Time Synchronization in Underwater Acoustic Sensor Networks

A Dissertation in
Electronics Engineering
by
Oriol Pallarés

Director:
Joaquín del Río

Co-Directors:
Pierre-Jean Bouvet
Antoni Mànuel

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Departament d'Enginyeria Electrònica

Time Synchronization in Underwater Acoustic Sensor Networks

Oriol Pallarés Valls

PhD Thesis
in Electronic Engineering

June 2016

Time Synchronization in Underwater Acoustic Sensor Networks

Oriol Pallarés Valls

PhD Thesis

Electronic Engineering Department

Universitat Politècnica de Catalunya

Supervisor:
Dr. Joaquín del Río

Co-Supervisors:
Dr. Pierre-Jean Bouvet
Dr. Antoni Mànuel

Thesis submitted in partial fulfillment of the requirement for the PhD Degree issued by the Universitat Politècnica de Catalunya, in its Electronic Engineering Program.

June 2016


Oriol Pallarés Valls, 2015

A copy of this PhD thesis can be downloaded from: http://www.tdx.cat/
http://upcommons.upc.edu/

Date of defense: September xx, 2016
Date of this version: June 30, 2016

Oriol Pallarés Valls
SARTI Research Group
Universitat Politècnica de Catalunya
Rambla de l'Exposició 24 Edifici C, 08800
Vilanova i La Geltrú, Spain
<oriol.pallares@upc.edu>

Abstract

This thesis deals with the development of a time synchronization algorithm for underwater sensor networks. The ease of deployment and maintenance of wireless networks leaded this research to the use of an acoustic communication sensor network to share a common base time between all nodes.

Acoustic signals are well adapted to the underwater medium but experience very challenging impairments such as Doppler, extensive multi-paths and low transmission speed that can nevertheless be corrected at the reception side.

Several acoustic waveforms can be invoked to transmit digital data through the underwater medium, without loss of generality, in this study is considered Orthogonal Frequency-Division Multiplexing (OFDM) communication scheme to exchange data between wireless underwater nodes containing sensor time references. This communication link will be used among others, because of its high data rate and its good performance in highly dispersive channels such as underwater acoustic channel,to carry time stamp message required for network synchronization.

Time synchronization is a critical piece of infrastructure of any distributed system. UWSN make extensive use of synchronized time for many services provided by a distributed network. In UWSN, Global Positioning System (GPS) signals are not available and synchronization systems are mostly based on acoustic communication. Owing to high latency of the underwater acoustic transmission channel with respect to cabled or radio network makes the use of conventional synchronization protocols even more challenging underwater.

Many time synchronization algorithms for underwater wireless sensor networks (UWSN) can be found in literature, such as TSHL, D-SYNC, DA-Sync. but only a few of them take into account all the water channel challenges, such as low available bandwidth, long propagation delays and sensor node mobility.

To solve this problem, in this research a further development of the existing time synchronization protocols found in literature is driven, yielding in a new approach to solve time synchronization problem in underwater sensor networks. To perform time synchronization we apply Precision Time Protocol (PTP) std. IEEE 1588, which is capable to synchronize two clocks with a precision below hundreds of nanoseconds in a point to point cabled Ethernet Network, and DA-Sync protocol, which is a bidirectional message exchange based method between a master clock and an slave one, and refines its time synchronization parameters by using medium kinematic models.

In cabled synchronization systems, such as PTP, time stamps are acquired in physical layer (PHY) in order to achieve maximum precision, avoiding indeterministic time like Operating System (OS) time slots or medium access protocols. Analogously, it happens in acoustic communication, time stamps are extracted from a large acquisition window, and the improvement of these time stamps is treated in this thesis.

Contrary to cable networks, the low celerity of wave sound makes underwater acoustic

communications system very sensitive to Doppler effect, yielding to non-uniform frequency scaling represented by compression or dilatation of the time axis. This frequency scaling can be induced by two factors: motion (sensor mobility, channel variation, etc...) and clock skew receiver between transmitter and receiver. Actually, in order to address this problem, some systems uses expensive inertial sensors for compensating Doppler scaling due to motion and temperature compensated low drift clocks. So in this thesis is evaluated the Doppler scaling caused by motion and skew in order to correct it.

Finally, several tests in the laboratory, test tank, and at sea are performed in order to check the performance of acoustic communication and time synchronization. Results show a correct behavior of hardware and software, and also validate the performance of the time synchronization applied to acoustic UWSN.

Keywords: OFDM, UWSN, Time synchronization, Doppler scale, PHY time-stamp.

Acknowledgements

I would like to thank a large number of people that in many different ways have helped me in the accomplishment of this PhD thesis.

In first place, I wish to thank my supervisor Doctor Joaquín del Río, for his valuable leadership, constant help and support. His enthusiasm and inspiration has always provided excellent scientific guidance to cover all the topics. I appreciate the confidence he has placed on me, and his patience in listening to all my questions and problems and his effort in revising all my work. I also would like to specially thank my co-directors Doctor Pierre-Jean Bouvet for his endless help in the resolution of acoustic communication, time synchronization tips, and his assistance in writing this work and all the publications written along three years of cooperation. And Doctor Antoni Mànuel for his confidence hiring me at SARTI research group and giving me a priceless guidance along my first steps in research field.

I would like to thank the people of SARTI research group, specially my friends Ivan Masmitjà, David Sarriá, Normandino Carreras and Julián Muñoz who directly contributed in some way in the development of this project, with whom I shared so many lunches, great laughs, and endless discussions on the most diverse topics. Thanks for sharing the enthusiasm and values of friendship and collaboration. My special thanks to David, who started with this research field in SARTI and provided me with interesting suggestions and tips on how to improve it.

I am indebted with Doctor Daniel Mihai Toma for the time he spent on reading this thesis. I would like to thank him for his valuable comments and discussions.

During the Summer of 2012 I had the chance to spend three wonderful months at the Monterey Bay Aquarium Research institute (MBARI), Moss Landing, California, USA . Thanks to Tom O'Reily, Kent Headley and Doctor Duane Edginton for making this internship possible, for their patience teaching me in microcontrollers software programming, and give me a more transversal sight of oceanic engineering applied to other fields besides electronics.

My special gratitude to the professor who put me in touch with the SARTI group, Doctor Spartacus Gomáriz of Tehcnical University of Catalunya. I cannot forget all my friends and professors I met in this university, great people and talents, who shared with me their motivation, good research, and invaluable appreciation and friendship.

Last but not least, I would like to deeply thank my family, specially my parents and sister, who supported and encouraged me from the beginning of this experience. None of this would have been possible without their help. Moltes gràcies Ramon, Mercè i Laura. This thesis is for you.

Oriol Pallarés Valls
June 30, 2016

Contents

List of Figures

List of Tables

Glossary

AWGN Additive White Gaussian Noise

BER Bit Error Rate

CFO Carrier Frequency Offset

CFR Channel Frequency Response

CIR Channel Impulse Response

CP Cyclic Prefix

DAC Digital to Analog Converter

FEC Forward Error Correction

FFT Fast Fourier Transform

FPGA Field-Programmable Gate Array

FSK Frequency Shift Keying

GPS Global Positioning System

HW Hardware

ICI Inter-Carrier Interference

IFFT Inverse Fast Fourier Transform

ISI Inter-Symbol Interference

LFM Linear Frequency Modulation

MAC Medium Access Control

MSE Mean Square Error

OFDM Orthogonal Frequency Division Multiplexing

PHY Physical layer

PSK Phase Shift Keying

PT Pure Tone

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase Shift Keying

RF Radio Frequency

RMSE Root Mean Squared Error

SNR Signal to Noise Ratio

SSB Single-Sideband

UWA Underwater Acoustics

UWAC Underwater Acoustic Channel

UWSN Underwater Sensor Network

Chapter 1

Introduction

This dissertation presents a contribution to time synchronization accuracy state of the art for underwater sensor networks. As will be discussed during the document, a key point for time synchronization in acoustic-Underwater Sensor Network (UWSN) is Doppler scaling and frame time stamping. Hence, a comparative between Doppler scale algorithms besides frame detection algorithms is developed and applied to a time synchronization protocol.

The overarching goal is to detect each algorithm best performance for different scenarios, and use this information to improve time synchronization accuracy, therefore the emphasis is on Doppler scale compensation and frame detection algorithms that can be integrated to time synchronization protocol.

The end result of this study is:

- Acoustic communication protocol enclosing time synchronization information.
- Simple solution to Medium Access Control (MAC) time stamp problem for underwater acoustic communication.
- Doppler scale estimation approaches applied to time synchronization.
- Complete time synchronization protocol for underwater sensor networks.

The research presented here draws from fields such as acoustic communication and time synchronization, obtaining information from the first field in order to improve timing accuracy. Time synchronization results obtained in this dissertation could be used as feedback to the communication protocol, since a good time synchronization accuracy between nodes in a network, allows a more efficient slotted access to the medium.

This chapter describes the motivation for developing a time synchronization algorithm for underwater sensor networks, then briefly reviews the state of the art describing other studies in the same field, in context section. The goal of the thesis follows, including specific objectives, and finally there is a section with the main contributions of this PhD thesis.

1.1 Motivation

Earth is "the water planet", ocean waters cover nearly 71 percent of Earth's surface, whereas fresh waters in lakes and rivers cover less than 1 percent. The necessity to study our oceans by marine researchers, oceanographers, marine commercial operators, off-shore oil industry and defense organizations, give rise to the necessity to communicate in the water channel.

Oceanographic studies are directly linked with the knowledge of tectonic movements, climate changes and all the fields concerning the biosphere [1]. This force us to be aware of marine research, to be able in the future to evaluate more precisely which factors mostly affect climate change, how can we preserve all kinds of life and ecosystems, which nowadays we are still discovering.

The high difficulty to deploy underwater cabled networks leads to the use of underwater wireless communications, which can be performed by an optical link, electromagnetic waves or acoustics. Optical communications requires perfect alignment between nodes and is sensitive to water turbidity [2] whereas electromagnetic waves suffer from large attenuation and are dedicated to low range applications [3]. Acoustic signals are well adapted to the underwater medium but experience very challenging impairments such as Doppler, multi-paths and low transmission speed that can nevertheless be corrected at the reception side, this makes acoustic communication very attractive and widely used in underwater scenario [4].

Underwater acoustic sensor networks have recently become a common research field in both industry and academia [5–8], starting by those early efforts by the USA around the Second World War developing first underwater acoustic submarine communication system [9]. It used analogue modulation in the 811kHz band (Single-Sideband (SSB) amplitude modulation) [10]. Research has since advanced, pushing digital modulation detection techniques into the forefront of modern acoustic communications. At present, several types of acoustic modems are available commercially enabling communication between sensors in an underwater wireless network, typically offering up to a few kilobits per second (kbps) over distances up to a few kilometers. Considerably higher bit rates have been demonstrated [11, 12].

To perform collaborative or distributed tasks in an Acoustic UWSN, such as vehicle positioning [13] or seismological networks [14], is necessary to share a common time base between all the nodes in the network.

In terrestrial networks this time synchronization can be performed by Global Positioning System (GPS) or timing protocols, where communication latency can be neglected or easily compensated. A clear example of a sensor network with GPS synchronization capability is the power efficient duty-cycling in electrical generators [15], and a widely known example for time synchronization protocol is Network Time Protocol (NTP) [16], which is used to synchronize devices through Internet, such as computers or cell phones.

Even having this time synchronization algorithms, this technology can not be directly ported to wireless UWSN, where GPS electromagnetic signals are strongly attenuated after 1 meter of water column, and communication latency is high enough to affect significantly time synchronization protocol.

Then there is a necessity to implement time synchronization protocols for acoustic-UWSN. This is what motivates the development of this dissertation, since in literature can be found several time synchronization approaches, but most of them are based on simulations or they do not take into account all the underwater communication and synchronization challenges, resulting in odd time accuracy results.

1.2 Goal of the thesis

The main goal of this dissertation is to evaluate a new approach concerning time synchronization accuracy for underwater sensor networks. To achieve this objective, the research is split in three different statements:

- Demonstrate Physical layer (PHY) time stamp performance in Underwater Acoustics (UWA) environment.

  - Characterize several frame detection algorithms performance

- Demonstrate time synchronization performance improvement when using acoustic communication physical medium information.

  - Characterize several Doppler scale compensation algorithms performance

- Characterize time synchronization algorithm performance when UWA channel correction factor's are applied.

  - Prove whole system performance in both simulation and real field tests.

To develop each one of previous statements is necessary to implement an acoustic communication system capable to enclose time stamps information, and to provide physical medium characteristics to timing algorithms.

Since a communication system is needed to reach all the objectives mentioned above, this dissertation will be divided in two parts. Part I presents acoustic communication algorithm design, and comparatives in frame detection and Doppler scale correction algorithms, due to this information is also used in communication stage. Then Part II, is dedicated to the development of time synchronization algorithm and time synchronization accuracy corrections given by information detailed in first part.

1.3 Thesis main contributions

The dissertation presents contributions to the art of time synchronization in Acoustic-UWSN. The theory of each of the contributions presented below will be explained in detail . And each contribution is demonstrated in simulation, laboratory and in the field:

- Acoustic communication protocol adapted to provide channel information for a time synchronization protocol.

- Dedicated Hardware (HW) for frame time stamp in PHY layer.

- Most used in UWSN Doppler scale correction algorithms comparison.

- Most used in UWSN frame detection algorithms comparison.

- Whole system design for time synchronization in Acoustic-UWSN

- Comparison between results in simulation, laboratory tests and real field tests

Laboratory tests are performed in a test tank filled with freshwater, test tank dimensions are 150 cm long, 40 cm tall and 40 cm width.

Field tests will be placed in a shallow water environment in Mediterranean Sea, close to Barcelona (Spain). In front of Vilanova i La Geltrú at 4 Km offshore and 20 m deep there is and underwater observatory OBSEA [17], which provides power supply and 1 Gbps connection to any underwater instrument. So it will be used for connecting our communication system to perform time synchronization tests.

1.4 Thesis structure

This chapter has provided a brief introduction to motivation and objectives of the thesis. The following chapters present each individual contribution in detail:

Chapter 1 Introduction, in this chapter are described the goals that the author pretend to accomplish along this thesis' development. Besides that, author's motivation to start working in this research field and thesis' main contributions are also explained.

Chapter 2 Acoustic communication, this chapter describes a general acoustic communication system, for doing so, first of all the state of the art is explained chronologically. Then underwater acoustic channel challenges are cited and disaggregated one by one explaining how do they affect the transmitted signal and how is it represented mathematically, this way the tranmitter and receiver modules are created to face all underwater channel challenges. Finally the real design of the acoustic communication script is explained following signal steps from transmitter to receiver.

Chapter 3 Frame detection. Once the communication module is implemented, an important part of communication is the detection of the frame, since in experimental tests, useful information will be enclosed inside of an acquisition window of raw data. Since frame detection will be also a critical part in time synchronization algorithm, a separate chapter is dedicated to this issue. Two different approaches for frame detection are described and compared to use later in time synchronization algorithm.

Chapter 4 Doppler scale compensation. In this chapter, five different Doppler scale estimators are explained and compared in order to use then as source of information for time synchronization calibration procedure. Besides transferring Doppler scale information to time synchronization mechanism, it is also used to correct frequency shifting in the main signal to be able to recover useful information.

Chapter 5 Experimental tests. This chapter contains extensive performance results based on whole chain communication described at previous chapters. It is also presented the transmitting and receiving hardware outline, which will be detailed in time synchronization part since it has been developed for improving time synchronization accuracy.

Chapter 6 Time synchronization. This section reviews the state of the art of time synchronization applied to acoustic-UWSN. It describes the structure of a common time synchronization protocol, besides some of the challenges of acoustic communication and how do they interfere synchronization protocol.

Chapter 7 Frame time stamp. This chapter contains the hardware description used for time stamping incoming frames.

Chapter 8 Time synchronization protocol. The algorithm used to provide accurate time synchronization to node's clocks is described in this chapter. In it is explained how physical layer parameters are applied to time synchronization protocol besides the calibration procedure using first order kinematic model.

Chapter 9 Experimental tests. This is one of the main contributions to the state of the art. Experimental tests showing time synchronization protocol described in previous chapter.

Chapter 10 Conclusions and future work. This chapter contains the conclusions of the work presented throughout this thesis, which is focused on the development of a time synchronization protocol for underwater sensor networks, as well as some suggestions about lines of improvement for the future.

Chapter 2

Acoustic communication

Wireless data transmission underwater over distances above 100 m relies mostly on acoustic waves [18]. Optical signals suffer from attenuation and requires perfect alignment, what makes them unsuitable for easy deployable UWSN [2]. Radio waves only propagate well at low frequencies and over short distances [3]. Hence, sound is used for underwater wireless communication for all purposes but very short distances, where it is possible to improve data rates with any of the other two technologies mentioned above.

Sound propagates underwater at low speed of approximately 1500 m/s, and propagation occurs over multiple paths leading to extreme delay spread compared to radio propagation. Moreover low propagation speed associated with sea movement creates large Doppler effect [19]. For these reasons the underwater acoustic channel is one of the most challenging communication media [20].

In this chapter is described a whole digital communication system to face mentioned Underwater Acoustic Channel (UWAC) challenges and enclose time stamping messages needed in time synchronization protocol. First a brief history and the actual state of the art will be described in section 2.1. Then an Orthogonal Frequency Division Multiplexing (OFDM) acoustic communication system is developed, section 2.2, which will be used on the one hand to transmit time stamping message required for the time synchronization algorithm in chapter 3 and on the other hand to provide clock skew information from Doppler scale estimation as described in chapter 4.

2.1 Context

The term communication derives from the Latin *comunicare* and is defined as the imparting or exchanging of information by speaking, writing or using some other medium. This dissertation describes a communication system for underwater medium, used to exchange time synchronization information.

The science of communication has a vast and rich history, and is directly linked with human technological evolution. Existing techniques for underwater wireless transmission are Radio Frequency (RF) waves, optical and acoustic waves.

2.1.1 Radio Frequency communication

RF waves are electromagnetic waves in the frequency band below 300 GHz that travel at speed of light. An electromagnetic wave is a wave of energy that has a frequency within

the electromagnetic spectrum and propagates as a periodic disturbance of the electromagnetic field when an electric charge oscillates or accelerates. Underwater radio frequency communications have been investigated since the very early days of radio [21], and received considerable attention during the 1970s. Even so, few underwater RF systems have been developed due to the highly conducting nature of sea water at high frequencies. By using low frequencies, of the order of Hz, it would be possible to communicate at larger distances, but the wave length would be of the order of Km, making practically impossible to design an antenna for this purpose. Few short range RF underwater communication modems are actually available [22] and [23].

2.1.2 Optical communication

Optical waves are electromagnetic waves that have wavelengths between 400 nm (blue light) and 700 nm (red light). Due to their short wavelength, high frequency and high speed ($3 \cdot 10^8$ m/s), optical waves are generally limited to short distances, when used as wireless communication carriers, because of its rapid absorption in water and optical scattering caused by suspended particles and plankton in significant [10]. Then, in spite of optical communication is sensitive to turbidity, it is well designed to low turbidity water like deep sea. Few commercial optical modems for underwater environments have been developed, such as Bluecomm Underwater optical modem of Sonardyne [24], because of its necessities to work at such specific channels. Besides commercial modems some research have been performed in this field such as the one conducted at MIT and the optical/acoustic modem designed by the Woods Hole Oceanographic Institution (WHOI) [25].

2.1.3 Acoustic communication

Acoustic waves propagate by means of adiabatic compression and decompression, are waves that have the same direction of vibration as their direction of travel. Due to the grater density of water, they travel five times faster in water that they do in air (1518 m/s and 343 m/s respectively at 20 C°), but are about five orders of magnitude slower than electromagnetic (EM) waves. The acoustic channel is widely used because of its relatively low attenuation, however being one of the most difficult media for wireless communication, due to frequency-dependent attenuation that affects especially the higher frequencies, noise, multi-path and non-uniform Doppler effect as will be described in section 2.2.

For more intuitive comprehension, major characteristics of acoustic, electromagnetic and optical carriers are summarized in Table 2.1.

Table 2.1: Comparison of acoustic, EM and optical waves in seawater environments [10]

	Acoustic	Electromagnetic	Optical
Nominal speed (m/s)	~ 1500	$\sim 3e8$	$\sim 3e8$
Power loss	Relatively small	Large	$\propto$ turbidity
Bandwidth	$\sim$ kHz	$\sim$ MHz	$\sim 10 - 150$ MHz
Frequency band	$\sim$ kHz	$\sim$ MHz	$\sim 10^{14} - 15^{15}$ MHz
Antenna size	~ 0.1 m	~ 0.5 m	~ 0.1 m
Effective range	$\sim$ km	~ 10 m	$\sim 10 - 100$ m

First communication efforts in underwater medium date back to the World War II for military purposes [9]. It was an underwater telephone, which was developed in 1945 by

the United States for submarine communication [5]. This device used a SSB suppressed carrier amplitude modulation with a bandwidth between 8 and 11 kHz. However, it wasn't until the development of VLSI technology [26] at the early 80's that a new generation of underwater acoustic communication systems emerged [27]. With the ability to integrate digital signal processing capabilities to the traditional circuit designs, made possible for first time to implement complex signal processing and data compression algorithms at the submerged ends of an underwater communication link [28]. This leaded a group of scientists of the Massachusetts Institute of Technology (MIT) and the Woods Hole Oceanographic Institution (WHOI) to develop a communication system based on Frequency Shift keying (FSK) modulation [29], what provides a reliable communication in noisy and reverberant offshore environment.

Since these first underwater communication systems, technology has clearly evolved to higher data throughputs or larger communication distances [5]. Making possible to transmit video in underwater acoustic communications [30] or connect UWSN such as seismological networks [14].

In telecommunications, signal modulations are chosen taking into account some parameters, such implementation complexity, supported data rate, and robustness against channel and noise effects. So recently, a worldwide convergence has occurred to use OFDM as high data rate communication technology [31] in underwater channel.

Multi-carrier modulation is an attractive alternative to single-carrier broadband modulation in channels with frequency-selective distortion, such underwater communication. It divides the total available bandwidth into many narrow sub-bands, such that the channel transfer function keeps constant (ideal) along one transmission, avoiding this way time-domain channel equalization [4].

Nowadays, many commercial acoustic modems are available using high data rates modulations. Most of them are developed in research laboratories becoming finally a commercial product. A large number of configurations are available when acquiring one of this systems, so in this section, most used commercial modems are presented and described its principal characteristics.

Table 2.2 contains acoustic modem's main characteristics. As can be seen, several frequency bands and data throughputs are available. They are designed to provide a full communication with a third-party sensor, giving a serial to acoustic link and vice-versa. But none of them provide frame time stamping information which would be really useful for developing a customized time synchronization algorithm.

An accurate time synchronization protocol, must be capable to enclose timing information in the communication layer, as well as be able to determine frame arrival exact time [32]. So this is the main lack of all commercial systems, and the reason why in this dissertation is designed an acoustic modem based on FPGA and a laptop as processor.

To address these issues, in this thesis is presented a communication system capable to inter-operate with a time synchronization algorithm, allowing this way to develop a comparison between actual time synchronization algorithms performance and improve them mixing their best characteristics.

2.2 Shallow Water Acoustic Communication

Given the main drawbacks of underwater acoustic medium and the low propagation speed of sound in water, the underwater acoustic channel is commonly regarded as one of the most challenging channels for communication. In addition to that, this dissertation study is thought to be deployed at shallow water, above 100 m deep, what causes long spread of

Table 2.2: Acoustic modems

Product name	Max. bit rate[kbps]	Range [km]	Freq. band [kHz]
Teledyne Benthos 960 [33]	15.36	6	22 - 27
WHOI Micromodem [34]	5.4	3	22.5 - 27.5
Linkquest UWN 1000 [35]	7	1.2	27 - 45
Evologics S2C R 48/78 [36]	31.2	2	48 - 78
Sercel MATS 3G 34 kHz [37]	24.6	5	30 - 39
L3 Oceania GPM-300 [38]	1	45	not specified
Tritech Micron Data Modem [39]	0.04	0.5	20 - 28
FAU Hermes [40]	87.768	0.18	262 - 375

Channel Impulse Response (CIR). Making it even more challenging to recover modulated data.

Next, principal characteristics of the underwater channel will be described as well as the channel communication model, what will be used for simulating acoustic communications.

2.2.1 Channel Characteristics and Passband Model

Communication simulations require a characterization of shallow water acoustic channel to be capable to emulate, in laboratory, a real communication between transmitter and receiver and consequently, obtain results as close to real channel as possible. So, in this section most influent challenges to underwater acoustic communication, such as slow sound velocity, propagation losses, multipath and channel noise, are presented.

Sound Velocity

Slow propagation speed of sound through water, compared to electromagnetic waves, is an important factor to take into account. The speed of sound in water depends on temperature, salinity and pressure; typically, these variables are designated when reporting sound velocity [41]. At sea level and 32% salinity, the speed of sound in water is 1518.06 m/s at 20 C°. Along last decades several sound speed equations in underwater channel have emerged, where most relevant are Del Grosso [42] and Chen and Millero [43] equations. This last equation was set by the United Nations Educational, Scientific and Cultural Organization (UNESCO) as the standard algorithm to compute sound speed in sub-aquatic medium, and is described in equation (2.1).

$$c(S,T,P) = C_w(T,P) + A(T,P)S + B(T,P)S^{3/2} + D(T,P)S^2 \tag{2.1}$$

Where the coefficients cited in the equation include a total of 42 variables, which are dependent of temperature (T), pressure (P), and salinity (S).

Under natural conditions, sound velocity within a medium is not uniform [44]. Variation in sound velocity in function of depth changes is known as Sound-Velocity Profile (SVP). An SVP is a very useful tool for being able to predict the path of propagation of sound in the ocean. A typical SVP as function of depth is shown in Figure 2.1, where three different thermocline layers produce sound speed gradient variations in function of pressure.

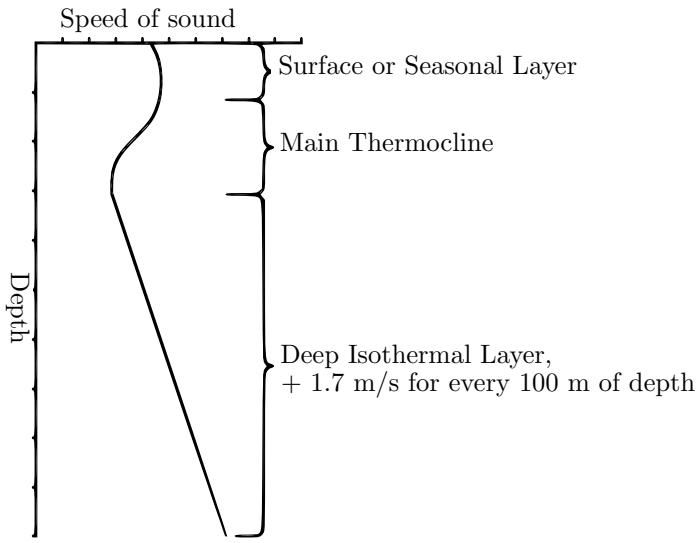


Figure 2.1: Typical Sound Velocity Profile in a water column

- Surface or seasonal layer is the top and the most variable part. Depending on the time of the day and the season, the heat from the sun will cause different water variations, making the very top to be warmer than the water below, and causing the greatest sound velocity gradient linked with temperature changes.

- Main thermocline connects the seasonal layer with the uniformly cold water of deep ocean. This layer is limited between 100 and 500 meters. And is affected by water currents caused by strong winds and waves in winter. This layer connect a warm water column and the cold deep ocean, so this profile, even suffering a gradient variation in it, is quite constant along day time and seasons.

- Deep isothermal layer only produces sound velocity variations along depth due to pressure increase, since in this layer there are not thermal changes, water temperature is nearly constant around 4 C°. Then sound speed increases 1.7 m/s per each 100 meters of depth.

Rays move into a medium which has a slower propagation speed, they tend to become more vertical as they get closer to the deep ocean, and then they bend back upwards to the main thermocline where the slower SVP is found.

In this thesis all simulations and tests are performed in a shallow water channel, where sound speed is usually constant throughout the water column. The acoustic signal propagates along straight lines only varying its direction by sea-floor, surface or thermoclines rebounds.

Propagation Loss

During acoustic waves propagation, there are three main factors of energy loss: absorptive loss, geometric spreading, and scattering loss. All of them affect in different manner acoustic wave energy during propagation between emitter and receiver, causing a power loss at the reception side, and limiting communication range.

Absorptive loss is given by energy conversion to other forms by the medium. Water channel inelasticity converts acoustic waves into heat during propagation. This energy loss is related to frequency, lower signal time period results in higher energy conversion to temperature, so the absorptive loss for acoustic wave propagation can be expressed as: $e^{\alpha(f)d}$, where d is propagation distance, and $\alpha(f)$ is the absorption coefficient at frequency f. One of the most used expressions to define absorption coefficient is Thorp's equation [45] (2.2), which is valid to frequencies up to tens of kilohertz [46].

$$\alpha = 0.11\frac{f^2}{1+f^2} + 44\frac{f^2}{4100+f} + 2.75\cdot 10^{-4} f^2 + 0.003 \tag{2.2}$$

Where α is given in dB·km^{-1} and f is signal's frequency in kHz.

Geometric spreading is the power loss due to the principle of energy conservation. Because of acoustic wave front occupies larger area coverage in function of propagation distance, energy must be divided by the same factor which multiplies the covered area. Hence, the wave energy in each unit surface becomes less and less. For instance, in a wave front cylindrical propagation, energy loss is proportional to the square of the distance, while a cylindrical dispersion is proportional to propagation distance. Note that geometric spreading is not dependent to frequency.

Scattering is a general physical process in which an incident wave is forced to deviate from a straight trajectory by one or more paths due to non-uniformities in the medium. These non-uniformities in the water can be given by non-ideal sea surfaces and bottoms, obstacles in the water column, such fish, plankton or bubble clouds. What introduces energy dispersion in spatial domain and also in the frequency domain, when these obstacles are moving.

Time varying Multipath

An acoustic signal to communicate from sender to receiver can propagate along several paths. Signals are reflected with sea-floor, sea-surface or even notorious thermoclines, a junction of these reflections can result in different paths to the receiver as shown in figure 2.2. At each reflection, the signal will suffer attenuation given by reflection losses, and at the same time path length extension will cause signal power losses. In this dissertation all tests will be performed in a shallow water environment, where signals will suffer from severe multipath.

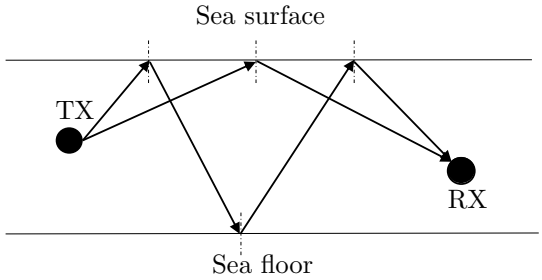


Figure 2.2: Multipath effect in a point to point communication

Each one of these N_p paths will have its own attenuation, causing different time-varying signal amplitudes $A_p(t)$, and different time varying delay $\tau_p(t)$ for each possible path at the receiver side. For instance, a transmitted passband signal going from node A to node B, defined as $\tilde{x}_{AB}(t)$, propagating along several paths, will be detected at node B as:

$$\tilde{y}_{AB}(t) = \sum_{p=0}^{N_p-1} A_p(t)\tilde{x}_{AB}(t - \tau_p(t)) \tag{2.3}$$

Ambient Noise

Noise is defined as a random fluctuation in an acoustic signal that accompany transmitted data and distorts the desired one. Acoustic noise may be given by three main sources: water motion, including also the effects of surf, rain, hail, and tides; man-made sources, including ships; and marine life [47]. Due to the multiple sources, ambient noise (ω) can be approximated as Gaussian, but not white noise. Usually four noise components are considered: turbulence, shipping, waves and thermal noise. These noise sources can be modeled with the following empirical power spectral density (p.s.d.) as a function of frequency in kHz [48] [46]:

$$\begin{aligned}
10\log N_t(f) &= 17 - 30\log f \\
10\log N_s(f) &= 40 + 20(s - 0.5) + 26\log f - 60\log(f + 0.03) \\
10\log N_w(f) &= 50 + 7.5w^{1/2} + 20\log f - 40\log(f + 0.4) \\
10\log N_{th}(f) &= -15 - 20\log f
\end{aligned} \tag{2.4}$$

where s is the shipping activity whose value ranges between 0 and 1 for low and high activity respectively whereas w is the wind speed expressed in m/s. The overall p.s.d. of the ambient noise noted $N(f)$ and expressed as the sum of the four above mentioned noise components [49]:

$$N(f) = N_t(f) + N_s(f) + N_w(f) + N_{th}(f) \tag{2.5}$$

Noise level is highly frequency-dependent. The noise power spectrum density almost monotonically decreases as frequency increases [50]. Then, when selecting a frequency band for communication, besides the frequency-dependent path loss, noise should also be taken into account [51].

Hence the received pass-band signal noted $\tilde{y}_{AB}(t)$ will be affected by the UWA channel as follows:

$$\tilde{y}_{AB}(t) = \sum_{p=0}^{N_p-1} A_p(t)\tilde{x}_{AB}(t - \tau_p(t)) + \tilde{\omega}(t) \tag{2.6}$$

Doppler effect

The last underwater channel challenge to face in this study is non-uniform Doppler scaling effect, which is induced by two factors:

- Inherent changes in the propagation medium: this produce random and path dependent varying delay (τ_p) leading to Doppler spread effect in the received signal. This effect cannot be compensated and is treated as noise.
- Relative motion between transmitter and receiver: producing a time-varying delay identical for each path thus a Doppler shift effect in the received signal that can be compensated.

Relative motion between transmitter and receiver is defined in equation (2.9). As shown in (2.6), varying delay $\tau_p(t)$ brings individual frequency shifting for each path leading to Doppler spread effect for the transmitted signal [52]. In case of motion between node A and node B, each $\tau_p(t)$ contains a time variation $a_m t$ identical for each path such [53]:

$$\tau_p(t) = \bar{\tau}_p + \delta\tau_p - a_m t \tag{2.7}$$

where $\bar{\tau}_p$ is the static delay of path p and $\delta\tau_p$ is the residual time varying delay coming from small scale fluctuation of propagation medium which is treated as random. Finally we have Doppler scale factor defined as $a_m = -\frac{v_r}{c_w}$ where v_r is the receiver (B)/ transmitter (A) relative velocity and c_w is wave celerity in water. By writing $\tau_p = \bar{\tau}_p + \delta\tau_p$, we can introduce the complex time-varying channel attenuation $h_p(t)$ such as:

$$h_p(t) = A_p(t)e^{-j2\pi f_0 \tau_p} \tag{2.8}$$

The significant variation of $h_p(t)$ comes from the phases $2\pi f_o \tau_p$ taken modulo 2π than can vary substantially in time and independently from one path to another producing Doppler spread in the received signal. With these notations equation (2.6) can be rewritten in baseband as:

$$y_{AB}(t) = \sum_{p=0}^{N_p-1} h_p(t)e^{j2\pi f_{d,m} t} x_{AB}((1+a_m)t - \tau_p) + \omega(t) \tag{2.9}$$

where $f_{d,m} = a_m f_0$, $\omega(t)$ is the baseband noise, and $\tilde{y}_{AB}(t) = \Re[y_{AB}(t)e^{j2\pi f_0 t}]$. As shown in the previous expression, motion provides a Doppler frequency shifting $f_{d,m}$ (identical for each path) and a time dilatation (or compression) by a factor $1+a_m$) in the received baseband signal. Doppler scale can be compensated at the receiver side by resampling the received baseband signal by a factor of $1/(1+a_m)$ and by compensating the phase rotation by a factor $\exp(-j2\pi f_{d,m} t)$ as detailed in [54].

Drift effect

To finish with channel characteristics and passband model, there is a last effect on the signal, which is not given by the underwater channel, but sensor clock.

Let consider two node A and B. In practice, each node has its own clock that can differ from the other one. Let's assume that node A has the master clock and B has the slave clock which has a drift of θ with respect to master clock. The time basis at B can be written as $t'_B = \theta t_A$

Then, equation (2.9) at node B can be reformulated as:

$$y_{AB}(t) = \sum_{p=0}^{N_p-1} h_p(t)e^{j2\pi f_{d,AB} t} x_{AB}(\theta(1+a_m)t - \tau_p) + \omega(t) \tag{2.10}$$

What demonstrates that Doppler scaling effect is induced by two factors: Doppler velocity of the medium and clock drift. Both of them affect the same way the received signal [55].

Where a_{AB} is the combined Doppler factor, and $f_{d,AB}$ is the Doppler shift in the received baseband signal $y_{AB}(t)$ going from node A to node B and vice versa. If A is the master node and B the slave one, the combined Doppler factor can be expressed as:

$$1 + a_{AB} = \theta(1 + a_m) \tag{2.11}$$

$$1 + a_{BA} = \frac{(1 + a_m)}{\theta} \tag{2.12}$$

Hence, combined Doppler factor affects y_{AB} signal by following expression

$$y_{AB}(t) = \sum_{p=0}^{N_p-1} h_p(t) e^{j2\pi f_{d,AB} t} x_{AB}((1 + a_{AB})t - \tau_p) + \omega(t) \tag{2.13}$$

One can easily show that the non uniform Doppler scale provides a frequency shift equal to $f_{d,AB} = a_{AB} f_0$ into the base band received signal. Therefore, Doppler scaling yields to a sampling rate deviation from the expected one in the received signal. So, if we are capable of estimating, both together, drift and Doppler shifting due to movement, it would be possible to correct $\tilde{y}_{AB}(t)$ from f_s to a new base time f'_s without Doppler scale :

$$f'_s = \frac{1}{1 + a_{AB}} f_s \tag{2.14}$$

2.2.2 Modulation Techniques for UWA Communications

Modulation is the process of varying one or more properties of a periodic waveform, called carrier signal, with a modulating signal which contains data to be transmitted, making it able to propagate over a channel.

In literature all kind of modulation techniques can be found for UWA communication, such as phase, frequency, amplitude or spread spectrum techniques. This is due to the inconsistent nature of underwater channel, that do not fix a clear optimum modulation method.

As introduced in section 2.1, first communication modulation used to communicate between submarines, was an analog carrier amplitude SSB modulation with carrier frequency between 8 and 11 kHz. This modulation technique is still in use by some submarines in large military usage and research purposes [56]. Besides SSB, in the analog modulation techniques can be found Pulse Position Modulation (PPM) [57], Amplitude Modulation (AM), Frequency modulation (FM) and Phase Modulation (PM) [58].

A common challenge to face in all UWA communication systems, is multipath, which directly affect analog modulation techniques in some way. Then, digital modulation technique can be used, since they present a major advantage in data equalization for reducing multipath effect.

Digital modulation is similar to analog one, but it can only transmit finite level of information bits. Most common single-carrier modulation systems are separated into two main groups, coherent and non-coherent digital modulation techniques. Where, in coherent modulation is needed to care about the phase at the receiving end [59]:

- Coherent digital modulation technique:
  - Phase Shift Keying (PSK), phase changes with respect to the digital information signal, keeping constant frequency.
  - Quadrature Amplitude Modulation (QAM), is a form of AM that represents digital data as variations of a carrier wave. Binary symbol 1 is transmitted as a fixed-amplitude carrier wave at a constant frequency for a bit duration of T, and binary symbol 0 is transmitted at a different fixed-amplitude, with same frequency and duration than the other symbols.
- Non-coherent digital modulation techniques:

– Frequency Shift Keying (FSK), frequency changes with respect to the digital information signal, regardless of signal phase or amplitude.

The emergent use of coherent phase-based systems in the early 90's was due to new capabilities for high-speed digital signal processing, that allowed the use of powerful receiver algorithms capable to couple a decision feedback adaptative equalizer with a second-order phase-locked loop, allowing improvements in bandwidth efficiency [60, 61].

Multi-carrier modulation is an attractive alternative to single carrier broadband modulation on channels with frequency-selective distortion. It is based on the division of the total bandwidth in small narrow-band carriers, converting the channel transfer function to an ideal response at each subband [4]. So that intersymbol interference can be less severe, which helps to simplify the receiver complexity of channel time-domain equalization.

Due to the existence of guard bands between neighboring subbands in the multicarrier approach with nonoverlapping subbands, band-pass filtering can be used to separate the signals. So, this approach is essentially a Frequency Division Multiplexing (FDM) approach. At each subband, one can adopt signaling schemes such as M-ary PSK or another modulation technique from the ones mentioned above.

To improve useful frequency band utilization, an FDM modulation with subcarrier overlapping can be also used; OFDM is one prevailing example of multicarrier modulation with overlapping subcarriers. The waveform is designed to maintain orthogonality between subcarriers along the whole communication, even propagating through long paths, avoiding the need of a time-domain equalizer [50, 62, 63]. This advantage, makes OFDM one of the most common modulations for broadband wireless applications, and will be used for this thesis development, with a QPSK modulation at each subcarrier.

2.3 OFDM Communication

This section describes OFDM waveform and how this modulation can be performed theoretically. Then, real implementation of OFDM transmitter and receiver will be explained following the ideal description.

2.3.1 OFDM Waveform

OFDM is a frequency-division multiplexing scheme used as a digital multi-carrier modulation method. FDM systems usually require a guard band between modulated subcarriers to prevent the spectrum of one subcarrier from interfering with another. These guard bands lower the system's effective information rate when compared to single carrier system with similar modulation [64].

Thus, it is necessary to receive all the subcarriers uncorrelated between them to be able to recover useful information. And it is possible by the principle of orthogonality between signals, where the dot product of two deterministic signals is equal to zero. To create an orthogonal basis set a Discrete Fourier Transform (DFT) can be used, which essentially correlates its input signal with each of the sinusoidal basis functions, and the dot product of all sinusoids of the DFT are equal to zero. Figure 2.3 shows the utilization of the available bandwidth for a 5 subcarriers OFDM signal.

Orthogonality principle requires that the subcarrier spacing is $\Delta f = \alpha/T_u$ Hz, where T_u is the useful symbol duration in seconds, and α is a positive integer (typically 1 to take advantage of the available bandwidth). Then, the total bandwidth is $B = K\Delta f$ Hz, where K is the total number of subcarriers. The baseband OFDM transmit signal can expressed as :

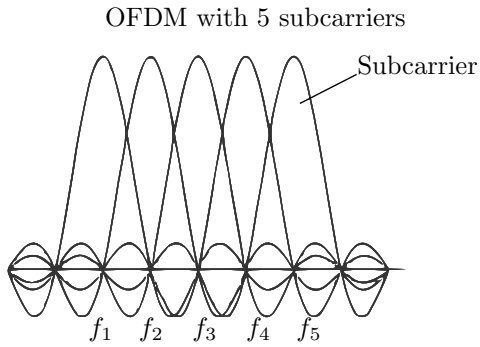


Figure 2.3: Bandwidth utilization for an OFDM signal

$$x(t) = \frac{1}{K} \sum_{k=0}^{K-1} d_k e^{j2\pi f_k t} \quad k = [0, \ldots, K-1], t \in [0, T] \tag{2.15}$$

where subcarrier frequencies f_k are defined as $f_k = (k - K/2)\Delta f$ and d_k represents the modulated frequency component.

The most important advantage of using OFDM instead of common FDM modulation, is that it allows high spectral efficiency, since almost the full available frequency band is used.

OFDM is very effective for communication over channels with frequency selective fading. It is challenging to handle frequency selective fading in the receiver, in which case, the design of the receiver is hugely complex. Then, instead of trying to handle frequency selective fading as a whole, OFDM mitigates the problem by converting the entire frequency selective fading channel into small flat fading channels, which is easier to combat by employing simple error correction and equalization schemes [65].

A major problem that results from the use of orthogonality is the need of a high accuracy frequency synchronization between transmitter and receiver. OFDM systems have low frequency deviation tolerance, due to it will cause the loss of orthogonality between subcarriers given by Inter-Carrier Interference (ICI), making it impossible to demodulate data at the receiver end.

Frequency deviations can be given by Doppler spread due to sea currents, motion, or transmitter and receiver clock impairments. Doppler shifting can be compensated at the receiver end by using specific signals robust against multipath.

2.3.2 OFDM modulation using FFT

Due to the orthogonality of OFDM subcarriers it can be implemented by a DFT, or in a more computationally efficient way by using an Fast Fourier Transform (FFT) at the receiver side, and the Inverse Fast Fourier Transform (IFFT), at the transmitter side.

At the transmitter side is used the IFFT to put raw binary data in the temporal domain, for doing so it is necessary to convert first this binary numbers into complex matrix symbols, what is done by a mapper. This process converts binary data to a time domain waveform composed by K orthogonal frequencies as shown in figure 2.4.

As it has been claimed, base-band OFDM transmit signal $x(t)$ sampled at nT_u/K can be generated by using an IFFT as follows:

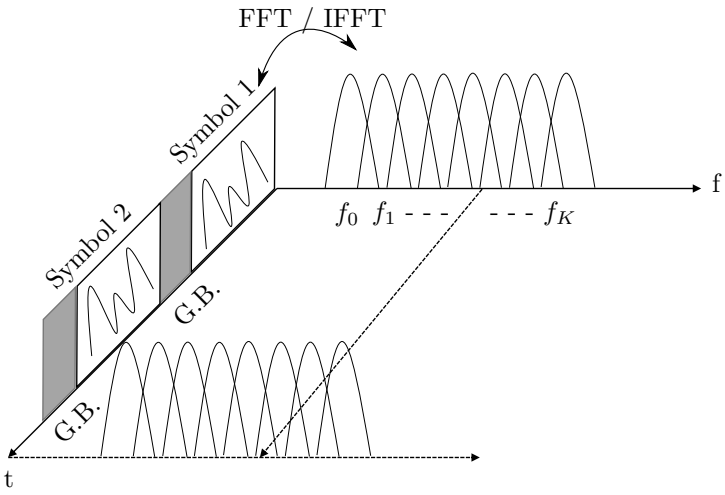


Figure 2.4: Efficient transmitter implementation using FFT

$$
\begin{aligned}
x[n] &= 1/K \sum_{k=0}^{K-1} d_k e^{j2\pi(k-K/2)\frac{n}{K}} \quad k = [0,\ldots,K-1], n = [0,\ldots,K-1] \\
&= \frac{e^{-j\pi n}}{K} \sum_{k=0}^{K-1} d_k e^{j2\pi k\frac{n}{K}} \quad k = [0,\ldots,K-1], n = [0,\ldots,K-1] \\
&= \frac{(-1)^n}{K} \sum_{k=0}^{K-1} d_k e^{j2\pi k\frac{n}{K}} \quad k = [0,\ldots,K-1], n = [0,\ldots,K-1]
\end{aligned}
\tag{2.16}
$$

Let $y[n]$ be the received discrete signal, in the absence of channel, to retrieve again the digital component noted as $\hat{d}_k$, the inverse equation must be used by invoking a direct FFT form at the receive side:

$$
\hat{d_k} = \sum_{k=0}^{K-1} (-1)^n y[n] e^{-j2\pi k\frac{n}{K}} \quad k = [0,\ldots,K-1], n = [0,\ldots,K-1] \tag{2.17}
$$

2.3.3 Cyclic prefix guard band

Given the dispersive nature of underwater channel, intersymbol interference appears in consecutive data streams. OFDM is a block transmission scheme, which partitions information symbols into blocks, and a guard interval is inserted between blocks before transmission to reduce intersymbol interference. There are two types of guard intervals: one is padding zeros at the end of each symbol, ant the other is the introduction of cyclic redundancy at the transmitter, which reduces the complexity to only FFT processing and one tap scalar equalization at the receiver.

In this work has been used a cyclic redundancy prior to each symbol, noted as Cyclic Prefix (CP), since it is more robust to noise by enlarging the number of useful samples to

be processed [64], figure 2.5. The derivations for zero padded OFDM can be carried out similarly.

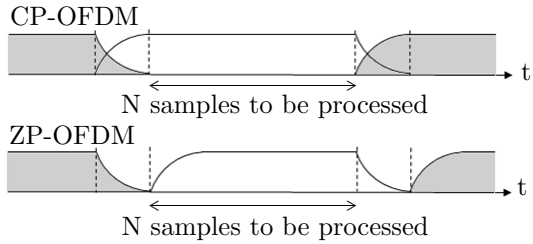


Figure 2.5: Received signal for CP-OFDM and ZP-OFDM

2.4 OFDM transmitter

The implemented OFDM system defined in figure 2.6 is composed by both: transmitter and receiver, separated by the channel described above (Subsection-2.2.1). In this section are described each one of the parts of the transmitter block diagram, from binary raw data to temporal OFDM modulated signal.

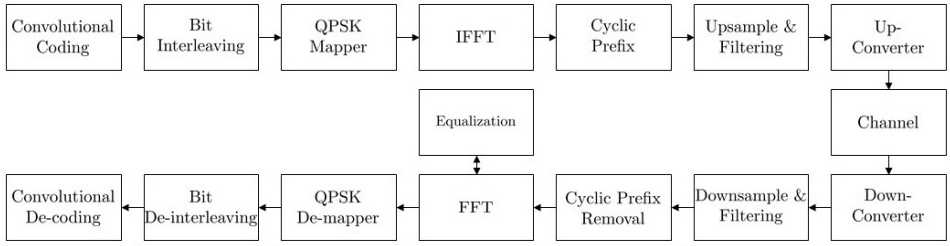


Figure 2.6: OFDM communication block diagram

Here are also taken into account communication parameters considerations to make it feasible for real tests, as well as simulation configuration.

2.4.1 Convolutional encoder

To control transmission errors the system is provided of a Forward Error Correction (FEC) algorithm. This is necessary to avoid packet retransmissions due to errors, what is a problem in long communication delays environments.

The central idea is the sender encodes the message in a redundant way by using an error-correcting code. The redundancy allows the receiver to detect a limited number of errors that may occur anywhere in the message, and often to correct these errors without any retransmission.

The cost of using FEC algorithm is a fixed higher forward channel bandwidth, or useful data transmission reduction.

In this study is used a convolutional code before the mapping, which will be decoded by a Viterbi decoder at the receiver side. And there is a two-fold approach when designing an

appropriate coding technique: first, the error correction capability, how many wrong bits can be corrected in one frame, and second, it has to provide enough coding and decoding speed so that it will not affect transmission timing.

The encoder uses a sliding window to calculate $r > 1$ parity bits by combining various subsets of bits in the window. The combining is performed by a simple exclusive-or operation. Unlike a block code, the windows overlap and slide by one at each movement. The size of the window, in bits, is called the code's constraint length (K_c), shown in the rectangular window of figure 2.7. The longer the constraint length, the larger the number of parity bits that are influenced by any given message bit.

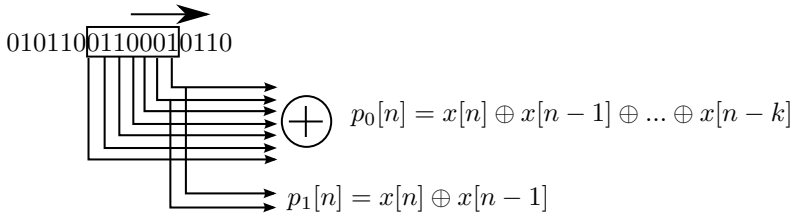


Figure 2.7: Convolutional code with two parity bits per message ($r = 2$), constraint length $K_c = 7$ and generator polynom: $(133, 171)_o$.

A larger constraint length implies a greater resilience to bit errors. Although the trade-off is that it will take way longer to decode codes of long constraint length as it will be presented in section 2.5, so one cannot increase the constraint length arbitrarily and expect fast decoding [66].

2.4.2 Bit Interleaver

Bit interleaving is another FEC approach besides convolutional coding. It is commonly used in digital communication and digital storage systems to improve performance of FEC codes. Most of communication channels are not memoryless, what means that errors typically occur in bursts rather than independently. Then, as mentioned in previous subsection, if the number of errors within a code word exceeds the error-correcting code's capability, it fails to recover the original message.

For avoiding error bursts within a code word, a block interleaver accepts a set of symbols and rearranges them. Symbol permutation is performed according to a mapping, then a corresponding deinterleaver uses the inverse mapping to restore the original sequence of symbols.

In this study is used a random interleaver block, which chooses a permutation table randomly using an initial seed parameter provided by the user in the block mask. By using the same initial seed value in the corresponding random deinterleaver block, it is possible to restore the permuted symbols to their original ordering.

Following example describes how a random interleaver can help FEC algorithms by spreading a burst error along several symbols,avoiding this way exceeding the error-correcting code's capacity:

Transmitted sentence:	ThisIsAnExampleOfInterleaving...
Error-free transmission:	TIEpfeaghsxlIrv.iAaenli.snmOten.
Received sentence with a burst error:	TIEpfe_____Irv.iAaenli.snmOten.
Received sentence after deinterleaving:	T_isI_AnE_amp_eOfInterle_vin_...

At the received sentence after deinterleaving one can observe how the burst error can be easily faced by a pair of convolutional coder and decoder.

2.4.3 QPSK Mapper

The modulation chosen in the current research is a Quadrature Phase Shift Keying (QPSK), which is a digital modulation scheme that conveys data by changing the phase of a reference signal or carrier wave. Four constellation points are used in this study, positioned 90° of angular spacing around a circle, with 2 bits assigned to each point (see constellation points in Fig. 2.8).

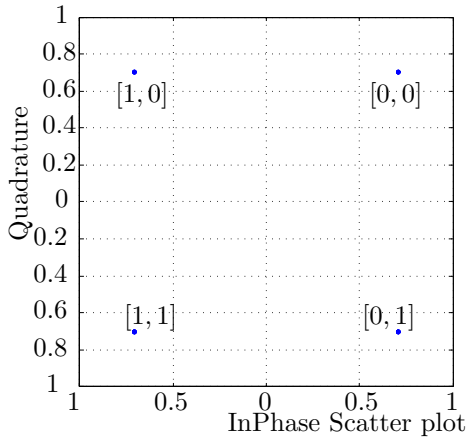


Figure 2.8: Ideal transmitted Q-PSK constellation. Labels of the constellation consists of providing the equivalence between complex symbols and bits, 01 is coded by 0.707-0.707j

Since the aim of this study is not communication modulation, but time synchronization, the chosen constellation is a low density one. In practice, as higher is the density of points in the constellation, higher is the probability that the symbols are wrongly detected in the receiver due to phase shift and amplitude scaling of the complex points after passing through the channel, so a QPSK mapper result in a good trade-off between complexity and robustness against underwater acoustic communications.

2.4.4 IFFT

Following Fig. 2.3 block diagram, after FEC algorithms and QPSK mapping, the OFDM blocks in temporal domain are obtained using the IFFT algorithm as presented in subsection 2.3.2.

An OFDM system treats the source symbols QPSK at the transmitter as though they are in the frequency-domain. These symbols are used as the inputs to an IFFT block that brings the signal into the time-domain.

Each input symbol acts like a complex weight, given by quadrature and phase from QPSK modulation, for the corresponding sinusoidal basis function. Since the input symbols are complex, the value of the symbol determines both the amplitude and phase of the sinusoid for that subcarrier.

The IFFT takes in N_{FFT} symbols at a time where N_{FFT} is the number of subcarriers in the system, which in this study is set to 512. More complex modulation mappings or higher N_{FFT} symbols could be used for enhancing communication performance in terms of throughput, but this work centers its focus to time synchronization improvements, so communication has been developed easy and fast to develop, making it strong enough for the underwater acoustic channel. All the algorithms presented in this thesis can be directly ported to higher modulations or bandwidth usage.

For practical implementation must be taken into account narrow filtering effect at the reception side, what can cause a loss of initial or ending subcarriers of the IFFT block, so for our design are only used 90 % of carriers which yields in $K = 460$ active carriers. Then zero carriers are added at the beginning and the end of the symbol to the useful carriers K in order to keep the sum of carriers N_{FFT} as 512.

As it may be expected, oversampling is required in order to assure the correct demodulation of the OFDM blocks. To correctly count for the oversampling, the IFFT algorithm does not operate with a number of samples N_{FFT}, but it operates with a number of samples $N = RN_{FFT}$, where R is the oversampling ratio and $(R-1)k$ zeros have been appended to the N_{FFT} subcarriers' symbols.

Then, the IFFT algorithm performs the following equation:

$$x[n] = x(n/f_s) = \frac{(-1)^n}{N} \sum_{k=0}^{N-1} \breve{X}(k) e^{j2\pi nk/N} \quad n = [0, \ldots, N-1] \tag{2.18}$$

where $\breve{X}[k]$ represents the k-th element of a vector of size N built from K useful subcarriers $X[k]$ as shown in figure 2.9

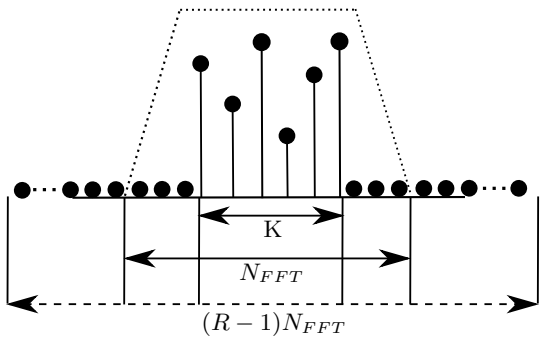


Figure 2.9: Vector construction for useful spectrum and oversampling.

The sampling frequency f_s is linked to the useful symbol duration T_u as follows :

$$T_u = \frac{N}{f_s} \tag{2.19}$$

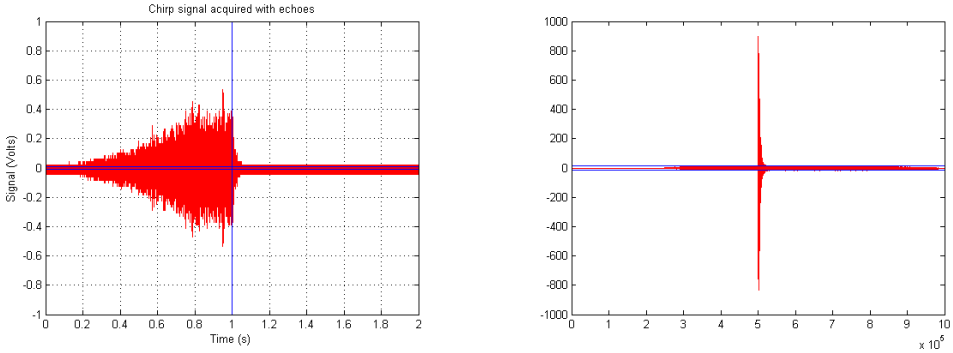
2.4.5 Cyclic Prefix

At the output of the IFFT block, a guard interval of N_{CP} samples is inserted at the beginning of each block. Since IFFT has performed an oversampling with ratio R, so cyclic prefix (CP) also has to take into account this sampling ratio. Then, CP has to cover an interval of $R \cdot N_{CP}$ samples.

For a proper scaling of CP length must be known the CIR. Multipath effect is directly linked to echoes stabilization, so before an OFDM symbol transmission previous transmission echoes have to be canceled or at least significantly reduced to do not interfere along each symbol. In open water multipath effect is only caused by thermoclines, so CIR is small and CP length takes only a tinny portion of the whole symbol in time domain. But for our study besides simulation we will work in a small test tank and shallow water environments, where CP can be composed by $\frac{1}{4}$ part of the useful OFDM symbol.

More restrictive CIR in our study is given by the test tank of dimensions 1m $\times$ 80cm $\times$ 60 cm, which has been estimated in 75 ms of reflection time spread. In order to estimate the CIR time spread, a chirp signal is transmitted through the acoustic water channel and is processed the receive side with his matched filter. The good autocorrelation properties of chirp signal provide a good estimation of the CIR span [67] as shown in figure 2.10

Right figure plots the received signal correlated with the sent chirp and in blue is shown the power of the received noise in order to estimate when the signal is completely extinguished. For calculating the exact point where the red signal crosses the blue signal, is performed the calculus of the envelope of the correlation and it provides the exact crossing point with the blue line.



(a) (b)

Figure 2.10: (a) Chirp signal reception. (b) Chirp correlation.

Difference between the maximum peak and the envelope versus noise level crossing is set in 75 ms at the test tank.

2.4.6 Upconverter

Upsampling process has been performed efficiently in IFFT and CP stages.

The generated baseband OFDM signals need to be shifted in frequency to the desired frequency band [4]. Furthermore, communication system hardware presents bandwidth limitations given by the transducer-hydrophone that must be taken into account.

Based on experimental tests using two cylindrical hydrophones developed by SARTI-UPC research group [68] our communication will be centered around 30 kHz with a total useful bandwidth of 20 kHz fixed by the hardware designed at section II.

For frequency adjustment from OFDM base-band signal center frequency to $f_0 = 30$ kHz pass-band frequency, Euler's polar form is used to perform the operation.

Given the base-band OFDM signal composed by all transmitter mentioned parts; Centered around f_0, duration of each OFDM data symbol is noted T_{ofdm} and can be decomposed into an useful part of length $T_u = N_{FFT}T_{sym}$ and CP part of duration $T_{CP} = L_{CP}T_{sym}$ where T_{sym} denotes duration of a complex cell. In each OFDM symbol, $K \leqslant N_{FFT}$ active carriers are modulated by using QPSK constellation. The inter-carrier spacing is computed as $\Delta f = \frac{1}{T_u}$ and signal bandwidth is $B_{OFDM} = \frac{K}{T_u}$. The fraction of CP is set according to the maximum delay spread of the underwater acoustic channel. The time domain modulated OFDM signal can be expressed as:

$$\tilde{x}(t) = \Re\Big(\frac{1}{K}\sum_{k=0}^{K-1} d_k e^{j2\pi(f_0+f_k)t}\Big) \quad k = [0,\dots,K-1], t \in [0, T_u] \tag{2.20}$$

With $f_k = (k - K/2)\Delta f$.

2.5 OFDM receiver

Transmitted signal is sent to the underwater acoustic channel where it will be modified by all the effects described in section 2.2.1, so the receiver has to be capable to clean up the signal from external noises as well as recover multiplexed and modulated information.

The aim of this study is to design a new approach for acoustic UWSN time synchronization. There are to key factors of communication highly linked to time synchronization, Doppler scaling and frame synchronization, so for the receiver design, we assume perfect frame detection and no Doppler scaling, and later, in chapters 3 and 4 are discussed these two topics.

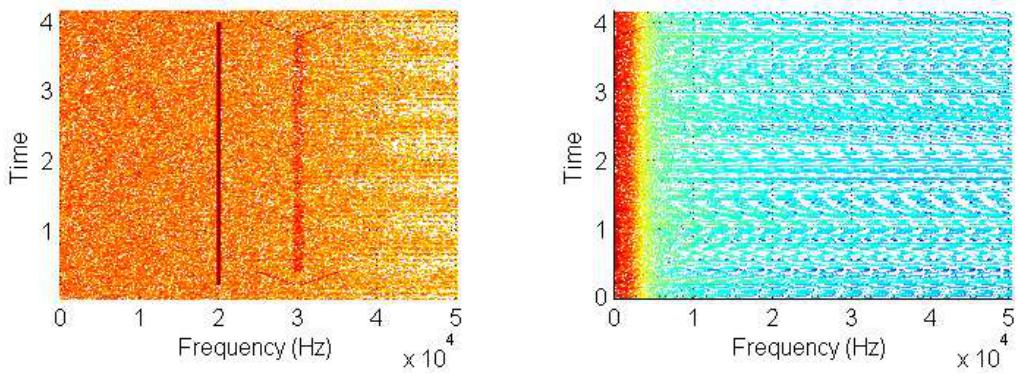
2.5.1 Downconverter, filtering and downsampling

Received signal, after communication through the acoustic channel will be distorted by noise, so we can have spectral power at non-desired frequencies. In order to delete undesired frequencies, firstly pass-band received signal is downconverted to base-band frequency, and then a low pass filter (LPF) is applied in order to work only with useful communication bandwidth.

Downconversion is performed by applying de inverse Euler polar form applied at the transmitter side followed by a LPF in order to reject high frequency harmonic components of f_0. Figure 2.11 plot the received signal centered at 30 kHz, with other signals that will be used at time synchronization part all mixed with channel noise (Figure a), then after down-conversion and filtering we can observe main communication signal centered at base-band (Figure b).

2.5.2 Cyclic Prefix removal

Cyclic Prefix is meant to decrease Inter-Symbol Interference (ISI), so once the signal is received, down-converted and filtered, guard band has to be removed for demodulating useful data. Extraction of cyclic prefix is performed after frame synchronization, which is detailed



(a) (b)

Figure 2.11: (a) Received signal. (b) Downconverted and filtered signal $y(t) := LPF\{\tilde{y}(t)e^{-j2\pi f_0 t}\}$.

in chapter 3. Same amount of samples added previously at the transmitter side have to be removed, but since it is a cyclic guard-band, them can be extracted after a time shift which can decrease signal Mean Square Error (MSE) compared to the transmitted one.

let $z[n]$ the baseband received discrete signal of length T_{OFDM} sampled at T_{sym} comprising the CP and useful signal. The useful part of the signal is extracted as :

$$y[n] = z[L_c p - \psi + n] \quad n \in [0, N_{FFT} - 1] \tag{2.21}$$

where ψ is the time-shift factor used to optimize CIR estimation. In the absence of echoes, due to the cyclic propertie of the CP, ψ can be chosen arbitrarly in the interval $[0, L_{cp} - 1]$ without degrading the equalization performance. In case of multi-path channel with pre-echoes (the main path is not the first path) and post echoes, if ψ is set too small, pre-echoes are not estimated in the CIR and performance are degraded. At the opposite side, if ψ is set to large, post echoes are not estimated and performance are also degraded. In practice the time-shift has to be set such that the CIR is centered within the CP window [69]. For our experimentation, time-shift factor is set as $\psi = L_{CIR}/4$

2.5.3 FFT & channel correction

FFT algorithm is used to retrieve the subcarriers' received symbols before channel treatment, by using the complementary method to the IFFT modulator, as shown in (2.17).

If the orthogonality of the OFDM signal is maintained and if channel delay spread is lower than the CP size, one can easily demonstrate that there is a linear relation between the transmit and the received frequency components:

$$Y[k] = H_k d_k + W[k] \quad k \in [0, K - 1] \tag{2.22}$$

where $W[k]$ is a complex valued Additive White Gaussian Noise (AWGN) and H_k is the Channel Frequency Response (CFR) component at the k-th subcarrier.

Due to the multi-path properties of the underwater acoustic channel, the CFR is frequency selective and must be compensated in order to retrieve the transmit frequency component $X[k]$. This process is called channel correction.

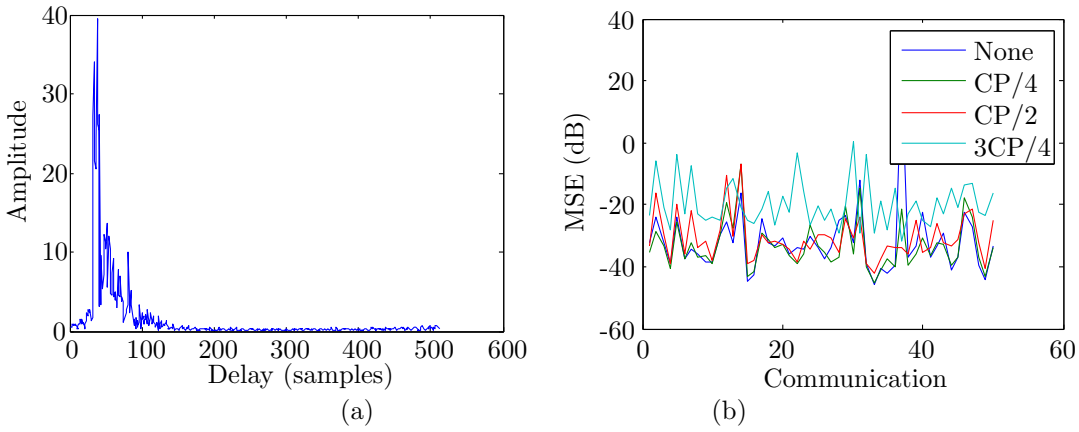


Figure 2.12: (a) Channel impulse response in function of delay. $f_s = 100$kS/s. (b) MSE in function of time shift of CP along several communication procedures.

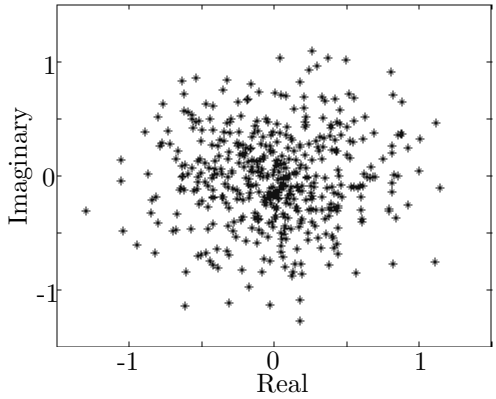


Figure 2.13: FFT constellation before frequency equalization. Simulated link 200m OFDM parameters: $B = 1333$Hz $f_0 = 30$KHz, Q-PSK, $K = 460$, $NFFT = 512$, $CP = 128$ samples

For channel correction, each frequency component $Y[k]$ has to be weighted in order to compensate the frequency fading of the channel. For doing so, Pilot tones are used to estimate at each subcarrier the CFR coefficient :

$$\hat{H}_k = \frac{Y[k]^* p_k}{|p_k|^2} \tag{2.23}$$

Where $\hat{H}_k$ is the estimated CFR at subcarrier k and p_k is a pilot tone known from the receiver.

The channel correction is thus performed as follows :

$$\hat{d}_k = \frac{\hat{H}_k^* Y[k]}{|H_k|^2]} \tag{2.24}$$

After equalization process, receiver computes again frame synchronization. Synchronization procedure can be repeated in order to achieve further improvements. The channel

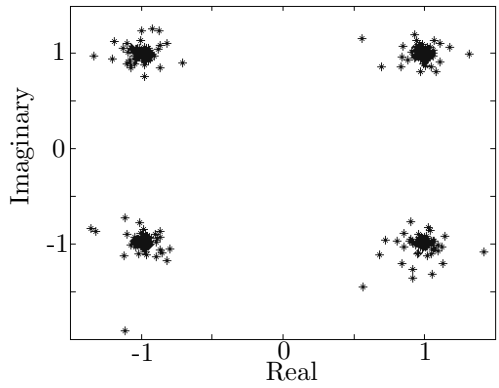


Figure 2.14: QPSK constellation after frequency equalization. Simulated link 200m OFDM parameters: $B = 1333$Hz $f_0 = 30$KHz, Q-PSK, $K = 460$, $NFFT = 512$, $CP = 128$ samples

impulse response can then be estimated again after performing frequency-offset compensation on the training symbol defined by the fine-timing estimate. It may be directly used or further processed for employment in channel equalization [70].

2.5.4 QPSK De-mapper

After equalization process, as shown in figure 2.14, is possible to convert information to a matrix representation, by setting both real and imaginary threshold division at 0. This decision can yield in errors in low Signal to Noise Ratio (SNR) scenarios, where point dispersion along the Q-PSK plane can be higher making possible the 0-point crossing.

To avoid false detection and improve robustness against noise by detecting an correcting burst errors, convolutional decoder associated with a bit deinterleaver is employed in our system.

In order to maximize the correction capability of the FEC decoder, the QPSK demapper provide to the decoding stage soft bits represented by Log Likelihood Ratio (LLR). Computation of LLRs from equalized frequency component can be found in [71].

2.5.5 Bit De-interleaver

According to transmitter mapping, the receiver rearranges data as it was at transmitter side before interleaving process by using the seed parameter provided by the user in the block mask.

2.5.6 Convolutional De-coding

At the receiver, we have a sequence of points to decode. These points have been extracted from a Q-PSK mapping in soft bits via LLRs, as shown in figure 2.14. Since Q-PSK points have not been digitized to a binary sequence, the decoder is a soft-decision decoding. The reason of working directly with soft bits instead of digitized binary sequence, which would be a hard-decision decoder, is because hard decision decoding makes an early decision regarding whether a bit is 0 or 1, it throws away information in the digitizing process. It might make a wrong decision, especially for points near the threshold, introducing a greater number of

bit errors in the received bit sequence. Although it still produces the most likely transmitted sequence given the received sequence, by introducing additional errors in the early digitization, the overall reduction in the probability of bit error will be smaller than with soft decision decoding. To implement this maximum likelihood decoder, in this research, a Viterbi decoder is used.

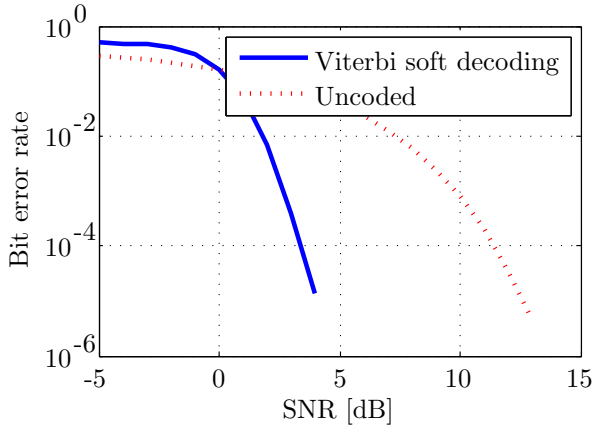


Figure 2.15: Viterbi Forward Error Correction compared to no codification at the receiver side in function of SNR. Viterbi soft-decision decoder with two parity bits per message ($r = 2$), constraint length $K = 7$, and $(133, 171)$ generator

Figure 2.15 plots a comparative between the Bit Error Rate (BER) at the reception side without applying any FEC algorithm and applying the viterby decoder, using convolutional encoder with two parity bits per message ($r = 2$), constraint length $K = 7$, and polynom generator $(133, 171)_o$.

2.6 Communication performance

In this section, communication performance assuming perfect synchronization and no Doppler scaling is evaluated. This communication protocol will be the basis for this thesis time synchronization research, so henceforth it will be used to add all the necessary algorithms related with time synchronization.

Communication performance has been evaluated using two parameters BER, which is computed after Viterbi decoding, and MSE (2.25) in simulation and real tests in a controlled test tank what emulates underwater real channel. The only way to create a channel without Doppler scaling and perfect frame synchronization is simulating it, due to it is not possible to parametrize perfectly sea currents and echoes.

$$\text{MSE} = \mathbb{E}[|d_k - \hat{d_k}|^2] \tag{2.25}$$

So first is pretended to validate a simple communication. Figure 2.16 shows communication performance for a frame containing 1 pilot for each OFDM useful data symbol.

For practical implementation several symbols can be added after each pilot, its just a matter of medium variability. In cases where the channel response do not change rapidly with time it is not necessary to provide new equalization parameters for each symbol. Figure 2.17 plots MSE evolution using one pilot and concatenating several OFDM symbols after

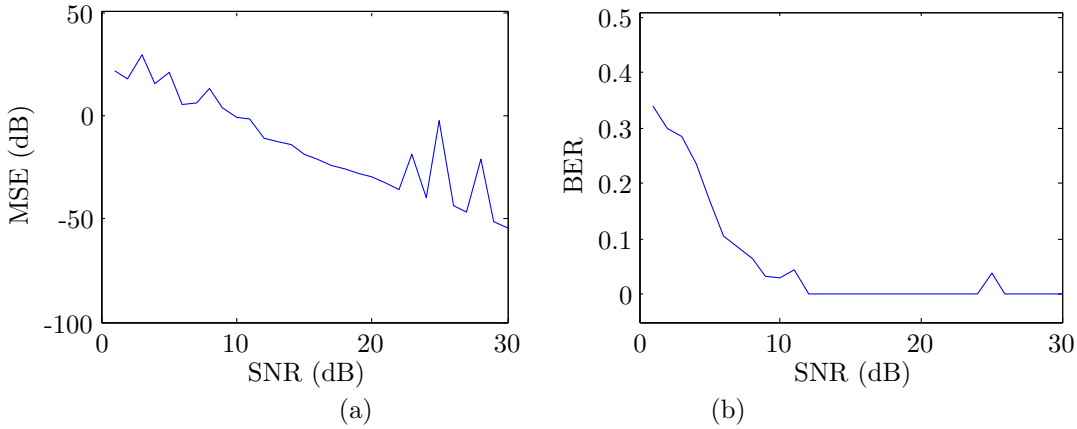


Figure 2.16: (a) MSE vs SNR for simulated acoustic communication with multipath distribution [1 0 0.5 0 0.25]. (b) BER vs SNR for simulated acoustic communication with multipath distribution [1 0 0.5 0 0.25]

it. Since it is only pretended to demonstrate communication quality worsening along time wiht 1 pilot per communication frame, it is not taken into account frequency fading or desynchronization.

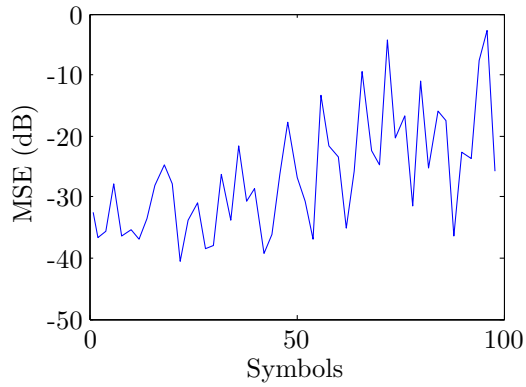


Figure 2.17: MSE evolution along a determined number of OFDM symbols concatenated after a pilot symbol. SNR= 15 dB

As it is expected as more symbols are added after one pilot, frame MSE increases. Process variability is due to it is a real test performed in a test tank with dimensions 150 cm large, 40 cm width and 40 cm tall, where we are not compensating frequency fading and each test is only performed one time due to authors purpose is only to demonstrate MSE trend.

These test verify the proper functionality of the underwater acoustic communication system which will be used hereinafter, so table 2.3 summarizes all the terms taken into account for communication matters.

Table 2.3: OFDM communication parameters

Description	Parameter	Simulation & Laboratory
Sampling frequency	f_s	100 kS/s
OFDM frequency center	f_0	30 kHz
OFDM BW	B_{OFDM}	1.19 kHz
Pure tone frequency center	f_{pt}	40 kHz
OFDM symbol duration	T_{OFDM}	480 ms
Useful part of OFDM signal	T_u	384 ms
Cyclic prefix period	T_{CP}	96 ms
FFT points	N_{FFT}	512
Active carriers	K	460
Signal to Noise Ratio	SNR	15 dB

Chapter 3

Frame Detection

One of the assumptions of previous chapter (Chapter 2) is perfect frame detection or synchronization. What means that the receiver knows perfectly when the first sample of the transmitted data is. Obviously, in real communication this is a fallacy, since the receiver has no information or trigger when the signal arrives to its PHY.

In this chapter frame detection will be addressed by studying several algorithms widely used in literature. First sample arrival estimation performance will be evaluated not only in simulation but also in laboratory real tests and shallow water field tests at OBSEA platform [17].

Since frame detection is a key point in time synchronization protocols, it is treated as an individual chapter, where it is pretended to get on to the best frame detection algorithm for this research purpose.

Symbol detection for an OFDM signal is significantly different than for a single carrier signal since there is not an "eye opening" where a best sampling time can be found. Rather there are hundreds or thousands of samples per OFDM symbol since the number of samples necessary is proportional to the number of subcarriers. Finding the symbol timing for OFDM means finding an estimate of where the symbol starts. There is usually some tolerance for symbol timing errors when a CP is used to extend the symbol [72].

To detect the arrival of the useful signal from the transmitter inside of an acquisition window, in OFDM there are two approaches: Preamble based approach and blind approach. The first method a preamble with special structure is used to detect the frame start. In this method Doppler-insensitive waveforms are usually adopted to account for channel variations. The second method is to use a cyclic structure in OFDM received signal due to CP can be used to detect the beginning of the OFDM symbol via autocorrelation of the received signal. In this method, the receiver can be ignorant of the transmitted preamble and only needs to know the preamble structure [50] additionally no overhead data is required.

- In section 3.1 (blind detection), the detection is developed for a specifically designed waveform, which consists of two OFDM symbols with similarities between its odd and even frequencies. It is based on Schmidl & Cox algorithm [72] [73].
- In section 3.2 (Preamble based detection), based on Doppler insensitive waveforms such as Linear Frequency Modulation (LFM), a cross-correlation based method for detection is presented.

In this study is chosen a preamble based approach for future developments, since we are looking for good accuracy and not overhead saving as a blind approach would provide.

3.1 Schmidl & Cox frame detection

Schmidl & Cox algorithm [72] is a complete chain for frame synchronization, making possible frame detection and Carrier Frequency Offset (CFO) compensation, which will be treated next chapter.

For frame detection is used a symbol with two identical halves in the time domain, which will remain identical after passing through the channel, except that there will be a phase difference between them caused by a frequency shifting along time.

To create two identical halves, even frequencies are filled with a pseudo-noise sequence and odd ones are filled with zeros. With these two identical halves is possible to extract an estimation of the frame arrival with a similar process to a cross-correlation of the symbol between its two halves.

If the conjugate of a sample from the first half is multiplied by the corresponding sample from the second half ($T/2$ seconds later), the effect of the channel should cancel. At the start of the fram, the products of each of these pairs of samples will have approximately the same phase, so the magnitude of the sum will be a large value.

Let's define L samples conforming one-half of symbol, and let the sum of the pairs be

$$P(d) = \sum_{m=0}^{L-1} (r^*_{d+m} r_{d+m+L}) \tag{3.1}$$

Where d is a time index corresponding to the first sample in a window of $2L$ samples.
The energy of the second half is defined by

$$R(d) = \sum_{m=0}^{L-1} |r_{d+m+L}|^2 \tag{3.2}$$

Window slides in time as the receiver searches for the training symbol. Then a timing metric can be defined as

$$M(d) = \frac{|P(d)|^2}{R(d)^2} \tag{3.3}$$

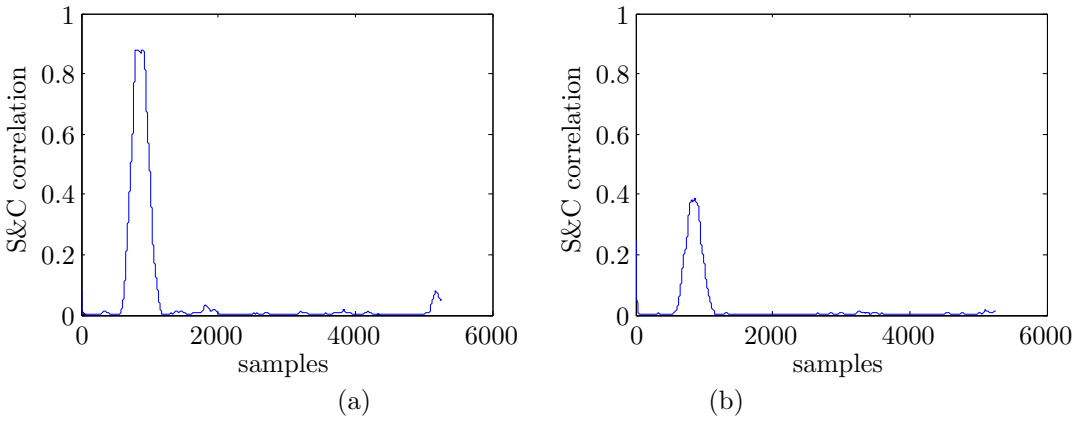


Figure 3.1: (a) Timing metric in simulation $SNR = 15$ dB. (b) Timing metric in test tank trial $SNR = 10$ dB.

Figure 3.1 shows an example of timing metric in both simulation and controlled field test, as a window slides past coincidence for the AWGN channel with 512 subcarriers and a SNR = 15 dB and SNR = 10 dB respectively. Timing metric reaches a plateau of the same length of the guard band minus CIR, as can be clearly observed in simulation metric. theoretically it means that the start of the symbol can be taken at any point of this plateau, since guard band redundancy allows this time shifting, but for practical implementation, as depicted in chapter 2 section 2.5 a time shifting was added at the receiver in order to increase frame MSE. So this kind of frame estimation can lead to frame detection inaccuracies.

Even its accuracy dependence to a high SNR and plateau detection, its pluses are that the receiver can be ignorant to the transmitted data since it knows the expected symbol structure, and this methodology also allows to compute CFO as will be discussed at chapter 4

3.2 Linear Frequency Modulation frame detection

In this section a cross-correlation based detection method is presented. For detecting the expected signal at the receiver side is used a matched filter, so first of all is described detection signal processing, and then the specific case for the LFM signal, defined in (3.4) [74].

$$x_{LFM}(t) = A\exp(j\pi\beta t^2) \tag{3.4}$$

In case of Doppler shift with $f_{d,AB} = a_{AB}f_0$, LFM signal is defined at the receiver side as:

$$y_{LFM}(t) = Ae^{j2\pi f_{d,AB}t}x_{LFM}((1+a_{AB})t) \tag{3.5}$$

Denote H_0 as the hypothesis that the received acquisition window is absent of useful frame, and H_1 as the hypothesis that the desired signal is present inside of an acquisition window. For a time-varying channel depicted in (2.13) receiver signals can be expressed as:

$$\tilde{y}_{AB}(t) = \begin{cases} \tilde{\omega}(t), & H_0 \\ \sum_{p=0}^{N_p-1} A_p(t)\tilde{x}_{AB}((1+a_{AB})t-\tau_p) + \tilde{\omega}(t), & H_1 \end{cases} \tag{3.6}$$

After down-converting the passband waveform to baseband and lowpass filtering, the baseband waveform can be formulated as

$$y_{AB}(t) = \begin{cases} \omega(t), & H_0 \\ \sum_{p=1}^{N_p} h_p(t)e^{j2\pi f_{d,AB}t}x_{AB}((1+a_{AB}t-\tau_p) + \omega(t), & H_1 \end{cases} \tag{3.7}$$

Using as local template, the transmitted signal in baseband $x_{AB}(t)$, the matched filter output at the receiver side is:

$$R_{MF}(\tau) = \int x^*_{AB}(t)y_{AB}(t+\tau)dt \tag{3.8}$$

Then the matched filter for each specific case at the receiver, substituting (3.7) into (3.8) yields:

$$R_{MF}(\tau) = \begin{cases} R_{xn}(t), & H_0 \\ \sum_{p=0}^{N_p-1} h_p(t)\exp(-j\pi\frac{f_d^2}{\beta})R_{xx}(\tau_p - \frac{f_d}{\beta}) + R_{xn}(t), & H_1 \end{cases} \tag{3.9}$$

Where $R_{xx}(t)$ is the auto-correlation function of $x_{AB}(t)$, and $R_{xn}(t)$ is the correlation between $x_{AB}(t)$ and the ambient noise $\omega(t)$, which are two independent variables following a non-centrally chi-squared distribution.

From equation 3.9, one can observe that a Doppler shift will shift the correlation peak by a delay of f_d/β, this effect is called the delay-Doppler coupling. Thus, Doppler scaling, presented in next chapter, has to be compensated to enhance frame detection accuracy.

Based on matched filter output, a common detection method for radar and sonar applications is used. It is thresholding method:

$$max\{|r_{MF}[n]|^2\} \lesseqgtr_{H_1}^{H_0} Th \tag{3.10}$$

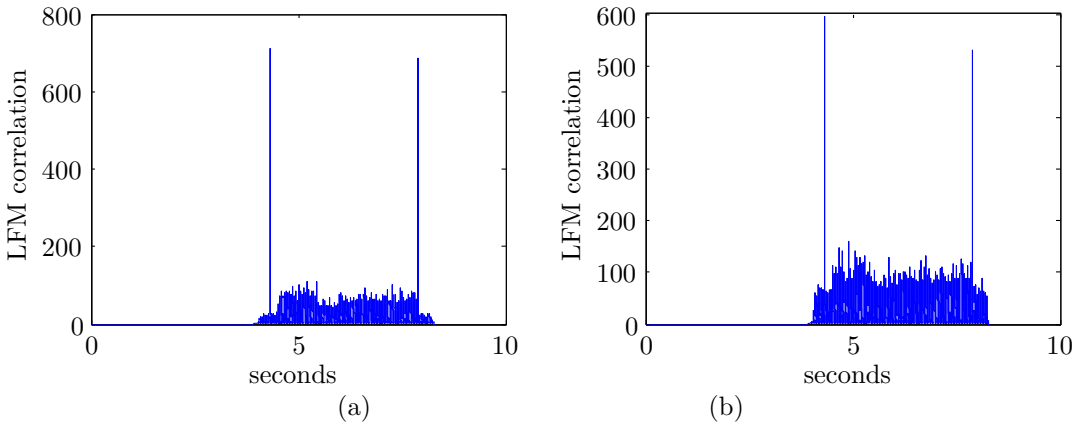


Figure 3.2: (a) LFM correlation in simulation $SNR = 15$ dB. (b) LFM correlation in test tank trial $SNR = 10$ dB.

In time varying channels Doppler-insensitive waveforms are desirable for frame detection and synchronization. The LFM waveform is one example of this frame detection algorithm and it can be applied to a narrow-band communication, which matches to this research requirements.

Chapter 4

Doppler scale compensation

Contrary to cable networks, the low celerity of wave sound makes underwater acoustic communications system very sensitive to Doppler effect, yielding to non-uniform frequency scaling represented by compression or dilatation of the time axis. This frequency scaling can be induced by 2 factors: motion (sensor mobility, channel variation, etc...) and clock skew receiver between transmitter and receiver. Actually, in order to address this problem some systems uses expensive inertial sensors for compensating Doppler scaling due to motion and temperature compensated low drift clocks [75].

The objective of this chapter is to design and to experiment in real condition a comparison of different Doppler scale estimators used in acoustic communication. Since we are working with frequency modulated communication, Doppler scaling will affect both communication metrics and time stamping accuracy. In time synchronization these two factors are what determine its performance. Communication metrics worsen with Doppler scaling by the fact that a frequency shift yields in a phase shift at QPSK constellation leading to BER degradation and resulting in the loss of useful data. On the other hand, Doppler shifting also affects time synchronization algorithms since it results in a dilatation/compression of the time basis which impacts directly the synchronization protocol accuracy. In the following we consider five algorithms for Doppler scale estimation exhibiting affordable complexity for real-time implementation. More sophisticated algorithm based on exhaustive search approach can be found in [53].

Then, at the receiver side after downconversion, and now taking into account all underwater channel challenges, (2.13) can be reformulated as:

$$y_{AB}(t) = \sum_{p=0}^{N_p-1} h_p(t) e^{j2\pi f_{d,AB} t} x_{AB}((1+a_{AB})t - \tau_p) + \omega(t) \tag{4.1}$$

where $f_{d,AB} = a_{AB} f_0$. Hence, is necessary to extract Doppler scaling at the reception side before proceeding to frame detection and data demodulation as described in chapter 2.

4.1 Pure tone Doppler shift estimation

This approach consists of sending a Pure Tone (PT) signal outside the useful spectrum and studying its phase variation at the reception side in order estimate the Doppler shift [25]. At the reception side, after base band conversion and narrow-band filtering around f_{pt}, the expression of the baseband received PT signal can be derived from (4.1) as follows:

$$y_{pt}(t) = e^{j2\pi a_{AB} f_{pt} t} \sum_{p=0}^{P-1} A_p(t) e^{-j2\pi f_{pt} \tau_p} + w(t) \tag{4.2}$$

Let's note $\psi(t) = \sum_{p=0}^{P-1} A_p(t) e^{-j2\pi f_{pt} \tau_p}$, the argument of $y_{pt}[n] = y_{pt}(n/f_s)$ can be expressed as:

$$\arg y_{pt}[n] = 2\pi f_{d,AB} \frac{f_{pt}}{f_0} \cdot \frac{n}{f_s} + \arg \psi[n] + \arg w[n] \tag{4.3}$$

Due to the properties of $\delta\tau_p$ which can be reasonably modeled as zero mean Gaussian process independent for each path, $\psi(t)$ can be treated as an uncorrelated random process that can be mitigated by low pass filtering. An estimation of the motion induced Doppler shift can be formed as [76]:

$$\hat{f}_d = \frac{f_s}{2\pi} \frac{f_0}{f_{pt}} \text{LPF} \left[\arg \left(y_{pt}[n] y_{pt}^*[n-1] \right) \right] \tag{4.4}$$

where LPF[.] denotes the low pass filtering operation. This algorithm allows communication system to be independent of Doppler shift estimation procedure, due to the fact the PT signal does not interfere with payload bandwidth. In this study we use a OFDM communication band centered at 30 kHz with 1.3 kHz Bandwidth, and a pure tone centered at 20 kHz for Doppler scale study, then while we are transmitting instrument information and synchronization framing, we are capable to add another tone without spreading communication time, and neither worsening OFDM robustness. When separating both signals, OFDM and pure tone, is to necessary to use a sharp pass band filter in order to avoid interferences while do not cutting the Doppler frequency shifting on the sides of the pure tone signal.

4.2 Schmidl & Cox CFO detection

Originally introduced in [72], the Schmidl & Cox algorithm is used to perform both time synchronization and carrier frequency offset (CFO) detection. This algorithm is based on two OFDM preambles, the first one is used for frame detection besides a fine CFO, and the second one is used for a coarse CFO estimation. in our case, CFO detection is to detect Doppler shift coming from motion and/or clock skew.

First preamble is a symmetric OFDM symbol which will suffer a phase difference between the first half and the second one:

$$\phi = \pi T_{sym} \Delta f \tag{4.5}$$

Which can be estimated as the angle resulting from the partial correlation $P(d)$ between the two halves near the best timing point [72].

$$P(d) = \sum_{m=0}^{L-1} y_{d+m}^* y_{d+m+L} \tag{4.6}$$

where y_n denotes the received baseband signal sampled at $1/T_{sym}$ and d is a time index corresponding to the first sample in a window of $2L$ samples. Phase variations are estimated as:

$$\hat{\phi} = \arg\left(P(d)\right) \tag{4.7}$$

An estimation of the actual frequency shifting is given by:

$$\hat{f}_d = \frac{\hat{\phi}}{\pi T_{sym}} + \frac{2\hat{g}}{T_{sym}} \tag{4.8}$$

Where g is an integer corresponding to the coarse CFO that will be estimated in the second phase.

The second training symbol contains a Pseudo-Noise (PN) sequence on the odd (x_1) frequencies to measure these sub-channels, and another PN sequence on the even frequencies (x_2) to help determine frequency offset. After computing the relation between odd frequencies and even ones at the transmitter side, we will obtain a conversion factor between pairs of frequencies, then at the receiver side, we can use this factor to estimate $\hat{x}_2$ from x_{m1} and vice versa. Then by computing the correlation between x_2 and $\hat{x}_2$, an estimation of the g factor is found by maximizing the following correlation metric:

$$B(g) = \frac{\left|\sum_{k\epsilon x} x^*_{1,k+2g} v^*_k x_{2,k+2g}\right|^2}{2(\sum_{k\epsilon x} |x_{2,k}|^2)^2} \tag{4.9}$$

Where v_k is the differentially-modulated PN sequence on the even frequencies of the second Schmidl & Cox symbol.

4.3 Preamble/Postamble Doppler scale estimation

As mentioned before, Doppler scaling performs similar to interpolating a signal into a different time base. Then if we compute an analysis of time variations between two known points in a frame, we could find a relation between this time variation and Doppler scaling, as described in equation (4.1). As originally introduced in [77], we add an LFM preamble and postamble to detect time compression or dilatation of the frame and then to estimate the Doppler scale factor. Main reason for using an LFM instead of another signal is its good robustness again Doppler effect as well as its cross correlation performance in environments corrupted by white Gaussian noise.

An example of cross-correlation results with LFM are plotted in Figure 3.2, the doppler scaled frame duration (noted t_{DS}) is estimated by computing time difference between the two correlation peaks given by the preamble and the postamble. Then by knowing the original frame duration t_{ideal} the Doppler shift is estimated as:

$$\hat{f}_d = f_0 \left(1 - \frac{t_{DS}}{t_{ideal}}\right) \tag{4.10}$$

Since the sampling frequency is a limiting factor, Doppler shift estimation accuracy can be improved significantly by using center of gravity computation for the time difference between correlation peaks.

4.4 Null carrier shift estimation

Doppler Effect causes a frequency shift, then if we know, in frequency domain, the power distribution of an OFDM symbol, we can estimate frequency shifting if this spectral distribution has been modified.

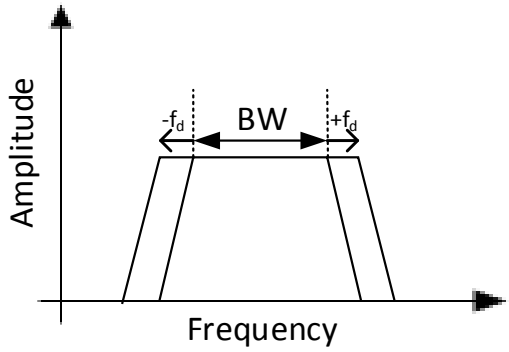


Figure 4.1: Null carrier approach description

So using an OFDM symbol, after computing the FFT, and estimating the frequency shifting in the null carriers we will be able to relate it with the Doppler Scaling. For doing so, we span a variable between the frequency shift detection range, in order to optimize frequencies centering in a symbol [54]. Then this value will be directly the equivalent to the Doppler scaling in Hz.

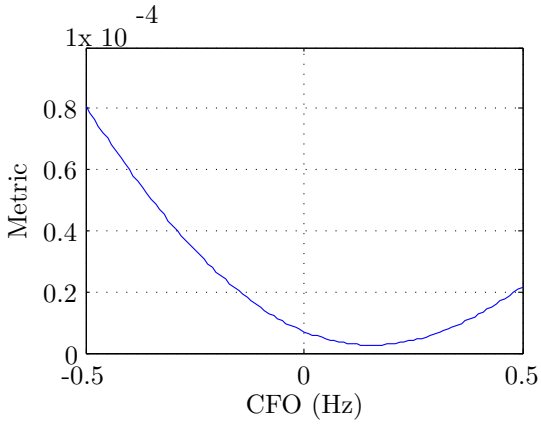


Figure 4.2: Null carrier approach Doppler shift estimation @ $f_{d,NC} = 0.15$ Hz, SNR = 15 dB

With this approach we are only capable to perform fine corrections of Doppler scale, so this approach should be used as a fine compensation of any of the previous ones. For this reason it will not be included in a direct comparison with the other three algorithms.

4.5 Time-Frequency plane shift estimation

In this approach is also used an LFM signal, but this time the point of interest will be its frequency shifting. We will compute the spectrogram of an LFM signal with BT=1000 t=0,2s, represented in Figure 4.3 as Tx Chirp. After sending it through a simulated channel we receive the chirp with noise and Doppler shifting in frequency plane (Rx Chirp), so if we compute the difference between TX and RX chirp in the frequency plane we will obtain the

value of Doppler shifting in Hz.

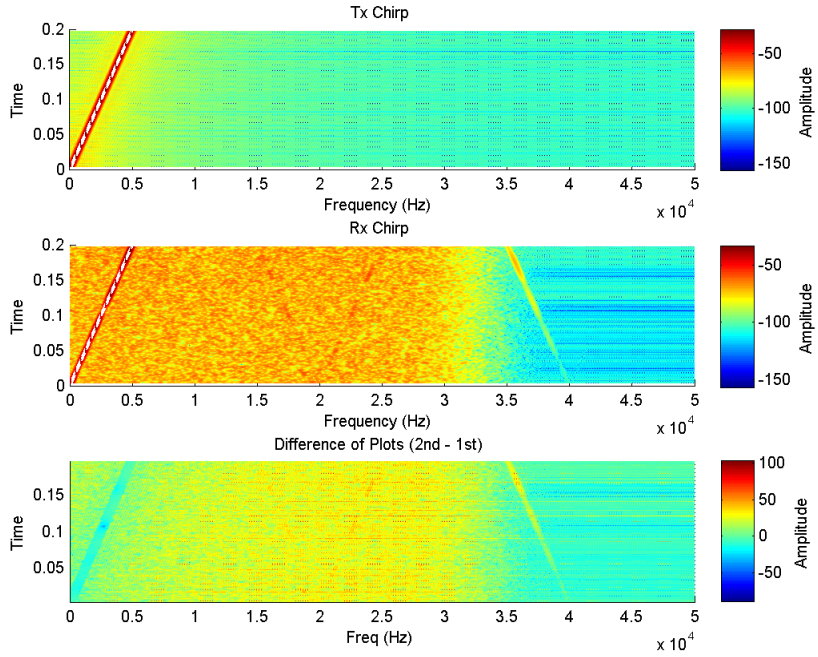


Figure 4.3: Spectrogram of TX chirp (up), Spectrogram of RX chirp (middle), Difference between subplot 1 and subplot 2 (down), SNR = 15 dB

In order to extract both chirps in frequency domain is computed a maximum detection in Amplitude axis to port data to 2D plane. After that, linear regression is performed to make the frequency detection smoother, then, only is necessary to compute mean difference between the two frequency evolutions along time. What results directly in Doppler scaling frequency. As can be observed in figure 4.3 the difference between transmitted time-frequency plane and received one, is highly dependent to SNR, in scenarios below 20 dB it yields in aberrant Doppler scale estimations. So for this reason this algorithm cannot be directly compared, in terms of accuracy, to the first three algorithms presented in this chapter.

Henceforth, updating table 2.3, parameters used for communication taking into account frame detection and Doppler scaling are summarized at table 4.1

4.6 Simulation

By simulating these five methods, we analyze the accuracy on Doppler shift detection, and how does it affect frame correction and synchronization. The parameters used for simulation are summarized in Table 4.1, and the frame structure for evaluating communication protocol is described in figure 4.4. First, we compute an SNR sweep 1000 times averaging the results at each SNR for avoiding aberrant errors in our results. We consider a Doppler shifting of $f_d = 20$ Hz at $f_0 = 30$ kHz which corresponds to a relative motion of 1 m/s. Figure 4.5 displays the Root Mean Squared Error (RMSE) of Doppler shift estimation defined as:

Table 4.1: Communication parameters summary [Typ. values]

Description	Parameter	Simulation & Laboratory	OBSEA
Sampling frequency	f_s	100 kS/s	100 kS/s
OFDM frequency center	f_0	30 kHz	30 kHz
OFDM BW	B_{OFDM}	1.19 kHz	1.18 kHz
Pure tone frequency center	f_{pt}	20 kHz	20 kHz
LFM BW	B_{LFM}	5 kHz	5 kHz
OFDM symbol duration	T_{OFDM}	480 ms	168 ms
Useful part of OFDM signal	T_u	384 ms	96 ms
Cyclic prefix period	T_{CP}	96 ms	72 ms
FFT points	N_{FFT}	512	128
Active carriers	K	460	114
Signal to Noise Ratio	SNR	15 dB	15 dB
Doppler frequency	f_d	20 Hz	20 Hz

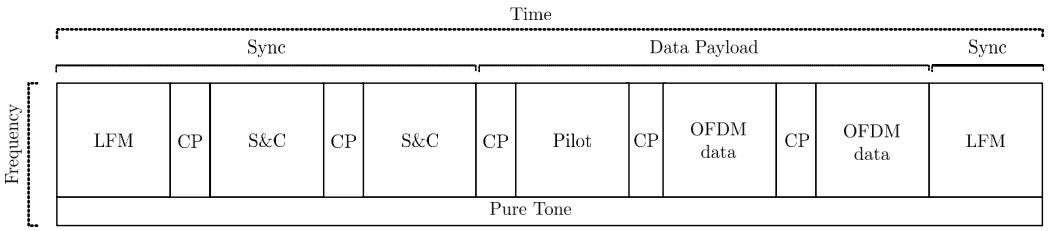


Figure 4.4: Overall structure of the transmitted signal

$$\text{RMSE} = \sqrt{\mathbb{E}\left[|f_d - \hat{f}_d|^2\right]} \tag{4.11}$$

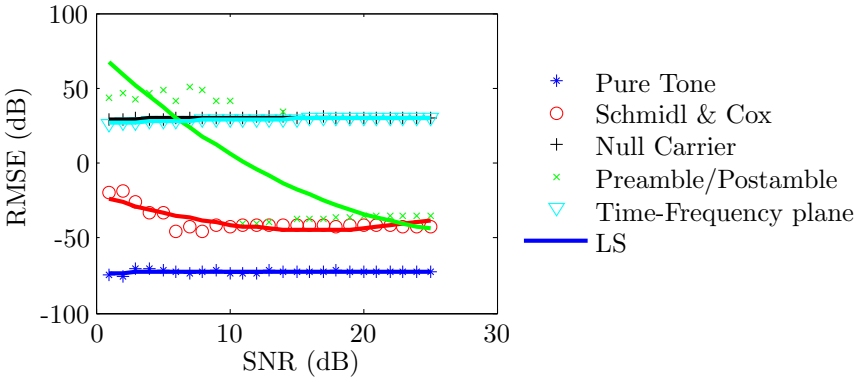


Figure 4.5: Doppler scale estimation simulation $f_d = 20$ Hz at $f_0 = 30$ kHz with SNR sweep

Then, in figure 4.6 , the same simulation is repeated, but this time we keep a constant

SNR of 15 dB and the sweep is performed along f_d from 0 to 70 Hz. By using the term LS, we mean that a second order Least Square Regression model has been applied in order to make results more readable, it is just an approximation, which in some cases differ from real acquired values given by raw data points.

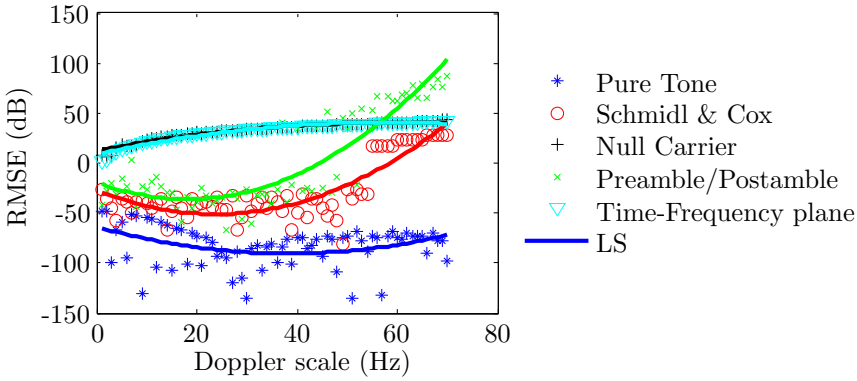


Figure 4.6: Doppler scale estimation simulation vs. SNR = 15 dB at f_0 = 30 kHz with f_d sweep

Figures 4.5 and 4.6 prove the low accuracy in Doppler scale estimation of Null carrier approach (section 4.4) and Time-Frequency plane approach (section 4.5), as it was expected after algorithm performance verification. Therefore, hereinafter all simulations and real tests will not take into account this two algorithms for Doppler scale estimation and compensation. On the other hand, presenting values above 0 dB regarding Doppler scale estimation RMSE is useless, since it means completely odd estimations, so for making graphs more readable, only meaningful results will be plot.

Thereby, figures 4.5 and 4.6 can be replotted as figures 4.7 and 4.8 respectively:

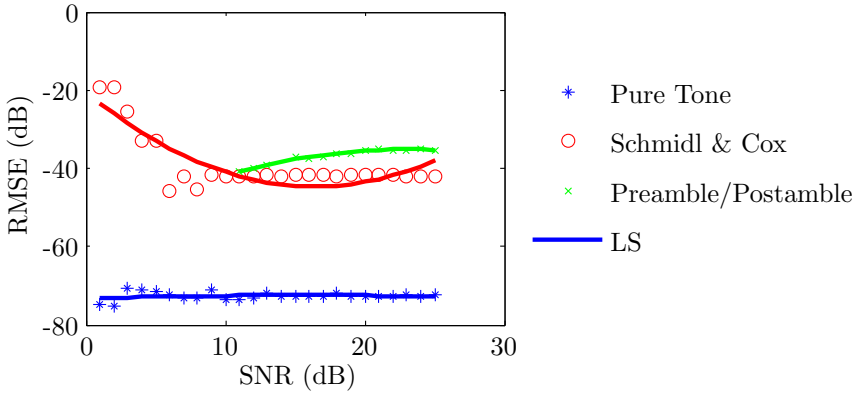


Figure 4.7: Doppler scale estimation simulation f_d = 20 Hz at f_0 = 30 kHz with SNR sweep

Hence, if we use this Doppler scale estimation for correcting frequency shifting in our communication we will be able to recover enclosed timing information. Figure 4.9 displays frame MSE, equation 4.12, after applying Doppler scale compensation and channel equalization with each described algorithm.

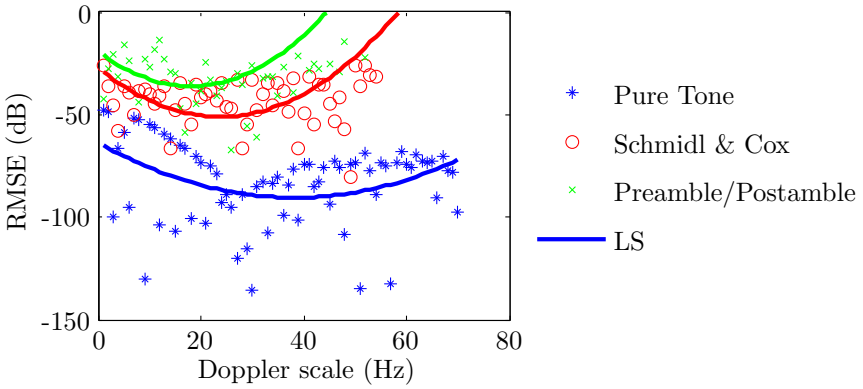


Figure 4.8: Doppler scale estimation simulation vs. SNR = 15 dB at $f_0 = 30$ kHz with f_d sweep

$$\text{MSE} = E[|\hat{d}_k - d_k|^2] \tag{4.12}$$

where $\hat{d}_k$ denotes the estimation of data cell d_k after OFDM equalization.

The curve labeled perfect correction provide a lower bound on MSE performance where the Doppler shift is perfectly known and removed. Finally in figure 4.10, we run a simulation where we keep a constant 15 dB SNR, and we perform a Doppler shift sweep.

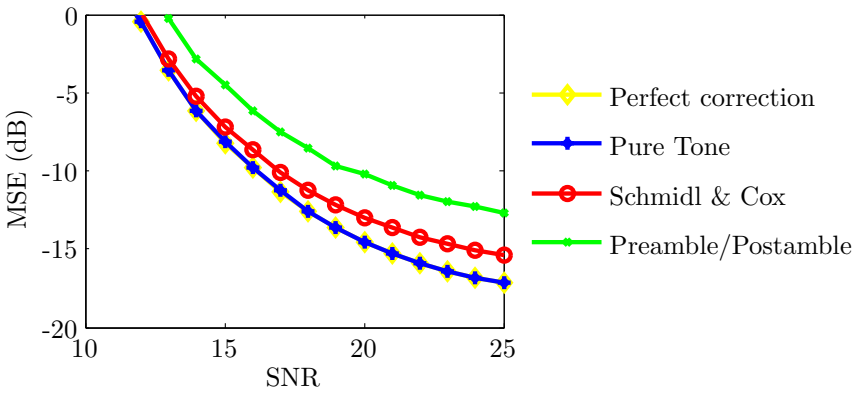


Figure 4.9: Frame MSE after Doppler scale compensation on an SNR sweep with $f_d = 20$ Hz in simulation

This simulation gives an idea of which algorithm will have better performance in an ideal scenario. Pure tone approach outperforms all the other algorithms amply. All three methods have a flat response for the Doppler frequency shift scenario, and they present a direct response with the SNR sweep, as the SNR increase the Doppler scale estimation improves and consequently the MSE in the frame after correction also improves.

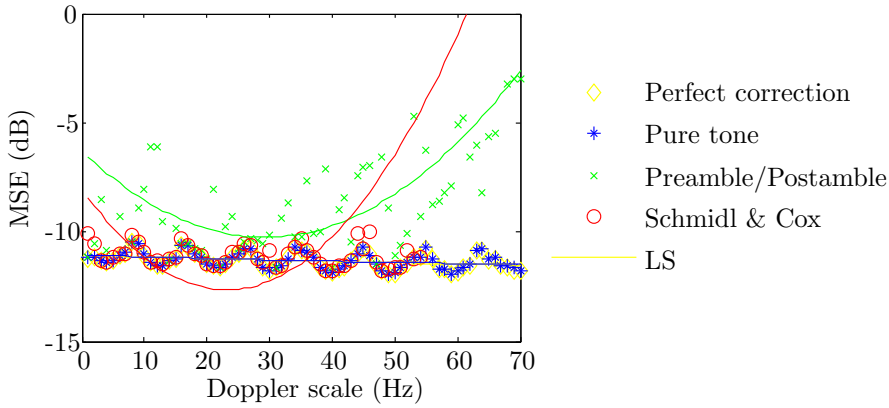


Figure 4.10: Frame MSE after Doppler scale compensation on a Doppler scale frequency sweep with SNR = 15 dB in simulation

Chapter 5

Experimental tests

This chapter contains extensive performance results based on whole chain communication described at previous chapters. It is also presented the transmitting and receiving hardware outline, which will be detailed in time synchronization part since it has been developed for improving time synchronization accuracy.

5.1 Communication Hardware

Once modulation and demodulation algorithms have been implemented and simulated, they must be coupled with hardware components. In order to bring and drive current advances in acoustic communication to time synchronization field, an specific hardware for communication purposes must be designed as will be described in chapter 7. Thus in this section is developed the communication hardware, which will be refurbished at time synchronization part to include new functionalities to enhance timing accuracy.

For high accuracy time synchronization algorithms is necessary to avoid indeterministic times when transmitting and receiving frames, so an Field-Programmable Gate Array (FPGA) based controller will be used. An FPGA is used for programming critical timing parts, as a software-defined hardware, and a real time controller controls communication data flow. The aim of this research is not to design a communication and time synchronization commercial product, but evaluate different approaches to enhance time synchronization capabilities. Then a distributed and a software-defined modem, with the Open Systems Interconnection (OSI) layer distributed in space is designed. Where MAC layer up to application layer will be set at laboratory computer and the PHY layer is set inside a vacuum cylinder deployed underwater as shown in figure 5.1. This approach ease testing different algorithms in experimental tests.

Since controller and FPGA have to be deployed in a rough environment place, such as sea floor, where currents can hit strongly the vacuum cylinder chassis, a robust solution must be used. For this reason, in this research, a National Instruments hardware has been chosen. It is a compact RIO model 9103 (cRIO-9103) [78] containing the FPGA with 3 M gates for a maximum processing power, and the controller model 9012 (cRIO-9012) [79] with 64 MB DRAM, 400 MHz processor and 128 MB nonvolatile storage, which is enough for controlling raw data flow between on-shore computer and FPGA as PHY layer. Principal characteristic of this controller/FPGA family is its robustness. cRIO products endure 3 G impacts, are suitable for use in Zone 2 hazardous locations and in ambient temperatures of $-40°C \leq Ta \leq 70°C$, besides that this products are tested and complies with the regulatory

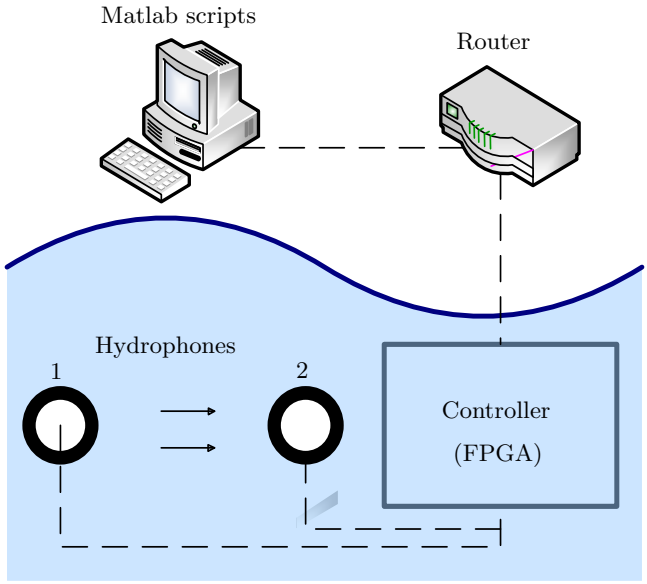


Figure 5.1: Distributed Radio-defined modem outline

requirements and limits for electromagnetic compatibility (EMC). So it is one of the best platforms for being enclosed in a vacuum cylinder, which can suffer impacts during deployment, and its temperature can increase in summer because of the lack of refrigeration. Figure 5.7 shows compact RIO design at its final stage.

Once controller and FPGA are chosen, is necessary to convert digital data to analog in order to transmit it through underwater acoustic channel. All our simulations are based in a 100 kS/s sampling rate. So, an analog input and output module compatible with cRIO architecture, and matching desired sampling rate frequency, have been selected. The analog input stage is a cRIO-9201 module [80] with a ±10 V 12-bit resolution input, capable to work up to 500 kS/s in an 8 sequential single-ended channel acquisition. The analog output is a cRIO-9263 module [81] with a ±10 V 16-bit resolution output, capable to work up to 100 kS/s in an 4 simultaneous analog output.

Once in analog domain, is necessary to condition signals to match hydrophones requirements. For transmission, analog signal power has to be increased using a power amplifier as described in subsection 5.1.1, and at the receiver side, a charge amplifier and low noise amplifier are needed to recover received signal, described in subsection 5.1.2 [68]. For research purposes, concerning time synchronization matters, is needed a bidirectional communication, so each transducer must be connected to both, charge amplifier and power amplifier. This half-duplex capability is possible by using a T/R switch as its shown in figure 5.4 where is outlined the amplifying prototype.

5.1.1 Power amplifier

Power amplifier stage is based in two parallel power amplifiers which provide 50 mA each one (see figure 5.2). Yielding in a maximum power supply of 100 mA capabilities up to ±50 V.

Since communication range is not one of the goals of this research, the power supply has been set to 10 V, due to the high cost of more powerful power supplies. Then the power

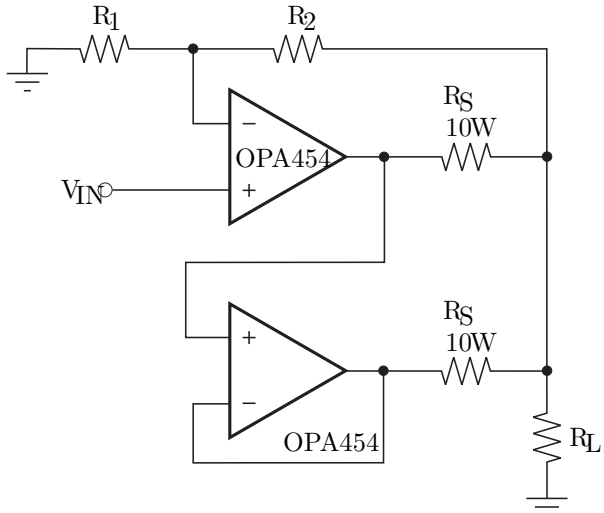


Figure 5.2: Power amplifier, 100 mA ±50 V [82].

amplifier stage is driving power output of 1W, which is enough for short range acoustic communication.

5.1.2 Charge amplifier

To condition piezoelectric hydrophones, has been used a conventional charge amplifier adapted to match transducers impedances, see figure 5.3. This stage works as a band pass filter, where the upper limit is set by piezoelectric and its wirings, and the lower limit is given by circuit components.

5.1.3 Prototype design

The amplifier module prototype is schematized in figure 5.4. With this design is possible to do a half-duplex communication, with a two-stage charge amplifier, placed at the upper side of the Transmission/Reception switch, where 'LNA' stands for Low Noise Amplifier and 'Amp' is a conventional Operational Amplifier for reception. And for transmission a power amplifier 'PA' is directly connected to the switch which handles directly with the piezoelectric transducer.

In appendix A can be found detailed design of each one of these stages in order to create the final PCB, represented in figure 5.5, capable to match figure 5.4 functionality with filtering limits given by this research work frequencies summarized at table 4.1.

This PCB design, has been developed to fit inside one of the c-Modules architectures of National Instruments. This means that it has same size than analog input and output modules, and can be directly plugged to one of the FPGA connectors in the cRIO.

Finally, as shown in figure 5.6, communication hardware block diagram will be composed by the cRIO module (Real time controller + FPGA), two analog modules for input and output, and two amplifiers, to reproduce a half-duplex communication using only one data flow controller. Black arrows indicate the connections between modules and the direction of the information. Since we implement bidirectional communication all wirings have an arrow

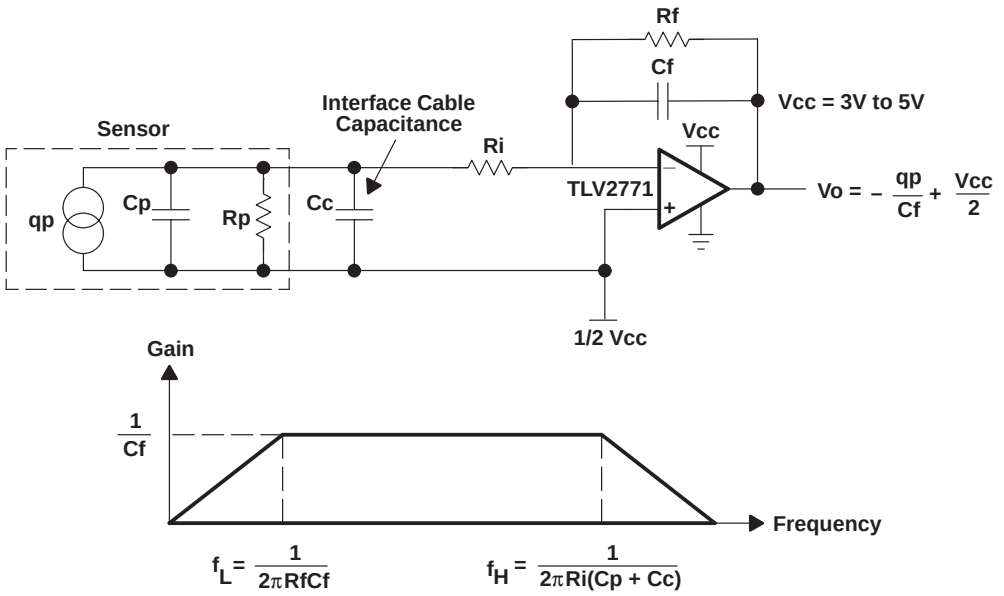


Figure 5.3: Charge amplifier [83]

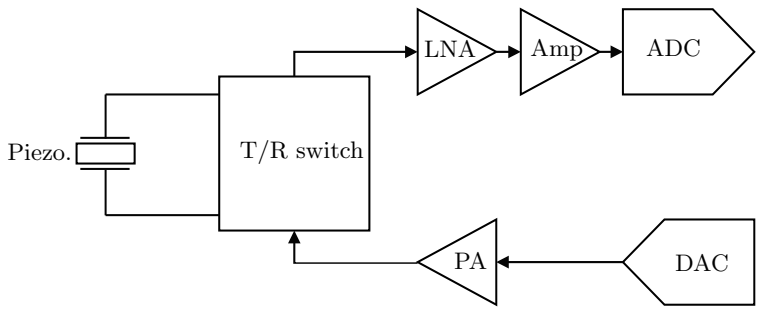


Figure 5.4: Amplifier module outline

at both sides.

Final appearance of communications controller hardware can be seen in figure 5.7. And this system will be used in both laboratory tests and underwater experimental tests, where the only difference will be the casing and wiring between the analog to digital converters and the amplifiers modules, which in underwater test should be expandable to test several distances keeping piezoelectric to amplifier wiring as short as possible to do not lose signal power.

5.2 Laboratory tests

Communication tests in controlled environment have been performed in a water test tank of dimensions 150 cm long, 40 cm tall and 40 cm width, as displayed in figure 5.6. Presented

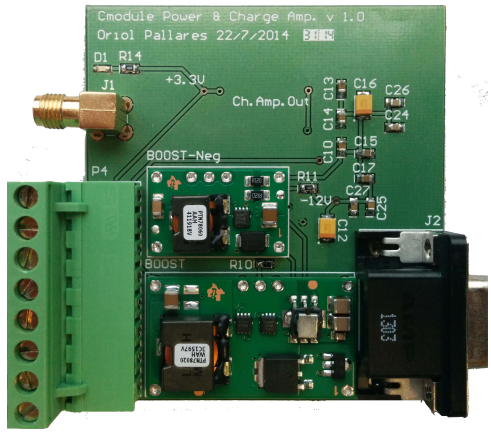


Figure 5.5: Amplifier module PCB

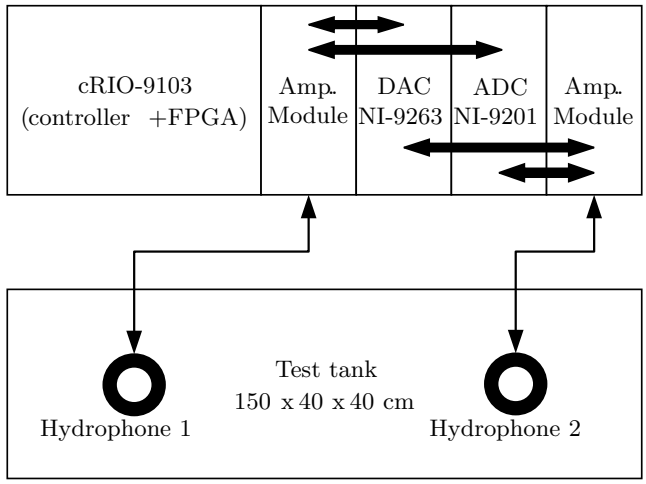


Figure 5.6: Communication hardware block diagram

tests pretend to evaluate proper functionality of simulated algorithms before developing real tests. It is a good scenario for evaluating communication performance, since it has strong multipath effect, and Doppler scaling can be controlled by applying wave interpolation, due to test tank water is motionless.

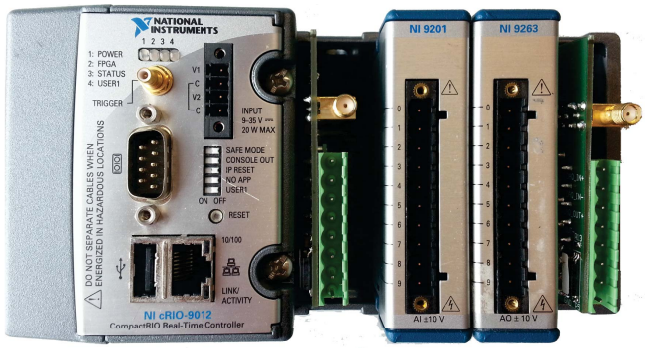


Figure 5.7: cRIO with analog modules besides amplifier module (Communication hardware)

5.2.1 Workbench

The signal is sent through the test tank, in a 1 meter distance communication, and recovered back by the NI device which will send the raw data to a computer running Matlab®. This raw data is saved and the same receiver script used in simulation will estimate communication performance.

Only the PHY has been modified from simulation to real tests. Laboratory tests workbench match exactly figure 5.6 disposition, with some extra components to allow autonomous functionality as shown in figure 5.8. A Relative Humidity (RH) sensor must to be added in order to detect water leaks inside of a tight cylinder ones a final deployment in the sea floor is performed. Then a MOXA system is added to remotely control a relay in charge of opening and closing power supply to the cRIO module. And finally a switch to redirect data to MOXA controller or to cRIO module.

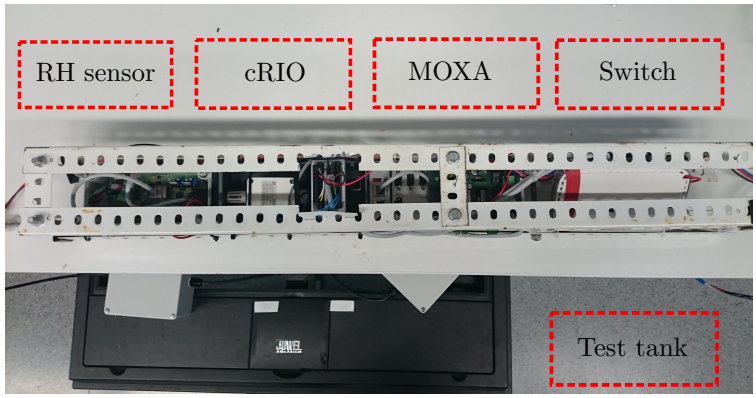


Figure 5.8: Laboratory workbench

5.2.2 Experimental Results

With this experiment we pretend to verify a proper functionality of Doppler scale estimation algorithms in a controlled environment. Where we do not have inherent sea currents, so

we can simulate Doppler shifting to estimate it without any additive sea factor. Simulated Doppler shift is added in the received signal by resampling the received pass-band signal by a factor of $1 + a$.

To compare each Doppler shift estimation algorithm, we use the same methodology as section 4.6 but this time the signal is transmitted in the 'test tank' where we have a very difficult channel condition with strong multipath of about 75 ms, and all tests are only averaged 10 times due to real test's processor timing constraints. Figures 5.9 and 5.10 show provides Root Mean Squared Error (RMSE) performance with SNR sweep and Doppler shift sweep respectively. In practice SNR variation is performed by modifying the transmit signal power, then the SNR displayed in aforementioned figures is an estimated SNR defined as the ratio between the signal and the noise of each channel.

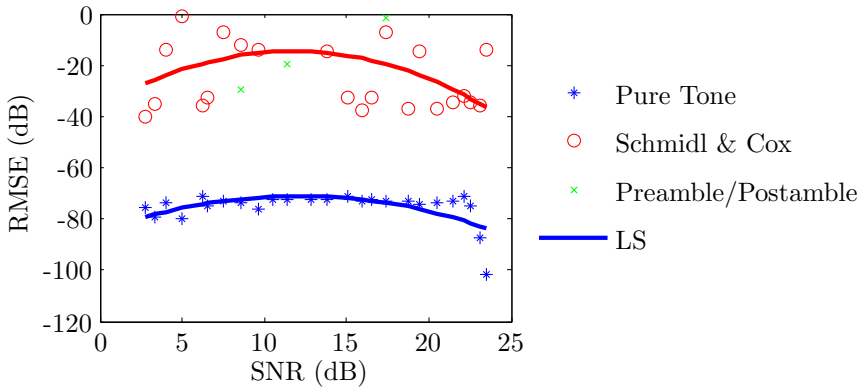


Figure 5.9: Doppler scale estimation Laboratory test $f_d = 20$ Hz at $f_0 = 30$ kHz with SNR sweep

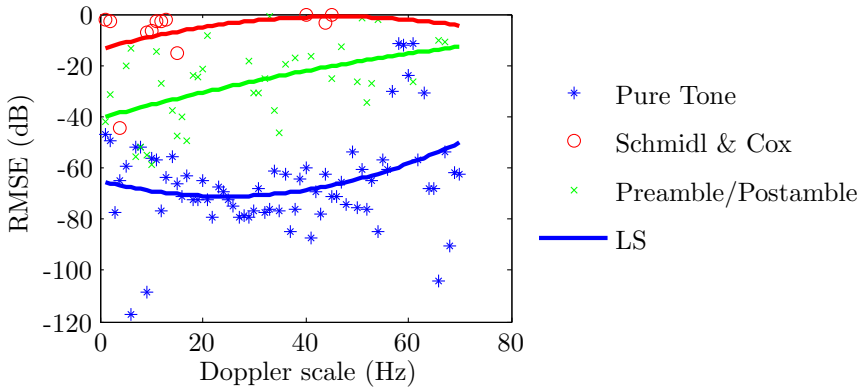


Figure 5.10: Doppler scale estimation Laboratory test SNR = 15 dB at $f_0 = 30$ kHz with f_d sweep

Figure 5.10 shows similar performance to figure 4.8, corresponding to Doppler scale estimation in both Laboratory and Simulation respectively, where RMSE is increasing or constant as f_d increase.

In a second step, we apply Doppler scale compensation to the OFDM frame with each frequency shifting estimation obtained in the previous stage, and we evaluate the MSE of the data frame after compensation and channel equalization, results are carried out in figures 5.11 and 5.12 respectively. For these laboratory tests, we repeat both SNR and Doppler frequency sweeps for observing the algorithms performance for any water channel status.

As in simulation, PT approach outperforms the other estimation algorithm, leading to near-perfect Doppler shift correction at a SNR of 15 dB.

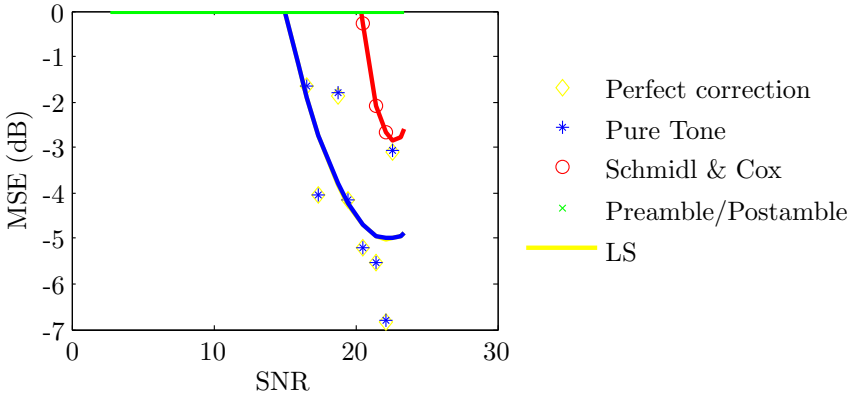


Figure 5.11: Frame MSE Laboratory test after Doppler scale compensation on an SNR sweep with constant $f_d = 20$ Hz

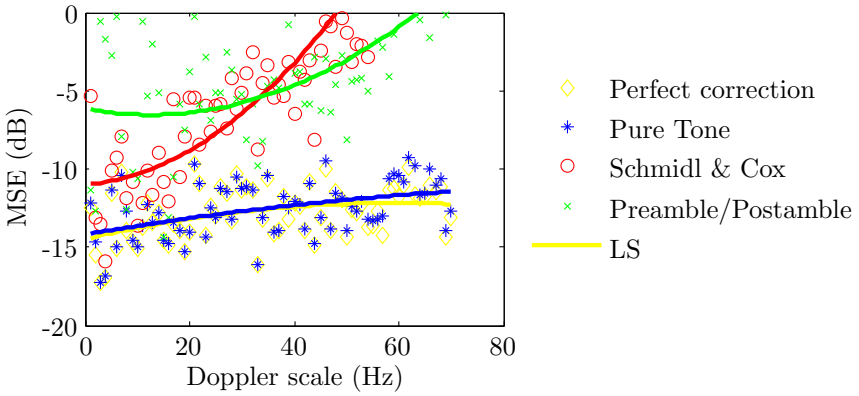


Figure 5.12: Frame MSE Laboratory test after Doppler scale compensation on a Doppler scale frequency sweep with constant SNR = 15 dB

In figure 5.12 a high variability of MSE along Doppler scale sweep can be observed. This is due to the low number of repetitions performed for each Doppler scale value causing high variance. For this reason the least square linearization can help us to validate the trend of MSE in function of Doppler scaling.

5.3 Underwater experimental results

Since the mid of 2009, (Sistemas de Adquisición Remota y Tratamiento de la Información) SARTI research group have developed OBSEA [17], which is an underwater platform that provides power supply and data connectivity between underwater systems and SARTI's laboratory. It is placed 4 Km offshore and 20 meters deep as schematized at figure 5.13. Thus, this platform is used for sending raw data from a computer to the compact RIO module deployed underwater.

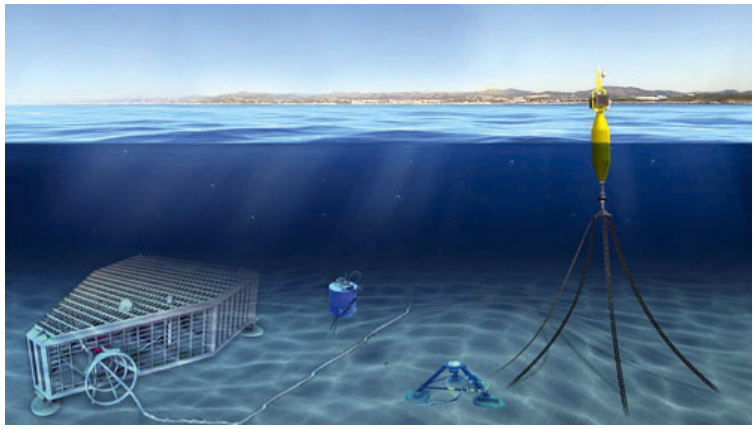


Figure 5.13: Instruments connectivity to laboratory using OBSEA platform [84]

5.3.1 Deployment

Due to OBSEA platform is placed at 20 meters deep, divers can install easily the acoustic modem developed in previous section. Before the deployment, the cRIO based acoustic modem has to be placed in a tight cylinder enclosing all the electronics but the amplifier stages, which are enclosed in vacuum cases close to the hydrophones. Figure 5.14 shows the water tight enclosing system setup used in subsection 5.2.1.

Figure 5.14: Water tight cylinder enclosing cRIO based acoustic modem electronics

Previous to final deployment, pressure tests inside of an hyperbaric chamber were done. These tests are based on several pressure variations between 0 bar and 5 bar, and next an static long time pressure test, set at 1.5 the maximum depth the instrument will be placed, which is $1.5x20$ meters = 3 bar, during one week. During these tests RH sensor must keep constant, what means that the system is ready to be deployed.

This experiment started on June 8, 2015. The same signal set as described in chapter 4 was used. The signal was transmitted from a depth of about 20 m and received by a single hydrophone at same depth at 1.5 m of separation of the transmitter. Both hydrophones where attached at 0.5 m of the seafloor. OFDM signals were transmitted while hydrophones were moving free in an horizontal axis by the sea currents. This research pretends to evaluate time synchronization accuracy, so long range experiments will not be conducted, but will be simulated.

Deployment at sea floor was performed by divers between buoy chains and OBSEA platform as shown in figure 5.13.

5.3.2 Experimental Results

Field testing is a key step in evaluating the performance of an underwater acoustic communication system. Because of the extensive cost of such tests (especially in deep waters), several numerical models for distance simulation, presented in subsection 2.2.1, have been used.

As in the previous subsections, we first evaluate the RMSE performance of Doppler scale algorithms as plotted in figure 5.15, and then MSE performance after correction and equalization is provided in figure 5.16. In this real test we only perform a Doppler scale sweep with a constant estimated SNR fixed at 20 dB, due to hardware limitations, the power amplifier clips the signal if we need to transmit high power to the water channel, so we are not able to increase the SNR up to the same levels shown in Laboratory tests. Again the Doppler scale is simulated by re-sampling the received pass-band signal.

These experimental tests confirm how Pure tone outperforms Schmidl & Cox algorithm and Preamble/Postamble LFM one, in all the possible scenarios. As in laboratory test, Doppler scale estimation and compensation lead to an MSE quite close to a perfect Doppler shift correction. So this algorithm will be the one used for the synchronization part of this research.

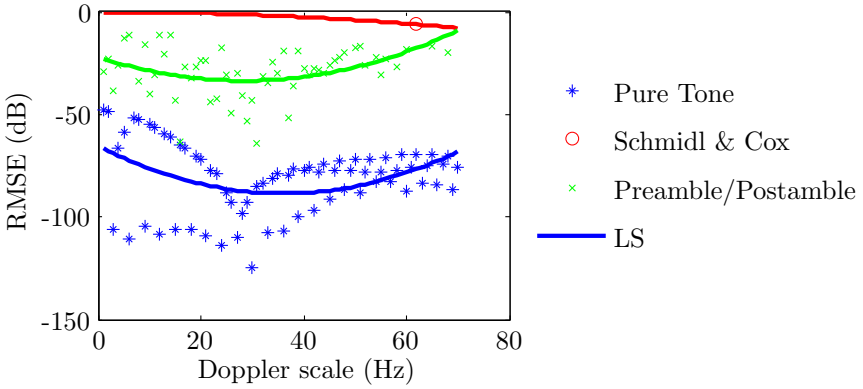


Figure 5.15: Doppler scale estimation at OBSEA SNR = 15 dB at f_0 = 30 kHz with f_d sweep

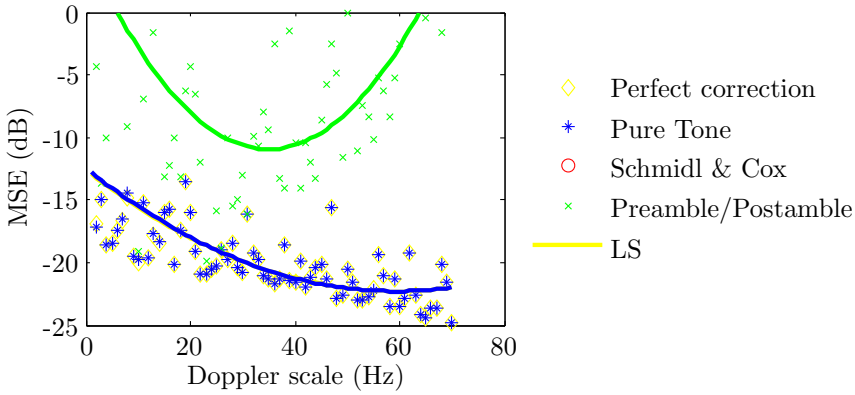


Figure 5.16: Frame MSE at OBSEA after Doppler scale compensation on a Doppler scale frequency sweep with estimated SNR = 15 dB

In figure 5.16 and odd behavior can be observed, since MSE decreases while Doppler shifting is increasing. If results given in figure 5.15 are used for Doppler scale compensation, an increase of MSE in function of f_d is not be possible.

This problem can be explained by the fact that Doppler scale compensation is performed in pass-band, with a sampling frequency only three times bigger than central communication frequency, frame interpolation to a new time base can yield in odd MSE results. This problem can be solved by compensating Doppler scaling in baseband where symbol frequency is 10 times smaller than sampling frequency. Furthermore, for this relative short ratio between sampling and communication frequencies, MSE can be decreased by using baseband compensation, as shown in conclusions chapter.

Since data link will be used in next part of the thesis for time stamp information communication inside OFDM data, BER as function of Doppler scale variations is also presented in figure 5.17. Again, due to Power amplifier constraints the SNR sweep is not possible because it can cause signal clipping what leads to errors in BER computation. Thereby next figure, shows how BER is close to 0, after averaging 10 repetitions for each f_d value, using a pass-band Doppler scale compensation, what means that time stamp information will be available at the receiver side, and partially validates our communication system for time synchronization purpose.

With QPSK modulation and MSE below 3.2 dB, BER after Viterby decoder has to be lower than 10^{-4} which is considered as Quasi Error Free transmission, as stated in chapter 2 figure 2.15. As shown in this chapter, pure tone approach outperforms this MSE level in all possible scenarios.

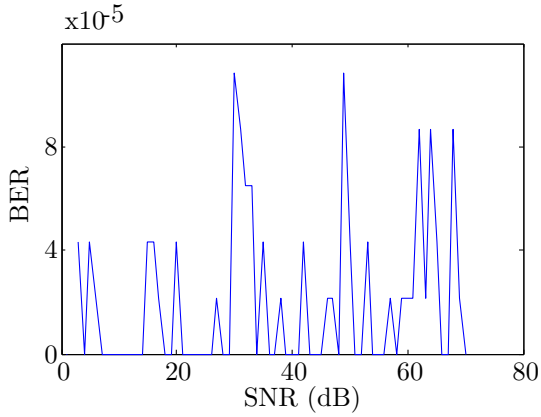


Figure 5.17: BER at OBSEA after Doppler scale pass-band compensation, using Pure Tone approach, on a Doppler scale frequency sweep with estimated SNR = 15 dB

Chapter 6

Time synchronization

6.1 Context

This section reviews the state of the art of time synchronization applied to acoustic-UWSN. It describes the structure of a common time synchronization protocol, besides some of the challenges of underwater acoustic communication and its implication with synchronization protocol.

Sharing the same time base is a key point in distributed networks for precise event time stamping or fine wake/sleep cycles in sensors, which can reduce significantly power consumption in an standalone network.

UWSN make extensive use of synchronized time for many services provided by a distributed network. In UWSN, GPS signals are not available and synchronization system are mostly based on acoustic communication. Owing to high latency of the underwater acoustic transmission channel with respect to cabled or radio networks makes the use of conventional synchronization protocol even more challenging in underwater.

Many time synchronization algorithms can be found in literature, and all of them have similar structures even being for terrestrial sensor networks or UWSN.

Below are described three different time synchronization algorithms —found in recent literature—for terrestrial cabled or radio networks, which will be the base for the following six algorithms, that are underwater time synchronization algorithms.

All the algorithms are listed as they were published chronologically. This way is possible to observe how the state of the art evolved to most recent approaches, where this research started its basis.

6.1.1 Synchronization problem

Time synchronization problem consists on giving all the nodes of a network a common time scale to operate.

Clock synchronization is a problem for computer science and engineering which deals with the idea that internal clocks of several devices may differ. Even starting with the same base time, they will differ along time due to drifts between each clock tick count. Time synchronization algorithms are those in charge to provide time information to any electronic device, in order to allow it to get synchronized to a reference system [85].

The differences on the clocks of sensor nodes at any time is referred as the offset error between them. There are three reasons for the nodes to be representing different times in their respective clocks. The nodes might be started at different times (clock offset) as

described below, the quartz crystals at each of these nodes might be running at slightly different frequencies, causing the clock values to gradually diverge or the frequency of the clocks can change variably over time because of environment conditions such as temperature (clock skew) [86].

Clock skew

Let a wireless network has N sensors, where each sensor has a discrete-time clock with period T_N. If the nodes are isolated, the timing clock of the node N evolves as:

$$t_N(t) = tT_N + \tau_N(0) \tag{6.1}$$

where $0 \leq \tau_N(0) \leq T_N$ is an initial arbitrary phase and $t = 1, 2, 3...$ runs over the periods of the timing signal [87].

Each sensor has a clock generator crystal in it and those crystals generates a clock for the sensor and the period is T_N. Even all the sensors have been set to the same working frequency they all are affected by a set of uncertainties. That means that they may not be running at same speeds. That difference then causes a clock skew (θ). As a result, after some time, although all the sensors are started exactly the same time, they may not be at the same clock. Basically, the clock skew is a hardware problem caused by variation in the crystal frequency due to noise, temperature, aging, voltage change etc. (6.2) and the cost of the crystal increases with the accuracy and the low-cost nodes in the sensor network generally use less accurate crystals which has larger margin of errors [88]. This is the main synchronization problem because most of the times a sensor data with a time data provides an information for an application. That is, without a correct time information, data is not valuable and even it might affect application's accuracy [89].

$$\theta_B = \frac{t}{t_B} \tag{6.2}$$

Where t is the perfect time evolution, and t_B is the time evolution in clock B.

Clock offset

Without skew effect all nodes would be started at the same exact time to do not have any clock difference in the network. If not this time difference defined as clock offset has to be treated by the application to be able to create collaborative tasks. Assume that a node starts at $t = 0$ and another node starts at $t = 5$. If those nodes have exactly same periods (T), there would be a fixed clock difference between two nodes:

$$\Delta(t_2 - t_1) = 5T + \tau_2 - \tau_1 \tag{6.3}$$

Clock offset defined as $5T$ must be handled with a synchronization algorithm. Even starting at the same time (t), there will be a tiny fixed clock offset defined as $\tau_2 - \tau_1$ given by a phase deviation between clocks.

6.1.2 Synchronization systems

Principle

Most of time synchronization algorithms [15,32,90] are based in a message exchange protocol. These protocols use to send a frame from slave clock, which needs to be synchronized to a reference master clock, and then the master node sends the response to the slave. With this

structure, it is possible to time stamp all the departure and arrival times of the messages, defined in Figure 6.1 as T_1, T_2, T_3 and T_4.

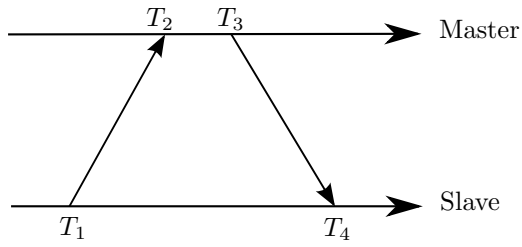


Figure 6.1: General time synchronization protocol structure, between Master node (A) and Slave node (B)

As shown in figure 6.1, a beacon initiates the synchronization process by broadcasting a request message and storing the send time of the message, T_1. Master node records the receive time of the message as per its local time, T_2. Then after a random time given by the MAC layer, master node transmits a reply message to the slave one at time t_3, time stamped by its local time. And slave clock will have all time stamps once it receives the reply message time stamped as T_4.

The local time of any node B is related to the true global time, t_A by equation (6.4).

$$t_B(t_A) = \theta_B t_A + \beta_B \tag{6.4}$$

Where, $t_B(t_A)$ denotes the local time of node B at time t_A, θ_B is the clock skew and β_B is the clock offset.

After the message exchange the slave clock can compute offset (β) and propagation time (τ) from equations 6.5 and 6.6 respectively:

$$\hat{\beta} = \frac{(T_2 - T_1) - (T_4 - T_3)}{2} \tag{6.5}$$

$$\hat{\tau} = \frac{(T_2 - T_1) + (T_4 - T_3)}{2} \tag{6.6}$$

Consequently the slave node can adjust its own clock from the reference one.

Then the main difference between cabled or RF wireless synchronization algorithms is the way how they use time stamps information, they can perform a master-slave synchronization, or a slave-slave synchronization and also try to improve time accuracy by computing time error with the neighborhood nodes besides a master clock.

Cabled systems

Timing-Sync Protocol for Sensor Networks (TPSN) TPSN protocol [32] provides synchronization in a network. The principle of the protocol is based in a pyramidal hierarchy. Foremost, one node is selected as master and it will be set at the top of the pyramid, from this node all the others will be synchronized.

In a descending hierarchy from master to salves, bidirectional synchronization frames will be sent from top to bottom. Master node handles all slave's synchronization times, in order to avoid uncertainties due to message transmission between nodes, all frames are time stamped at the MAC layer at both, reception and transmission. Furthermore, all nodes have

bi-directional links with their neighbors, which can be connected to another network master clock, correcting this way errors due to frame propagation.

With this protocol two adjacent nodes in a cabled network can be synchronized with an error against the master clock below 17 μs, in a sender-receiver synchronization that requires two sent and received messages is not the best algorithm in terms of energy efficiency for an UWSN.

Reference-Broadcast Synchronization (RBS) RBS [15] provides synchronization all over the network. Synchronization method is based on broadcast messaging from a master to a group of slave nodes. Each message sent by a master clock is received by all the salves nodes at different times, then each slave repeat the procedure, making possible to the other slave clocks to correct its time base. By receiving time references from all the nodes in the network, and applying a time synchronization regression with all the information, after 30 messages exchanges between nodes, it is possible to synchronize two clocks with an accuracy below 11 μs.

This algorithm has high power consumption because of the high number of messages exchanged to achieve good performance. Then, in a wireless stand alone network, where each node has its own limited power supply, RBS would not be the best solution for time synchronization challenge.

IEEE 1588 Precision Time Protocol (PTP) PTP is a network synchronization standard, with an accuracy down to microseconds range.

A time offset between an slave clock and a master clock is due to a combination of an initial difference between two base times, skew between clocks and a transmission delay in synchronization algorithm framing. In order to correct this time offset between two clocks PTP has two stages in the message exchange, first an offset correction and then a frame propagation time correction.

This protocol is the one with better accuracy between the cabled studied algorithms. According to studies found in literature [90], version 2 of PTP (PTPv2) presents point-to-point synchronization accuracy improvements, which results in time correction error between two clock below tens of nanoseconds.

To improve this thesis' author knowledge on time synchronization field, PTP was studied and ported to an underwater seismometer. In this study the author pretended to use a Commercial seismometer and provide the sensor clock from an external source, since this is the aim of this research. Knowing the disadvantages of cabled sensor networks, cited at introduction of this thesis this methodology was ported to acoustic wireless communication.

These three protocols are designed for irrelevant message propagation times relative to time synchronization accuracy, and this is what make them useless for acoustic communication. In underwater acoustics we have slow propagation times, of the order of 1500 m/s, what give relevance to skew effect along a time synchronization procedure, besides different paths for message transmission what causes impairments in both messages propagation times.

Underwater systems

Next six subsections detail the evolution of underwater wireless time synchronization algorithms in recent literature, where high accuracy in time synchronization was achieved because of proper compensation of both clock offset and skew.

Time Synchronization for High Latency acoustic networks (TSHL) TSHL is the first time synchronization technique introduced for high latency networks. TSHL is a two phase technique for time synchronization. The main idea in TSHL [91] is that it splits time synchronization process in the mobile UWSN into two phases. In the first phase, the nodes estimate clock skew. In the second phase in order to determine the offset, they swap skew compensated synchronization messages within the network. After the completion of the 2 phases a model is obtained that maps the local, inaccurate clock to the reference time base. Thus it is possible to compute a global time for synchronization.

In this method is assumed that the propagation is constant over the message exchange. The second assumption made in TSHL is that clocks are short term stable, which means clock frequency and skew remains constant for a short period of time (typically 5-10 min). The factors enabling the short term instability includes environmental factors such as sudden variation in temperature, supply voltage or shock. This assumption enables the use of linear regression for modeling the clock skew.

Underwater acoustic mobile networks (MU-Sync) MU-Sync [92] is a cluster-based synchronization algorithm for underwater acoustic mobile networks. Its design avoids frequent re-synchronization by estimating both the clock skew and offset.

In this protocol, the clock skew is estimated by performing the linear regression twice over a set of local time information gathered through message exchanges. The first linear regression enables the cluster head to offset the effect of long and varying propagation delays; the second regression in turn obtains the estimated skew and offset, as shown in figure 6.2.

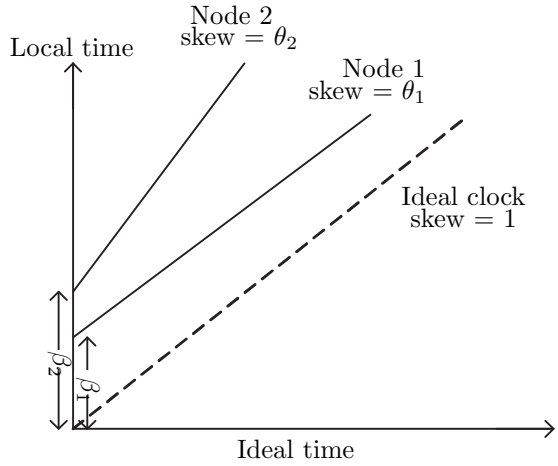


Figure 6.2: Clock offset β and skew θ computation from linear regression

This protocol overcomes previous underwater time synchronization approaches, such as TSHL, where clock skew was poorly compensated, simulation shown in figure 6.3 illustrates how MU-Sync protocol keeps its time synchronization accuracy flat after a synchronization procedure. . In fact, the TSHL is even worse than the case where no synchronization is performed. Its poor performance arises from its poor accuracy in estimating the skew. This is due to its assumption that the inter-node propagation delay is constant during the skew estimation process. When a node moves, the propagation delay varies with time; hence, if the linear regression is applied blindly without taking this into consideration, it causes inaccurate skew estimation. The small error in skew estimation can cause severe drift as

time progresses. For example, the error can grow as large as 8 s within a day of operation even with a skew error that is as small as 0.0001.

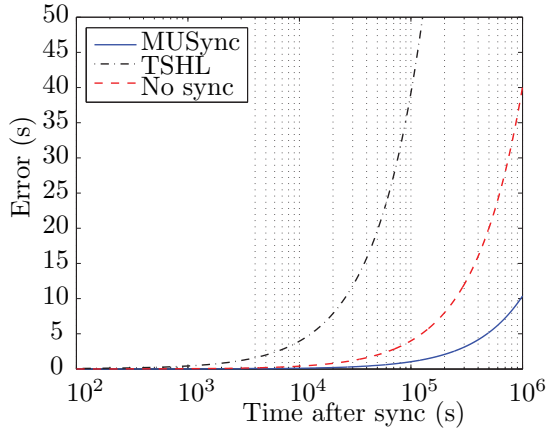


Figure 6.3: Error in time estimation as function of the time elapsed since synchronization

But the accuracy of MU-Sync is highly dependent on the accuracy of the propagation delay estimation, as it is a major contributor to synchronization error for underwater acoustic networks. They are using half of the round-trip time as an estimation of the one-way propagation delay. This may result in low accuracy if the propagation delay varies significantly within the round trip message exchange, as it is given in high reverberant scenarios.

Time synchronization method for underwater wireless sensor networks (UA-TSP) This protocol present a different approach, instead of propagating a time reference from master to slaves, it pretends to create a convergence to an averaged time between all nodes in the network [93]. Consequently it is not capable to create a master-slave architecture, where all slaves track master base-time. Furthermore, it does not correct Doppler scale effect, what it is resulting in time accuracy errors above 10 ms which is not feasible for an underwater seismologic network, or other application where time synchronization accuracy must go below 1 ms and it must track a global base time.

Doppler-based time Synchronization for mobile underwater sensor networks (D-Sync) Then, going back to a master-slave time synchronization approach, and improving MU-Sync performance, was presented D-Sync.

D-Sync synchronization algorithm [94] is also based on a frame exchange, but in this case, instead of two synchronization stages, one for offset correction, and another one for propagation time correction, it is simplified to only one message exchange.

Sending a round-trip message and computing Doppler scaling in this message besides channel impulse response estimation, leads to a determination of propagation time. So, only with a message concerning offset computation, is also possible to obtain latency information.

To sum up, this protocol computes each one of the propagation times without averaging it from round-trip message exchange which makes D-Sync to outperform MU-Sync as shown in figure 6.4 where the mean and the standard deviation of the synchronization error of both algorithms is plotted as function of the response time of a time. At their simulations, the relative speed of the node is set to 2 m/s, which is nominal velocity for self-propelled vehicles and corresponds to a higher mobility regime for devices that are not propelled. For

these simulations, it is observed that both D-Sync and MU-Sync show an increasing trend with the response time, however for different reasons. In case of MU-Sync the error increases because it assumes that nodes remain stationary during a message exchange. While in reality nodes would have moved a distance proportional to the response time. This translates to an error in the one-way propagation delay for MU-Sync. The error in D-Sync increases with the response time because Doppler measurements are available only when a message is exchanged between nodes, as opposed to a continuous estimate. When the response time grows, so does the interpolation error. However, D-Sync still outperforms MU-Sync because unlike MU-Sync, it takes node mobility into account.

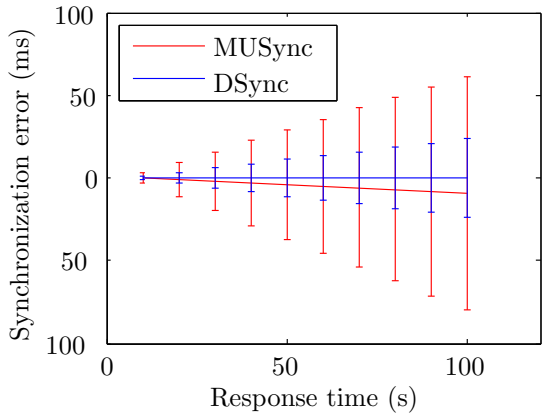


Figure 6.4: Performance with response time

Main shortcoming of this protocol is that it overlooks the effect of clock skew when estimating Doppler shift.

Efficient time synchronization for Mobile underwater sensor networks (Mobi-Sync) The distinguishing attribute of Mobi-Sync [95], is how it utilizes information about the spatial correlation of mobile sensor nodes to estimate the long dynamic propagation delays among nodes. The time synchronization procedure consists of three phases: delay estimation, linear regression, and calibration. In these three phases it uses best characteristics of the last two protocols. It computes propagation time at each message sent by using neighbors messaging in a whole network, besides using Doppler scale information for time stamps compensation.

Furthermore, this protocol adds a calibration process, where Doppler scale information is computed after a previous skew correction, which was the main leakage of D-Sync protocol.

Then, this time synchronization scheme for mobile UWSNs is the first one to utilize the spatial correlation characteristics of underwater objects, improving the synchronization accuracy as well as energy efficiency. But the main problem is the same that it is found in UA-TSP protocol, it needs a whole network to be capable to estimate an accurate time correction to a master clock.

A Doppler assisted time synchronization scheme for mobile underwater sensor networks (DA-Sync) Same authors who developed Mobi-Sync approach have designed a new methodology for acoustic UWSN time synchronization named DA-Sync [96]. It has similar design than previous one, but this time a first order kinematic model has been used

to compute propagation time at each message, instead of using neighboring messages. Then this protocol would allow to reach a high accuracy time synchronization without the necessity of using a big network.

DA-Sync is a fundamental cross-layer-designed time synchronization protocol for mobile UWSNs, with high accuracy and high energy efficiency as its major design goals, overcoming all previous time synchronization approaches.

This protocol takes into account skew effect during time synchronization procedure and it does not trust completely the measured speed with Doppler shift, which leads to synchronization errors, but rather uses a kinematic model to refine velocity estimation. DA-Sync makes a data collection process where time stamps and measured velocities are saved, with these data a velocity estimation refinement is computed by using the first order kinematic model. After this process refined velocities and time stamps are given to compute propagation delay estimation. In order to take skew into account during this process, a calibration loop is performed where skew is initialized to 1 at the first loop iteration and then by performing a weighted linear regression of propagation delays, skew is refined at each iteration up to a stabilization time.

As presented in [96] this methodology overcomes all existing time synchronization approaches.

6.2 Synchronization methodology

For our research, we will take DA-Sync protocol as reference and exploit communication layer information in order to enhance synchronization accuracy.

The two main factor that are used as information source in DA-Sync protocol are time stamps and Doppler shift information. As presented in chapters 3 and 4 respectively, these two factors have been widely studied in this thesis. So using best approach between all the presented in previous chapters, for both time stamping procedure and Doppler scale estimation, it is possible to enhance DA-Sync protocol performance.

Frame detection, described in chapter 3, is treated as a relative detection inside of an acquisition window. It means that the detection value is given in samples after the start of the acquisition, not given as a global time stamp of the sensor clock. Since it is pretended to give a global time synchronization to all the nodes of the sensor, this frame detection must be relative to the sensor clock. Then, in order to avoid nondeterministic times in time stamping procedure, given by Operating Systems (OS) or MAC layer a hardware time stamp methodology is presented in chapter 7.

Doppler scale estimation, described in chapter 4, is used for velocity and skew first estimations, so in order to reach maximum possible accuracy the best approach between all presented is chosen. Pure tone approach has concluded in the best results in all comparisons, whether in simulation, laboratory tests or OBSEA field tests. For this reason, it will be used hereafter in time synchronization protocol.

Finally in chapter 8 DA-Sync protocol will be described in detail, including on it research's new approaches in terms of time stamping and Doppler scale estimation procedures, creating this way a new time synchronization protocol for mobile underwater sensor networks capable to work not only in simulation, but also in real tests.

Chapter 7

Frame time stamp

This chapter contains the hardware description used for time stamping incoming frames.

Considering a non-deterministic time stamping for the message input or output, equations (6.5) and (6.6) are reformulated as follows:

$$\beta = \frac{((T_2 + \epsilon_2) - (T_1 + \epsilon_1)) - ((T_4 + \epsilon_4) - (T_3 + \epsilon_3))}{2} \tag{7.1}$$

$$\tau = \frac{((T_2 + \epsilon_2) - (T_1 + \epsilon_1)) + ((T_4 + \epsilon_4) - (T_3 + \epsilon_3))}{2} \tag{7.2}$$

Where ϵ_1, ϵ_2, ϵ_3 and ϵ_4 stand for the time stamp uncertainty in both incoming and outgoing messages. This uncertainty use to be around hundreds of microseconds for software time stamps taken at application layer and some nanoseconds for cabled hardware time stamps, such as the one presented in PTP [90]. Hardware time stamping has higher precision because the messages are referenced to a time base in the PHY layer, avoiding this way unpredictability introduced by OS or medium access algorithms, which can be about hundreds of microseconds.

Hence, it is necessary to port the mechanisms used in cabled time synchronization algorithms for hardware time stamping to the underwater communication algorithms in order to enhance time synchronization accuracy.

Cabled networks use a frame preamble which is identified at the PHY layer and triggers a clock acquisition. In underwater wireless networks, this method has to face with all channel challenges, such as high attenuation, multipath, frequency shifting... so it has to be redesigned to match underwater channel effects.

In literature, most of the synchronization protocols do not manage time stamping issue, as can be observed with all presented approaches in section 6.1.2. In this research, this problem is handled by creating an hybrid time synchronization model, mixing hardware time stamp, as cabled precise timing protocol do, and software frame detection, for adjusting the first sample arrival inside of an acquisition window, as presented in chapter 3. Figure 7.1 describe hybrid time synchronization methodology.

Some recent studies, such as Evologics' latest development in underwater acoustic modems for synchronization of underwater acoustic network clocks [97], use a similar approach to the one presented in this thesis. They use a hardware time stamp at the physical layer and then at the application layer global time stamp is computed.

The advantage of using this methodology is the possibility to create a distributed network. What means that the physical layer can be deployed underwater and all the other

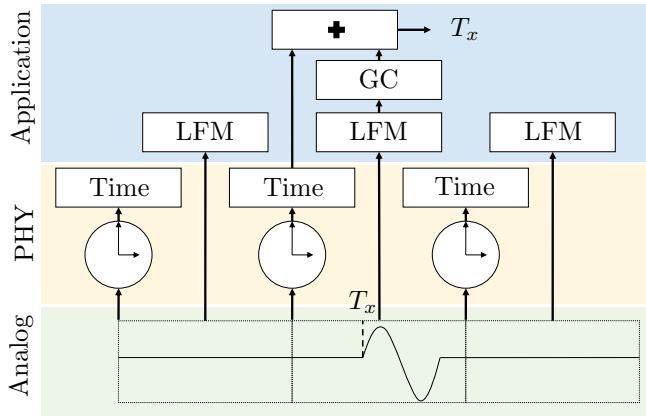


Figure 7.1: Hybrid time stamp procedure

OSI layers can be executed in a computer on-shore. By doing this propagation times between the physical layer and the rest of the system, or network throughput uncertainties are avoided. This is an optimum design for testing new algorithms and do not have to modify any underwater part to test new time synchronization designs.

Output framing detection or time stamps, can be performed by most of the commercial (Data Acquisition) DAQ systems, being capable to trigger the Digital to Analog Converter (DAC) output at a certain time with a hardware pulse. This means that the output time stamping has a deterministic and negligible error.

7.1 Hardware time stamp

In section 5.1, an FPGA and real time controller based communication system is designed. As introduced before, the main reason of using an FPGA is its re-programming functionality to match each one of the requirements of this thesis. Maybe the use of an FPGA for transmitting raw data, from an on-shore computer to a communication off-shore hardware, was not completely necessary. But for the new functionality of triggering a sensor clock acquisition at a deterministic time, and emulate a clock as described in section 6.1.1 must be done by directly programming a HW interface.

To reproduce figure 7.1 PHY functionality, the FPGA hardware description will be composed by four different loops, two of them running a clock and the other ones for acquisition and generation respectively. These loops are triggered in pair wise, one trigger for clock and generation module and the second trigger for the other clock and acquisition device. This hardware description, represented in figure 7.2, allows to emulate two different clock acquisition for master clock and slave one, what is optimum for testing all possible clock impairments. And also have separate trigger for master node transmission and slave node reception, what allows to test proper functionality of synchronization protocol, which has to compensate trigger offset.

In order to reduce hardware costs, both master and slave clocks are designed at the same FPGA, which as introduced before is capable to create a bidirectional message exchange.

With National Instruments cRIO-9103 [78], which is the FPGA module used previously in communication part, the maximum FPGA frequency is 40 MHz, what means that the minimum allowed period for the clock emulation is 50 ns. This clock speed is small enough for

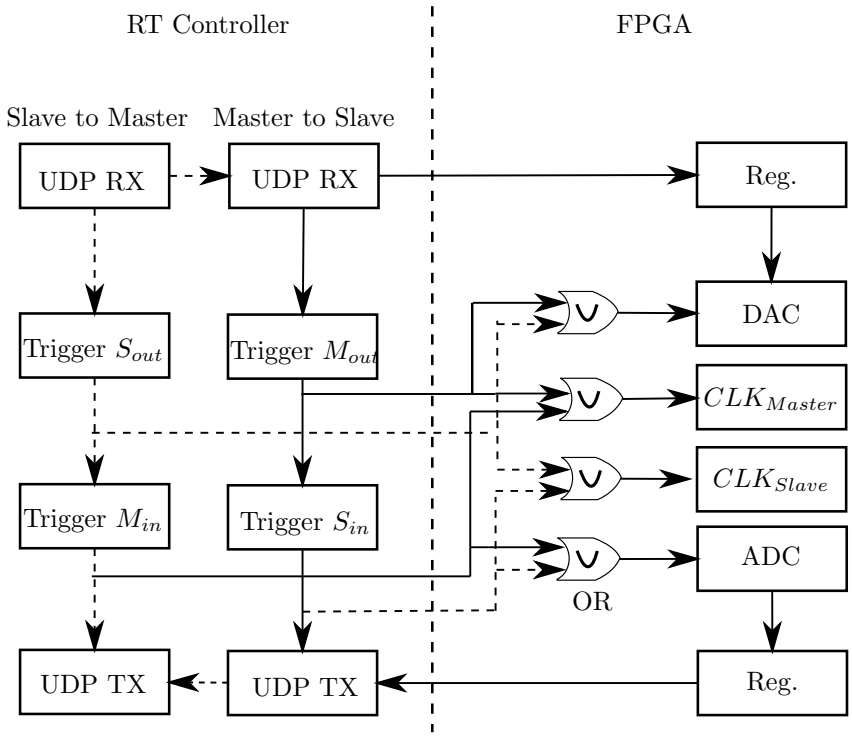


Figure 7.2: FPGA and Real Time controller work-flow

developing an underwater time synchronization algorithm, which due to high communication latency and channel challenges, use to reach a maximum accuracy of the order of tens of microseconds.

Then, for a pure hardware time stamp, the maximum generation or acquisition engine time stamp accuracy is ±50 ns.

7.2 Hybrid time stamp

Previous section has described a hardware time stamp method, but it only time stamp the generation or acquisition engine. This means that for a signal generation it can be considered as a pure hardware time stamp, but for the acquisition process, it is necessary to link the real arrival of the frame inside of an acquisition window.

Thus an hybrid time stamp procedure is used at the reception side. It mixes a hardware time stamp for the acquisition start and a software time stamp for the first sample detection inside the acquisition window, as shown in figure 7.1.

To evaluate frame detection maximum accuracy it is necessary to take into account both hardware time stamp and software frame detection accuracies. Hardware time stamping accuracy has been defined as ±50 ns in previous section, and for the software frame detection there are two methods described in this thesis (Chapter 3). The accuracy in its methods is given by the sampling rate or the timing metric plateau in Schmidl & Cox algorithm, section 3.1, and the gravity center (GC) algorithm precision in the LFM algorithm, section 3.2. These two parameters are not quantifiable, so to demonstrate the precision and robustness

against noisy environments of these two methods is performed a simulation sweeping the SNR.

For this simulation is sent a bidirectional OFDM message, keeping figure 4.4 format, enclosing time stamps information. This message is sent through a simulated physical layer, introducing multi-paths, white Gaussian noise and the time equivalent to 1 meter of distance between two sensors before the frame.

On the receiver side is performed a frame detection twice with the same incoming message, one time for the LFM algorithm and another time for the S&C one, this way it is possible to directly compare both results.

In order to simulate a real clock time readings, we reference T2, T3 and T4 to T1 as is detailed in equations (7.3), (7.4), (7.5) and (7.6). Equations (6.5) and (6.6) are the description of how do we estimate the offset β and propagation time τ between master and slave nodes respectively, which are used to correct slave node clock and evaluate synchronization accuracy between two nodes.

$$T1 = t_0 + \varepsilon_1 \tag{7.3}$$

$$T2 = T1 + \tau_1 + \beta + \varepsilon_2 \tag{7.4}$$

$$T3 = T2 + t_{wait} + \varepsilon_3 \tag{7.5}$$

$$T4 = T1 + \tau_1 + \tau_2 + t_{wait} + \varepsilon_4 \tag{7.6}$$

Table 7.1: Notation Summary

$\hat{\beta}$	Estimated clock offset
β	Real clock offset
$\hat{\tau}$	Estimated propagation delay
τ_1, τ_2	Propagation delay of A-B and B-A
ε_1, ε_2, ε_3, ε_4	Time stamp uncertainty
t_0	Initial time
t_{wait}	User defined time between messages

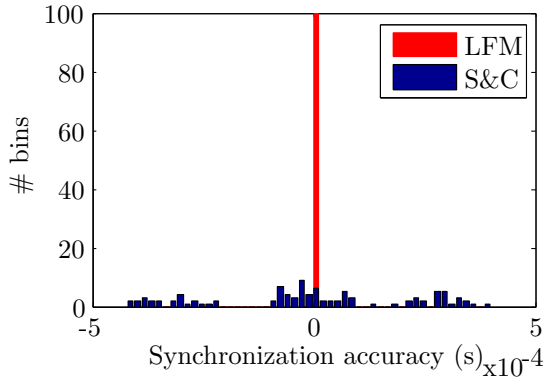


Figure 7.3: Frame arrival detection simulation

We repeat this bidirectional synchronization 100 times as can be seen in figure 7.3, which is an histogram representing the distribution of timing accuracy, defined by the difference,

after synchronize, between the master clock and the slave one. These results prove the higher performance of LFM versus S&C frame detection algorithm, due to LFM works at F_s, while S&C works at F_{sym} and plateau reduces the accuracy. LFM for frame arrival estimation applied to synchronization algorithm result in a time accuracy below 10 microseconds, while S&C plus channel equalization leads to an accuracy below 400 microseconds if we take into account the flat distribution of the synchronization accuracy represented in figure 7.3.

Using the best algorithm between the two tested in this research, the hybrid time stamp method will be highly affected by the software frame detection, while the hardware time stamping accuracy can be neglected, thus the hybrid frame time stamp procedure yields in 10 μs of time accuracy.

Chapter 8

Time Synchronization protocol

This study aims to synchronize two different sensor clocks by performing a bidirectional message exchange enclosing timing data. For doing so, we use a widely known scheme in time synchronization approaches [96] described in figure 8.2. This bidirectional message exchange encloses data transmission and reception times, besides frame relative velocities. This information is used by the slave clock estimating two key points in time synchronization: clock offset and clock skew. Figure 8.1 represents all synchronization information sources described in previous chapters and that will be used in this chapter.

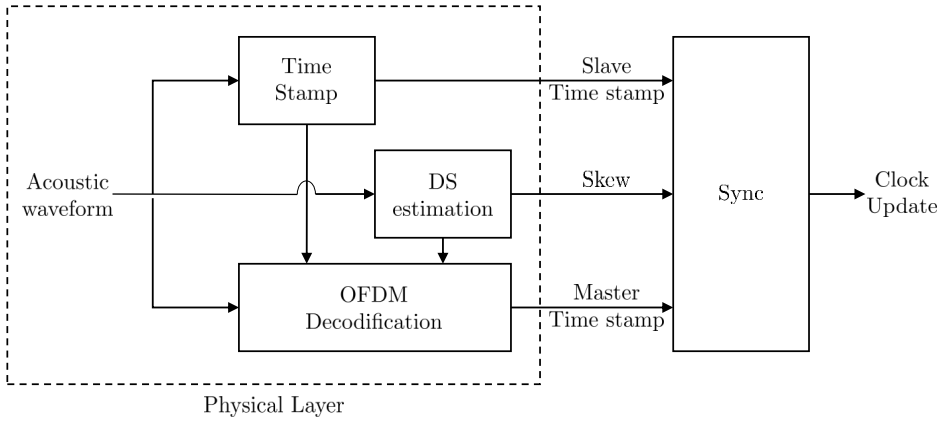


Figure 8.1: Time synchronization schema

The offset is the difference between a slave clock and a master one, this can be estimated by computing propagation times to compensate the time stamps delay due to the message exchange. With this, it is possible to compensate clock error and set the slave clock to the same time than the master one, as presented in chapter 6.

The second part, regarding clock skew, must be also taken into account to set both clocks running with the same frequency, described in section 6.1.1. If this clock skew, is not compensated the slave clock will suffer a clock drift relative to the master clock, which will not be compensated until next clock offset estimation.

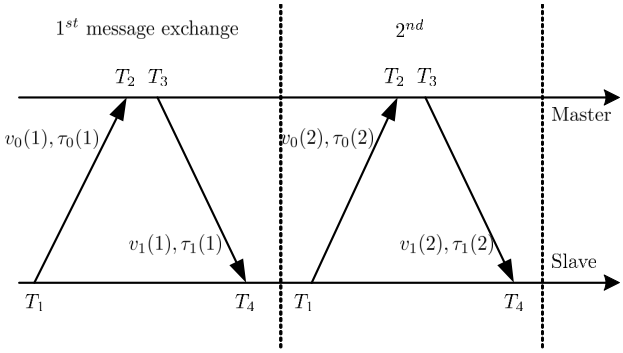


Figure 8.2: Synchronization schema.

8.1 Clock offset estimation

The slave node records the sending time stamp T_1, obtained at the PHY layer, right before the message leaves. For being able to acquire a clock reading just at the PHY, in this study is presented a new approach using both hardware time stamp by using an FPGA to determine acquisition start time, and then a software tunning in order to locate the starting sample of the message inside of an acquisition window, where besides useful information there is also channel noise. This frame time stamp method has been defined as Hybrid time stamp in previous chapter.

Upon receiving the message, the reference node also time stamps T_2, using FPGA time stamping as described in chapter 7, and estimates and records the ordinary node's relative velocity v_0 with Doppler shifts as specified in chapter 4 and message propagation time τ_0 as well. Then after a time interval t_r, the reference node sends back a message which will generate T_3, T_4, τ_0 and v_1.

Once this bidirectional messaging procedure is completed, the system is able to compute clock offset (β) and propagation time (τ), that will allow time synchronization between clocks. Depending on the accuracy requirement, the above message exchange process can be run multiple times.

Message exchange process is shown in Figure 8.2

8.2 Clock skew estimation

Clock skew is extracted from a bidirectional message exchange for initialization and after several message exchanges it is computed by performing a weighted linear regression.

First estimation comes from Doppler scale estimation, which is composed by node celerity and sensor clock skew. The impact of clock skew to Doppler scaling in equation (4.1) detailed in chapter 4 between two nodes A and B is given by equation 8.1.

Let consider two nodes A and B. In practice, each node has his own clock that can differ from the other one. Let's assume that node A has the master clock and B has the slave clock which has a drift of θ with respect to master clock. The time basis in B can be written as : $t[k] = k\theta/f_s$. The baseband received signal in B from node A noted $\tilde{y}_{AB}(t)$ is expressed as:

$$y_{AB}(t) = \sum_{p=1}^{N_p} h_p(t) e^{j2\pi f_{d,AB} t} x\big((1 + a_{AB})t - \tau_p\big) + w(t) \tag{8.1}$$

where a_{AB} (respectively a_{BA}) is the combined Doppler scale factor going from node A to node B (respectively from node B to node A) defined as [52]:

$$1 + a_{AB} = \theta(1 + a_m) \tag{8.2}$$

$$1 + a_{BA} = \frac{(1 + a_m)}{\theta} \tag{8.3}$$

and $f_{d,AB}$ (respectively $f_{d,BA}$) is the Doppler shift in the received baseband signal $y_{AB}(t)$ (respectively in $y_{BA}(t)$) defined as:

$$f_{d,AB} = \theta f_{d,m} \tag{8.4}$$

$$f_{d,BA} = \frac{f_{d,m}}{\theta} \tag{8.5}$$

As a result, by assuming that relative motion v_r is constant in a message exchange, the clock skew can be estimated from the estimates of Doppler scale factor in both nodes.

By combining (8.2) and (8.3), clock skew (θ) can be isolated from the Doppler scaling factor which has been estimated previously by one of the algorithms evaluated in section 4.

This first approach provides clock skew information, even though this is strongly affected by one frame quality. Then, by adding more information to this estimation, such as kinematic model of the nodes and a linearization of several computations, can reach a better performance on this estimation.

8.3 Data collection

For time synchronization between pair of clocks DA-Sync relies on estimating the clock offset and skew, which present the relation between the time measured by two different clocks.

For doing so, a bidirectional message exchange between nodes is used, as it has been presented above.

8.4 Velocity estimation refinement

Since velocity will be used in our synchronization algorithm for computing propagation times, and it will affect on the linear regression used for skew estimation, is necessary to estimate it as fine as possible.

Then by using kinematic equations is possible to refine the initial velocity estimation obtained from DS factor calculated in chapter 4. Assuming a first-order kinematic model with a constant acceleration between consecutive sampling times, as described in [96] we have the dynamic equation

$$x(k+1) = F(k)x(k) + \Gamma(k)w(k) \tag{8.6}$$

Where $w(k)$ denotes the discrete-time process noise, which is supposed to follow a Gaussian distribution. Then as presented in [96],

$$x(k+1) = [v(k+1) \quad \alpha(k+1)]^T \tag{8.7}$$

in which $v(k+1)$ and $\alpha(k+1)$ denote velocity and acceleration respectively. $\Delta T(k)$ can be determined in a two-point differencing procedure as

$$\Delta T(k) = \frac{\tau_0[k] + \tau_1[k]}{2} + T_3[k] - T_2[k] \tag{8.8}$$

The estimation of the state (k+1) based on the measurement of v_0 and v_1 can be obtained with Kalman filter, which is an optimal Minimum Mean Square Error (MMSE) state estimator under the Gaussian assumption of both the process noise and measurement noise, figures 8.3 and 8.4 display the estimation of velocity and acceleration refinement by using Kalman filtering in first kinematic model equations.

Regarding the kinematic model for the relative velocity we define:

$$v = k_1 t + k_2 \tag{8.9}$$

Where k_1 is a parameter closely related to environmental factors such as tides and bathymetry set at 10^{-3} , and k_2 is used to simulated some random factors. In our simulations k_2 is a random variable following a normal distribution with π m/s as the variance value and 0.1π m/s as the mean.

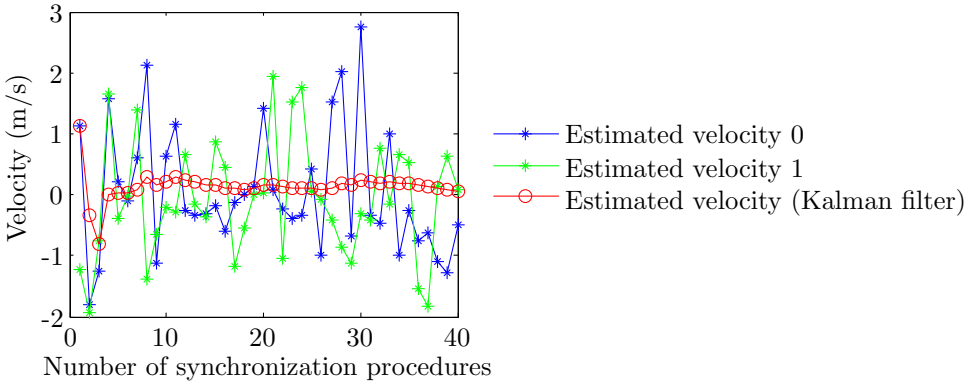


Figure 8.3: Kalman filter for velocity refinement

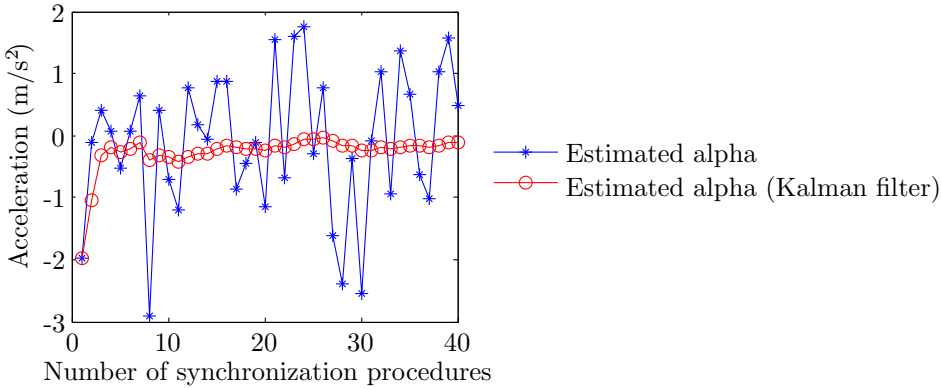


Figure 8.4: Kalman filter for acceleration refinement

For filter initialization we use measured velocities in message exchange [96]:

$$\hat{v}(1) = v_0[1] \tag{8.10}$$

$$\hat{\alpha}(1) = \frac{v_1[1] - v_0[1]}{\Delta T[1]} \tag{8.11}$$

8.5 Propagation time estimation

This phase aims to estimate the long and dynamic propagation delays.

In our synchronization schema we have to handle with τ_1 and τ_2. Then these propagation delays will be used to compute Weighted Least-Squares Estimation (WLSE) what will result in an skew estimation by computing linear regression slope.

In order to compute propagation delays, we use velocity and acceleration obtained in sections 8.2 and 8.4. Since acoustic waves propagation time is not linked with individual moving pattern of any sensor, but a relative movement between two nodes, we can work directly with relative velocities obtained from physical layer, instead of using a three dimensional velocity schema.

Following DA-Sync protocol [96] we obtain the propagation delays τ_1 and τ_2:

$$\begin{cases} \tau_1 &= \frac{(T_4-T_1)c_w-\theta(T_3-T_2)(c_w+\hat{v}_0)-\frac{1}{2}\hat{\alpha}(T_3-T_2)^2}{2\theta c_w} \\ \tau_2 &= \frac{(T_4-T_1)c_w+\theta(T_3-T_2)(\hat{v}_0-c_w)+\frac{1}{2}\hat{\alpha}(T_3-T_2)^2}{2\theta c_w} \end{cases} \tag{8.12}$$

In (8.12), θ is needed to estimate propagation times, since in this algorithm step this value is not known yet, we will set it as '1', what will be corrected in calibration procedure.

8.6 Linear regression

In this section the clock skew (θ) is estimated by using previous time stamps and propagation time computation (8.13).

Parameter β and θ are obtained by linear regression:

$$T_2[k] = \theta(T_1[k] + \tau_1[k]) + \beta \tag{8.13}$$

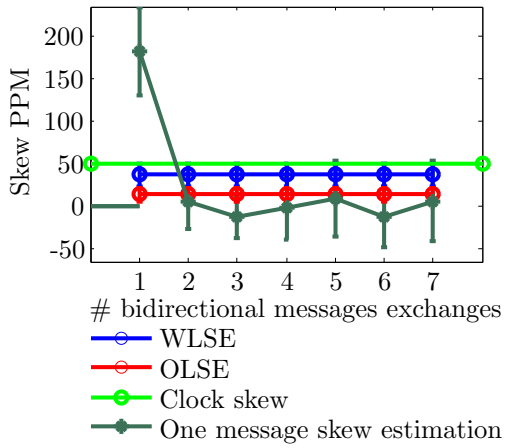


Figure 8.5: Simulation of clock skew estimation

By default, both axis of the linear regression should be identical in an ideal scenario, but they are affected by clock skew and offset.

With linear regression we obtain the offset from zero to the initial value of the regression line, which is the offset between clocks (β), and the slope of the line as clock skew (θ), where regression line is defined:

$$y = \theta x + \beta \tag{8.14}$$

In figure 8.5 is plotted the accuracy of skew estimation by using three different approaches: 1) Direct skew estimation at each message exchange; 2) Ordinary Least Squares Estimation (OLSE) [98]; 3) WLSE [99]. This test has been run 10 times for observing the variance of the estimations, then we can ensure a better performance of the WLSE for any scenario.

8.7 Calibration

For achieving this skew estimation, we have assumed in previous section $\theta = 1$ in order to estimate propagation times. So now we will need to repeat propagation delay estimation and linear regression with actual estimated skew values. This procedure will be repeated up to 10 times or θ stabilization, as described in Figure 8.6.

Table 8.1: Time synchronization parameters summary [Typ. values]

Description	Parameter	Simulation	Laboratory	OBSEA
Signal to Noise Ratio	SNR	15 dB	15 dB	15 dB
Doppler frequency shift	f_d	20 Hz	20 Hz	20 Hz
Clock offset	β	800 ms	800 ms	800 ms
Clock skew	θ	50 PPM	50 PPM	50 PPM
Node relative movement	v_0, v_1	1 m/s	1 m/s	1 m/s
Sound propagation speed in sea	c_w	1500 m/s	1500 m/s	1500 m/s
Distance between nodes	d	300 m	2 m	1.5 m
Frame propagation time	$\tau_0, \tau_1, \tau_2, \tau_3$	200 ms	1.3 ms	1 ms

8.8 Simulation

After running DA-Sync algorithm with this research own frame detection procedure and Doppler Scale factor estimation. Which are Linear Frequency Modulation and Pure tone method respectively. Figure 8.7 synchronization results are obtained after 10 seconds since last synchronization procedure. Parameters used for simulation are described in table 8.1. The inherent skew of the ordinary clock is set at 50 PPM, clock offset is initialized as 800 μs. Response time is fixed as 1 s, and the propagation speed (c_w) is 1500 m/s.

As can be observed in figure 8.7, after 8 messages exchanged an accuracy below 400 μs can be reached, going below of 200 μs after 22 messages exchanged due to Kalman filter predictor enhancement with the increment of initialization points. If system behavior after 8 messages exchanges is analyzed, time offset estimation error can be computed just after a synchronization procedure and compare its performance by following figure 8.6 refinement, or directly computing offset without any compensation. This statement is shown in figure 8.7.

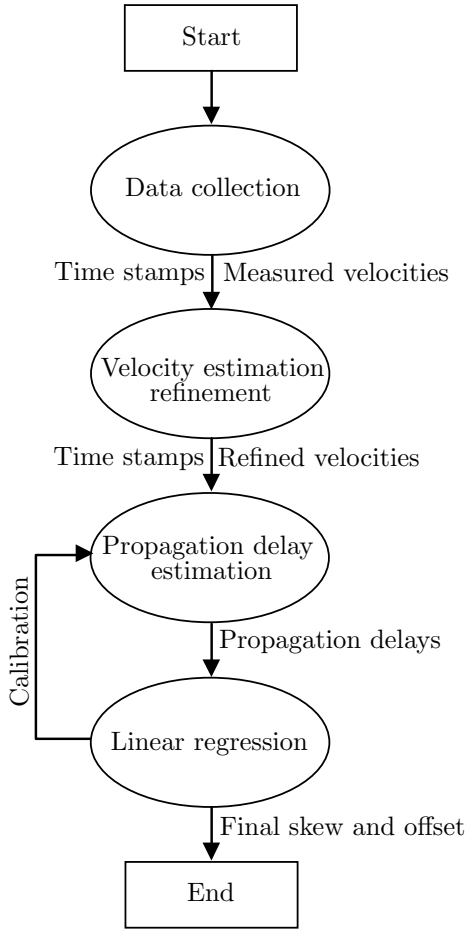


Figure 8.6: Calibration procedure

By repeating 100 times this synchronization algorithm (made up 8 messages exchange), can be observed how Kalman filtering applied in form of correction for a weighted linear regression, is close to double synchronization accuracy versus not applying any correction. Figure 8.8 shows the mean and the variation of the offset estimation after applying skew compensation, in an 8 message exchange algorithm repeated 100 times.

On the other hand, in figure 8.9, same simulation histogram is presented but this time without skew correction. An offset correction is set but skew will be drifting the clock during synchronization procedure. In this study are necessary $t_{sync} = 59$ seconds to exchange 8 bidirectional messages, so offset computation will be affected by the factor of θt_{sync}, as can be seen in the plot, where cn be observed an histogram center shifting due to clock skew.

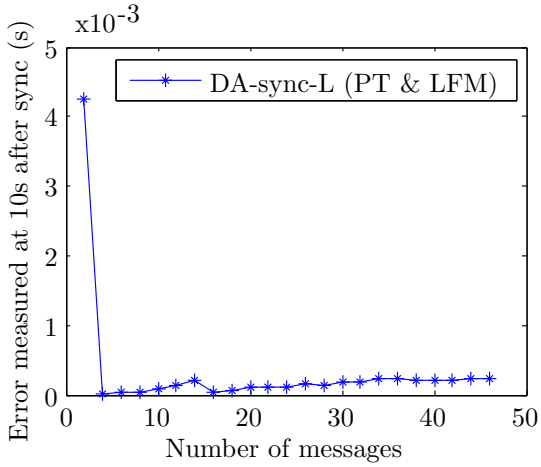


Figure 8.7: Simulation of clock offset error after 10 s versus number of messages, where DA-Sync-L means DA-Sync like protocol, that is our own application of DA-Sync protocol.

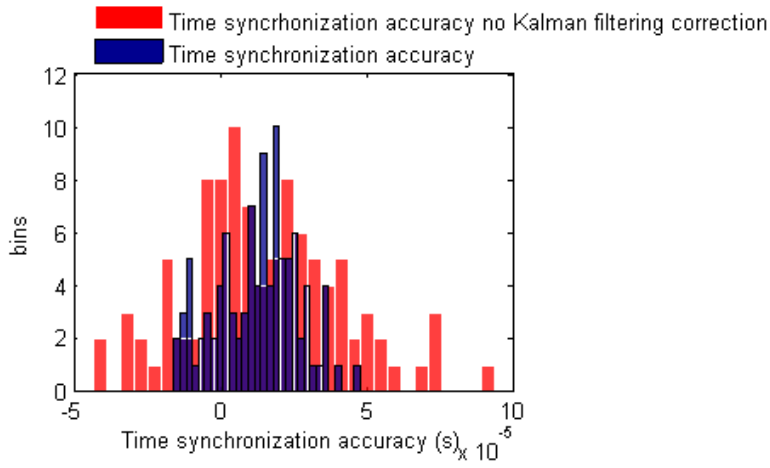


Figure 8.8: Simulation of time synchronization accuracy after 8 message exchange procedure repeated 100 times

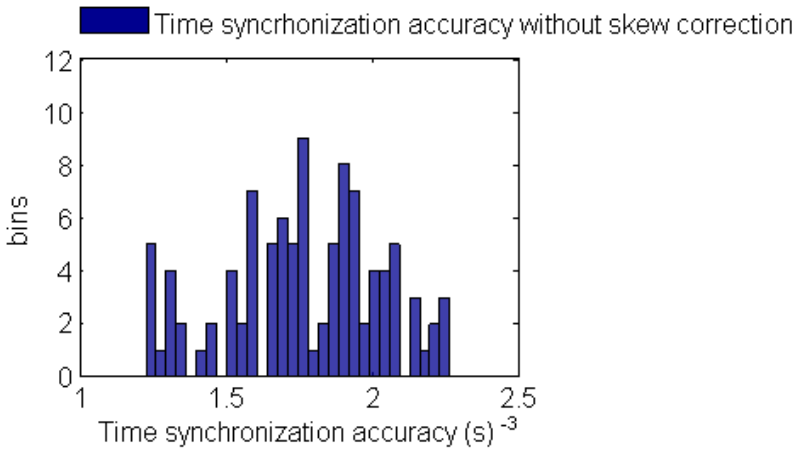


Figure 8.9: Simulation of time synchronization accuracy after 8 message exchange procedure without correcting clock skew repeated 100 times

Chapter 9

Experimental tests

This chapter contains detailed tests that pretend to demonstrate proper functionality of time synchronization protocol developed along this thesis. All tests are performed in simulation, test tank and OBSEA platform [17], and a time synchronization comparison in all these channels will be presented at the end of this chapter.

As described in chapter 8 there are two critical factors in underwater acoustic time synchronization algorithms; one is time stamping procedure and the second one is Doppler scale estimation. Since Doppler scale performance are presented in chapter 5, here are reflected hybrid time stamping results in both field tests and simulation. Then, global performance of the whole time synchronization algorithm chain are displayed.

9.1 Hybrid time stamp

Hybrid time synchronization, presented in 7.2, is now evaluated in shallow water field tests, where multipath effect is strong and can be variant along a message exchange procedure, yielding in a worsening of the performance of both LFM and S&C algorithms.

Figure 9.1 describes time synchronization accuracy as function of SNR, where one can observe that this experimental test provides similar results that simulated ones in terms of timing accuracy. SNR is modified by applying different power output to the power amplifier stage, what causes variations at the receiver side. Obviously, the SNR oscillation depends on the maximum output power of the system , and the distance between hydrophones due to the dispersion of the acoustic signal in a water channel, what fix this test SNR sweep between 5 and 20 dB

This demonstrate similar performance than in ideal hardware time stamp while the communication SNR do not decrease below 5 dB, where the receiver starts to loose frame information containing time stamps data.

Figure 9.2 shows in detail the LFM approach which is 10 times preciser than S&C one, although this second one is not affected by SNR variations.

This prove the correct behavior of hybrid time stamp procedure in real experimental tests besides simulation.

9.2 Laboratory tests

Laboratory tests are performed in a water test tank in order to evaluate proper functionality of simulated algorithms before performing sea tests in OBSEA. Same laboratory workbench

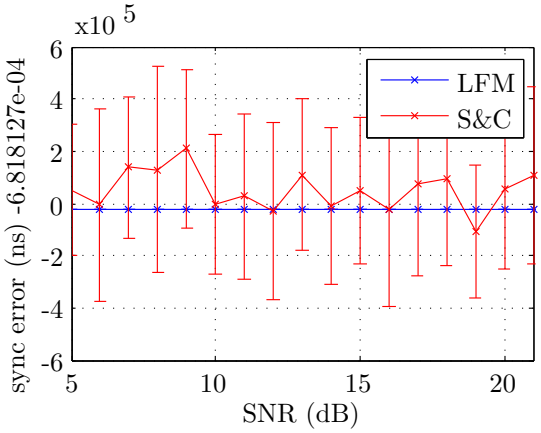


Figure 9.1: Time stamp accuracy Vs. SNR sweep from 5 dB to 20 dB

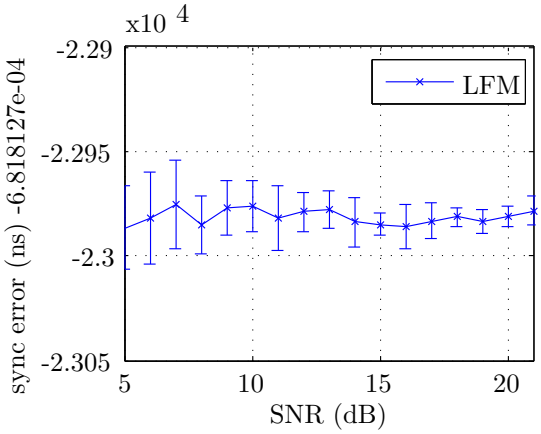


Figure 9.2: LFM Time stamp accuracy Vs. SNR sweep from 5 dB to 20 dB

of communication as section 5.2.1 is used for this trial. For this laboratory experiment, we use the same time synchronization protocol simulated in section 8.8,but instead of simulating the physical channel, this time the transmitted signal is sent through the test tank by using cRIO modules.

Since there is no water movement in the test tank, we can simulate both currents and skew by adding Doppler scaling to the OFDM signal. Since frequency shifting is constant along the frame, the results have similar accuracy than simulation ones, as shown in figure 9.3

9.3 Underwater experimental results

In real tests, time synchronization accuracy should be close to previous trial. But for this scenario sea currents control is not possible, then along one frame propagation Doppler shifting may vary, causing a Doppler scale estimation algorithm's performance loss. Moreover,

Kalman filtering and WLSE have to be limited along time in order to avoid extrapolating Doppler scaling from different sea current sources. As can be observed in figure 9.3, time synchronization accuracy begins to decrease after the 8th message, what means that after two minutes approximately since the beginning of the test. So for this case, sea currents vary significantly every two minutes. In a real application this time should be auto-adjusted in function of time synchronization error evolution, what is linked with the length of Kalman estimator and WLSE.

For this trial initial offset was set to 1 and initial skew to 1, with an estimated SNR of 15 dB, in simulation an accuracy of 179 μs is reached after 8 messages exchanged, in test tank a minimum accuracy of 170 μs is achieved, and finally in OBSEA platform, after 10 seconds running time synchronization protocol a minimum accuracy of 400 μs is reached. For all scenarios, the error is growing after the $8th$ message exchange and keeps worsening as the algorithm average during more time channel characteristics. This is due to the fact that the channel is varying and it cannot be estimated as constant along several messages exchanged as described above.

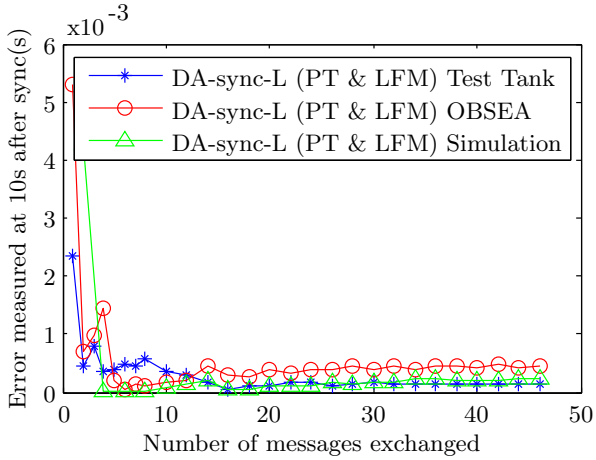


Figure 9.3: Clock offset after 10 s after last synchronization procedure Vs. number of messages exchanged for time synchronization

Chapter 10

Conclusions and future work

This chapter contains the conclusions of the work presented throughout this thesis, which is focused on the development of a time synchronization protocol for underwater sensor networks, as well as some suggestions about lines of improvement for the future.

In this research we investigated the performance of time synchronization algorithms in UWA channels with Doppler shifts. To compensate Doppler distortion we have evaluated five different frequency shift estimators, which have been applied to a time synchronization algorithm. The proposed work was tested in simulation and in a water test tank, before real-time shallow water experiments.

Regarding communication, using a SNR of 15 dB and 20 Hz of acoustic signal frequency shifting, Pure tone approach is demonstrated to outperform Preamble and Postamble, Schmidl & Cox, Time-Frequency plane and Null carrier scale estimators. Then, this Doppler scale estimator is used as source of information for the time synchronization protocol, presenting a new approach to the refinement process in the synchronization field not presented in other studies.

Over clock skew of 50 PPM, using Pure tone algorithm for Doppler scale estimation and LFM for frame detection, since this scenario provided the best results in the presented study, time synchronization accuracy after 10 seconds was 179 μs in simulation and 170 μs in Laboratory.

Finally the synchronization protocol was tested 4 km off the coast of Vilanova i La Geltrú, where, time synchronization accuracy after 10 seconds was 400 μs in OBSEA tests. Good performance was achieved even at high frequency shifting (up to 70 Hz) and low SNR (less than 15 dB). Experimental results suggest that time synchronization algorithm decrease its performance with large message exchange procedures. Since water channel medium changes its current characteristics during synchronization procedure, the Kalman filtering for velocity estimation, as described in section 8.7, is worthless.

Experimental and laboratory tests also allowed the authors to verify the proper functionality of the Hardware designed for communication and synchronization purposes. The FPGA has been proved as a good approach for time stamp procedure, overcoming other methods such as MAC layer time stamping, or software frame detection in the Operating System layer in terms of time accuracy. Communication charge and power amplifiers for a half-duplex communication are also valid for this thesis purpose. To achieve higher communication ranges, only is necessary to replace the power supply with a more powerful supply, since all the other components are designed to work up to a 5 W instead of the actual 1W.

As evidence that the study is framed within actual research studies on underwater sensor networks, this year (2016) one of the principal developers in underwater communica-

tion products, Evologics [100], has presented a novel time synchronization protocol which is integrated in one of its acoustic modems and will be used for time synchronization and positioning. This study [97], also presents a distributed OSI layer for the time stamp procedure, as it is presented in this thesis, where they use their own patented frame detection algorithm and Doppler scale estimator, yielding in a time synchronization accuracy around tens of microseconds with channel SNR set at 15 dB and Doppler shifting ±4 Hz. So similar performance to our study is achieved.

We conclude that the time synchronization protocol is valid for timing constraints above 400 μs, but it can worsen its performance in high mobility nodes, such as a fast Autonomous Underwater Vehicle (AUV), since multipath's fast variability can decrease LFM matched filter accuracy, resulting in false peak detections, what creates this way a performance worsening in time stamp procedure.

Summarizing, in Introduction (chapter 1) next items where presented as goals of the thesis. Below each ones of the items the contributions on the field are described.

- Demonstrate PHY time stamp performance in UWA environment.
  - Linear Frequency modulation and Schmidl & Cox algorithms for frame time stamp have been evaluated in both simulation and experimental tests. Tests in UWA channel resulted in an hybrid time stamp accuracy of 10 μs for the LFM approach and 400 μs for the S&C. This comparison contributes to the time synchronization algorithm to choose properly which algorithm to use at each specific scenario.
  - For this specific goal 10 publications associated to time stamping applied to time synchronization field have been published in both journals and conferences: (IMEKO'15), (IV'14), (IV'12a), (IV'12b), (SSCO'14), (IMEKO'14), (OCEANS'13a), (OCEANS'13b), (OCEANS'13c) and (MARTECH'11) as detailed in section 10.2.
  - Main contributions reached for this specific goal, from the mentioned in chapter 1 are:
    * Dedicated HW for frame time stamp in PHY layer.
    * Most used in UWSN frame detection algorithms comparison.
- Demonstrate time synchronization performance improvement when using acoustic communication physical medium information.
  - Five different Doppler scale estimators have been evaluated and compared at several SNR and frequency shifting. Between all the algorithms tested in this thesis, Pure tone approach gave best results in both simulation and experimental tests. Doppler scale information provides the source data for skew computation, thus using the most accurate Doppler scale estimator in our channel characteristics resulted in better skew estimation, and consequently better time synchronization accuracy. In this thesis has been compared time synchronization accuracy using acoustic communication physical medium information and without using it, showing this way the improvement of the order of hundreds of microseconds.
  - For this specific goal 1 publication associated to time synchronization enhancement when using acoustic communication physical medium information have been published in a conference: (OCEANS'16) as detailed in section 10.2.
  - Main contributions reached for this specific goal, from the mentioned in chapter 1 are:
    * Acoustic communication protocol adapted to provide channel information for a time synchronization protocol.

    * Most used in UWSN Doppler scale correction algorithms comparison.

- Characterize time synchronization algorithm performance when UWA channel correction factor's are applied.
  - One of the main contributions in the field of time synchronization algorithms, is the experimentation part in OBSEA platform. The novelty relies on a complete time synchronization chain, using physical channel information for time synchronization refinement, besides first order kinematic model, designed for real implementation in a distributed OSI layer along off-shore hardware and on-shore computer.
  - For this specific goal 1 publication associated to time synchronization performance characterization, when using acoustic communication physical medium information, and its functionality verification in experimental tests have been published in a journal: (JOE'16) as detailed in section 10.2.
  - Main contributions reached for this specific goal, from the mentioned in chapter 1 are:
    * Whole system design for time synchronization in Acoustic-UWSN
    * Comparison between results in simulation, laboratory tests and real field tests

10.1 Future work

The development of a research project always brings about the discovery of new problems, as well as new interesting research projects. As future work there are some lines to continue the development of the time synchronization protocol.

As mentioned in conclusions section, the problem with mobile nodes is that in channel with severe multipath they rapidly cross (small-size) coherence zones, and therefore the multipath intensity profile rapidly change. This multipath variability (changes in number and strength of arrivals, disappearing fist arrival, etc...) is one of the main contributors to the loss of hybrid frame detection accuracy. Thus, as a future work the software frame detection should be improved to handle with severe multipath environments.

At this moment, this research presents a point to point time synchronization approach, what can be ported to:

- *Network synchronization*: Time synchronization propagation along a whole network, requires to create a protocol to interact between all nodes, create a system to identify the master/slave role at each moment, handle with down nodes, etc...
- *Node positioning*: Time synchronization between two nodes permits to compute acoustic messages Time Of Flight (TOF), which is the propagation time of a frame. In the case of an underwater node can compute the TOF between itself and four surface nodes, then its position can be triangulated and related to surface nodes position. Hence, more accuracy in time synchronization yields in more accuracy in positioning algorithms. This line of research is taken by Ivan Masmitjá with his thesis:"Diseño de sistemas de posicionamiento acústico para la monitorización de especies marinas", based on acoustic positioning for marine species tracking".

Another point to study, as presented in chapter 5, is Doppler scale compensation in the communication side. In this study several methods to estimate Doppler scaling are presented and used at time synchronization protocol, but communication raw data have to be compensated too. In this research Doppler shifting is compensated in pass-band by applying

a time interpolation process. But as described in subsection 5.3.2 a base-band compensation could lead to some benefits. When the sampling frequency is close to communication central frequency, interpolation in pass-band fall behind base-band Doppler scale compensation.

Figure 10.1, is a comparative between pass-band and base-band compensation in our scripts, where we can easily observe that frame MSE decreases with a base-band correction. Then in this line of study, communication metrics could be improved.

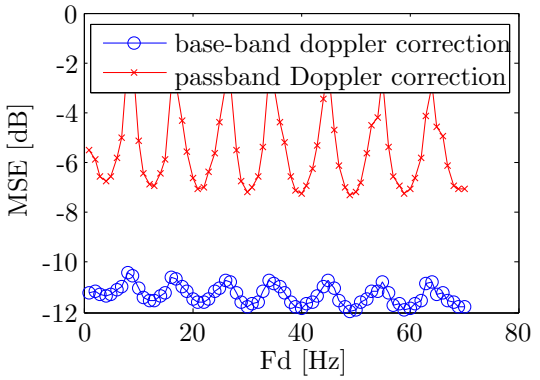


Figure 10.1: Comparative of frame MSE using Doppler scale compensation in Pass-band and in Base-band with sampling frequency set at $F_s = 100$ kHz

10.2 Publications associated to the thesis

10.2.1 Journals

(JOE'16) O. Pallarés, P.J. Bouvet, J. del Río. "TS-MUWSN: Time Synchronization for Mobile Underwater Sensor Networks". Journal of Oceanic Engineering. Accepted pending publication. ISSN 0364-9059. June 2016

(IMEKO'15) O. Pallarés, P.J. Bouvet, J. del Río. "Hybrid time synchronization for Underwater Sensor Networks". Acta Imeko. 4(3):30-35. ISSN: 2221-870X. September 2015.

(IV'14) D. Sarriá, J. del Río, O. Pallarés. "Acoustic modem for underwater sensor networks". Instrumentation Viewpoint 17. ISSN 1886-4864. Winter 2014.

(IV'12a) O. Pallarés, F.J. Cadena, D. Mihai-Toma. "COBS Time synchronization for data acquisition". Instrumentation Viewpoint 13. ISSN 1886-4864. Winter 2012.

(IV'12b) D. Sarriá, O. Pallarés, A. Mánuel. "Low-cost acoustic transmitter for underwater sensor networks". Instrumentation Viewpoint 13. ISSN 1886-4864. Winter 2012.

10.2.2 Conferences

(IMEKO'16) I. Masmitja, O. Pallarés, S. Gomáriz, J. Del Río, T. OReilly, B. Kieft. Range-Only Underwater Target Localization: Error Characterization. 21st IMEKO TC4 and 19th International workshop on ADC Modelling and testing. Budapest, Hungary, September 2016. Pending acceptation.

(OCEANS'16) O. Pallarés, P.J. Bouvet, J. del Río. "Time synchronization accuracy refinement for mobile shallow water acoustic sensor network". Oceans MTS/IEEE. Shanghai, China. April 2016.

(SSCO'14) O. Pallarés, P.J. Bouvet, J. del Río. "Underwater acoustic communication messaging time stamp applied to global time synchronization". Sensor Systems for a Changing Ocean (SSCO) - Sea tech week. Brest, France. October 2014.

(IMEKO'14) O. Pallarés, P.J. Bouvet, J. del Río. "Hybrid time synchronization for Underwater Sensor Networks". 20th Imeko TC4 and 1st TC4 workshop on metrology for Geothechnics. Benevento, Italy. September 2014.

(OCEANS'13a) O. Pallarés, F.J. Cadena, N. Carreras, D. Mihai-Toma, J. del Río. "Contribution to COBS synchronization with PTP IEEE std. 1588". Oceans 2013 MTS/IEEE. Bergen, Norway. June 2013.

(OCEANS'13b) O. Pallarés, D. Sarriá, C. Violo, J. del Río. "Preliminary OFDM based acoustic communication for underwater sensor networks synchronization". Oceans 2013 MTS/IEEE. Bergen, Norway. June 2013.

(OCEANS'13c) D. Sarriá, O. Pallarés, J. del Río. "Low cost OFDM based transmitter for underwater acoustic communications". Oceans 2013 MTS/IEEE. Bergen, Norway. June 2013.

(MARTECH'11) O. Pallarés, S. Shariat-Panahi, J. del Río, A. Mánuel. "Time synchronization of a Commercial Seismometer through IEEE-1588". Martech 2011. Cádiz, Spain. September 2011.

10.3 Publications derived from this thesis

Furthermore to publications reviewing processes, which are pretended to guarantee the quality of the research, five different articles have cited parts of the studies carried on along this thesis completion, what can be considered as a quality and novelty indicator of the thesis. (BSSA'15) and (NIMPRS'14) have cited the work presented in (MARTECH'11), which presented our first approach for an underwater cabled system with PTP IEEE std. 1588 time synchronization. (MPE'15) cites (OCEANS'13a), where experimental tests of time synchronization accuracy in OBSEA platform are presented. (ICTES'15) cites (OCEANS'13b), where an LFM frame detection method is verified in experimental tests for time synchronization purposes. Finally (NETWORKS'15) cites our work (OCEANS'13c), where was presented the first design of hardware devices for charge and power amplifying.

(ICTES'15) C. Klungmontri, I. Nilkhamhang, W. Covanich, T. Isshiki. "Underwater positioning systems for underwater robots using trilateration algorithm" In Information and Communication Technology for Embedded Systems (IC-ICTES), 6th International Conference of (pp. 1-5). IEEE. March 2015

(BSSA'15) Y. Xia, S. Ni, X. Zeng, J. Xie, B. Wang, S. Yuan. "Synchronizing intercontinental seismic networks using the 26 s persistent localized microseismic source" Bulletin of the Seismological Society of America, 105(4), 2101-2108. August 2015

(NETWORKS'15) W. Bin Abbas, N. Ahmed, C. Usama, A.A. Syed. "Design and evaluation of a low-cost, DIY-inspired, underwater platform to promote experimental research in UWSN". Ad Hoc Networks. Elsevier 34:239-251. November 2015

(MPE'15) C. Levy, M. Pinchas. "Maximum Likelihood Estimation of Clock Skew in IEEE 1588 with Fractional Gaussian Noise". Mathematical Problems in Engineering. Volume 2015. Hindawi Article ID: 174289. February 2015

(NIMPRS'14) P. Wlodarczyk, S. Pustelny, D. Budker, M. Lipińsky."Multi-channel data acquisition system with absolute time synchronization" Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 763:150-154. 2014

Appendices

Appendix A

Amplifier schematics

This appendix contains Altium Designer (14.3) schematics for the design of the amplifier module used for acoustic communication.

A.1 Power Amplifier

see figure A.1.

A.2 Charge Amplifier

see figure A.2.

A.3 Power Supply

see figure A.3.

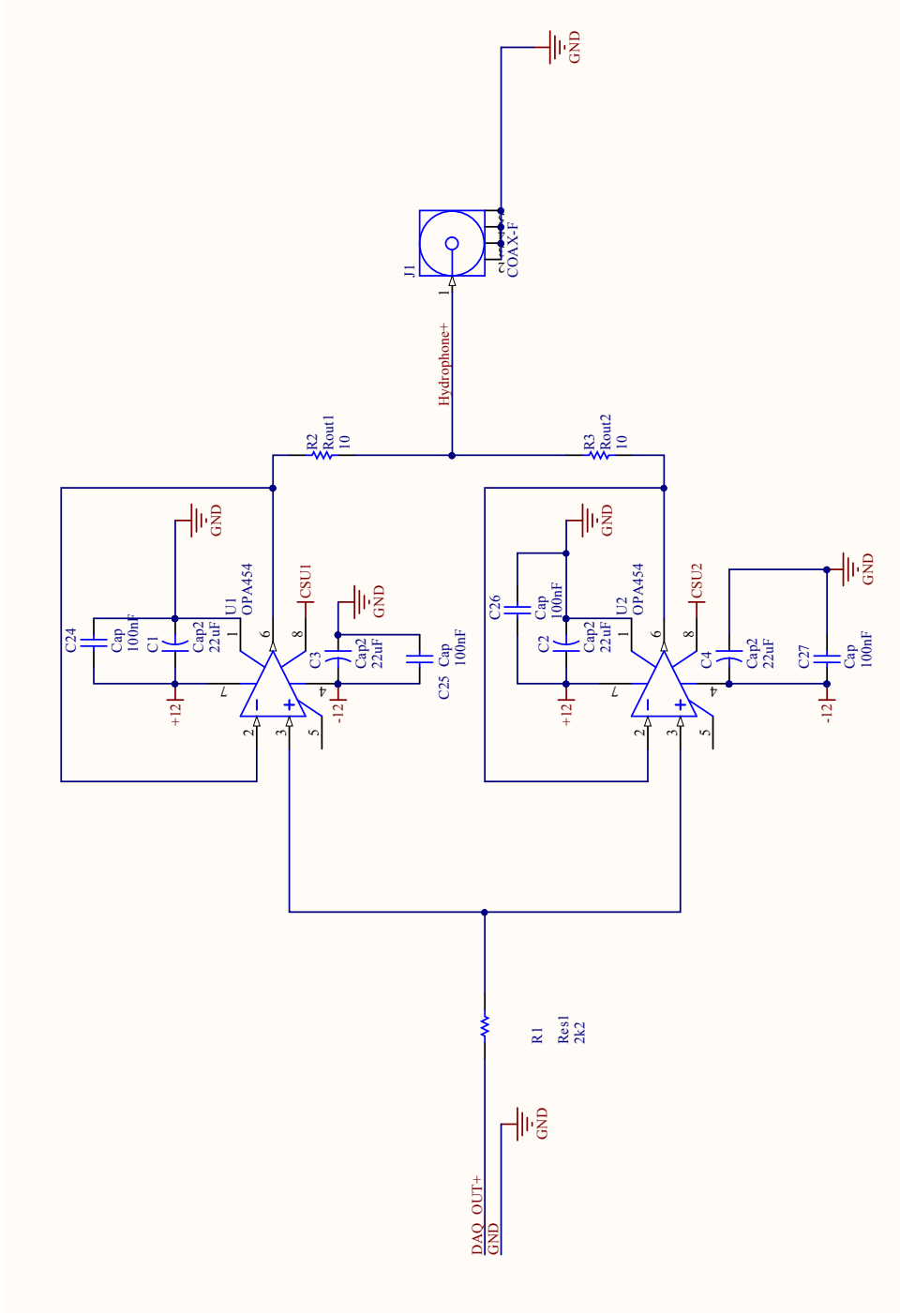


Figure A.1: Power Amplifier schematic

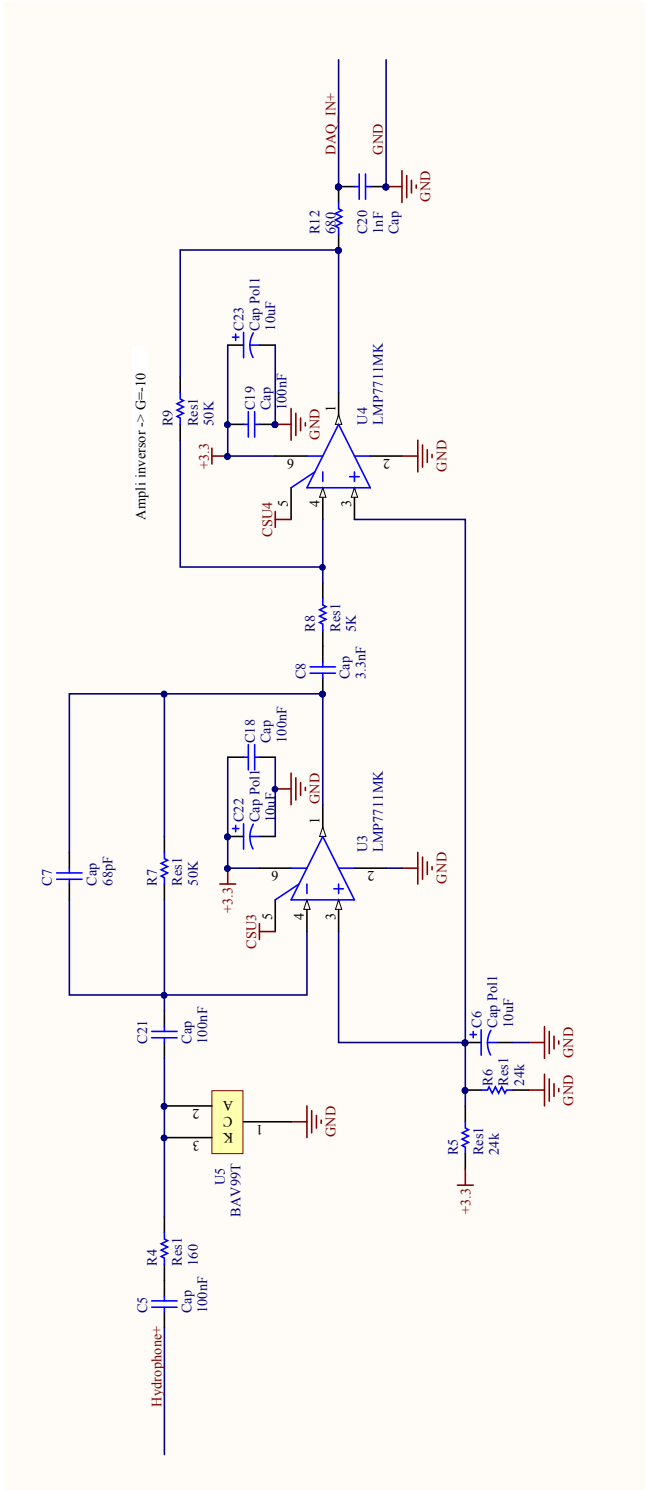


Figure A.2: Charge amplifier schematic

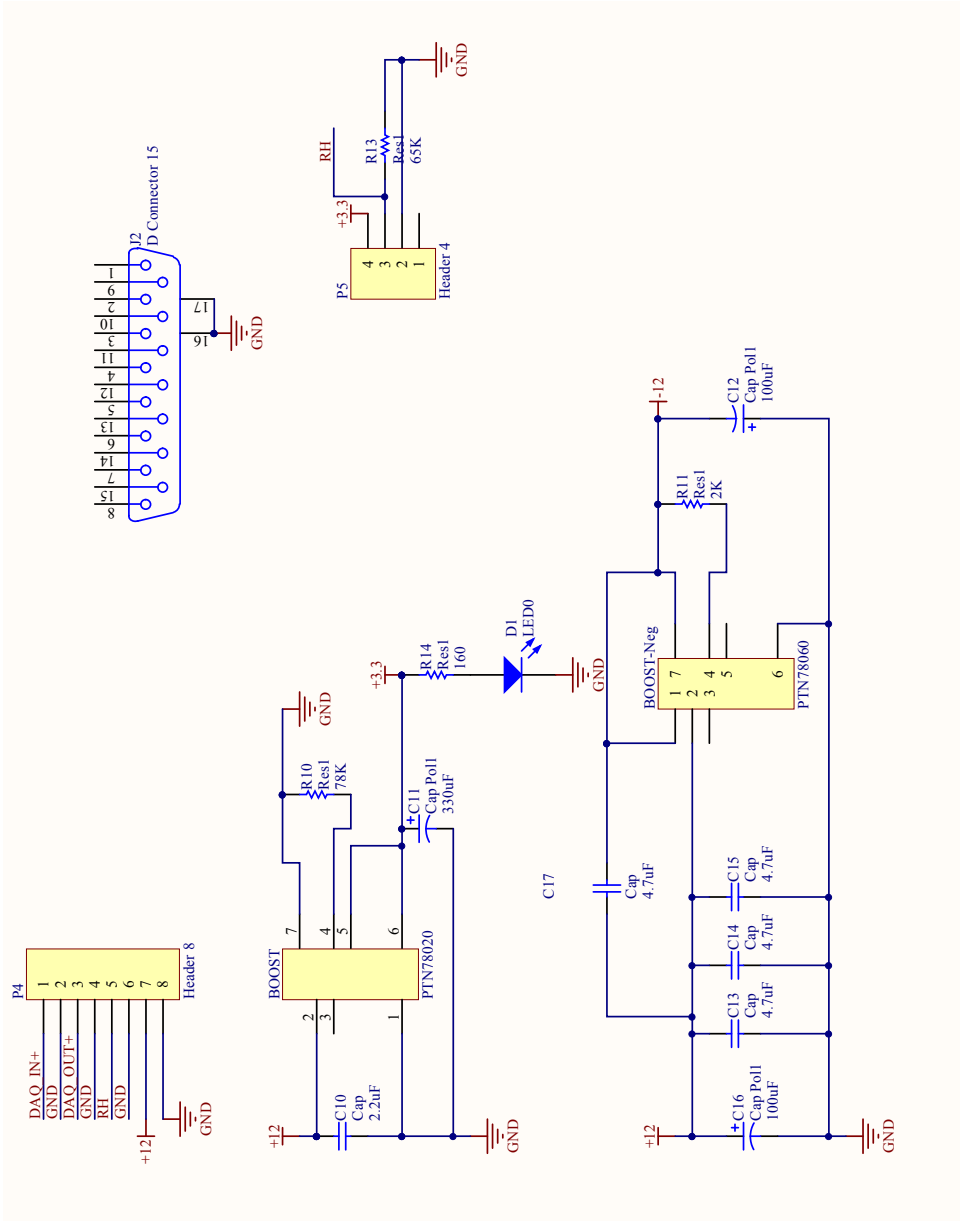


Figure A.3: Power supply schematic

Bibliography

[1] Matthew Fontaine Maury. *The physical geography of the sea and its meteorology.* Courier Corporation, 2003.

[2] N Farr, A Bowen, J Ware, C Pontbriand, and M Tivey. An integrated, underwater optical/acoustic communications system. In *OCEANS 2010 IEEE-Sydney*, pages 1–6. IEEE, 2010.

[3] Xianhui Che, Ian Wells, Gordon Dickers, Paul Kear, and Xiaochun Gong. Re-evaluation of RF electromagnetic communication in underwater sensor networks. *Communications Magazine, IEEE*, 48(12):143–151, 2010.

[4] J. Ribas and Dr M Stojanovic. *Underwater wireless video transmission using acoustic OFDM.* PhD thesis, Masters thesis, Massachusetts Institute of Technology, 2009.

[5] Ian F Akyildiz, Dario Pompili, and Tommaso Melodia. Underwater acoustic sensor networks: research challenges. *Ad hoc networks*, 3(3):257–279, 2005.

[6] John Heidemann, Wei Ye, Jack Wills, Affan Syed, and Yuan Li. Research challenges and applications for underwater sensor networking. In *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*, volume 1, pages 228–235. IEEE, 2006.

[7] Jim Partan, Jim Kurose, and Brian Neil Levine. A survey of practical issues in underwater networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(4):23–33, 2007.

[8] Bar-Shalom Yaakov, XR Li, and Kirubarajan Thiagalingam. Estimation with applications to tracking and navigation. *New York: Johh Wiley and Sons*, 245, 2001.

[9] A Quazi and W Konrad. Underwater acoustic communications. *IEEE Communications Magazine*, 2(20):24–30, 1982.

[10] Lanbo Liu, Shengli Zhou, and Jun-Hong Cui. Prospects and problems of wireless communication for underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8):977–994, 2008.

[11] P.C. Carrascosa and M. Stojanovic. Adaptive Channel Estimation and Data Detection for Underwater Acoustic MIMO-OFDM Systems. *Oceanic Engineering, IEEE Journal of*, 35(3):635–646, July 2010.

[12] S. Roy, T.M. Duman, and V.K. McDonald. Error Rate Improvement in Underwater MIMO Communications Using Sparse Partial Response Equalization. *Oceanic Engineering, IEEE Journal of*, 34(2):181–201, April 2009.

[13] Joseph Curcio, John Leonard, Jerome Vaganay, Andrew Patrikalakis, Alexander Bahr, David Battle, Henrik Schmidt, and Matthew Grund. Experiments in moving baseline navigation using autonomous surface craft. In *OCEANS, 2005. Proceedings of MTS/IEEE*, pages 730–735. IEEE, 2005.

[14] Oriol Pallarés, Javier Cadena-Muñoz, Normandino Carreras, Daniel M Toma, and Joaquin del Rio-Fernandez. Contribution to COBS synchronization with PTP IEEE std. 1588. In *OCEANS-Bergen, 2013 MTS/IEEE*, pages 1–4. IEEE, 2013.

[15] Jeremy Elson, Lewis Girod, and Deborah Estrin. Fine-grained network time synchronization using reference broadcasts. *SIGOPS Oper. Syst. Rev.*, 36(SI):147–163, December 2002.

[16] David L Mills. Internet time synchronization: the network time protocol. *Communications, IEEE Transactions on*, 39(10):1482–1493, 1991.

[17] SARTI-UPC. OBSEA Expandable seafloor observatory. http://www.obsea.es, 2009. [Online; accessed 03-May-2016].

[18] F Dunn, WM Hartmann, DM Campbell, NH Fletcher, and Thomas Rossing. *Springer handbook of acoustics*. Springer, 2015.

[19] Milica Stojanovic and James Preisig. Underwater acoustic communication channels: Propagation models and statistical characterization. *Communications Magazine, IEEE*, 47(1):84–89, 2009.

[20] D. Brady Catipovic, J. and S. Etchemendy. Development of underwater acoustic modems and networks. *Oceanography 6*, (3):112–119, 1993.

[21] J.B. Lindsay. *Mr. lindsay's marine telegraph*. Dundee Advertiser, 12th April 1853.

[22] WFS Defense. Seatext ®Product Datasheet. http://www.wfs-tech.com/, 2015. [Online; accessed 02-October-2015].

[23] WFS Defense. Seatooth ®Product Datasheet. http://www.wfs-tech.com/, 2015. [Online; accessed 02-October-2015].

[24] Sonardyne. Bluecomm Underwater Optical Modem. http://www.sonardyne.com/products/all-products/instruments/1148-bluecomm-underwater-optical-modem.html, 2015. [Online; accessed 06-June-2016].

[25] Heather Brundage. *Designing a wireless underwater optical communication system*. PhD thesis, Massachusetts Institute of Technology, 2010.

[26] JS THOMPSON and RE OWEN. VLSI in Communications. *IEEE journal on selected areas in communications*, 3(2):345–346, 1985.

[27] Samar Kaddouri. Estimated Underwater Acoustic Channel Using Multiple Sources and Multiple Broad-band receivers in Environment Fund Small and Very High Speed.

[28] Hirohisa Gambe, Toshi Ikezawa, Toshihiko Matsumura, Toshitaka Tsuda, and Shigeru Fuji. On the design of a high-performance LSI circuit digital signal processor for communication. *Selected Areas in Communications, IEEE Journal on*, 3(2):357–368, 1985.

[29] AB Baggeroer, Donald E Koelsch, Keith von der Heydt, and Josko Catipovic. DATS-a digital acoustic telemetry system for underwater communications. In *OCEANS 81*, pages 55–60. IEEE, 1981.

[30] Milica Stojanovic. Recent advances in high-speed underwater acoustic communications. *Oceanic Engineering, IEEE Journal of*, 21(2):125–136, 1996.

[31] Mérouane Debbah. Short introduction to ofdm. *White Paper, Mobile Communications Group, Institut Eurecom*, 2004.

[32] Saurabh Ganeriwal, Ram Kumar, and Mani B Srivastava. Timing-sync protocol for sensor networks. In *Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 138–149. ACM, 2003.

[33] Teledyne Benthos. 960 Series ATM-966. https://teledynebenthos.com/product/acoustic_modems/960-series-atm-966, 2015. [Online; accessed 21-September-2015].

[34] Woods Hole Oceanographic Institute. Micromodem. http://acomms.whoi.edu/micro-modem/, 2015. [Online; accessed 21-September-2015].

[35] LinkQuest Inc. UWM1000. http://www.link-quest.com/html/uwm1000.htm, 2015. [Online; accessed 21-September-2015].

[36] Evologics. S2CR 48/78. http://www.evologics.de/en/products/acoustics/s2cr_48_78.html, 2015. [Online; accessed 21-September-2015].

[37] Sercel. MATS 3G. http://www.sercel.com/products/Pages/mats3g.aspx, 2015. [Online; accessed 21-September-2015].

[38] L3 Oceania. GPM-300. http://www2.l-3com.com/oceania/products/uc_modem.htm, 2015. [Online; accessed 21-September-2015].

[39] Tritech. Micron. http://www.tritech.co.uk/product/micron-data-modem, 2015. [Online; accessed 21-September-2015].

[40] Pierre-Philippe J Beaujean and Edward A Carlson. Hermesa high bit-rate underwater acoustic modem operating at high frequencies for ports and shallow water applications. *Marine Technology Society Journal*, 43(2):21–32, 2009.

[41] Beranek LL. Acoustical measurements. New York: American Institute of Physics, 1988.

[42] VA Del Grosso and CW Mader. Speed of sound in pure water. *the Journal of the Acoustical Society of America*, 52(5B):1442–1446, 1972.

[43] Chen-Tung Chen and Frank J Millero. The use and misuse of pure water pvt properties for lake waters. 1977.

[44] Andrew H Bass and Christopher W Clark. The physical acoustics of underwater sound communication. In *Acoustic communication*, pages 15–64. Springer, 2003.

[45] Arthur B Baggeroer. Acoustic telemetry-an overview. *Oceanic Engineering, IEEE Journal of*, 9(4):229–235, 1984.

[46] Milica Stojanovic. On the relationship between capacity and distance in an underwater acoustic communication channel. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(4):34–43, 2007.

[47] Gordon M Wenz. Acoustic ambient noise in the ocean: spectra and sources. *The Journal of the Acoustical Society of America*, 34(12):1936–1956, 1962.

[48] Robert J Urick. Ambient noise in the sea. Technical report, DTIC Document, 1984.

[49] Pierre-Jean BOUVET and Alain LOUSSERT. Mimo underwater acoustic communications over shallow water channel.

[50] Shengli Zhou and Zhaohui Wang. *OFDM for Underwater Acoustic Communications*. John Wiley & Sons, 2014.

[51] James Preisig. Acoustic propagation considerations for underwater acoustic communications network development. *ACM SIGMOBILE Mobile Computing and Communications Review*, 11(4):2–10, 2007.

[52] Jun Liu, Zhaohui Wang, Zheng Peng, Michael Zuba, Jun-Hong Cui, and Shengli Zhou. TSMU: A time synchronization scheme for mobile underwater sensor networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–6. IEEE, 2011.

[53] Lei Wan, Zhaohui Wang, Shengli Zhou, TC Yang, and Zhijie Shi. Performance comparison of doppler scale estimation methods for underwater acoustic ofdm. *Journal of Electrical and Computer Engineering*, 2012:1, 2012.

[54] Baosheng Li, Shengli Zhou, Milica Stojanovic, Lee Freitag, and Peter Willett. Multicarrier communication over underwater acoustic channels with nonuniform doppler shifts. *Oceanic Engineering, IEEE Journal of*, 33(2):198–209, 2008.

[55] M Stovanovic and L Freitag. Integrated doppler tracking and efficient resampling for phase coherent acoustic communication. *IEEE of Oceanic Engineering*.

[56] Woods Hole Oceanographic Institution (WHOI). ALVIN. https://www.whoi.edu/main/hov-alvin, 2014. [Online; accessed 02-November-2015].

[57] G Cariolaro and T Erseghe. Pulse position modulation. *Encyclopedia of Telecommunications*, 2003.

[58] Harold S Black. *Modulation theory*. van Nostrand, 1953.

[59] Daniel B Kilfoyle and Arthur B Baggeroer. The state of the art in underwater acoustic telemetry. *Oceanic Engineering, IEEE Journal of*, 25(1):4–27, 2000.

[60] Milica Stojanovic, J Catipovic, and John G Proakis. Adaptive multichannel combining and equalization for underwater acoustic communications. *The Journal of the Acoustical Society of America*, 94(3):1621–1631, 1993.

[61] Milica Stojanovic, Josko Catipovic, and John G Proakis. Phase-coherent digital communications for underwater acoustic channels. *Oceanic Engineering, IEEE Journal of*, 19(1):100–111, 1994.

[62] John AC Bingham. Multicarrier modulation for data transmission: An idea whose time has come. *Communications Magazine, IEEE*, 28(5):5–14, 1990.

[63] Zhengdao Wang and Georgios B Giannakis. Wireless multicarrier communications where fourier meets shannon, department of ece. *University of Minnesota, Minneapolis MN*, pages 1–21, 2000.

[64] Louis Litwin and Michael Pugel. The principles of OFDM. *RF signal processing*, 2:30–48, 2001.

[65] Mathuranathan Viswanathan. *Simulation of Digital Communication Systems Using Matlab*. eBook, 2013.

[66] Hari Balakrishnan, Christopher J. Terman, and George C. Verghese. *Bits, Signals, and Packets: An introduction to digital communications & networks*. M.I.T. Department of EECS, 2012.

[67] Brian Borowski. *Application of Channel Estimation to Underwater Acoustic Communication*. PhD thesis, Stevens Institute of Technology, 2010.

[68] David Sarria. *Sistemas inalmbricos para la monitorizacin continua del comportamiento de especies marinas*. PhD thesis, Universitat Politecnica de Catalunya, 2014.

[69] Mingqi Li, Lin Tang, Xiaodong Zhang, Honglin Hu, and Haifeng Wang. A channel estimation method based on sample-shift of time-domain signal for ofdm systems. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pages 1430–1434. IEEE, 2007.

[70] Hlaing Minn, Vijay K Bhargava, and Khaled Letaief. A robust timing and frequency synchronization for ofdm systems. *Wireless Communications, IEEE Transactions on*, 2(4):822–839, 2003.

[71] Filippo Tosato and Paola Bisaglia. Simplified soft-output demapper for binary interleaved cofdm with application to hiperlan/2. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 2, pages 664–668. IEEE, 2002.

[72] Timothy M Schmidl and Donald C Cox. Robust frequency and timing synchronization for ofdm. *Communications, IEEE Transactions on*, 45(12):1613–1621, 1997.

[73] Fan Wu and Mosa Ali Abu-Rgheff. Time and frequency synchronization techniques for OFDM systems operating in gaussian and fading channels: A tutorial. In *Annual Postgraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting*, pages 1–6, 2007.

[74] Wei Wei, Hu Xiaoyi, Wang Deqing, Xu Ru, and Sun Haixin. Performance comparison of time synchronization algorithms for ofdm underwater communication system. In *Mechatronics and Machine Vision in Practice, 2007. M2VIP 2007. 14th International Conference on*, pages 104–107. IEEE, 2007.

[75] Jinwang Yi, Diba Mirza, Curt Schurgers, and Ryan Kastner. Joint time synchronization and tracking for mobile underwater systems. In *Proceedings of the Eighth ACM International Conference on Underwater Networks and Systems*, page 38. ACM, 2013.

[76] Poorani Kathiroli, Pierre-Philippe J Beaujean, and Nikolaos Xiros. *Source speed estimation using a pilot tone in a high frequency acoustic modem*. IEEE, 2011.

[77] Bayan S Sharif, Jeff Neasham, Oliver R Hinton, and Alan E Adams. A computationally efficient doppler compensation system for underwater acoustic communications. *Oceanic Engineering, IEEE Journal of*, 25(1):52–61, 2000.

[78] National Instruments. cRIO-9103. http://sine.ni.com/nips/cds/view/p/lang/es/nid/14158, 2010. [Online; accessed 10-May-2016].

[79] National Instruments. cRIO-9012. http://sine.ni.com/nips/cds/view/p/lang/es/nid/203347, 2010. [Online; accessed 10-May-2016].

[80] National Instruments. cRIO-9201. http://sine.ni.com/nips/cds/view/p/lang/en/nid/208798, 2010. [Online; accessed 10-May-2016].

[81] National Instruments. cRIO-9263. http://sine.ni.com/nips/cds/view/p/lang/en/nid/208806, 2010. [Online; accessed 10-May-2016].

[82] Texas Instruments. OPA-454. http://www.ti.com/lit/ds/symlink/opa454.pdf, 2007. [Online; accessed 21-September-2015].

[83] Texas Instruments. TLV2771. http://www.ti.com/lit/an/sloa033a/sloa033a.pdf, 2007. [Online; accessed 05-May-2016].

[84] I Masmitja, J del Rio, JJ de Damborenea, and A Conde. Underwater potentiostat for real-time electrochemical corrosion measurements. In *Sensor Systems for a Changing Ocean (SSCO), 2014 IEEE*, pages 1–4. IEEE, 2014.

[85] Andrew S Tanenbaum and Maarten Van Steen. *Distributed systems*. Prentice-Hall, 2007.

[86] Saurabh Ganeriwal, Srdjan Čapkun, Chih-Chieh Han, and Mani B Srivastava. Secure time synchronization service for sensor networks. In *Proceedings of the 4th ACM workshop on Wireless security*, pages 97–106. ACM, 2005.

[87] Osvaldo Simeone and Umberto Spagnolini. Distributed time synchronization in wireless sensor networks with coupled discrete-time oscillators. *EURASIP Journal on Wireless Communications and Networking*, 2007(1):1–13, 2007.

[88] Randolph Tjoa, Kim Loon Chee, PK Sivaprasad, SV Rao, and Joo Ghee Lim. Clock drift reduction for relative time slot tdma-based sensor networks. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, volume 2, pages 1042–1047. IEEE, 2004.

[89] Ali Burak Kulakli and Kayhan Erciyes. Time synchronization algorithms based on timing-sync protocol in wireless sensor networks. In *Computer and Information Sciences, 2008. ISCIS'08. 23rd International Symposium on*, pages 1–5. IEEE, 2008.

[90] Technology update on ieee 1588: The second edition of the high precision clock synchronization protocol.

[91] Affan A Syed, John S Heidemann, et al. Time synchronization for high latency acoustic networks. In *INFOCOM*, 2006.

[92] Nitthita Chirdchoo, Wee-Seng Soh, and Kee Chaing Chua. Mu-sync: a time synchronization protocol for underwater mobile networks. In *Proceedings of the third ACM international workshop on Underwater Networks*, pages 35–42. ACM, 2008.

[93] Chao Lu, Shuo Wang, and Min Tan. A time synchronization method for underwater wireless sensor networks. In *Control and Decision Conference, 2009. CCDC'09. Chinese*, pages 4305–4310. IEEE, 2009.

[94] Feng Lu, Diba Mirza, and Curt Schurgers. D-sync: Doppler-based time synchronization for mobile underwater sensor networks. In *Proceedings of the Fifth ACM International Workshop on UnderWater Networks*, page 3. ACM, 2010.

[95] Jun Liu, Zhong Zhou, Zheng Peng, Jun-Hong Cui, Michael Zuba, and Lance Fiondella. Mobi-sync: efficient time synchronization for mobile underwater sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, 24(2):406–416, 2013.

[96] Jun Liu, Zhaohui Wang, Michael Zuba, Zheng Peng, Jun-Hong Cui, and Shengli Zhou. DA-Sync: A Doppler-assisted time-synchronization scheme for mobile underwater sensor networks. *Mobile Computing, IEEE Transactions on*, 13(3):582–595, 2014.

[97] Konstantin G Kebkal, Veronika K Kebkal, Oleksiy G Kebkal, and Roberto Petroccia. Underwater acoustic modems (s2cr series) for synchronization of underwater acoustic network clocks during payload data exchange.

[98] Fumio Hayashi. Econometrics. 2000.

[99] Brent R Moulton. Random group effects and the precision of regression estimates. *Journal of econometrics*, 32(3):385–397, 1986.

[100] Evologics. Evologics GmbH. https://www.evologics.de/, 2015. [Online; accessed 08-June-2016].