}(j) \tag{7.13}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{a'd}(j)|^2 = \begin{bmatrix} |j| \text{ zeros} \\ a'_1 \\ a'_2 \\ \vdots \\ a'_{n+j} \end{bmatrix} \mathbf{C}^*_{a'd}(j) \tag{7.14}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{b'c}(j)|^2 = \begin{bmatrix} |j| \text{ zeros} \\ b'_1 \\ b'_2 \\ \vdots \\ b'_{n+j} \end{bmatrix} \mathbf{C}^*_{b'c}(j) \tag{7.15}$$

In the iterations, the derivatives are scaled by learning rate μ_2 and subtracted from the mismatched filter coefficients. The updated mismatched filter is then subjected to the optimisation

1: *Load Doppler tolerant set of codes (**a** and **b** of length n) from memory;*
2: *Define mask for preserving autocorrelation mainlobe width, mainlobe symmetric to the optimised codes;*
3: *choose learning rates μ_2 ;*
4: *initialise mismatched filters (**c** and **d**) from pseudo-random sequences, with length $f > n$;*
5: *define filter for transmitter, receiver and channel effects;*
6: *calculate $\left\{ \mathbf{T}^1, \mathbf{T}^2, ..., \mathbf{T}^l \right\}$ matrices for the Doppler scaled lengths of interest*
7: *convolve **a** and **b** with the channel filter;*
8: **for** iter=1 **to** *K* **do**
9:   *normalise set of codes to unit norm*
10:   *calculate $\mathbf{a}'$ and $\mathbf{b}'$ using (6.11) and (6.12)*
11:   *rotate between $\mathbf{T}^l$ in subsequent iterations*
12:   *calculate aperiodic correlations $\mathbf{C_{a'c}}$, $\mathbf{C_{b'd'}}$, $\mathbf{C_{a'c}}$, $\mathbf{C_{b'c}}$ due to respective $\mathbf{T}^l$*
13:   *multiply the autocorrelations $\mathbf{C_{a'c}}$ and $\mathbf{C_{b'd}}$ with the mask*
14:   *Find maximum correlation value and the location of offset j for this maximum value , identify whether it belongs to or $\mathbf{C_{a'c}}$, $\mathbf{C_{b'd}}$, crosscorrelation $\mathbf{C_{a'd}}$, $\mathbf{C_{b'c}}$,;*
15:   *compute derivative of maximum correlation value wrt **c** and / or **d***
16:   *subtract μ_2 scaled derivative from **c** and / or **d***
17:   *check stopping criteria, maximum iterations or difference between successive maximum SLLs less than some threshold;*
18: **end for**

Algorithm 5: Pseudo-code for design of mismatched filters for Doppler tolerant codes.

iteration again. In most cases one million iterations are used, and the algorithm converges on the global minimum within these iterations. The algorithm implementing this optimisation is shown in Algorithm 5.

7.2   Results and Discussion

7.2.1   Results from Convex Optimisation Method

The optimisation problem was first implemented using CVX Toolbox and MATLAB 2018a. Using an optimised set of two codes each of length $n = 100$ from Chapter 6, we designed mismatched filters of different lengths $f > n$ using the algorithm. We retained the Doppler tolerant properties of the code and Doppler shifted the codes **a** and **b** by generating the T^l matrices. We then defined the convolution matrices $\Gamma_f(\mathbf{a}')$ and $\Gamma_f(\mathbf{b}')$, where $\mathbf{a}'$ and $\mathbf{b}'$ are Doppler shifted versions of code **a** and **b**, and given by (6.11) and (6.12). The mismatched filters are then designed for the Doppler shifted codes as already outlined in Section 7.1.1.

The results from the CVX model are shown in Figure 7.1. For the purpose of clarity, only the autocorrelations are shown, but the crosscorrelation SLLs are the same as the autocorrelation SLLs.

We first define the permissible loss in PG (or loss in autocorrelation mainlobe peak) as -1 dB. For a mismatched filter of length $f = 100$ (the same as a matched filter), the LPG being constrained to -1 dB, the sidelobes are at -23 dB, and the SLLs are consistently low and flat.

When the mismatched filter length is increased to $f = 150$, the sidelobes are at -24.4 dB. For a mismatched filter of length $f = 200$, the sidelobes are at -26.7 dB. For mismatched filters length of $f = 300$ and $f = 350$, the sidelobes are consistent at -29.4 dB. As expected, the sidelobes do not change for a mismatched filter with length greater than $f = 300$, that is, three times the length of the code. Similar results are obtained when the LPG is constrained to -1.5 dB, and results are shown in Figure 7.1(b). Figures 7.1(c) and (d) show the results for a set of two codes with length $n = 200$, length of mismatched filters f as shown in the legend and a loss in PG of -1 dB and -1.5 dB, respectively. Lower SLLs are achieved at the cost of an increased length of the mismatched filter or a greater loss in processing gain. The increase due to the length is restricted to a limiting value for a mismatched filter of length $f = 3n$ ($f = 3m$ in case of Doppler shifted codes). Increasing the filter length f beyond this does not present any further lowering of PSL.

This method is very effective for short code lengths, that is, for $n < 200$. The sidelobes of the optimised set of Doppler tolerant codes from Chapter 6 when correlated with the designed mismatched filter, are flat and low, as was intended when the PSL metric was chosen for the optimisation in Chapter 4.

The CVX Toolbox of MATLAB is an effective and efficient way to optimise convex problems. An important advantage of this approach is the ease with which a maximum permissible loss of processing gain can be specified. However, this QCQP method has two limitations. First, it is only effective for relatively short codes and can become memory-constrained for longer lengths of codes due to the size of the convolution matrix Γ_f. Second, our optimised codes already have a wider mainlobe and with low sidelobes and we do not benefit by adding additional constraints of mainlobe and sidelobe shaping in the optimisation problem. This limitation suggests finding an alternate, economical solution to the problem.

7.2.2 Results from Gradient-Descent Method

We implement the gradient-descent algorithm outlined in Section 7.1.2 on sets of Doppler tolerant optimised codes from Chapter 6. Figure 7.2 shows the correlations for a set of two codes with lengths $n = 100, 200, 500, 1000$ and mismatched filters of lengths f as shown in the legend. The SLLs are flat and low throughout the range of offsets.
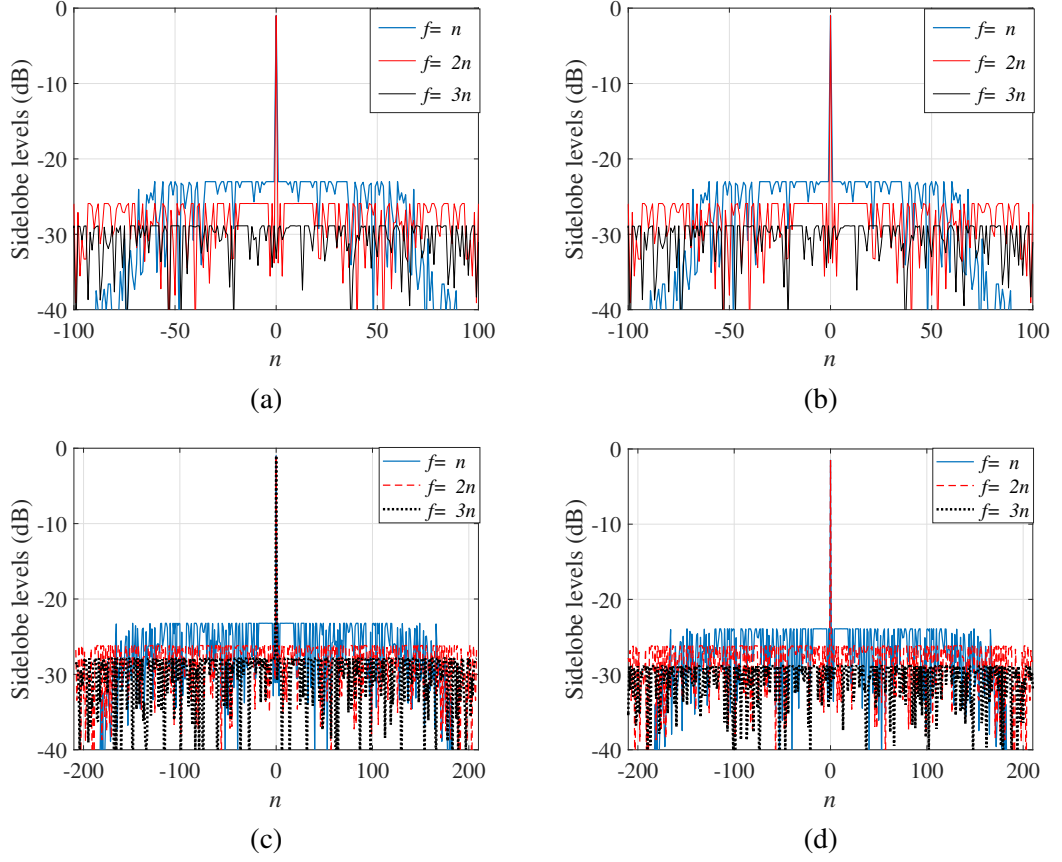
Figure 7.1: Autocorrelations for Doppler tolerant optimised codes and designed mismatched filters using CVX method. The legend shows the length of mismatched filters. Plots are for set of two codes with (a) Length $n = 100$ and loss in PG -1 dB. (b) Length $n = 100$ and loss in PG -1.5 dB. (c) Length $n = 200$ and loss in PG -1 dB. (d) Length $n = 200$ and loss in PG -1.5 dB.

Comparing the SLLs of the set of optimised codes of length $n = 1000$ with a mismatched filter of length $f = 3n$, with the SLLs of the same set of codes due to a matched filter, we see an improvement of almost 2.1 dB with a loss in processing gain of -1 dB. The SLLs of the Doppler tolerant codes were at -24.5 dB with a matched filter. With a mismatched filter, the SLLs are at -26.6 dB. SLLs for other length of codes may be compared on the same lines.

We also investigate the SLLs achieved for a set of optimised codes length $n = 500$ and $n = 1000$, when correlated with mismatched filters of different lengths, designed for different mainlobe widths d, and compare the results with an equivalent matched filter. We fix the loss in PG at -1.2 dB. The results are shown in Figure 7.3.

The SLLs decrease with the increase in the mainlobe width. For the case of a matched filter,
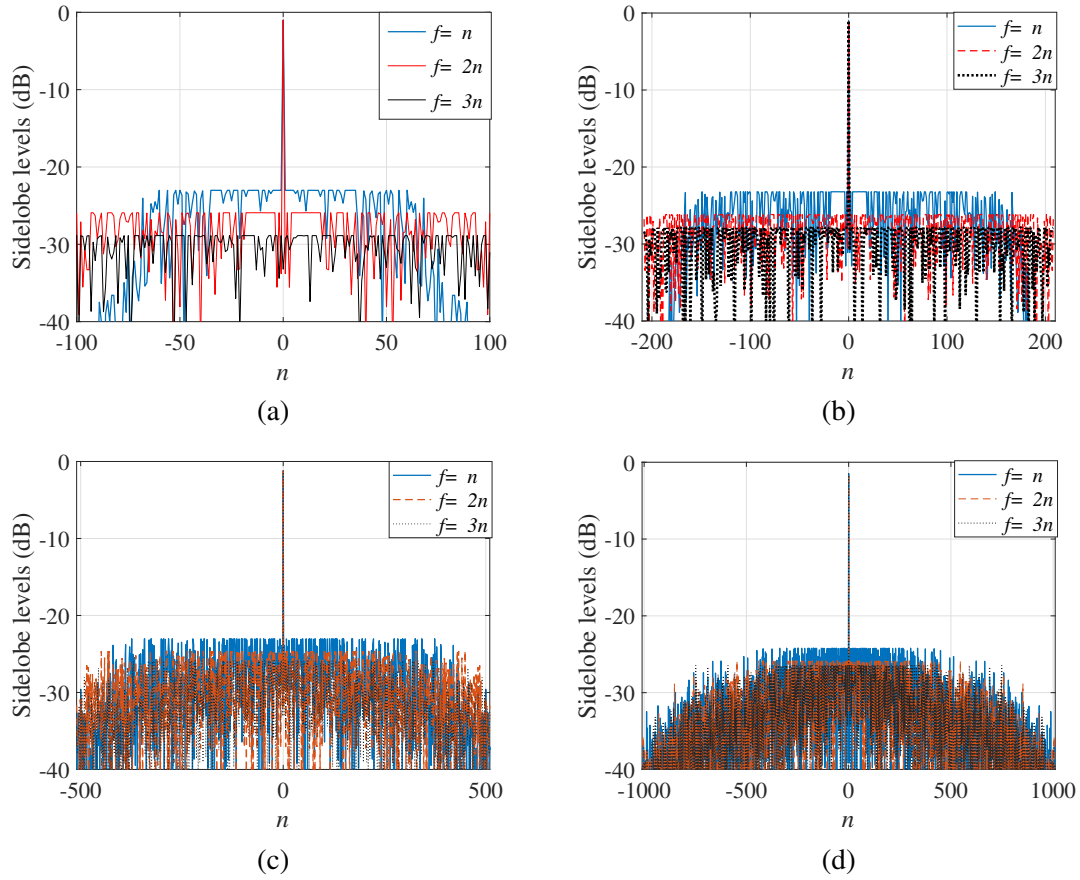
Figure 7.2: Autocorrelations for Doppler tolerant optimised set of codes, with mismatched filter designed using gradient-descent method. Length of the filters is as shown in the legend. The loss in PG is normalised to -1.5 dB for standardised results. Results are shown for a set of two codes with length (a) $n = 100$. (b) $n = 200$. (c) $n = 500$. (d) $n = 1000$.

the SLLs for a set of two codes with length $n = 500$ are at -22.1 dB for a mainlobe width of $d = 0$. The SLLs go as low as -24.5 dB for a mainlobe width of $d = \pm 50$. Similarly, for a set of two codes with length $n = 1000$, the SLLs vary from from -24.2 dB for a mainlobe width of $d = 0$ to -26.9 dB for a mainlobe width of $d = \pm 50$. The SLLs besides being not so low were also not flat. In case of a mismatched filter, the same set of codes with length $n = 500$ has SLLs varying from -23.8 dB for a mainlobe width of $d = 0$ to -28.2 dB for a mainlobe width of $d = \pm 50$. For the set of two codes with length $n = 1000$, the SLLs vary from -26.1 dB to -29 dB for mainlobe widths of $d = 0$ to $d = \pm 50$ respectively. There is an improvement of almost 2 dB when a mismatched filter is used for any case, and the sidelobes are flat.

One disadvantage associated with this method is lack of control over the loss in PG. We
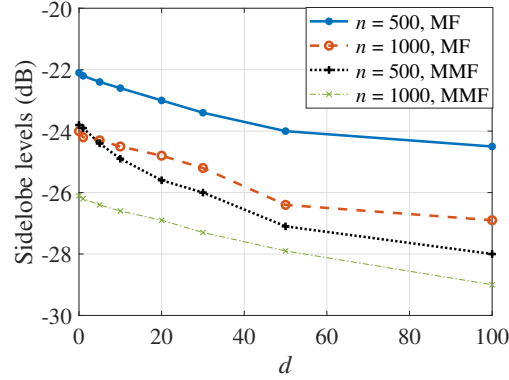
Figure 7.3: SLLs for Doppler tolerant codes of length n versus the mainlobe width d, when correlated with matched filter (MF) and mismatched filter (MMF) of length $f = 3n$. Use of a MMF reduces the SLLs noticeably as compared to a MF.

cannot fix this as was the case in the Convex Optimisation method suggested earlier in Section 7.1.1.

7.2.3 CVX versus Gradient-Descent Method

Design of mismatched filters has been achieved using two methods in the previous subsections. Both methods have proved effective in designing mismatched filters for the Doppler tolerant codes. In both the methods, the optimal properties of the set of codes is preserved, and flat and low SLLs are achieved when the code is correlated with the designed mismatched filter. Also, in both cases, the PSL of the autocorrelations and all sidelobes of crosscorrelations are optimised across the entire range of offset. The SLLs achieved in both cases are an improvement over the case of a matched filter. These lower sidelobes, however, come at the expense of some loss in the mainlobe peak (SNR or PG).

Using the convex optimisation and optimising the sets of codes as a QCQP model is very efficient and was found to work well for sets of codes with short lengths, that is, $n \leq 200$. The method is flexible in that multiple constraints such as the threshold of the SLLs, the threshold of loss in the mainlobe peak, the mainlobe and the sidelobe shaping can be added to the optimisation problem. Additional constraints increase the computational complexity of the optimisation process. These additional constraints could work for a set of codes that do not already have desirable mainlobe and sidelobe characteristics. This is an advantage over other designs suggested in the literature. For the Doppler tolerant codes designed in this thesis, we do not need to add these constraints. These codes have been optimised for a monotonic wider mainlobe and flat and low sidelobes across the entire range offsets. The method involves creating the convolution
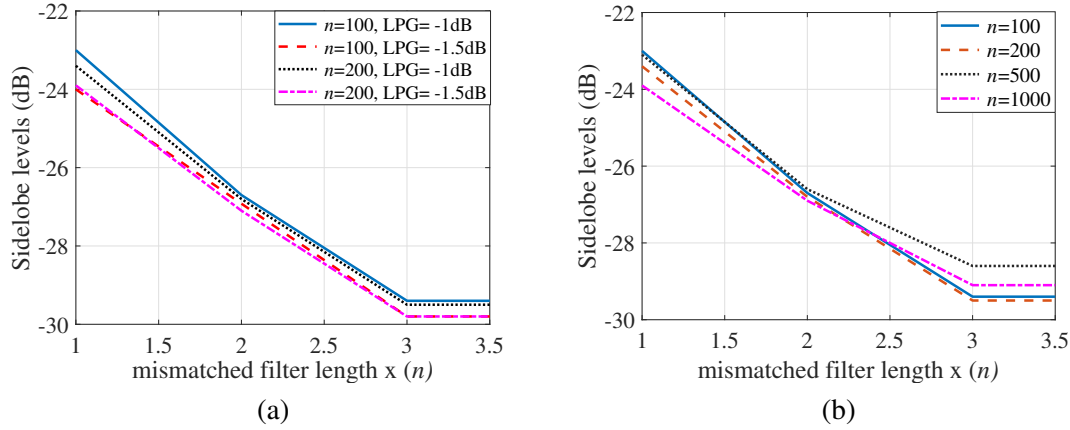
Figure 7.4: SLLs for different length of codes plotted against the mismatched filter lengths. (a) SLLs achieved from CVX Method, legend shows the length of codes and the respective LPG. (b) SLLs achieved from Gradient Descent method, legend shows the length of codes, LPG normalised to -1.5 dB for all codes.

matrix, which becomes a memory constraint for the length of codes $n > 200$.

The gradient-descent method, on the other hand, is a simple, flexible and economical way to design the mismatched filter. The optimisation iterations can work for any length n of codes, and results for sets of codes with length $n \leq 1000$ have been presented in this thesis. Since the problem is still convex, the gradient method should converge to a global minimum, though this may take a lot of time. A drawback associated with this method in its current formulation is the inability to control or limit the threshold on the loss in processing gain.

7.2.4 Comparative Results

The two methods have their own advantages and disadvantages. Based on the application or nature, size and complexity of the problem, either of the methods may be used to design an effective mismatched filter. SLLs achieved from either of the methods are the same, as the problem formulation in either case now is convex; the iterations always end up in the global minimum. Results from the CVX method are presented in Figure 7.4(a) and results from the Gradient Descent method are presented in Figure 7.4(b).

7.3 Summary

In this chapter, we presented a short introduction to the design of mismatched filters. Our Doppler tolerant codes from Chapter 6 had low but no longer flat SLLs. We suggested two methods to

design mismatched filters for optimised codes from this thesis. The first method was based on convex optimisation. The method was very effective and flexible, and a number of constraints can be added to it. The second method was based on the gradient-based optimisation algorithm, similar to previously proposed algorithms in this thesis. This algorithm, like its previous variants, was simple, flexible and could work with any practical length of code. Results from both the methods were presented, along with merits and demerits of each method. Lastly, we compared and analysed both methods so as to establish a guideline on application of a method based on the nature of the design problem.

Chapter 8

Conclusions and Future Work

This thesis has addressed an important research topic in the field of underwater acoustic sensing – code optimisation for application in a multicode sonar system. In particular, the thesis has laid out a general framework for developing sets of codes that are optimised for a number of constraints that are posed by an underwater channel. The aim of the research in this thesis has been to optimise sets of codes for optimal correlation properties and ascertain the performance of these sets of codes against the well known Welch bound [25].

The research has focused on various factors that affect an acoustic wave during the course of its propagation in the UWA channel. The effects of these factors were mathematically modelled. The model was then used to suggest constraints to be applied in the optimisation methods. Consolidated results have been presented to prove the viability and effectiveness of the optimisation approach.

In this chapter, important conclusions from the thesis are drawn. The implications of these conclusions are then presented, followed by suggestions for future work based on the work produced in this thesis.

8.1 Conclusions

The research in this thesis sets out a method to optimise sets of codes for a given set of constraints. Given the prevailing UWA channel conditions in the form of an impulse response, a set of transducers with their frequency responses, a power profile followed by the sonar system power supply, an anticipated speed of the target (say fish), a chosen carrier frequency and bandwidth, an optimised set of codes may be developed which will be optimal for the prevailing conditions and given transceiver.

The findings in this thesis are based on the gradient-descent algorithm we have presented in

Chapter 4. Gradient-descent algorithms are simple and intuitive methods to find solutions to an optimisation problem. They are not memory constrained and so can find an optimal solution to problems involving long codes. The time taken to reach the (local) minimum may be affected by the size of the problem. There are many local minima but all exhibit good performance, with very little variation between them.

The gradient-descent algorithm we have presented is quite flexible so that it can cover a number of aspects for the purpose of optimising sets of codes. These include

1. Conformance of the magnitude of the samples of the code to a given target power profile;

2. Channel frequency response, so that codes are optimised for the given channel characteristics;

3. Transmit amplifier, and the (angle-dependent) projector and hydrophone frequency response;

4. Doppler effects;

5. Minimal PSL of autocorrelations and all crosscorrelations;

In this thesis, we have used an arbitrary filter denoting the channel, the projector and the hydrophone impulse response, and did not use a measured and realistic frequency response. If these characteristics are known, then the Algorithms 1 to 5 are able to design sets of codes that are particularly suited to the given hardware and prevailing channel conditions.

One drawback of the gradient-descent optimisation method we have presented is that it is not particularly fast, so it would be unrealistic to design and update codes in real-time for prevailing circumstances. The UWA channel suffers from a low channel coherence time and it may not be possible to update codes with the changing channel conditions.

From a sonar's perspective, targets are detected based on the strength of reflection from the target. Sonars generally use matched filtering to correlate the received signal with the transmitted signal. Sonars can benefit from the use of multiple orthogonal codes to allow the concurrent use of multiple projectors covering a large area. This provides a greater coverage of the surveying area with reduced time and cost. However, in the case of multiple codes, the SLLs of the matched filter are restricted by the Welch bound [25]. The sidelobes of stronger reflections often mask weaker targets. In order to achieve SLLs lower than the Welch bound [25] for a matched filter, a compromise has to be made. Work in this thesis suggests a compromise that is different from previously proposed compromises in the literature.

In this thesis, we optimise the PSL while widening the autocorrelation mainlobe symmetrically around the zero-offset in a controlled manner, while ensuring the autocorrelation mainlobe is

monotonic. We achieve low and flat SLLs across the entire range of correlation SLLs, which are in some cases, beyond the Welch bound.

For optimising the SLLs, we chose to minimise the PSL instead of the ISL. Optimising the PSL is often more important for a sonar system, which must detect targets in the presence of large discrete clutter. The alternate approach of optimising the ISL results in an overall low sum of correlation sidelobes, with a mix of higher SLLs at some offsets and lower SLLs in the remaining regions. Optimising the PSL results in an overall low value of the SLLs, and usually a flat profile. Allowing some of the SLLs to be higher than the remaining region, we have shown that the best results in terms of lower SLLs are achieved when the region of higher SLLs are just around the zero-offset of autocorrelations. These findings strengthen our idea of permitting a wider autocorrelation mainlobe. The concept can be contrasted with the LCZ/ZCZ where low SLLs are achieved in a specific region of correlations offsets at the cost of higher SLLs in the remaining region. The widening of the autocorrelation mainlobe and optimising the magnitude of the code values to comply with a target power profile are novel contributions from this thesis.

We also investigated the performance of the matched filter for white non-stationary noise due to application of a TVG. A matched filter is the optimal filter to maximise the SNR of a signal received in white and stationary noise. We found that the matched filter is still a suitable filter for SNR improvement of a signal when the noise is white but non-stationary due to TVG. A filter designed to optimise the SNR of the received signal in the modified noise does give a slightly better SNR for long codes and at short ranges, but the slight improvement in SNR is at the cost of considerable degradation of SLLs.

The Doppler effect in an UWA channel is also a cause of degraded SLLs at the receiver. The optimisation algorithm presented in this thesis was modified to include the Doppler model. Sets of codes were optimised for multiple Doppler shifts and for delays which are non-integer multiples of the sample period. It was observed that the the SLLs were considerably degraded. We also found that codes with inherent Doppler tolerance but poor SLLs provide good initialisations for the algorithm presented in this research. These codes retain their Doppler tolerance while improvement in SLLs is achieved.

Finally we investigated mismatched filters to further reduce the SLLs. The mismatched filters from the two design methods we suggested provide lower and flatter SLLs when compared to the matched filters at the cost of some loss in autocorrelation mainlobe peak. The two designs have their own merits and demerits, and an appropriate design may be be chosen based on the size of set of codes and the desired correlation features.

8.2   Implications

The conclusions from this research show that code optimisation is a productive research area with many areas open for improvements. The proposed algorithms in this thesis are simple, economical and flexible. The algorithms are not limited by the number of codes or the length of each code in the set of codes. Minor modifications in the algorithm allow designing of sets of codes for multiple constraints, such as an optimised PAPR, channel, projector and hydrophone frequency responses, size of the monotonic mainlobe width, optimised flat and low SLLs, and the Doppler effect.

Gradient-descent is a local optimisation method, so a global minimum for the solution is not guaranteed, but the results have shown that there are many local minima that exhibit good performance. These codes can then be used in Algorithm 5 to design optimal mismatched filters for the optimised sets of codes. The designing of codes is a non-convex problem, hence a global minimum may not be guaranteed. Designing of a mismatched filter however, is a convex problem, and so a global minimum is always found.

Each additional constraint adds to the computational complexity of the optimisation problem.

8.3   Future Areas of Research

This section presents suggestions for future work that can be undertaken on the topic of code optimisation of multicode sonar systems, based on the findings of the research in this thesis.

The code optimisation algorithm presented in Chapter 4 does not take into account the real-time channel response while optimising the set of codes. It is already established that due to complexities involved in the UWA channel, no standard channel model exists that would be applicable for a generalised case. Where an attempt has been made to develop a model of an UWA channel based on local data, the model is restricted to the local area for the given time, and so is not guaranteed to be a model that can be applied in general. Interesting results are expected when the algorithm is used with a data-set of real-life UWA channels, and the set of transducers and the hydrophones that are intended to be used.

The algorithm has been implemented in double-precision floating points. The numerical robustness of the proposed algorithms should be evaluated for fixed point precision.

In Chapter 5, a matched filter was found to be the optimal receiver filter for maximising SNR of the received signal even in non-stationary white noise. The findings are consistent with the TVG model as shown in Figure 5.1 only, and the optimised codes from Chapter 4. However, it could be interesting to see the results for other models of TVG and other codes that may be used in sonars.

Change in the carrier frequency and the length of the actual signal due to the Doppler effect is inevitable in an UWA channel. The change in the carrier frequency due to the Doppler effect results in a mismatch at the receiver, which is tuned to the transmitted carrier frequency. We have presented a mathematical model for Doppler tolerant code optimisation scheme, and modified our optimisation algorithm to incorporate the model. Our method targets a range of Doppler shifts in a given interval. The optimal SLLs are achieved for the lengths chosen for optimisation. A method has been suggested to minimise the difference between the best SLLs (at Doppler shifts chosen for optimisation) and worst SLLs (at Doppler shifts in between the chosen optimised Doppler shifts). Future work include looking for alternate ways to decrease the difference between the worst and best SLLs and perhaps improve the ambiguity functions of the optimised sets of codes. Another line of research could be to estimate the Doppler shift and optimise the codes for this known Doppler shift, which could show improved performance of the sonar, both in terms of a matched filter and a mismatched filter, for the given UWA channel environment.

Another avenue for future work is the design of mismatched filters for the optimised codes, using methods other than the QCQP approach. The lack of applicability of CVX for longer codes, and the slow convergence of the gradient-based algorithm pose a limitation in their own way. This field is still open to the invention of faster and more memory efficient algorithms for code design. We have presented methods to design mismatched filters for Doppler tolerant codes that have been optimised for a range of Doppler tolerant shifts. The results from this thesis may be compared to an alternate method where a large bank of mismatched filters is used with each filter separately for any Doppler shift that may be expected.

The proposed work in this thesis may be extended to design of sets of codes for SAS. Effects of dispersion, coherent speckle, and distributed scatterers may be incorporated within the suggested framework so as to make the sets of codes robust to these factors.

Appendix A

Proof : The Welch Bound

A.1   The Welch Bound for k=1

Proofs for Welch bound [25] can be found in many places in the literature [126,128,131,258]. The "First Welch Bound" corresponding to $k = 1$ is the one most commonly used in communication applications and a simpler proof of the same as adapted from [259] is presented below. A geometric perspective involving Grammian operators is used to derive the bounds. The proof proceeds in two steps; each depending on a further basic mathematical inequality.

Consider an $m \times m$ Gram Matrix $\mathbf{G}$ of vectors $\mathbf{x} = \{x_i\}$ for $1 \leq i \leq m$, and $\|x_i\| = 1$ as follows

$$\mathbf{G} = \begin{bmatrix} \langle \mathbf{x}_1, \mathbf{x}_1 \rangle & ... & \langle \mathbf{x}_1, \mathbf{x}_m \rangle \\ .. & .. & .. \\ \langle \mathbf{x}_m, \mathbf{x}_1 \rangle & .. & \langle \mathbf{x}_m, \mathbf{x}_m \rangle \end{bmatrix}$$

where $< .,. >$ denotes the inner product. We know that Trace of $\mathbf{G}$ is equal to the sum of its eigenvalues. Also, $\mathbf{G}$ is a non-negative semi-definite matrix, meaning it is a Hermitian matrix with all eigenvalues being non-negative. And since the rank of $\mathbf{G}$ is at most n, it will have n positive eigenvalues with all its remaining eigenvalues equal to zero. So, representing the non-negative eigenvalues as $\lambda_1, ..., \lambda_r$, with $r \leq n$, we have

$$(\mathrm{Tr}\, \mathbf{G})^2 = (\sum_{i=1}^{r} \lambda_i)^2 \tag{A.1}$$

The first step recalls the Cauchy Schwarz's Inequality which is further elaborated in Section A.2 below.

A.2   Cauchy-Schwarz Inequality

The Cauchy-Schwarz Inequality states that for all vectors $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{C}^n$,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{v}, \mathbf{v} \rangle \tag{A.2}$$

Taking the square root of both sides and referring to norms of the vectors yields

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \, \|\mathbf{v}\| \tag{A.3}$$

The condition of equality is satisfied when $\mathbf{u}$ and $\mathbf{v}$ are linearly dependent, i.e., they are parallel (or one is a scalar multiple of the other vector), or one of them has a magnitude zero.

If $u_1, u_2, ..., u_n \in \mathbb{C}$ and $v_1, v_2, ..., v_n \in \mathbb{C}$ and the inner product is the standard complex inner product, the inequality may be restated explicitly as

$$|u_1 v_1^* + ... + u_n v_n^*|^2 \leq (|u_1|^2 + ... + |u_n|^2)(|v_1|^2 + ... + |v_n|^2) \tag{A.4}$$

or

$$\left| \sum_{i=1}^{n} u_i v_i^* \right|^2 \leq \sum_{i=1}^{n} |u_i|^2 \cdot \sum_{k=1}^{n} |v_k^*|^2 \tag{A.5}$$

Proof

A simple proof of the Cauchy-Schwarz Inequality is reproduced here for the purpose of understanding and further use in deriving Welch bounds. In order to prove that

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \, \|\mathbf{v}\|$$

we assume that $\langle \mathbf{u}, \mathbf{v} \rangle \neq 0$ otherwise the inequality is obviously true, because neither $\|\mathbf{u}\|$ nor $\|\mathbf{v}\|$ can be negative.

We now define

$$\mathbf{z} = \mathbf{u} - \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \mathbf{v} \tag{A.6}$$

We then have

$$\langle \mathbf{z}, \mathbf{v} \rangle = \left\langle \mathbf{u} - \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \mathbf{v}, \mathbf{v} \right\rangle = \langle \mathbf{u}, \mathbf{v} \rangle - \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \langle \mathbf{v}, \mathbf{v} \rangle = 0$$

This implies that $\mathbf{z}$ is a vector that is orthogonal to vector $\mathbf{v}$. By applying the Pythagorean theorem to

$$\mathbf{u} = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \mathbf{v} + \mathbf{z}$$

we get

$$\|\mathbf{u}\|^2 = \left|\frac{\langle \mathbf{u}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle}\right|^2 \|\mathbf{v}\|^2 + \|\mathbf{z}\|^2$$

and simplifying, we get

$$\|\mathbf{u}\|^2 = \frac{|\langle \mathbf{u}, \mathbf{v}\rangle|^2}{\|\mathbf{v}\|^2} + \|\mathbf{z}\|^2 \geq \frac{|\langle \mathbf{u}, \mathbf{v}\rangle|^2}{\|\mathbf{v}\|^2}$$

The above mentioned inequality holds by a factor of $\|\mathbf{z}\|^2$, and multiplying both sides of this equation by $\|\mathbf{v}\|^2$ gives

$$\|\mathbf{u}\|^2 \|\mathbf{v}\|^2 \geq |\langle \mathbf{u}, \mathbf{v}\rangle|^2$$

which completes the proof of equation (A.2).

A.3 Back to the First Welch Bound

Applying the Cauchy-Schwarz Inequality to equation (A.1) above yields

$$(\text{Tr } \mathbf{G})^2 = (\sum_{i=1}^{r} \lambda_i)^2 \leq r \sum_{i=1}^{r} \lambda_i^2 \leq n \sum_{i=1}^{m} \lambda_i^2 \qquad (A.7)$$

for $r \leq n$

The Frobenius norm of $\mathbf{G}$ is given by

$$\|\mathbf{G}\|_{\text{F}} = \sqrt{Tr\,(\mathbf{G}\,\mathbf{G}^H)}$$

So the square of the Frobenius norm of $\mathbf{G}$ is given by

$$\|\mathbf{G}\|_{\text{F}}^2 = \sum_{i=1}^{m}\sum_{j=1}^{m} |\langle \mathbf{x}_i, \mathbf{x}_j\rangle|^2 = \sum_{i=1}^{m} \lambda_i^2 \qquad (A.8)$$

From (A.7) and (A.8), we get

$$\sum_{i=1}^{m}\sum_{j=1}^{m} |\langle \mathbf{x}_i, \mathbf{x}_j\rangle|^2 = \sum_{i=1}^{m} \lambda_i^2 \geq \frac{(Tr\,\mathbf{G})^2}{n} \qquad (A.9)$$

We know that $Tr\,\mathbf{G} = m$, so the above inequality reduces to

$$\sum_{i=1}^{m}\sum_{j=1}^{m} |\langle \mathbf{x}_i, \mathbf{x}_j\rangle|^2 \geq \frac{m^2}{n} \qquad (A.10)$$

The left side of equation (A.10) can be divided into two sums, one for $i = j$ and one for $i \neq j$.

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 = \sum_{i=j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 + \sum_{i\neq j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2$$

For $i = j$, we know that each $\mathbf{x}_i$ has unit length so that $\left|< \mathbf{x}_i, \mathbf{x}_j >\right|^2 = 1$ for all the diagonal values of $\mathbf{G}$. So the summation for $i = j$ is equal to m. Thus we have

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 = m + \sum_{i\neq j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 \tag{A.11}$$

From (A.10) and (A.11), we have

$$\sum_{i=1}^{m}\sum_{j=1}^{m}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 = m + \sum_{i\neq j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 \geq \frac{m^2}{n} \tag{A.12}$$

The second part of the proof uses the another simple mathematical inequality, based on the fact that the average of a set of non-negative numbers cannot be greater than the largest number in the set. This can be mathematically represented as : if $a_l \geq 0$ for $l = 1, 2, ...L$, then

$$\frac{1}{L}\sum_{p=1}^{L} a_p \leq \max a_p$$

We know that there are total of $m \times m = m^2$ values in $\mathbf{G}$, of which m are eigenvalues for $i = j$, so for $i \neq j$, there are $m^2 - m$ values or $m(m-1)$ non-negative terms. We denote the largest of these as $c_{\max}^2$, so that each of these $m(m-1)$ term contributes a maximum of $c_{\max}^2$. We can now say that

$$\frac{1}{m(m-1)}\sum_{i\neq j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 \leq c_{\max}^2$$

which is simplified to

$$\sum_{i\neq j}\left|\langle \mathbf{x}_i, \mathbf{x}_j \rangle\right|^2 \leq m(m-1)c_{\max}^2 \tag{A.13}$$

Using (A.12) and (A.13) we get

$$m + m(m-1)c_{\max}^2 \geq \frac{m^2}{n} \tag{A.14}$$

or

$$1 + (m-1)c_{\max}^2 \geq \frac{m}{n}$$

which reduces to

$$c_{\max}^2 \geq \frac{1}{m-1}\left(\frac{m}{n}-1\right) \tag{A.15}$$

or

$$c_{\max} \geq \sqrt{\frac{1}{m-1}\left(\frac{m}{n}-1\right)} \tag{A.16}$$

which completes the proof of the Welch bound [25].

Appendix B

Derivatives for Aperiodic Correlations

Consider for illustration two codes **a** and **b**, each of length say $n = 4$, and defined as

$$\mathbf{a} = [a_1, a_2, a_3, a_4]$$

and

$$\mathbf{b} = [b_1, b_2, b_3, b_4]$$

such that all $a_1, a_2, ..., a_n$, and $b_1, b_2, ..., b_n \in \mathbb{C}$ and $\|\mathbf{a}\| = \|\mathbf{b}\| = 1$

Let $a_n{}^*$ represent the complex conjugate of a_n and $b_n{}^*$ represent the complex conjugate of b_n. The correlations at various lags and the respective derivatives can be computed as explained below.

B.1   Auto Correlations

For lag $j \geq 0$

Let $\mathbf{C_{aa}}(j)$ represent the autocorrelation of code **a** at j-th lag, which is defined as

$$\mathbf{C_{aa}}(j) = \sum_{i=1}^{n} a_i a_{i-j}^* \tag{B.1}$$

The square of autocorrelation $\mathbf{C}_{aa}(j)$ is

$$|\mathbf{C_{aa}}(j)|^2 = \mathbf{C_{aa}}(j)\ \mathbf{C_{aa}^*}(j) \tag{B.2}$$

and the derivative of square of autocorrelations $\mathbf{C}_{aa}(j)$ is

$$\frac{\delta}{\delta \mathbf{a}^*}|\mathbf{C_{aa}}(j)|^2 = \left[\frac{\delta}{\delta \mathbf{a}^*}\mathbf{C_{aa}}(j)\right]\ \mathbf{C_{aa}^*}(j) + \left[\frac{\delta}{\delta \mathbf{a}^*}\mathbf{C_{aa}^*}(j)\right]\ \mathbf{C}_{aa}(j) \tag{B.3}$$

At lag $j = 0$, the autocorrelation $\mathbf{C}_{aa}(j)$ is given by

$$\mathbf{C_{aa}}(0) = a_1 a_1{}^* + a_2 a_2{}^* + a_3 a_3{}^* + a_4 a_4{}^* \tag{B.4}$$

and

$$\mathbf{C^*_{aa}}(0) = a_1^* a_1 + a_2^* a_2 + a_3^* a_3 + a_4^* a_4 \tag{B.5}$$

The derivative of (B.4) and (B.5) with respect to the complex conjugate[1] of each of the element of the code a is

$$\frac{\delta}{\delta \mathbf{a}_1^*} \mathbf{C_{aa}}(0) = a_1$$

$$\frac{\delta}{\delta \mathbf{a}_2^*} \mathbf{C_{aa}}(0) = a_2$$

$$\frac{\delta}{\delta \mathbf{a}_3^*} \mathbf{C_{aa}}(0) = a_3$$

$$\frac{\delta}{\delta \mathbf{a}_4^*} \mathbf{C_{aa}}(0) = a_4$$

$$\frac{\delta}{\delta \mathbf{a}_1^*} \mathbf{C_{aa}}(0) = a_1$$

$$\frac{\delta}{\delta \mathbf{a}_2^*} \mathbf{C_{aa}}(0) = a_2$$

$$\frac{\delta}{\delta \mathbf{a}_3^*} \mathbf{C_{aa}}(0) = a_3$$

$$\frac{\delta}{\delta \mathbf{a}_4^*} \mathbf{C_{aa}}(0) = a_4$$

so that

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{aa}}(0) = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \tag{B.6}$$

---

[1] See Appendix E for details of derivatives with respect to complex conjugate

and

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C}_{\mathbf{aa}}^*(0) = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \tag{B.7}$$

Substituting in (B.3), we get

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C}_{\mathbf{aa}}(0)|^2 = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \mathbf{C}_{\mathbf{aa}}(0) + \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \mathbf{C}_{\mathbf{aa}}(0) \tag{B.8}$$

which simplifies to

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C}_{\mathbf{aa}}(0)|^2 = a \left[\mathbf{C}_{\mathbf{aa}}(0) + \mathbf{C}_{\mathbf{aa}}^*(0) \right] \tag{B.9}$$

Similarly, the derivative of autocorrelation $\mathbf{C_{bb}}$ with respect to $\mathbf{b}^*$ is

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C}_{\mathbf{bb}}(0)|^2 = b \left[\mathbf{C}_{\mathbf{bb}}(0) + \mathbf{C}_{\mathbf{bb}}^*(0) \right] \tag{B.10}$$

For $j = 1$, the code will be offset by a value of j, and prefixed by j zeros accordingly. For $j = 1$ the autocorrelation is

$$\mathbf{C}_{\mathbf{aa}}(1) = a_1 a_2{}^* + a_2 a_3{}^* + a_3 a_4{}^* \tag{B.11}$$

and

$$\mathbf{C}_{\mathbf{aa}}^*(1) = a_1^* a_2 + a_2^* a_3 + a_3^* a_4 \tag{B.12}$$

The derivatives of (B.11) and (B.12) with respect to complex conjugate of each of the element of $\mathbf{a}$ are

$$\frac{\delta}{\delta \mathbf{a}_1^*} \mathbf{C}_{\mathbf{aa}}(1) = 0$$

$$\frac{\delta}{\delta \mathbf{a}_2^*} \mathbf{C}_{\mathbf{aa}}(1) = a_1$$

$$\frac{\delta}{\delta \mathbf{a}_3^*} \mathbf{C_{aa}}(1) = a_2$$

$$\frac{\delta}{\delta \mathbf{a}_4^*} \mathbf{C_{aa}}(1) = a_3$$

$$\frac{\delta}{\delta \mathbf{a}_1^*} \mathbf{C_{aa}^*}(1) = 0$$

$$\frac{\delta}{\delta \mathbf{a}_2^*} \mathbf{C_{aa}^*}(1) = a_1$$

$$\frac{\delta}{\delta \mathbf{a}_3^*} \mathbf{C_{aa}^*}(1) = a_2$$

$$\frac{\delta}{\delta \mathbf{a}_4^*} \mathbf{C_{aa}^*}(1) = a_3$$

so that

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{aa}}(1) = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{B.13}$$

and

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{aa}}(1) = \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \tag{B.14}$$

Substituting in (B.3), we have

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C_{aa}}(1)|^2 = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \mathbf{C_{aa}^*}(1) + \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \mathbf{C_{aa}}(1) \tag{B.15}$$

Similarly, the derivative of $\mathbf{C_{bb}}(1)$ with respect to complex conjugate of each element of $\mathbf{b}$ is

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C_{bb}}(1)|^2 = \begin{bmatrix} 0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \mathbf{C_{bb}^*}(1) + \begin{bmatrix} b_2 \\ b_3 \\ b_4 \\ 0 \end{bmatrix} \mathbf{C_{bb}}(1) \tag{B.16}$$

Generalising for all offsets $j \geq 0$, we have

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C_{aa}}(j)|^2 = \begin{bmatrix} \mathbf{z}_j \\ a_1 \\ \vdots \\ a_{n-j} \end{bmatrix} \mathbf{C_{aa}^*}(j) + \begin{bmatrix} a_{j+1} \\ \vdots \\ a_n \\ \mathbf{z}_j \end{bmatrix} \mathbf{C_{aa}}(j) \tag{B.17}$$

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C_{bb}}(j)|^2 = \begin{bmatrix} \mathbf{z}_j \\ b_1 \\ \vdots \\ b_{n-j} \end{bmatrix} \mathbf{C_{bb}^*}(j) + \begin{bmatrix} b_{j+1} \\ \vdots \\ b_n \\ \mathbf{z}_j \end{bmatrix} \mathbf{C_{bb}}(j) \tag{B.18}$$

where $\mathbf{z}_j$ is a length j vector of zeros.

Lag $j \leq$ -1

For lag of $j \leq$ -1, the codes will be offset by a value of j, with j zeros appended. For $j = -1$, the autocorrelation $\mathbf{C_{aa}}$ of code $\mathbf{a}$ is given by

$$\mathbf{C_{aa}}(-1) = a_2 a_1{}^* + a_3 a_2{}^* + a_4 a_3{}^* \tag{B.19}$$

and

$$\mathbf{C_{aa}}(-1) = a_2^* a_1 + a_3^* a_2 + a_4^* a_3 \tag{B.20}$$

The derivatives of (B.19) and (B.20) with respect to complex conjugate of each elements of $\mathbf{a}$ are given by

$$\frac{\delta}{\delta a_1{}^*}(\mathbf{C_{aa}}(-1)) = a_2$$

$$\frac{\delta}{\delta a_2{}^*}(\mathbf{C_{aa}}(-1)) = a_3$$

$$\frac{\delta}{\delta a_3{}^*}(\mathbf{C_{aa}}(-1)) = a_4$$

$$\frac{\delta}{\delta a_4{}^*}(\mathbf{C_{aa}}(-1)) = 0$$

$$\frac{\delta}{\delta a_1{}^*}(\mathbf{C_{aa}^*}(-1)) = 0$$

$$\frac{\delta}{\delta a_2{}^*}(\mathbf{C_{aa}^*}(-1)) = a_1$$

$$\frac{\delta}{\delta a_3{}^*}(\mathbf{C}_{\mathbf{aa}}^*(-1)) = a_2$$

$$\frac{\delta}{\delta a_4{}^*}(\mathbf{C}_{\mathbf{aa}}^*(-1)) = a_3$$

so that

$$\frac{\delta}{\delta \mathbf{a}^*}\mathbf{C}_{\mathbf{aa}}(-1) = \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \tag{B.21}$$

and

$$\frac{\delta}{\delta \mathbf{a}^*}\mathbf{C}_{\mathbf{aa}}^*(-1) = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{B.22}$$

so that

$$\frac{\delta}{\delta \mathbf{a}^*}|\mathbf{C}_{\mathbf{aa}}(-1)|^2 = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \mathbf{C}_{\mathbf{aa}}(-1) + \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \mathbf{C}_{\mathbf{aa}}(-1)^* \tag{B.23}$$

We can generalise this for all negative lags $j < 0$ for code $\mathbf{a}$ as

$$\frac{\delta}{\delta \mathbf{a}^*}|\mathbf{C}_{\mathbf{aa}}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ a_1 \\ \vdots \\ a_{n+j} \end{bmatrix} \mathbf{C}_{\mathbf{aa}}(j) + \begin{bmatrix} a_{|j|+1} \\ \vdots \\ a_n \\ \mathbf{z}_{|j|} \end{bmatrix} \mathbf{C}_{\mathbf{aa}}^*(j) \tag{B.24}$$

Similarly, the derivative of autocorrelation of code $\mathbf{b}$ at all lags $j < 0$ lag can be shown to be

$$\frac{\delta}{\delta \mathbf{b}^*}|\mathbf{C}_{\mathbf{bb}}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ b_1 \\ \vdots \\ b_{n+j} \end{bmatrix} \mathbf{C}_{\mathbf{bb}}(j) + \begin{bmatrix} b_{|j|+1} \\ \vdots \\ b_n \\ \mathbf{z}_{|j|} \end{bmatrix} \mathbf{C}_{\mathbf{bb}}^*(j) \tag{B.25}$$

B.2 Cross Correlations

<u>Lag $j \geq 0$</u>

Let $\mathbf{C_{ab}}(j)$ represent the crosscorrelation of code $\mathbf{a}$ and code $\mathbf{b}$ at $j - th$ lag, which is defined as

$$\mathbf{C_{ab}}(j) = \sum_{i=1}^{n} a_i b_{i-j}^* \tag{B.26}$$

The square of autocorrelation $\mathbf{C_{ab}}(j)$ is

$$|\mathbf{C_{ab}}(j)|^2 = \mathbf{C_{ab}}(j) \ \mathbf{C_{ab}^*}(j) \tag{B.27}$$

and the derivative of square of autocorrelations $\mathbf{C_{ab}}(j)$ is

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C_{ab}}(j)|^2 = \left[\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{ab}}(j) \right] \ \mathbf{C_{ab}^*}(j) + \left[\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{ab}^*}(j) \right] \ \mathbf{C_{ab}}(j) \tag{B.28}$$

Then for $j = 0$,

$$\mathbf{C_{ab}}(0) = a_1 b_1{}^* + a_2 b_2{}^* + a_3 b_3{}^* + a_4 b_4{}^* \tag{B.29}$$

and

$$\mathbf{C_{ab}^*}(0) = a_1^* b_1 + a_2^* b_2 + a_3^* b_3 + a_4^* b_4 \tag{B.30}$$

The derivatives of (B.29) and (B.30) with respect to complex conjugate of $\mathbf{a}$ are

$$\frac{\delta}{\delta a^*} (\mathbf{C_{ab}}(0)) = 0 \tag{B.31}$$

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C_{ab}^*}(0) = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \tag{B.32}$$

Similarly, the derivatives of (B.29) and (B.30) with respect to complex conjugate of $\mathbf{b}$ are

$$\frac{\delta}{\delta \mathbf{b}^*} \mathbf{C_{ab}}(0) = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \tag{B.33}$$

and

$$\frac{\delta}{\delta b^*}(\mathbf{C}_{\mathbf{ab}}^*(0)) = 0 \tag{B.34}$$

so that we have

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C}_{\mathbf{ab}}(0)|^2 = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \mathbf{C}_{\mathbf{ab}}(0) \tag{B.35}$$

and

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C}_{\mathbf{ab}}(0)|^2 = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \mathbf{C}_{\mathbf{ab}}^*(0) \tag{B.36}$$

Similarly, for lag $j = 1$, the code $\mathbf{b}$ will be shifted right by one offset and appended by one zero, so that

$$\mathbf{C}_{\mathbf{ab}}(1) = a_1 b_2^* + a_2 b_3^* + a_3 b_4^* \tag{B.37}$$

and

$$\mathbf{C}_{\mathbf{ab}}^*(1) = a_1^* b_2 + a_2^* b_3 + a_3^* b_4 \tag{B.38}$$

The derivatives of (B.37) and (B.38) with respect to complex conjugate of $\mathbf{a}$ are

$$\frac{\delta}{\delta a^*}(\mathbf{C}_{\mathbf{ab}}(1)) = 0 \tag{B.39}$$

$$\frac{\delta}{\delta \mathbf{a}^*} \mathbf{C}_{\mathbf{ab}}^*(1) = \begin{bmatrix} b_2 \\ b_3 \\ b_4 \\ 0 \end{bmatrix} \tag{B.40}$$

Similarly, the derivatives of (B.37) and (B.38) with respect to complex conjugate of **b** are

$$\frac{\delta}{\delta \mathbf{b}^*} \mathbf{C_{ab}}(1) = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{B.41}$$

and

$$\frac{\delta}{\delta b^*} (\mathbf{C}^*_{\mathbf{ab}}(1)) = 0 \tag{B.42}$$

so that we have

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C_{ab}}(1)|^2 = \begin{bmatrix} b_2 \\ b_3 \\ b_4 \\ 0 \end{bmatrix} \mathbf{C_{ab}}(1) \tag{B.43}$$

and

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C_{ab}}(1)|^2 = \begin{bmatrix} 0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \mathbf{C}^*_{\mathbf{ab}}(1) \tag{B.44}$$

Generalising for lags $j \geq 0$, we have

$$\frac{\delta}{\delta \mathbf{a}^*} |\mathbf{C_{ab}}(j)|^2 = \begin{bmatrix} b_{j+1} \\ \vdots \\ \mathbf{z}_j \end{bmatrix} \mathbf{C_{ab}}(j) \tag{B.45}$$

and

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C_{ab}}(j)|^2 = \begin{bmatrix} \mathbf{z}_j \\ a_1 \\ \vdots \\ a_{n-j} \end{bmatrix} \mathbf{C}^*_{\mathbf{ab}}(j) \tag{B.46}$$

<u>Lag $j < 0$</u>

For all negative offsets, the codes **a** will be shifted left by j offsets, with j zeros appended. At lag $j = $ -1, the crosscorrelation $\mathbf{C_{ab}}(-1)$ is

$$\mathbf{C_{ab}}(-1) = a_2 b_1{}^* + a_3 b_2{}^* + a_4 b_3{}^* \tag{B.47}$$

and

$$\mathbf{C^*_{ab}}(-1) = a_2 b_1{}^* + a_3 b_2{}^* + a_4 b_3{}^* \tag{B.48}$$

The derivatives of (B.47) and (B.48) with respect to complex conjugate of **a** are

$$\frac{\delta}{\delta a^*}(\mathbf{C_{ab}}(-1)) = 0 \tag{B.49}$$

$$\frac{\delta}{\delta \mathbf{a}^*}\mathbf{C^*_{ab}}(-1) = \begin{bmatrix} 0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{B.50}$$

Similarly, the derivatives of (B.37) and (B.38) with respect to complex conjugate of **b** are

$$\frac{\delta}{\delta \mathbf{b}^*}\mathbf{C_{ab}}(-1) = \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \tag{B.51}$$

and

$$\frac{\delta}{\delta b^*}(\mathbf{C^*_{ab}}(-1)) = 0 \tag{B.52}$$

so that we have

$$\frac{\delta}{\delta \mathbf{a}^*}|\mathbf{C_{ab}}(-1)|^2 = \begin{bmatrix} 0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \mathbf{C_{ab}}(-1) \tag{B.53}$$

and

$$\frac{\delta}{\delta \mathbf{b}^*} |\mathbf{C_{ab}}(-1)|^2 = \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ 0 \end{bmatrix} \mathbf{C_{ab}^*}(-1) \tag{B.54}$$

Generalising for all lags $j < 0$, we have

$$\frac{\delta}{\delta \mathbf{a}^*} (\mathbf{C_{ab}}(j)) = \begin{bmatrix} \mathbf{z}_{|j|} \\ b_1 \\ \vdots \\ b_{n+j} \end{bmatrix} \mathbf{C_{ab}}(j) \tag{B.55}$$

and

$$\frac{\delta}{\delta \mathbf{a}^*} (\mathbf{C_{ab}}(j)) = \begin{bmatrix} a_{|j|+1} \\ \vdots \\ a_n \\ \mathbf{z}_{|j|} \end{bmatrix} \mathbf{C_{ab}}(j) \tag{B.56}$$

Appendix C

Matched Filter in White Stationary Noise

Consider a general linear time-invariant (LTI) filter with impulse response $h(t)$, input signal $x(t)$, and output signal $y(t)$ and corresponding Fourier transforms $H(f)$, $X(f)$ and $Y(f)$, so that

$$x(t) = s_1(t) + v_1(t) \tag{C.1}$$

$$y(t) = s_o(t) + v_o(t) \tag{C.2}$$

where $s_1(t)$ is the signal component at the input, $v_1(t)$ is the noise component at the input, $s_o(t)$ is signal component at the output, and $v_o(t)$ is the noise component at the output.

The frequency domain representation of (C.1) and (C.2) is

$$X(f) = S_1(f) + V_1(f) \tag{C.3}$$

$$Y(f) = S_o(f) + V_o(f) \tag{C.4}$$

The output signal $s_o(t)$ from the filter $h(t)$ is

$$s_o(t) = s_1(t) * h(t) = \int_{-\infty}^{\infty} H(f) S_1(f) e^{j2\pi ft} df \tag{C.5}$$

where $H(f)$ is the Fourier transform of of $h(t)$. At any time t, the Total Power P_{s_o} of the output

signal $s_{o(t)}$ is

$$P_{s_0} = |s_0(t)|^2 = \left| \int_{-\infty}^{\infty} S_1(f)H(f)e^{j2\pi ft}df \right|^2 \tag{C.6}$$

The input noise $v_1(t)$ is assumed to be white and stationary, and the double-sided spectral power P_{v_1} is

$$P_{v_1} = \int_{-\infty}^{\infty} |V_1(f)|^2 df = \frac{K}{2} \tag{C.7}$$

where K is an arbitrary constant.

The total output noise power P_{v_0} of the output noise v_o is

$$P_{v_0} = E\left\{v_o^2(t)\right\} = \frac{K}{2} \int_{-\infty}^{\infty} |H(f)|^2 df \tag{C.8}$$

The output SNR$_o$ is then

$$\text{SNR}_o = \frac{P_{s_0}(t)}{E\left\{v_1^2(t)\right\}} = \frac{\left|\int_{-\infty}^{\infty} S_1(f)H(f)e^{j2\pi ft}df\right|^2}{\frac{K}{2}\int_{-\infty}^{\infty} |H(f)|^2 df} \tag{C.9}$$

From the Cauchy – Schwarz inequality [260], we have

$$\left| \int_{-\infty}^{\infty} a(f)b(f)df \right|^2 \leq \int_{-\infty}^{\infty} |a(f)|^2 df \left| \int_{-\infty}^{\infty} |b(f)|^2 df \right. \tag{C.10}$$

Let $a(f) = H(f)$ and $b(f) = S_1(f)e^{j2\pi ft}df$. (C.9) now becomes

$$\text{SNR}_o = \frac{\left|\int_{-\infty}^{\infty} S_1(f)H(f)e^{j2\pi ft}df\right|^2}{\frac{K}{2}\int_{-\infty}^{\infty} |H(f)|^2 df} \leq \frac{\int_{-\infty}^{\infty} \left|S_1(f)e^{j2\pi ft}\right|^2 df \int_{-\infty}^{\infty} |H(f)|^2 df}{\frac{K}{2}\int_{-\infty}^{\infty} |H(f)|^2 df} \tag{C.11}$$

Since $\int_{-\infty}^{\infty} \left|S_1(f)e^{j2\pi ft}\right|^2 df = \int_{-\infty}^{\infty} |S_1(f)|^2 df$, (C.11) simplifies to

$$\text{SNR}_o \leq \frac{2}{K} \int_{-\infty}^{\infty} |S_1(f)|^2 df \tag{C.12}$$

with equality if

$$H(f) = kS_1^*(f) \tag{C.13}$$

where k is a scalar. $H(f)$ the optimal filter that will maximise the output SNR$_o$ of the received

signal. In the time domain, the optimal filter h_{opt} is

$$h_{\text{opt}} = \mathscr{F}^{-1}\left[kS_1^*(f)\right] = k \int_{-\infty}^{\infty} S_1(-f)e^{-j2\pi ft}df \tag{C.14}$$

and since $S^*(f) = S(-f)$, we have

$$h_{\text{opt}} = ks_1^*(-t) \tag{C.15}$$

Appendix D

Derivatives for Doppler Tolerant Code Optimisation

Consider a code **a** with length $n = 4$ as the originally transmitted code and **a'** as the Doppler shifted code with length $m = 5$. We define the codes as

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

and

$$\mathbf{a'} = \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \\ a'_4 \\ a'_5 \end{bmatrix}$$

where

$$\mathbf{a'} = \mathbf{T} \, \mathbf{a} \tag{D.1}$$

and **T** is the transformation matrix as already explained in Section 6.1.

D.1   Autocorrelations

We calculate the derivatives of $\mathbf{C_{aa'}}$, the correlation of the transmitted code **a** and the reflected code **a'** as follows. We first append $m - n$ zeros to the code **a** so that it is of the same length as **a'**.

We have

$$\mathbf{C_{aa'}}(j) = \sum_{i=1}^{m} a_i a_{i-j}'^* \tag{D.2}$$

Substituting (D.1) in (D.2), we have

$$\mathbf{C_{aa'}}(j) = \sum_{i=1}^{m} a_i \left(T_{i-j,:} \, \mathbf{a}\right)^* \tag{D.3}$$

where $\mathbf{T}_{i-j,:}$ denotes all the columns of the ij-th row. (D.3) simplifies to

$$\mathbf{C_{aa'}}(j) = \sum_{i=1}^{m} a_i \, \mathbf{a}^* \, T_{i-j,:}^* \tag{D.4}$$

Similarly, we can calculate $\mathbf{C}_{\mathbf{aa'}}^*(j)$ which is

$$\mathbf{C}_{\mathbf{aa'}}^*(j) = \sum_{i=1}^{m} T_{i-j,:} \, \mathbf{a} \, a_i^* \tag{D.5}$$

The derivative of the squared correlation of code $\mathbf{a}$ and code $\mathbf{a'}$ is

$$\frac{\partial}{\partial \mathbf{a}^*} \left|\mathbf{C_{aa'}}(j)\right|^2 = \left[\frac{\partial}{\partial \mathbf{a}^*} \left(\mathbf{C_{aa'}}(j)\right)\right] \mathbf{C}_{\mathbf{aa'}}^*(j) + \left[\frac{\partial}{\partial \mathbf{a}^*} \left(\mathbf{C}_{\mathbf{aa'}}^*(j)\right)\right] \mathbf{C_{aa'}}(j) \tag{D.6}$$

Now we calculate the derivative of $\mathbf{C_{aa'}}$ for various lags j as follows.
For lag $j \geq 0$

Consider the case for lag j=0. From (D.4)

$$\mathbf{C_{aa'}}(0) = \sum_{i=1}^{m} a_i \, \mathbf{a}^* \, T_i^* \tag{}$$

or

$$\mathbf{C_{aa'}}(0) = a_1 \, \mathbf{a}^* \, (T_{1,:})^* + a_2 \, \mathbf{a}^* \, (T_{2,:})^* + a_3 \, \mathbf{a}^* \, (T_{3,:})^* + a_4 \, \mathbf{a}^* \, (T_{4,:})^* + a_5 \, \mathbf{a}^* \, (T_{5,:})^* \tag{D.7}$$

Similarly,

$$\mathbf{C}_{\mathbf{aa'}}^*(0) = T_{1,:} \, \mathbf{a} \, a_1^* + T_{2,:} \, \mathbf{a} \, a_2^* + T_{3,:} \, \mathbf{a} \, a_3^* + T_{4,:} \, \mathbf{a} \, a_4^* + T_{5,:} \, \mathbf{a} \, a_5^* \tag{D.8}$$

The derivatives of (D.7) and (D.8) are

$$\frac{\partial}{\partial \mathbf{a}^*} \left(\mathbf{C_{aa'}}(0)\right) = a_1 \, (T_{1,:})^* + a_2 \, (T_{2,:})^* + a_3 \, (T_{3,:})^* + a_4 \, (T_{4,:})^* + a_5 \, (T_{5,:})^* \tag{D.9}$$

where $\left(T_{(i,:)}\right)^*$ is simply the conjugate transpose of $T_{(i,:)}$, and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C}^*_{\mathbf{aa}'}(0)\right) = \begin{bmatrix} T_{1,:}\,\mathbf{a} \\ T_{2,:}\,\mathbf{a} \\ T_{3,:}\,\mathbf{a} \\ T_{4,:}\,\mathbf{a} \\ T_{5,:}\,\mathbf{a} \end{bmatrix} \tag{D.10}$$

Now consider the case for lag j=1. From (D.4)

$$\mathbf{C}_{\mathbf{aa}'}(1) = \sum_{i=1}^{m} a_i\,\mathbf{a}^*\,T^*_{i-1}$$

or

$$\mathbf{C}_{\mathbf{aa}'}(1) = a_1\,\mathbf{a}^*\,(T_{0,:})^* + a_2\,\mathbf{a}^*\,(T_{1,:})^* + a_3\,\mathbf{a}^*\,(T_{2,:})^* + a_4\,\mathbf{a}^*\,(T_{3,:})^* \tag{D.11}$$

Similarly,

$$\mathbf{C}^*_{\mathbf{aa}'}(1) = T_{0,:}\,\mathbf{a}^*\,a_1 + T_{1,:}\,\mathbf{a}^*\,a_2 + T_{2,:}\,\mathbf{a}^*\,a_3 + T_{3,:}\,\mathbf{a}^*\,a_4 + +T_{4,:}\,\mathbf{a}^*\,a_5 \tag{D.12}$$

The derivatives of (D.11) and (D.12) are

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C}_{\mathbf{aa}'}(1)\right) = a_1\,(T_{0,:})^* + a_2\,(T_{1,:})^* + a_3\,(T_{2,:})^* + a_4\,(T_{3,:})^* + a_5\,(T_{4,:})^* \tag{D.13}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C}^*_{\mathbf{aa}'}(1)\right) = \begin{bmatrix} T_{0,:}\,\mathbf{a} \\ T_{1,:}\,\mathbf{a} \\ T_{2,:}\,\mathbf{a} \\ T_{3,:}\,\mathbf{a} \\ T_{4,:}\,\mathbf{a} \end{bmatrix} \tag{D.14}$$

Generalising for any code $\mathbf{a}$ with length n, and any matrix $\mathbf{T}$ with size $m \times n$, where m is the length of Doppler shifted code $\mathbf{a}'$, the derivative of $\mathbf{C}_{\mathbf{aa}'}$ for any lag $j \geq 0$

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C}_{\mathbf{aa}'}(j)\right) = \sum_{i=1}^{m} a_i\,\left(T^*_{(:,i-j)}\right) \tag{D.15}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C}^*_{\mathbf{aa}'}(j)\right) = \begin{bmatrix} \mathbf{z}_j \\ T_{(1,:)}\,\mathbf{a} \\ \vdots \\ T_{(m-j,:)}\,\mathbf{a} \end{bmatrix} \tag{D.16}$$

Substituting in (D.6), we have

$$\frac{\partial}{\partial \mathbf{a}^*}|\mathbf{C}_{\mathbf{aa}'}(j)|^2 = \left[\sum_{k=0}^{m-1} a_{k+1}\left(T^*_{(:,k-j)}\right)\right]\mathbf{C}^*_{\mathbf{aa}'}(j) + \begin{bmatrix} \mathbf{z}_j \\ T_{(1,:)}\,\mathbf{a} \\ \vdots \\ T_{(m-j,:)}\,\mathbf{a} \end{bmatrix}\mathbf{C}_{\mathbf{aa}'}(j) \tag{D.17}$$

Similarly, the derivatives for autocorrelations $\mathbf{C}_{\mathbf{bb}'}$ for lag $j \geq 0$ are

$$\frac{\partial}{\partial \mathbf{b}^*}\left(\mathbf{C}_{\mathbf{bb}'}(j)\right) = \sum_{i=1}^{m} b_i\left(T^*_{(:,i-j)}\right) \tag{D.18}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*}\left(\mathbf{C}^*_{\mathbf{bb}'}(j)\right) = \begin{bmatrix} T_{(|j|+1:)}\,\mathbf{b} \\ \vdots \\ T_{(m,:)}\,\mathbf{b} \\ \mathbf{z}_j \end{bmatrix} \tag{D.19}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*}|\mathbf{C}_{\mathbf{bb}'}(j)|^2 = \sum_{i=1}^{m-|j|} b_i\left(T^*_{(:,i+|j|)}\right)\mathbf{C}^*_{\mathbf{bb}'}(j) + \begin{bmatrix} T_{(|j|+1:)}\,\mathbf{b} \\ \vdots \\ T_{(m,:)}\,\mathbf{b} \\ \mathbf{z}_j \end{bmatrix}\mathbf{C}_{\mathbf{bb}'}(j) \tag{D.20}$$

For lag $j < 0$

Consider the case for lag $j = -1$. From (D.4)

$$\mathbf{C}_{\mathbf{aa}'}(-1) = \sum_{k=1}^{m} a_k\,\mathbf{a}^*\,T^*_{k+1}$$

or

$$\mathbf{C_{aa'}}(-1) = a_1\,\mathbf{a}^*\;(T_{2,:})^* + a_2\,\mathbf{a}^*\;(T_{3,:})^* + a_3\,\mathbf{a}^*\;(T_{4,:})^* + a_4\,\mathbf{a}^*\;(T_{5,:})^* + + a_5\,\mathbf{a}^*\;(T_{6,:})^* \quad \text{(D.21)}$$

Similarly,

$$\mathbf{C_{aa'}^*}(-1) = T_{(2,:)}\,\mathbf{a}\,a_1^* + T_{(3,:)}\,\mathbf{a}\,a_2^* + T_{(4,:)}\,\mathbf{a}\,a_3^* + T_{(5,:)}\,\mathbf{a}\,a_4^* T_{(6,:)}\,\mathbf{a}\,a_5^* \quad \text{(D.22)}$$

The derivatives of (D.21) and (D.22) are

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C_{aa'}}(-1)\right) = a_1\left(T_{(2,:)}\right)^* + a_2\left(T_{(3,:)}\right)^* + a_3\left(T_{(4,:)}\right)^* + a_4\left(T_{(5,:)}\right)^* + a_5\left(T_{(6,:)}\right)^* \quad \text{(D.23)}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C_{aa'}^*}(-1)\right) = \begin{bmatrix} T_{2,:}\,\mathbf{a} \\ T_{3,:}\,\mathbf{a} \\ T_{4,:}\,\mathbf{a} \\ T_{5,:}\,\mathbf{a} \\ T_{6,:}\,\mathbf{a} \end{bmatrix} \quad \text{(D.24)}$$

Generalising for any lag $j < 0$, we have

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C_{aa'}}(j)\right) = \sum_{i=1}^{m-|j|} a_i\left(T_{(:,i+|j|)}^*\right) \quad \text{(D.25)}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C_{aa'}^*}(j)\right) = \begin{bmatrix} T_{(|j|+1:)}\,\mathbf{a} \\ \vdots \\ T_{(m,:)}\,\mathbf{a} \\ \mathbf{z}_j \end{bmatrix} \quad \text{(D.26)}$$

Substituting in (D.6) and for all lags $j < 0$, we have

$$\frac{\partial}{\partial \mathbf{a}^*}\left|\mathbf{C_{aa'}}(j)\right|^2 = \sum_{i=1}^{m-|j|} a_i\left(T_{(:,i+|j|)}^*\right)\mathbf{C_{aa'}^*}(j) + \begin{bmatrix} T_{(|j|+1:)}\,\mathbf{a} \\ \vdots \\ T_{(m,:)}\,\mathbf{a} \\ \mathbf{z}_j \end{bmatrix}\mathbf{C_{aa'}}(j) \quad \text{(D.27)}$$

Similarly, the derivatives for $\mathbf{C_{bb'}}(j)$ for any lag $j < 0$, we have

$$\frac{\partial}{\partial \mathbf{b}^*}\left(\mathbf{C_{bb'}}(j)\right) = \sum_{k=-1}^{-m+1} b_{|k|}\left(T^*_{(:,|k|+1)}\right) \tag{D.28}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*}\left(\mathbf{C^*_{bb'}}(j)\right) = \begin{bmatrix} T_{(|j|+1:)}\,\mathbf{b} \\ \vdots \\ T_{(m,:)}\,\mathbf{b} \\ |j|\ zeros \end{bmatrix} \tag{D.29}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*}|\mathbf{C_{bb'}}(j)|^2 = \left[\sum_{k=0}^{m-1} b_{k+1}\left(T^*_{(:,k)}\right)\right]\mathbf{C^*_{bb'}}(j) + \begin{bmatrix} |j|\ zeros \\ T_{(1,:)}\,\mathbf{b} \\ \vdots \\ T_{(m-j,:)}\,\mathbf{b} \end{bmatrix}\mathbf{C_{bb'}}(j) \tag{D.30}$$

D.2 Crosscorrelations

The derivatives for $\mathbf{C_{ab'}}$ with respect to $\mathbf{a}^*$ and $\mathbf{b}^*$ can be calculated on the same lines, as follows.

For $j \geq 0$

The derivatives for $\mathbf{C_{ab'}}$ with respect to $\mathbf{a}^*$ are

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C_{ab'}}(j)\right) = 0 \tag{D.31}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*}\left(\mathbf{C^*_{ab'}}(j)\right) = \begin{bmatrix} \mathbf{z}_j \\ T_{1,:}\mathbf{b} \\ T_{2,:}\mathbf{b} \\ \cdot \\ T_{m-j,:}\mathbf{b} \end{bmatrix} \tag{D.32}$$

so that

$$\frac{\partial}{\partial \mathbf{a}^*} |\mathbf{C}_{\mathbf{ab'}}(j)|^2 = \begin{bmatrix} \mathbf{z}_j \\ T_{1,:}\mathbf{b} \\ T_{2,:}\mathbf{b} \\ . \\ T_{m-j,:}\mathbf{b} \end{bmatrix} \mathbf{C}_{\mathbf{ab'}}(j) \tag{D.33}$$

where $\mathbf{z}_j$ is a length j vector of zeros.

The derivatives for $\mathbf{C}_{\mathbf{ab'}}$ with respect to $\mathbf{b}^*$ are

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ab'}}(j)) = \sum_{i=1}^{m} a_i \left(T^*_{(:,i-j)} \right) \tag{D.34}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}^*_{\mathbf{ab'}}) = 0 \tag{D.35}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*} |\mathbf{C}_{\mathbf{ab'}}(j)|^2 = \sum_{i=1}^{m} a_i \left(T^*_{(:,i-j)} \right) \mathbf{C}^*_{\mathbf{ab'}}(j) \tag{D.36}$$

<u>For lag $j < 0$</u>
The derivatives for $\mathbf{C}_{\mathbf{ab'}}$ with respect to $\mathbf{a}^*$ are

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}^*_{\mathbf{ab'}}(j)) = 0 \tag{D.37}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}^*_{\mathbf{ab'}}(j)) = \begin{bmatrix} T_{|j|+1,:}\ \mathbf{b} \\ \vdots \\ T_{m,:}\ \mathbf{b} \\ \mathbf{z}_j \end{bmatrix} \tag{D.38}$$

so that

$$\frac{\partial}{\partial \mathbf{a}^*} |\mathbf{C}_{\mathbf{ab}'}(j)|^2 = \begin{bmatrix} T_{|j|+1,:} \, \mathbf{b} \\ \vdots \\ T_{m,:} \, \mathbf{b} \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{\mathbf{ab}'}^*(j) \tag{D.39}$$

The derivatives for $\mathbf{C}_{\mathbf{ab}'}$ with respect to $\mathbf{b}^*$ are

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ab}'}(j)) = \sum_{i=1}^{m-|j|} a_i \left(T_{i+|j|,:}^* \right) \tag{D.40}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ab}'}^*) = 0 \tag{D.41}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*} |\mathbf{C}_{\mathbf{ab}'}(j)|^2 = \sum_{i=1}^{m-|j|} a_i \left(T_{i+|j|,:}^* \right) \mathbf{C}_{\mathbf{ab}'}^*(j) \tag{D.42}$$

For $j \geq 0$

The derivatives for $\mathbf{C}_{\mathbf{ba}'}$ with respect to $\mathbf{a}^*$ are

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}_{\mathbf{ba}'}(j)) = \sum_{i=1}^{m} b_i \left(T_{i-j,:} \right)^* \tag{D.43}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = 0 \tag{D.44}$$

so that

$$\frac{\partial}{\partial \mathbf{a}^*} |\mathbf{C}_{\mathbf{ba}'}(j)|^2 = \sum_{i=1}^{m} b_i \left(T_{i-j,:}^* \right) \mathbf{C}_{\mathbf{ab}'}^*(j) \tag{D.45}$$

Similarly, the derivatives for $\mathbf{C}_{\mathbf{ba}'}$ with respect to $\mathbf{b}^*$ are

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = 0 \tag{D.46}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = \begin{bmatrix} \mathbf{z}_j \\ T_{1,:}\mathbf{a} \\ \vdots \\ T_{m-j,:}\mathbf{a} \end{bmatrix} \tag{D.47}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*} |\mathbf{C}_{\mathbf{ba}'}(j)|^2 = \begin{bmatrix} \mathbf{z}_j \\ T_{1,:}\mathbf{a} \\ \vdots \\ T_{m-j,:}\mathbf{a} \end{bmatrix} \mathbf{C}_{\mathbf{ab}'}(j) \tag{D.48}$$

For $j < 0$

The derivatives for $\mathbf{C}_{\mathbf{ba}'}$ with respect to $\mathbf{a}^*$ are

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}_{\mathbf{ba}'}(j)) = \sum_{i=1}^{m} b_i \left(T_{i+|j|,:}^* \right) \tag{D.49}$$

and

$$\frac{\partial}{\partial \mathbf{a}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = 0 \tag{D.50}$$

so that

$$\frac{\partial}{\partial \mathbf{a}^*} |\mathbf{C}_{\mathbf{ba}'}(j)|^2 = \sum_{i=1}^{m} b_i \left(T_{i+|j|,:}^* \right) \mathbf{C}_{\mathbf{ab}'}^*(j) \tag{D.51}$$

Similarly, the derivatives for $\mathbf{C}_{\mathbf{ba}'}$ with respect to $\mathbf{a}^*$ are

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = 0 \tag{D.52}$$

and

$$\frac{\partial}{\partial \mathbf{b}^*} (\mathbf{C}_{\mathbf{ba}'}^*(j)) = \begin{bmatrix} T_{|j|+1,:} \mathbf{a} \\ \vdots \\ T_{m,:} \mathbf{a} \\ \mathbf{z}_j \end{bmatrix} \tag{D.53}$$

so that

$$\frac{\partial}{\partial \mathbf{b}^*} |\mathbf{C}_{\mathbf{ba}'}(j)|^2 = \begin{bmatrix} T_{|j|+1,:}\ \mathbf{a} \\ \vdots \\ T_{m,:}\ \mathbf{a} \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{\mathbf{ba}'}(j) \tag{D.54}$$

These derivatives can be easily implemented in the optimisation algorithm. It is interesting to note that only eight basic derivatives need to be calculated; all other derivatives can be substituted from these.

Appendix E

Derivatives for Mismatched Filter Design

Consider we have two optimised codes **a** and **b**, of length n that are Doppler tolerant and have a wider mainlobe, with flat and low sidelobes of autocorrelations and all crosscorrelations.

Consider two mismatched filters **c** and **d** of length m, where $m > n$, that we want to design for each of the codes **a** and **b** respectively.

For $n = 4$, we have codes **a** and **b** as

$$\mathbf{ab} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ a_4 & b_4 \end{bmatrix} \tag{E.1}$$

Considering $m = 6$, we have mismatched filters **c** and **d** as

$$\mathbf{cd} = \begin{bmatrix} c_1 & d_1 \\ c_2 & d_2 \\ c_3 & d_3 \\ c_4 & d_4 \\ c_5 & d_5 \\ c_6 & d_6 \end{bmatrix} \tag{E.2}$$

Before computing correlations, we make size of $[\mathbf{ab}]$ equal to size of $[\mathbf{cd}]$ by appending zeros

at the start and end of the code, so that codes **a** and **b** are now

$$\mathbf{ab} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ a_4 & b_4 \\ a_5 & b_5 \\ a_6 & b_6 \end{bmatrix} \tag{E.3}$$

where $a_1 = b_1 = a_6 = b_6 = 0$.

We now compute the correlations of codes **a** and **b** with the mismatched filters **c** and **d** taht are randomly initialised. Let all possible correlations be denoted by $\mathbf{C}_{ac}$, $\mathbf{C}_{bc}$, $\mathbf{C}_{ad}$, and $\mathbf{C}_{bd}$.

At the receiver end, we want to maximise the mainlobe of $\mathbf{C}_{ac}$ and $\mathbf{C}_{bd}$, while minimising all sidelobes. We also want to minimise all $\mathbf{C}_{ad}$ and $\mathbf{C}_{bc}$. Computing the correlations and derivatives is as follows. The length of each of the above correlations will be equal to $2m - 1$, with offset j varying from $\{-m+1 \leq j \leq m-1\}$. The derivatives for various correlations are calculated as follows.

For lag $j = 0$

$$\mathbf{C}_{ac}(0) = a_1 c_1^* + a_2 c_2^* + a_3 c_3^* + a_4 c_4^* + a_5 c_5^* + a_6 c_6^* \tag{E.4}$$

and

$$\mathbf{C}_{ac}^*(0) = a_1^* c_1 + a_2^* c_2 + a_3^* c_3 + a_4^* c_4 + a_5^* c_5 + a_6^* c_6 \tag{E.5}$$

Now the derivatives of $\mathbf{C}_{ac}(0)$ and $\mathbf{C}_{ac}^*(0)$ with respect of each of the coefficients of **c** is

$$\frac{\delta}{\delta c_m^*}\left(\mathbf{C}_{ac}(0)\right) = a_m \tag{E.6}$$

$$\frac{\delta}{\delta c_m^*}\left(\mathbf{C}_{ac}^*(0)\right) = 0 \tag{E.7}$$

Note here that the derivative from (E.7) is valid for all offsets of j. The derivative of squared of correlation of $\mathbf{C}_{ac}(0)$ is then

$$\frac{\delta}{\delta \mathbf{c}^*}\left|\mathbf{C}_{ac}(0)\right|^2 = \left(\frac{\delta}{\delta \mathbf{c}^*}\mathbf{C}_{ac}^*\right)\mathbf{C}_{ac}(0) + \left(\frac{\delta}{\delta \mathbf{c}^*}\mathbf{C}_{ac}\right)\mathbf{C}_{ac}^*(0) \tag{E.8}$$

Since the first expression on the Right Hand Side of the equation is zero, (E.8) reduces to

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{ac}(0)|^2 = \left(\frac{\delta}{\delta \mathbf{c}^*} \mathbf{C}_{ac} \right) \mathbf{C}_{ac}^*(0) \tag{E.9}$$

or

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{ac}(0)|^2 = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} \mathbf{C}_{ac}^*(0) \tag{E.10}$$

For lag $j = 1$

$$\mathbf{C}_{ac}(1) = a_2 c_1^* + a_3 c_2^* + a_4 c_3^* + a_5 c_4^* + a_6 c_5^* + (0) c_6^* \tag{E.11}$$

and

$$\mathbf{C}_{ac}^*(1) = a_2^* c_1 + a_3^* c_2 + a_4^* c_3 + a_5^* c_4 + a_6^* c_5 + (0) c_6 \tag{E.12}$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{ac}(1)|^2 = \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ 0 \end{bmatrix} \mathbf{C}_{ac}^*(1) \tag{E.13}$$

Generalising for all $j \geq 0$,

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{ac}(j)|^2 = \begin{bmatrix} a_{j+1} \\ . \\ . \\ . \\ a_m \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{ac}^*(j) \tag{E.14}$$

where $\mathbf{z}_j$ is a length j vector of zeros.

The derivatives for correlations $\mathbf{C}_{\mathrm{bd}}$ are

$$\frac{\delta}{\delta \mathbf{d}^*} |\mathbf{C}_{\mathrm{bd}}(j)|^2 = \begin{bmatrix} b_{j+1} \\ . \\ . \\ . \\ b_m \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{\mathrm{bd}}^*(j) \qquad \text{(E.15)}$$

The derivatives of $\mathbf{C}_{\mathrm{ad}}$ for all offsets $j \geq 0$ are

$$\frac{\delta}{\delta \mathbf{d}^*} |\mathbf{C}_{\mathrm{ad}}(j)|^2 = \begin{bmatrix} a_{j+1} \\ . \\ . \\ . \\ a_m \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{\mathrm{ad}}^*(j) \qquad \text{(E.16)}$$

The derivatives for $\mathbf{C}_{\mathrm{bc}}$ for all offsets $j \geq 0$ are

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{\mathrm{bc}}(j)|^2 = \begin{bmatrix} b_{j+1} \\ . \\ . \\ . \\ b_m \\ \mathbf{z}_j \end{bmatrix} \mathbf{C}_{\mathrm{bc}}^*(j) \qquad \text{(E.17)}$$

For lag $j < 0$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{\mathrm{ac}}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ a_1 \\ a_2 \\ . \\ . \\ a_{m+j} \end{bmatrix} \mathbf{C}_{\mathrm{ac}}^*(j) \qquad \text{(E.18)}$$

$$\frac{\delta}{\delta \mathbf{d}^*} |\mathbf{C}_{bd}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ b_1 \\ b_2 \\ . \\ . \\ . \\ b_{m+j} \end{bmatrix} \mathbf{C}_{bd}^*(j) \qquad (E.19)$$

$$\frac{\delta}{\delta \mathbf{d}^*} |\mathbf{C}_{ad}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ a_1 \\ a_2 \\ . \\ . \\ . \\ a_{m+j} \end{bmatrix} \mathbf{C}_{ad}^*(j) \qquad (E.20)$$

$$\frac{\delta}{\delta \mathbf{c}^*} |\mathbf{C}_{bc}(j)|^2 = \begin{bmatrix} \mathbf{z}_{|j|} \\ b_1 \\ b_2 \\ . \\ . \\ . \\ b_{m+j} \end{bmatrix} \mathbf{C}_{bc}^*(j) \qquad (E.21)$$

These derivatives can be easily implemented in the optimisation algorithm. It is interesting to note that only four basic derivatives need to be calculated; all other derivatives can be substituted from these.

Appendix F

Derivatives With Respect to Complex Conjugate

In order to understand why we must take the derivative of a complex function with respect to the conjugate of the variable, and not the variable itself, we refer to [261]. We reproduce the mathematical derivation to show the derivatives with respect the variable and with respect to its conjugate. It is then proved that the desired expression can only be obtained by taking the derivative with the conjugate of the variable.

In order to differentiate an expression $f(z)$, with respect to a complex variable z, the Cauchy-Riemann equations [262, 263] have to be satisfied:

$$\frac{\partial f(z)}{\partial z} = \frac{\partial \mathbb{R}(f(z))}{\partial \Re z} + i \frac{\partial \Im(f(z))}{\partial \Re z} \tag{F.1}$$

and

$$\frac{\partial f(z)}{\partial z} = -i \frac{\partial \mathbb{R}(f(z))}{\partial \Im z} + \frac{\partial \Im(f(z))}{\partial \Im z} \tag{F.2}$$

A complex function that satisfies the Cauchy-Riemann equations for a set of points in region R is said to be an *analytic* in the region R. In general, expressions involving complex conjugate or conjugate transpose do not satisfy the Cauchy-Riemann equations. In order to avoid this, a more generalised definition of complex derivative is used.

The Generalised Complex Derivative is

$$\frac{df(z)}{dz} = \frac{1}{2} \left(\frac{\partial f(z)}{\partial \Re z} - i \frac{\partial f(z)}{\partial \Im z} \right) \tag{F.3}$$

and the Conjugate Complex Derivative is

$$\frac{df(z)}{dz^*} = \frac{1}{2}\left(\frac{\partial f(z)}{\partial \Re z} + i\, \frac{\partial f(z)}{\partial \Im z} \right) \tag{F.4}$$

The generalised complex derivative equals the normal derivative when f is an analytic function. For non-analytic function such as $f(z) = z^*$, the derivative equals zero.

The conjugate complex derivative equals zero when f is an analytic function. The conjugate complex derivative is used when deriving a complex gradient.

Also note that

$$\frac{df(z)}{dz} \neq \frac{\partial f(z)}{\partial \Re z} + i\, \frac{\partial f(z)}{\partial \Im z} \tag{F.5}$$

as it would apparently appear to be.

If f is a real function of a complex vector/ matrix z, then the complex gradient vector is given by

$$\Delta f(z) = 2\,\frac{df(z)}{dz^*} = \frac{\partial f(z)}{\partial \Re z} + i\, \frac{\partial f(z)}{\partial \Im z} \tag{F.6}$$

which is the expression that we actually desire.

Bibliography

[1] W. Avera, M. Harris, L. Bibee, S. Lingsch, and J. Sample, "Multibeam Bathymetry from a Mine-hunting Military Sonar," tech. rep., Naval Research Lab, Stennis Space Center, MS, 2002.

[2] Z. Fang, "Sonar," *International Encyclopedia of Geography: People, the Earth, Environment and Technology*, pp. 1–7, 2016.

[3] M. Ainslie, "A Century of Sonar: Planetary Oceanography, Underwater Noise Monitoring, and the Terminology of Underwater Sound," *Acoustics Today*, vol. 11, pp. 12–19, 2015.

[4] V. Voloshchenko, "The Multifrequency Sonar Equipment on the Self-action Nonlinear Effect," in *Advanced Materials*, pp. 659–668, Springer, 2016.

[5] P. Etter, "Advanced Applications for Underwater Acoustic Modeling," *Advances in Acoustics and Vibration*, vol. 2012, 2012.

[6] C. Lal, R. Petroccia, M. Conti, and J. Alves, "Secure Underwater Acoustic Networks: Current and Future Research Directions," in *Underwater Communications and Networking Conference (UComms), 2016 IEEE Third*, pp. 1–5, 2016.

[7] I. Akyildiz, P. Wang, and Z. Sun, "Realizing Underwater Communication through Magnetic Induction," *IEEE Communications Magazine*, vol. 53, no. 11, pp. 42–48, 2015.

[8] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.

[9] M. Chitre, S. Shahabudeen, L. Freitag, and M. Stojanovic, "Recent Advances in Underwater Acoustic Communications & Networking," in *IEEE/MTS OCEANS 2008-Kobe*, pp. 1–10, 2008.

[10] X. Xerandy, T. Znati, and L. Comfort, "A Cost-Effective Underwater Communication Infrastructure for Near-Field Tsunami Detection," in *Global Communications Conference (GLOBECOM), 2015 IEEE*, pp. 1–6, 2015.

[11] M. Murad, A. Sheikh, M. Manzoor, E. Felemban, and S. Qaisar, "A Aurvey on Current Underwater Acoustic Sensor Network Applications," *International Journal of Computer Theory and Engineering*, vol. 7, no. 1, p. 51, 2015.

[12] H. Lee, J. An, J. Seo, and J. Chung, "Multipath Selection Method for Maximum Ratio combining in Underwater Acoustic Channels," in *Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 519–521, 2018.

[13] P. Etter, *Underwater Acoustic Modeling and Simulation*. CRC Press, 2018.

[14] M. Stojanovic, "On the Relationship between Capacity and Distance in an Underwater Acoustic Communication Channel," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 34–43, 2007.

[15] B. Borowski, "Characterization of a Very Shallow Water Acoustic Communication Channel," in *IEEE/MTS OCEANS 2009-Biloxi*, pp. 1–10, 2009.

[16] J. Preisig, "Acoustic Propagation Considerations for Underwater Acoustic Communications Network Development," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 2–10, 2007.

[17] M. Stojanovic and J. Preisig, "Underwater Acoustic Communication Channels: Propagation Models and Statistical Characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, 2009.

[18] T. Yang, "The Effect of Internal Waves on Low-Frequency Underwater Acoustic Communications," *Journal of the Acoustical Society of America*, vol. 115, no. 5, pp. 2469–2469, 2004.

[19] S. Banerjee and M. Agrawal, "Underwater Acoustic Noise with Generalized Gaussian Statistics: Effects on Error Performance," in *IEEE/MTS OCEANS-Bergen*, pp. 1–8, 2013.

[20] D. Bertilone and D. Killeen, "Statistics of Biological Noise and Performance of Generalized Energy Detectors for Passive Detection," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 2, pp. 285–294, 2001.

[21] M. Chitre, J. Potter, and S. Ong, "Optimal and Near-Optimal Signal Detection in Snapping Shrimp Dominated Ambient Noise," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 2, pp. 497–503, 2006.

[22] W. Carey and R. Evans, *Ocean Ambient Noise: Measurement and Theory*. Springer Science & Business Media, 2011.

[23] F. Qu, Z. Wang, L. Yang, and Z. Wu, "A Journey toward Modeling and Resolving Doppler in Underwater Acoustic Communications," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 49–55, 2016.

[24] W. Chang, J. Tarng, *et al.*, "Effects of Bandwidth on Observable Multipath Clustering in Outdoor/Indoor Environments for Broadband and Ultrawideband Wireless Systems," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 4, pp. 1913–1923, 2007.

[25] L. Welch, "Lower Bounds on the Maximum Crosscorrelation of Signals (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, no. 3, pp. 397–399, 1974.

[26] J. Din, A. Yudhana, R. Hassan, *et al.*, "Target Strength for Fish Identification Using Echo Sounder," *Applied Physics Research*, vol. 1, no. 2, p. 92, 2009.

[27] K. Foote, "Fish Target Strengths for Use in Echo Integrator Surveys," *The Journal of the Acoustical Society of America*, vol. 82, no. 3, pp. 981–987, 1987.

[28] R. Love, "Measurements of Fish Target Strength: A Review," *Fish. Bull*, vol. 69, no. 4, pp. 703–715, 1971.

[29] R. Pratap, "Low Correlation Codes for Sonar Systems," Master's thesis, Victoria University of Wellington, 2016.

[30] R. Urick, *Principles of Underwater Sound for Engineers*. McGraw-Hill Education, 1967.

[31] M. Hafsteinsson and O. Misund, "Recording the Migration Behaviour of Fish Schools by Multi-beam Sonar during Conventional Acoustic Surveys," *ICES Journal of Marine Science*, vol. 52, no. 6, pp. 915–924, 1995.

[32] F. Martignac, A. Daroux, J. Bagliniere, D. Ombredane, and J. Guillard, "The use of Acoustic Cameras in Shallow waters: New Hydroacoustic Tools for Monitoring Migratory Fish Population: A Review of DIDSON Technology," *Fish and Fisheries*, vol. 16, no. 3, pp. 486–510, 2015.

[33] S. De Beukelaer, I. MacDonald, N. Guinnasso, and J. Murray, "Distinct Side-scan Sonar, RADARSAT SAR, and Acoustic Profiler Signatures of Gas and Oil seeps on the Gulf of Mexico slope," *Geo-Marine Letters*, vol. 23, no. 3-4, pp. 177–186, 2003.

[34] L. Morgan, W. C. Shanks III, D. Lovalvo, S. Johnson, W. Stephenson, K. Pierce, S. Harlan, C. Finn, G. Lee, M. Webring, *et al.*, "Exploration and Discovery in Yellowstone Lake: results from High-resolution Sonar imaging, Seismic Reflection Profiling, and Submersible Studies," *Journal of Volcanology and Geothermal Research*, vol. 122, no. 3-4, pp. 221–242, 2003.

[35] M. Ainslie, *Principles of Sonar Performance Modelling*. Springer, 2010.

[36] J. S. Bird and G. K. Mullins, "Analysis of Swath Bathymetry Sonar Accuracy," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 372–390, 2005.

[37] G. Acar and A. Adams, "Acmenet: An Underwater Acoustic Sensor Network Protocol for Real-time Environmental Monitoring in Coastal Areas," *IEE Proceedings–Radar, Sonar and Navigation*, vol. 153, no. 4, pp. 365–380, 2006.

[38] J. Damuth, R. Flood, R. Kowsmann, R. Belderson, and M. Gorini, "Anatomy and Growth Pattern of Amazon Deep-Sea Fan as revealed by Long-Range Side-Scan Sonar GLORIA and High-Resolution Seismic Studies," *AAPG Bulletin*, vol. 72, no. 8, pp. 885–911, 1988.

[39] R. Busnel, *Animal Sonar Systems*, vol. 28. Springer Science & Business Media, 2013.

[40] G. Carter, S. Mehta, and B. McTaggart, "Sonar Systems," *Advanced Signal Processing: Theory and Implementation for Sonar, Radar, and Non-Invasive Medical Diagnostic Systems*, p. 381, 2009.

[41] X. Lurton, *An Introduction to Underwater Acoustics: Principles and Applications*. Springer Science & Business Media, 2002.

[42] M. Kim, T. Lee, T. Im, and H. Ko, "The Analysis of Coherence Bandwidth and Coherence Time for Underwater Channel environments using Experimental Data in the West Sea, Korea," in *IEEE/MTS OCEANS 2016-Shanghai*, pp. 1–4, 2016.

[43] J. Butler and C. Sherman, *Transducers and Arrays for Underwater Sound*. Springer, 2016.

[44] N. Mohan and T. Undeland, *Power Electronics: Converters, Applications, and Design*. John Wiley & Sons, 2007.

[45] P. Misra, N. Kottege, B. Kusy, D. Ostry, and S. Jha, "Acoustical Ranging techniques in Embedded Wireless Sensor Networked Devices," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 1, p. 15, 2013.

[46] L. Lanbo, Z. Shengli, and C. Jun-Hong, "Prospects and Problems of Wireless Communication for Underwater Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 8, pp. 977–994, 2008.

[47] K. Mackenzie, "Discussion of Sea Water Sound-speed Determinations," *The Journal of the Acoustical Society of America*, vol. 70, no. 3, pp. 801–806, 1981.

[48] K. Mackenzie, "Nine-term Equation for Sound Speed in the Oceans," *The Journal of the Acoustical Society of America*, vol. 70, no. 3, pp. 807–812, 1981.

[49] L. Brekhovskikh, Y. Lysanov, and R. Beyer, "Fundamentals of Ocean Acoustics," *The Journal of the Acoustical Society of America*, vol. 90, no. 6, pp. 3382–3383, 1991.

[50] N. Simao, *Seismicity of the Mid-Atlantic Ridge in the MoMAR Area at a Regional Scale, Observed by Autonomous Hydrophone Arrays*. PhD thesis, Université de Bretagne Occidentale-Brest, 2009.

[51] I. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in Protocol Research for Underwater Acoustic Sensor Networks," in *Proceedings of the 1st ACM International Workshop on Underwater Networks*, pp. 7–16, ACM, 2006.

[52] M. Ainslie and J. McColm, "A Simplified Formula for Viscous and Chemical Absorption in sea water," *The Journal of the Acoustical Society of America*, vol. 103, no. 3, pp. 1671–1672, 1998.

[53] T. Gill, *The Doppler Effect*. Logos Press, 1965.

[54] M. Skolnik, "Introduction to Radar," *Radar Handbook, McGraw Hill Book Co.*, 1962.

[55] J. Alsup, "Comb Waveforms for Sonar," in *Signals, Systems, and Computers. Conference Record of the Thirty-Third Asilomar Conference on*, vol. 2, pp. 864–869, 1999.

[56] H. Cox and H. Lai, "Geometric Comb Waveforms for Reverberation Suppression," in *Signals, Systems and Computers. Conference Record of the Twenty-Eighth Asilomar Conference on*, vol. 2, pp. 1185–1189, 1994.

[57] A. Grami, *Introduction to Digital Communications*. Academic Press, 2015.

[58] D. Kilfoyle and A. Baggeroer, "The State of the Art in Underwater Acoustic Telemetry," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 4–27, 2000.

[59] R. Hodges, *Underwater Acoustics: Analysis, Design and Performance of Sonar*. John Wiley & Sons, 2011.

[60] C. McKinney and C. Anderson, "Measurements of Backscattering of Sound from the Ocean Bottom," *The Journal of The Acoustical Society of America*, vol. 36, no. 1, pp. 158–163, 1964.

[61] R. Urick, "The Backscattering of Sound from a Harbor Bottom," *The Journal of the Acoustical Society of America*, vol. 26, no. 2, pp. 231–235, 1954.

[62] A. Dowling, "Underwater Flow Noise," *Theoretical and Computational Fluid Dynamics*, vol. 10, no. 1-4, pp. 135–153, 1998.

[63] W. Lauterborn, *Cavitation and Inhomogeneities in Underwater Acoustics: Proceedings of the First International Conference, Göttingen, Fed. Rep. of Germany, July 9–11, 1979*, vol. 4. Springer Science & Business Media, 2012.

[64] V. Knudsen, R. Alford, and J. Emling, *Survey of Underwater Sound: Report No. 3 Ambient Noise*. National Defense Research Committee, Office of Scientific Research and Development, 1944.

[65] V. Knudsen, R. Alford, and J. Emling, "Underwater Ambient Noise," *J. Mar. Res.*, vol. 7, pp. 410–429, 1948.

[66] L. Berkhovskikh and Y. Lysanov, "Fundamentals of Ocean Acoustics. 1982," 1982.

[67] J. Hovem, *Marine Acoustics - The Physics of Sound in Marine Environments*. Peninsula Publishing, Los Altos Hills, CA, USA, 2010.

[68] P. Dahl, J. Miller, D. Cato, and R. Andrew, "Underwater Ambient Noise," *Acoustics Today*, vol. 3, no. 1, pp. 23–33, 2007.

[69] S.-T. Dictionary, "McGraw-Hill Dictionary of Scientific and Technical Terms," 2003.

[70] P. He, "Measurement of Acoustic Dispersion using both Transmitted and Reflected Pulses," *The Journal of the Acoustical Society of America*, vol. 107, no. 2, pp. 801–807, 2000.

[71] A. Gerlach, K. Flowers, R. Johnson Jr, W. Anderson, and E. Kunz, "Frequency Dispersion of Sound in Undersea Propagation.," tech. rep., Naval Research Lab Washington DC, 1982.

[72] R. J. Urick, *Sound Propagation in the Sea*, vol. 2. Peninsula Publishing, Los Altos Hills, CA, USA, 1982.

[73] J. W. Goodman, "Some Fundamental Properties of Speckle," *JOSA*, vol. 66, no. 11, pp. 1145–1150, 1976.

[74] J. G. Abbott and F. Thurstone, "Acoustic Speckle: Theory and Experimental Analysis," *Ultrasonic imaging*, vol. 1, no. 4, pp. 303–324, 1979.

[75] S. Bianchi and E. Giacomozzi, "Long-range Detection of Acoustic Vibrations by Speckle Tracking," *Applied Optics*, vol. 58, no. 28, pp. 7805–7809, 2019.

[76] J. Chanussot, F. Maussang, and A. Hétet, "Scalar Image Processing Filters for Speckle Reduction on Synthetic Aperture Sonar Images," in *IEEE/MTS OCEANS 2002-Biloxi*, vol. 4, pp. 2294–2301, 2002.

[77] D. Hyun, L. L. Brickson, K. T. Looby, and J. J. Dahl, "Beamforming and Speckle Reduction using Neural Networks," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 66, no. 5, pp. 898–910, 2019.

[78] M. A. Bruna, D. J. Pate, and D. Cook, "Synthetic Aperture Sonar Speckle Noise Reduction Performance Evaluation," *The Journal of the Acoustical Society of America*, vol. 143, no. 3, pp. 1856–1856, 2018.

[79] M. Dhanushree, R. Priyadharsini, and T. S. Sharmila, "Acoustic Image Denoising using various Spatial Filtering Techniques," *International Journal of Information Technology*, vol. 11, no. 4, pp. 659–665, 2019.

[80] A. Winder, "II. Sonar System Technology," *IEEE Transactions on Sonics and Ultrasonics*, vol. 22, no. 5, pp. 291–332, 1975.

[81] E. Britannica, "Sonar." https://www.britannica.com/technology/sonar, 2019. [Online; accessed 1-August-2019].

[82] J. Curie and P. Curie, "Développement par Compression de l'électricité Polaire Dans les Cristaux hémièdres à faces Inclinées," *Bulletin de minéralogie*, vol. 3, no. 4, pp. 90–93, 1880.

[83] R. Pach, J. Legutko, and P. Kulig, "History and Future Directions in Ultrasonography," *Polish Journal of Surgery*, vol. 84, no. 10, pp. 535–545, 2012.

[84] U. of Alberta, "Inventor of Sonar Ignored by History." https://phys.org/news/2008-02-inventor-sonar-history.html, Feb 2008.

[85] M. Thiery, "Seeing with Sounds: From Bats to Medical Ultrasonography," *Sarton Chair of the History of Sciences University of Ghent, Belgium, Vol 17, 2004*, p. 43, 2004.

[86] W. Arrasmith, E. Coots, J. Olson, and E. Skowbo, "Analyzing Infrasound and Seismic Signals emanating from a Waterborne System using Canonical Modeling and Analysis Methods," *International Journal of Modeling and Optimization*, vol. 4, no. 3, p. 176, 2014.

[87] J. J. Bernstein, S. L. Finberg, K. Houston, L. C. Niles, H. Chen, L. Cross, K. Li, and K. Udayakumar, "Micromachined High Frequency Ferroelectric Sonar Transducers," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 44, no. 5, pp. 960–969, 1997.

[88] A. Trokhan, "Forty Years of Underwater Acoustical Measurements at VNIIFTRI (All-Russian Institute of Physicotechmcal and Radio Engineering Measurements)," *Measurement Techniques*, vol. 38, no. 11, pp. 1276–1286, 1995.

[89] J. Gabriel, "Lippmann," *Principe de la Conservation de l'électricité ou second principe de la théorie des phénomènes électriques," "Annales de Chimie et de Physique.*, vol. 24, pp. 145–177, 1881.

[90] P. Nuyts, P. Reynaert, and W. Dehaene, "Digital Transmitter Architectures: Overview," in *Continuous-Time Digital Front-Ends for Multi-standard Wireless Transmission*, pp. 15–50, Springer, 2014.

[91] F. Raab, P. Asbeck, S. Cripps, P. Kenington, Z. Popovic, N. Pothecary, J. Sevic, and N. Sokal, "Rf and Microwave Power Amplifier and Transmitter Technologies-part 1," *High Frequency Electronics*, vol. 2, no. 3, pp. 22–36, 2003.

[92] P. Nuyts, P. Reynaert, and W. Dehaene, *Continuous-time Digital front-ends for Multistandard WirelessTtransmission.* Springer, 2014.

[93] D. Chu and T. Stanton, "Application of Pulse Compression Techniques to Broadband Acoustic Scattering by Live Individual Zooplankton," *The Journal of the Acoustical Society of America*, vol. 104, no. 1, pp. 39–55, 1998.

[94] G. Turin, "An Introduction to Matched Filters," *IRE Transactions on Information Theory*, vol. 6, no. 3, pp. 311–329, 1960.

[95] A. Verma, R. Kloser, and A. Duncan, "Potential Use of Broadband Acoustic Methods for Micronekton Classification," *Acoustics Australia*, vol. 45, no. 2, pp. 353–361, 2017.

[96] H. Blinchikoff and A. Zverev, "Filtering in the Time and Frequency Domains," *Inc, New York*, 1976.

[97] R. Heusdens, S. Engelen, P. J. Buist, A. Noroozi, P. Sundaramoorthy, C. Verhoeven, M. Bentum, and E. Gill, "Match Filtering Approach for Signal Acquisition in Radio-Pulsar Navigation," in *International Astronautical Federation (IAF)*, 10, 2012.

[98] D. Moore, *The Basic Practice of Statistics.* Palgrave Macmillan, 2010.

[99] M. Skolnik, "Introduction to Radar Systems," *New York, McGraw Hill Book Co.*, 1980.

[100] K. Foote, "Importance of the Swimbladder in Acoustic Scattering by Fish: A Comparison of Gadoid and Mackerel Target Strengths," *The Journal of the Acoustical Society of America*, vol. 67, no. 6, pp. 2084–2089, 1980.

[101] L. Rayleigh, "Xxxi. Investigations in Optics, with Special Reference to the Spectroscope," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 8, no. 49, pp. 261–274, 1879.

[102] M. Crocker, *Handbook of Acoustics*. John Wiley & Sons, 1998.

[103] M. Born and E. Wolf, "Principles of Optics, 7th (expanded) Edition," *United Kingdom: Press Syndicate of the University of Cambridge*, vol. 461, 1999.

[104] D. Chu and L. Hufnagle, "Time Varying Gain (tvg) Measurements of a Multibeam Echo Sounder for Applications to Quantitative Acoustics," in *IEEE/MTS OCEANS 2006-Boston*, pp. 1–5, 2006.

[105] A. C. Schimel, J. Beaudoin, I. M. Parnum, T. Le Bas, V. Schmidt, G. Keith, and D. Ierodiaconou, "Multibeam Sonar Backscatter Data Processing," *Marine Geophysical Research*, vol. 39, no. 1-2, pp. 121–137, 2018.

[106] D. Gabor, "Theory of Communication Part 1: The Analysis of Information," *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, vol. 93, no. 26, pp. 429–441, 1946.

[107] V. Čížek, "Numerische Hilbert-Transformation," *Proc. Inst. Radio Eng. Electron., Czechoslovak Academy of Sciences*, no. 11, 1961.

[108] X. Sun and P. Dang, "Numerical Stability of Circular Hilbert Transform and its Application to Signal Decomposition," *Applied Mathematics and Computation*, vol. 359, pp. 357–373, 2019.

[109] S. Pelzer and M. Vorländer, "Frequency-and Time-dependent Geometry for Real-time Auralizations," in *Proceedings of 20th International Congress on Acoustics, ICA*, pp. 1–7, 2010.

[110] F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for Efficient Communication in Underwater Acoustic Sensor Networks," *ACM Sigbed Review*, vol. 1, no. 2, pp. 3–8, 2004.

[111] M. Kheirabadi and M. Mohamad, "Greedy Routing in Underwater Acoustic Sensor Networks: A Survey," *International Journal of Distributed Sensor Networks*, vol. 9, no. 7, p. 701834, 2013.

[112] T. Yang, "Measurements of Temporal Coherence of Sound Transmissions through Shallow Water," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2595–2614, 2006.

[113] J. Rudander, T. Hus, P. Orten, P. van Walree, *et al.*, "Shallow-water Channel Sounding for High Speed Acoustic Communication," in *IEEE/MTS OCEANS 2017-Aberdeen*, pp. 1–8, 2017.

[114] P. van Walree, "Channel Sounding for Acoustic Communications: Techniques and Shallow-water Examples," *Norwegian Defence Research Establishment (FFI), Tech. Rep. FFI-Rapport*, vol. 7, 2011.

[115] J. Han, J. Huang, and M. Ran, "Design and Simulation of Underwater Acoustic Communication Network Based on OPENT," *Journal of System Simulation*, vol. 17, 2009.

[116] K. Kemih, M. Benslama, S. Filali, W. Liu, and H. Baudrand, "Synchronization of Chen System Based on Passivity Technique for CDMA Underwater Communication," *International Journal of Innovative Computing, Information And Control*, vol. 3, no. 5, pp. 1301–1308, 2007.

[117] P. Fan and M. Darnell, "Construction and Comparison of Periodic Digital Sequence Sets," *IEE Proceedings-Communications*, vol. 144, no. 6, pp. 361–366, 1997.

[118] Y. Han and K. Yang, "New *m*-ary Sequence Families with Low Correlation and Large Size," *IEEE Transactions on Information Theory*, vol. 55, no. 4, pp. 1815–1823, 2009.

[119] P. Kumar and O. Moreno, "Prime-phase Sequences with Periodic Correlation Properties better than Binary Sequences," *IEEE Transactions on Information Theory*, vol. 37, no. 3, pp. 603–616, 1991.

[120] P. Stoica, H. He, and J. Li, "Sequence Sets with Optimal Integrated Periodic Correlation Level," *IEEE Signal Processing Letters*, vol. 17, no. 1, pp. 63–66, 2010.

[121] M. Soltanalian, M. Naghsh, and P. Stoica, "Approaching Peak Correlation Bounds via Alternating Projections," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 5317–5312, 2014.

[122] M. Soltanalian, M. Naghsh, and P. Stoica, "On Meeting the Peak Correlation Bounds.," *IEEE Transactions on Signal Processing*, vol. 62, no. 5, pp. 1210–1220, 2014.

[123] H. He, P. Stoica, and J. Li, "On Aperiodic-correlation Bounds," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 253–256, 2010.

[124] R. Capocelli, A. DeSantis, and U. Vaccaro, *Sequences II: Methods in Communication, Security, and Computer Science*. Springer Science & Business Media, 2012.

[125] C. Ding and T. Feng, "A Generic Construction of Complex Codebooks Meeting the Welch Bound," *IEEE Transactions on Information Theory*, vol. 53, no. 11, pp. 4245–4250, 2007.

[126] J. Massey and T. Mittelholzer, "Welch's bound and Sequence Sets for Code-division Multiple-access Systems," *Sequences II: Methods in Communication, Security and Computer Sciences*, pp. 63–78, 1993.

[127] D. Sarwate, "Meeting the Welch Bound with Equality," in *Sequences and their Applications*, pp. 79–102, Springer, 1999.

[128] T. Strohmer and R. Heath, "Grassmannian Frames with Applications to Coding and Communication," *Applied and Computational Harmonic Analysis*, vol. 14, no. 3, pp. 257–275, 2003.

[129] S. Waldron, "Generalized Welch bound Equality Sequences are Tight Frames," *IEEE Transactions on Information Theory*, vol. 49, no. 9, pp. 2307–2309, 2003.

[130] P. Xia, S. Zhou, and G. Giannakis, "Achieving the Welch bound with Difference Sets," *IEEE Transactions on Information Theory*, vol. 51, no. 5, pp. 1900–1907, 2005.

[131] S. Datta, S. Howard, and D. Cochran, "Geometry of the Welch bounds," *Linear Algebra and its Applications*, vol. 437, no. 10, pp. 2455–2470, 2012.

[132] D. Sarawate, "Bounds on Crosscorrelation and Autocorrelation of Sequences (corresp.)," *IEEE Transactions on Information Theory*, vol. 25, no. 6, pp. 720–724, 1979.

[133] V. Levenshtein, "New Lower Bounds on Aperiodic Crosscorrelation of Binary Codes," *IEEE Transactions on Information Theory*, vol. 45, no. 1, pp. 284–288, 1999.

[134] P. Fan and L. Hao, "Generalized Orthogonal Sequences and their Applications in Synchronous CDMA Systems," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 83, no. 11, pp. 2054–2069, 2000.

[135] D. Li, "The Perspectives of Large Area Synchronous CDMA Technology for the Fourth – Generation Mobile Radio," *IEEE Communications Magazine*, vol. 41, no. 3, pp. 114–118, 2003.

[136] N. Suehiro, "A Signal Design without Co-channel Interference for Approximately Synchronized CDMA Systems," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 5, pp. 837–841, 1994.

[137] X. Tang and P. Fan, "A Class of Pseudonoise Sequences over GF(P) with Low Correlation Zone," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1644–1649, 2001.

[138] D. Peng and P. Fan, "New Theoretical Bounds on the Aperiodic Correlation Functions of Binary Sequences," *Science in China Series F: Information Sciences*, vol. 48, no. 1, pp. 28–45, 2005.

[139] S. Boztas, "New Lower Bounds on Aperiodic Cross-correlation of Codes over n-th roots of Unity," *Research Report 13, Department of Mathematics, Royal Melbourne Institute of Technology, Australia*, 1998.

[140] P. Z. Fan, N. Suehiro, N. Kuroyanagi, and X. M. Deng, "Class of Binary Sequences with Zero Correlation Zone," *Electronics Letters*, vol. 35, no. 10, pp. 777–779, 1999.

[141] D. Peng and P. Fan, "Bounds on Aperiodic Auto-and Cross-correlations of Binary Sequences with Low or Zero Correlation Zone," in *Parallel and Distributed Computing, Applications and Technologies, 2003. PDCAT'2003. Proceedings of the Fourth International Conference on*, pp. 882–886, 2003.

[142] H. Chen, "A: Relation between Periodic and Aperiodic Correlation functions," *The Next Generation CDMA Technologies*, 2007.

[143] H. Wen, F. Hu, and F. Jin, "Odd-Periodic and Periodic Complementary Binary Sequences," *Journal of Electronics (China)*, vol. 22, no. 1, pp. 42–46, 2005.

[144] Y. Yang, X. Tang, and G. Gong, "Even Periodic and Odd Periodic Complementary Sequence Pairs from Generalized Boolean Functions.," *Adv. in Math. of Comm.*, vol. 7, no. 2, pp. 113–125, 2013.

[145] X. Tang and P. Fan, "Bounds on Aperiodic and Odd Correlations of Spreading Sequences with Low and Zero Correlation Zone," *Electronics Letters*, vol. 37, no. 19, pp. 1201–1203, 2001.

[146] D. Peng, P. Fan, P. Fan, N. Suehiro, and M. Darnell, "Lower Bounds on the Aperiodic Correlation of LCZ and ZCZ Sequences," *Sequence Design and Applications for CDMA Systems (Southwest Jiaotong University Press, 2001, ISBN 7-81057-609/TN.264)*, 2001.

[147] D. Peng and P. Fan, "Generalised Sarwate Bounds on the Aperiodic Correlation of Sequences over Complex Roots of Unity," *IEE Proceedings-Communications*, vol. 151, no. 4, pp. 375–382, 2004.

[148] M. Rupf and J. Massey, "Optimum Sequence Multisets for Synchronous Code-division Multiple-access Channels," *IEEE Transactions on Information Theory*, vol. 40, no. 4, pp. 1261–1266, 1994.

[149] D. Peng and P. Fan, "Generalised Sarwate bounds on Periodic Autocorrelations and Crosscorrelations of Binary Sequences," *Electronics Letters*, vol. 38, no. 24, pp. 1521–1523, 2002.

[150] V. Sidelnikov, "Cross-Correlation of Sequences," *Probl Kybem (in Russian)*, vol. 24, pp. 15–42, 1971.

[151] V. Sidelnikov, "On the Mutual Correlation of Sequences," in *Dokl. Akad. Nauk SSSR*, vol. 196, pp. 531–534, 1971.

[152] P. V. Kumar and C. Liu, "On Lower Bounds to the Maximum Correlation of Complex Roots-of-Unity Sequences," *IEEE Transactions on Information Theory*, vol. 36, no. 3, pp. 633–640, 1990.

[153] D. Love, R. Heath, and T. Strohmer, "Grassmannian Beamforming for Multiple-Input Multiple-Output Wireless Systems," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2735–2747, 2003.

[154] R. Frank, "Polyphase Codes with Good Nonperiodic Correlation Properties," *IEEE Transactions on Information Theory*, vol. 9, no. 1, pp. 43–45, 1963.

[155] R. Barker, "Group Synchronisation of Binary Digital System," *Communication Theory*, pp. 273–287, 1953.

[156] R. Gold, "Optimal Binary Sequences for Spread Spectrum Multiplexing (corresp.)," *IEEE Transactions on Information Theory*, vol. 13, no. 4, pp. 619–621, 1967.

[157] S. Boztas and P. Kumar, "Binary Sequences with Gold-like Correlation but Larger Linear Span," *IEEE Transactions on Information Theory*, vol. 40, no. 2, pp. 532–537, 1994.

[158] T. Kasami, "Weight Distribution Formula for Some Class of Cyclic Codes," *Coordinated Science Laboratory Report no. R-285*, 1966.

[159] R. Heimiller, "Phase Shift Pulse Codes with Good Periodic Correlation Properties," *IRE Transactions on Information Theory*, vol. 7, no. 4, pp. 254–257, 1961.

[160] R. Frank, "Polyphase Complementary Codes," *IEEE Transactions on Information Theory*, vol. 26, no. 6, pp. 641–647, 1980.

[161] M. Schroeder, "Synthesis of Low-peak-factor Signals and Binary Sequences with Low Autocorrelation (corresp.)," *IEEE Transactions on Information Theory*, vol. 16, no. 1, pp. 85–89, 1970.

[162] D. Chu, "Polyphase Codes with Good Periodic Correlation Properties (corresp.)," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 531–532, 1972.

[163] B. Lewis and F. Kretschmer, "A New Class of Polyphase Pulse Compression Codes and Techniques," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-17, no. 5, pp. 364–372, 1981.

[164] N. Suehiro and M. Hatori, "N-shift Cross-orthogonal Sequences," *IEEE Transactions on Information Theory*, vol. 34, no. 1, pp. 143–146, 1988.

[165] S. W. Golomb, "Two-valued Sequences with Perfect Periodic Autocorrelation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 2, pp. 383–386, 1992.

[166] T. Lang and X. Chen, "Comparison of Correlation Parameters of Binary Codes for DS/CDMA Systems," in *IEEE Singapore International Conference on Communication Systems (ICCS)*, vol. 3, pp. 1059–1063, 1994.

[167] W. Alltop, "Complex Sequences with Low Periodic Correlations (corresp.)," *IEEE Transactions on Information Theory*, vol. 26, no. 3, pp. 350–354, 1980.

[168] H. Luke, "Sequences and Arrays with Perfect Periodic Correlation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 3, pp. 287–294, 1988.

[169] M. Ackroyd, "Huffman Sequences with Approximately Uniform Envelopes or Cross-correlation Functions (corresp.)," *IEEE Transactions on Information Theory*, vol. 23, no. 5, pp. 620–623, 1977.

[170] C. Ding, "Complex Codebooks from Combinatorial designs," *IEEE Transactions on Information Theory*, vol. 52, no. 9, pp. 4229–4235, 2006.

[171] P. Stoica, H. He, and J. Li, "New Algorithms for Designing Unimodular Sequences with Good Correlation Properties," *IEEE Transactions on Signal Processing*, vol. 57, no. 4, pp. 1415–1425, 2009.

[172] S. Kocabas and A. Atalar, "Binary Sequences with Low Aperiodic Autocorrelation for Synchronization Purposes," *IEEE Communications Letters*, vol. 7, no. 1, pp. 36–38, 2003.

[173] J. Li, P. Stoica, and X. Zheng, "Signal Synthesis and Receiver Design for MIMO Radar Imaging," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3959–3968, 2008.

[174] P. Stoica, H. He, and J. Li, "On Designing Sequences with Impulse-like Periodic Correlation," *IEEE Signal Processing Letters*, vol. 16, no. 8, pp. 703–706, 2009.

[175] M. Engeli, T. Ginsburg, H. Rutishauser, and E. Stiefel, *Refined Iterative Methods for Computation of the Solution and the Eigenvalues of Self-adjoint Boundary Value Problems*. Springer, 1959.

[176] F. Chen, Y. Li, and L. Lv, "Analysis of Shallow Water Sound Velocity Profile Impact on Detection Performance of Active Sonar Signal," in *IEEE/OES China Ocean Acoustics (COA)*, pp. 1–4, 2016.

[177] D. MacLennan, "Time Varied Gain Functions for Pulsed Sonars," *Journal of Sound and Vibration*, vol. 110, no. 3, pp. 511–522, 1986.

[178] J. Marszal, Z. Ostrowski, J. Schmidt, L. Kilian, A. Jedel, and A. Schmidt, "Time Variable Gain for Long Range Sonar with Chirp Sounding Signal," *Hydroacoustics*, vol. 10, pp. 117–122, 2007.

[179] M. Moszynski and A. Stepnowski, "Time-Varied-Gain Correction for Digital Echosounders," *European and Japanese Symposium on Acoustics*, vol. 33, 2012.

[180] E. Simmonds, I. Petrie, F. Armstrong, and P. Copland, "High Precision Calibration of a Vertical Sounder System for Use in Fish Stock Estimation," *Proceedings of the Institute of Acoustics*, vol. 6, no. 5, pp. 129–138, 1984.

[181] W. Au and K. Benoit-Bird, "Automatic Gain Control in the Echolocation System of Dolphins," *Nature*, vol. 423, no. 6942, p. 861, 2003.

[182] J. Baden, "Efficient Optimization of the Merit Factor of Long Binary Sequences," *IEEE Transactions on Information Theory*, vol. 57, no. 12, pp. 8084–8094, 2011.

[183] A. Boehmer, "Binary Pulse Compression Codes," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 156–167, 1967.

[184] T. Xiong and J. Hall, "Construction of Even Length Binary Sequences with Asymptotic Merit Factor 6," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 931–935, 2008.

[185] W. Remley, "Doppler Dispersion Effects in Matched Filter Detection and Resolution," *Proceedings of the IEEE*, vol. 54, no. 1, pp. 33–39, 1966.

[186] S. Blunt, K. Smith, and K. Gerlach, "Doppler-Compensated Adaptive Pulse Compression," in *IEEE International Radar Conference*, pp. 6–pp, 2006.

[187] J. Johnston and A. Fairhead, "Waveform Design and Doppler Sensitivity Analysis for Nonlinear FM Chirp Pulses," in *IEE Proceedings F (Communications, Radar and Signal Processing)*, pp. 163–175, IET, 1986.

[188] B. Lewis and F. Kretschmer, "Linear Frequency Modulation Derived Polyphase Pulse Compression Codes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-18, no. 5, pp. 637–641, 1982.

[189] J. Yang and T. Sarkar, "A New Doppler-Tolerant Polyphase Pulse Compression Codes based on Hyperbolic Frequency Modulation," in *IEEE International Radar Conference*, pp. 265–270, 2007.

[190] M. Jamil, H. Zepernick, and M. Pettersson, "Performance Assessment of Polyphase Pulse Compression Codes," in *2008 IEEE 10th International Symposium on Spread Spectrum Techniques and Applications*, pp. 166–172, 2008.

[191] P. Teal, "Tracking Wide-band Targets having Significant Doppler Shift," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 2, pp. 489–497, 2007.

[192] H. Deng, "Polyphase Code Design for Orthogonal Netted Radar Systems," *IEEE Transactions on Signal Processing*, vol. 52, no. 11, pp. 3126–3135, 2004.

[193] B. Liu, "Orthogonal Discrete Frequency-Coding Waveform Set Design with Minimized Autocorrelation Sidelobes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 4, 2009.

[194] G. Babur, O. Krasnov, A. Yarovoy, and P. Aubry, "Nearly Orthogonal Waveforms for MIMO FMCW Radar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 3, pp. 1426–1437, 2013.

[195] B. Liu, Y. Pi, X. Yang, R. Min, and W. Wu, "A Novel Approach to Improve Doppler Tolerance of Polyphase Code," in *2009 International Conference on Communications, Circuits and Systems*, pp. 558–561, 2009.

[196] F. Qazi and A. Fam, "Doppler Detection Capable Good Polyphase Code Sets based on Piecewise Linear FM," in *IEEE International Radar Conference*, pp. 0212–0217, 2014.

[197] J. Yang, Z. Qiu, W. Jiang, and X. Li, "Poly-phase Codes Optimisation for Multi-Input–Multi-Output Radars," *IET Signal Processing*, vol. 7, no. 2, pp. 93–100, 2013.

[198] F. Arlery, R. Kassab, U. Tan, and F. Lehmann, "Efficient Gradient Method for Locally Optimizing the Periodic/Aperiodic Ambiguity Function," in *IEEE International Radar Conference*, pp. 1–6, 2016.

[199] G. Cui, Y. Fu, X. Yu, and J. Li, "Local Ambiguity Function Shaping via Unimodular Sequence Design," *IEEE Signal Processing Letters*, vol. 24, no. 7, pp. 977–981, 2017.

[200] H. Esmaeili-Najafabadi, M. Ataei, and M. Sabahi, "Designing Sequence with Minimum PSL using Chebyshev Distance and its Application for Chaotic MIMO Radar Waveform Design," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 690–704, 2017.

[201] F. Wang, C. Pang, Y. Li, and X. Wang, "Designing Constant Modulus Sequences with Good Correlation and Doppler properties for simultaneous Polarimetric Radar," *Electronics*, vol. 7, no. 8, p. 153, 2018.

[202] J. Song, P. Babu, and D. Palomar, "Sequence Set Design with Good Correlation Properties via Majorization-Minimization," *IEEE Transactions on Signal Processing*, vol. 64, no. 11, pp. 2866–2879, 2016.

[203] H. He, P. Stoica, and J. Li, "Designing Unimodular Sequence Sets with Good Correlations – Including an Application to MIMO Radar," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4391–4405, 2009.

[204] F. Wang, C. Pang, Y. Li, and X. Wang, "Algorithms for Designing Unimodular Sequences with High Doppler Tolerance for Simultaneous Fully Polarimetric Radar," *Sensors*, vol. 18, no. 3, p. 905, 2018.

[205] P. Kajenski, "Mismatch Filter Design via Convex Optimization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1587–1591, 2016.

[206] M. H. Ackroyd and F. Ghani, "Optimum Mismatched Filters for Sidelobe Suppression," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-9, No 2, no. 2, pp. 214–218, 1973.

[207] F. Kretschmer and L. Welch, "Sidelobe Reduction Techniques for Polyphase Pulse Compression Codes," in *IEEE International Radar Conference*, pp. 416–421, 2000.

[208] A. Brenner, K. Eck, W. Wilhelm, and T. Noll, "Improved Resolution and Dynamic Range in Medical Ultrasonic Imaging using Depth-dependent Mismatched Filtering," in *Ultrasonics Symposium, 1997. Proceedings., 1997 IEEE*, vol. 2, pp. 1475–1480, 1997.

[209] R. Nuthalapati, "Design of Mismatched Filters for Long Binary Codes," in *IEEE International Radar Conference*, pp. 1–6, 2008.

[210] S. Haykin, *Adaptive Filter Theory*. Pearson Education India, 2008.

[211] S. Kesler and S. Haykin, "Mismatched Filtering of Sonar Signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-17,No 5, no. 5, pp. 730–734, 1981.

[212] J. Baden and M. Cohen, "Optimal Sidelobe Suppression for Biphase Codes," in *NTC'91-National Telesystems Conference Proceedings*, pp. 127–131, 1991.

[213] H. Luke, "Mismatched Filtering of Periodic Quadriphase and 8-phase Sequences," *IEEE Transactions on Communications*, vol. 51, no. 7, pp. 1061–1063, 2003.

[214] A. Fam, F. Qazi, and R. Kadlimatti, "Zero Sidelobe Aperiodic Codes via Additive-multiplicative Mismatched Filtering," in *IEEE Military Communications Conference*, pp. 829–836, 2013.

[215] O. Rabaste and L. Savy, "Mismatched Filter Optimization via Quadratic Convex Programming for Radar Applications," in *IEEE International Radar Conference*, pp. 1–6, 2014.

[216] C. Nunn, "Constrained Optimization Applied to Pulse Compression Codes and Filters," in *IEEE International Radar Conference*, pp. 190–194, 2005.

[217] A. Householder, *Principles of Numerical Analysis*. Courier Corporation, 2006.

[218] G. Venter, "Review of Optimization Techniques," *Encyclopedia of Aerospace Engineering*, 2010.

[219] R. Fletcher and M. Powell, "A Rapidly Convergent Descent Method for Minimization," *The Computer Journal*, vol. 6, no. 2, pp. 163–168, 1963.

[220] J. Nelder and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[221] J. Holland *et al.*, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT press, 1992.

[222] J. Kennedy, "Particle Swarm Optimization," *Encyclopedia of Machine Learning*, pp. 760–766, 2010.

[223] J. Jahn, "Karush–Kuhn–Tucker Conditions in Set Optimization," *Journal of Optimisation Theory and Applications*, vol. 172, no. 3, pp. 707–725, 2017.

[224] A. Cauchy, "Mémoire on a New Theory of Imaginaries, and on the Symbolic Roots of Equations and Equivalences," *CR Acad. Sci. Paris*, vol. 24, pp. 1120–1130, 1847.

[225] S. Haykin and B. Widrow, *Least-mean-Square Adaptive Filters*, vol. 31. John Wiley & Sons, 2003.

[226] S. Blackburn, "Non-overlapping Codes," *IEEE Transactions on Information Theory*, vol. 61, no. 9, pp. 4890–4894, 2015.

[227] D. Jakubisin and R. Buehrer, "Performance, Complexity, and Receiver Design for Code-aided Frame Synchronization in Multipath Channels," *IEEE Transactions on Communications*, vol. 63, no. 9, pp. 3363–3376, 2015.

[228] O. Omomukuyo, D. Chang, O. Dobre, R. Venkatesan, and T. Ngatched, "Robust Frame and Frequency Synchronization Based on Alamouti Coding for RGI-CO-OFDM," *IEEE Photonics Technology Letters*, vol. 28, no. 24, pp. 2783–2786, 2016.

[229] D. Ross, *Mechanics of Underwater Noise*. Elsevier, 2013.

[230] P. Qarabaqi and M. Stojanovic, "Statistical Characterization and Computationally Efficient Modeling of a Class of Underwater Acoustic Communication Channels," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 4, pp. 701–717, 2013.

[231] S. Barbarossa and A. Scaglione, "Theoretical Bounds on the Estimation and Prediction of Multipath Time-varying Channels," in *IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP*, vol. 5, pp. V–2545, Citeseer, 2000.

[232] C. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Prentice Hall PTR, 1992.

[233] C. Pan, L. Jia, R. Cai, and Y. Ding, "Modeling and Simulation of Channel for Underwater Communication Network," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 3, 2012.

[234] D. Fink, "Coherent Detection Signal-to-Noise," *Applied Optics*, vol. 14, no. 3, pp. 689–690, 1975.

[235] P. Woodward, *Probability and Information Theory, with Applications to Radar: International Series of Monographs on Electronics and Instrumentation*, vol. 3. Elsevier, 2014.

[236] M. B. Pursley, "Digital Communications," in *Reference Data for Engineers*, pp. 24–1, Elsevier, 2002.

[237] W. K. Chen, *The Electrical Engineering Handbook*. Elsevier, 2004.

[238] C. Therrien, "Discrete Random and Statistical Signal Processing," *International Journal of Adaptive Control and Signal Processing*, vol. 11, no. 3, pp. 256–256, 1997.

[239] W. Au, "Echolocation Signals of Wild Dolphins," *Acoustical Physics*, vol. 50, no. 4, pp. 454–462, 2004.

[240] Y. Pailhas, C. Capus, K. Brown, and Y. Petillot, "Biosonar: A Bio-mimetic Approach to Sonar Systems Concepts and Applications," in *On Biomimetics*, InTech, 2011.

[241] J. Blaxter and W. Dickson, "Observations on the Swimming Speeds of Fish," *ICES Journal of Marine Science*, vol. 24, no. 3, pp. 472–479, 1959.

[242] J. Yang and T. Sarkar, "Doppler-Invariant Property of Hyperbolic Frequency Modulated Waveforms," *Microwave and Optical Technology Letters*, vol. 48, no. 6, pp. 1174–1179, 2006.

[243] L. Kocjancic, A. Balleri, and T. Merlet, "Multibeam Radar based on Linear Frequency Modulated Waveform Diversity," *IET Radar, Sonar & Navigation*, vol. 12, no. 11, pp. 1320–1329, 2018.

[244] D. Russo and C. Bartberger, "Ambiguity Diagram for Linear FM Sonar," *The Journal of the Acoustical Society of America*, vol. 38, no. 2, pp. 183–190, 1965.

[245] P. Woodward, "Radar Ambiguity Analysis," tech. rep., Royal Radar Establishment Malvern (United Kingdom), 1967.

[246] R. McAulay and J. Johnson, "Optimal Mismatched Filter Design for Radar Ranging, Detection, and Resolution," *IEEE Transactions on Information Theory*, vol. 17, no. 6, pp. 696–701, 1971.

[247] N. Levanon and E. Mozeson, *Radar Signals*. John Wiley & Sons, 2004.

[248] Y. Wang, Y. He, J. Fu, G. Liang, and N. Zou, "Optimal Mismatched Filter Design with a Controllable Maximum Loss," in *IEEE/MTS OCEANS 2018-Charleston*, pp. 1–6, 2018.

[249] Y. Wang, Y. He, S. Sun, and J. Fu, "2d Optimal MMF Design with a Controllable Maximum Loss for Doppler Signals," *IET Radar, Sonar & Navigation*, vol. 12, no. 6, pp. 585–592, 2018.

[250] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[251] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming, Version 2.1." http://cvxr.com/cvx, March 2014.

[252] N. Pandey and L. Roy, "Convex Optimisation based Transmit Beampattern Synthesis for MIMO Radar," *Electronics Letters*, vol. 52, no. 9, pp. 761–763, 2016.

[253] L. Hu, H. Liu, D. Feng, B. Jiu, X. Wang, and S. Wu, "Optimal Mismatched Filter Bank Design for MIMO Radar via Convex Optimization," in *2010 International Waveform Diversity and Design Conference*, pp. 000126–000131, 2010.

[254] O. Rabaste and L. Savy, "Mismatched Filter Optimization for Radar Applications using Quadratically Constrained Quadratic Programs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 4, pp. 3107–3122, 2015.

[255] O. Aldayel, V. Monga, and M. Rangaswamy, "Successive QCQP Refinement for MIMO Radar Waveform Design under Practical Constraints," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3760–3774, 2016.

[256] A. De Maio, Y. Huang, D. Palomar, S. Zhang, and A. Farina, "Fractional QCQP with Applications in ML Steering Direction Estimation for Radar Detection," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 172–185, 2010.

[257] A. De Maio, Y. Huang, and M. Piezzo, "A Doppler Robust Max-Min Approach to Radar Code Design," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4943–4947, 2010.

[258] P. Delsarte and V. Levenshtein, "Association Schemes and Coding Theory," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2477–2504, 1998.

[259] Wikipedia, "Welch bounds — Wikipedia, The Free Encyclopedia." http://en.wikipedia.org/w/index.php?title=Welch%20bounds&oldid=906964640, 2019. [Online; accessed 18-September-2018].

[260] A. Cauchy, "Cours d'Analyse," *Oeuvres Completes, Bd*, vol. 3, p. 1821, 1821.

[261] K. Petersen and M. Pedersen, "The Matrix Cookbook, version: February 16, 2006," 2006.

[262] B. Riemann, *Grundlagen für Eine Allgemeine Theorie der Functionen Einer veränderlichen Complexen Grösse*. PhD thesis, EA Huth, 1851.

[263] H. Loomann, "Über die Cauchy-Riemannschen Differentialgleichungen," *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, vol. 1923, pp. 97–108, 1923.