Research in Industrial Projects for Students

Sponsor:
ePropulsion Limited

Optimization of an Acoustic Communication Protocol for Underwater Wireless Communication

Final Report
August 12, 2016

Tyler Chen
Xinzhi Fan
Jianhui Li
Roy Rinberg (Project Manager)

Dr. Avery Ching (Academic Mentor)
Andreas Widy (Industry Mentor)

This project was jointly supported by ePropulsion Ltd. and NSF Grant 1460018.

ABSTRACT

In this report provide an introduction to digital signal processing, specifically in the context of underwater communication. We introduce modulation, multiple access and information encoding, including the challenges faced in acoustic channels and possible solutions to such challenges. The intention is that the reader, with some background in math, be brought up to speed on the basics of underwater communication.

ACKNOWLEDGMENTS

We would like use this space to thank IPAM and HKUST for sponsoring the RISP-HK program. In particular, thanks to Stacey Beggs, Jorge Balbas, Albert Ku, Shingyu Leung, and everyone else involved in the planning and oversight of the program. We would also like to thank ePropulsion for providing us with this project. Finally, we would like to thank Andreas Widy and Dr. Avery Ching for mentoring us as we worked on this project.

CONTENTS

I Introduction . . . **2**

II The Underwater Channels . . . **3**

Electromagnetic Considerations . . . 3

Attenuation . . . 3

Noise . . . 5

Doppler Shifting . . . 6

Multipath Interference . . . 6

Calculating Usable Domains . . . 7

III Modulation . . . **8**

Orthogonal Frequencies . . . 8

Digital Modulation . . . 8

Recovering a Signal . . . 9

Multiple Frequency Shift Keying (M-FSK) . . . 10

Orthogonal Frequency Division Modulation (OFDM) . . . 10

Quadrature Amplitude Modulation (QAM) . . . 12

Multiple Domain Hybridization . . . 14

Orthogonality of Time Shifted signals. . . . 16

Accounting for Doppler Shifting in Modulation and Demodulation . . . 19

IV Multiple Access . . . **21**

V Information Accuracy . . . **24**

Linear Codes . . . 24

Effective Dictionary Creation . . . 25

VI Real Implementations . . . **26**

QAM . . . 26

OFDM . . . 26

VII Appendices . . . **33**

A Frequency Dependent α Values . . . 33

B Possible Minimization of Problems arising from Multipathing . . . 33

C Multipath Attenuation Values . . . 35

D Nearest Standard Basis . . . 35

E A Consideration on the Effective Usage of Constellations . . . 36

F Github . . . 37

I INTRODUCTION

ePropulsion Limited

Based in Hong Kong, ePropulsion Limited design and manufacture clean electric outboard motors. Originally a start-up at the Hong Kong University of Science and Technology, ePropulsion now has offices in Hong Kong and mainland China. More recently, ePropulsion have begun to branch into related fields, exploring underwater robotics and communication.

Goal

Underwater environments attract interest though an array of human activities, such as scuba-diving, submarine operation, and autonomous robotics. The demand to communicate has presented a difficult-to-answer problem. Currently, divers largely communicate using a series of prespecified hand signals; divers traveling in large groups have been known to communicate through a basic variation of morse code by hitting a metal rod against their aqua-tank to produce a loud clanging sound. This paper seeks to address the problems and possible solutions to communication in underwater environments.We seek to design a method to communicate underwater within a network of users.

Our Problem

Through the lens of information theory, our task is to transfer a message from a source to its intended receiver through a physical medium.

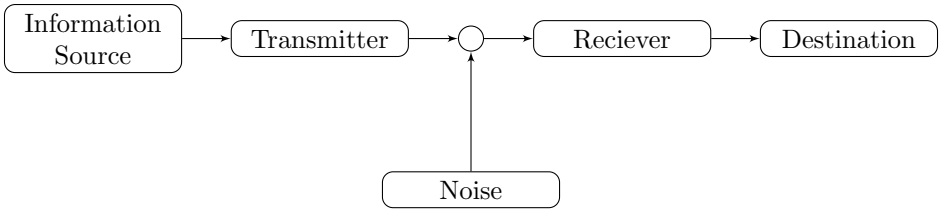


Figure 1: Model of Communication

This problem can be further decomposed into the following aspects: the physical environment, digital modulation, multi-access, and informational accuracy.

Modulation is the process of converting digital bits into a physical signal to be transmitted, and we will discuss the various methods of maximizing communication capacity through utilizing the time-frequency domain, as well as phase, and amplitude domains.

Network protocol analysis is a vital part of our project. In a group of users there must be exact communication protocol to insure accurate communication. In a group of N users, there is no way to avoid the fact that there are $N(N-1)$ unidirectional communication channels that need to be established in order for all users to be able to communicate with one another.

The last major part of our project is to consider the robustness of the information in the messages sent. In taking into account the informational accuracy of our system, we consider error-correcting codes. Additionally, it is also possible to create an effective dictionary to minimize the valuable information sent along error-prone channels.

II THE UNDERWATER CHANNELS

The goal of this project is to create an effective communication method in an underwater environment - why choose acoustics? Let us consider two methods of sending signals: through acoustic waves, and through electromagnetic (EM) waves. The physical considerations of an underwater environment are primarily: attenuation, Doppler shifting, multipath interference, and noise [1].

Electromagnetic Considerations

EM waves are used for underwater communication, and they certainly offers some advantages. The reasons people use EM waves in air largely stem from the extremely high propagation rate of EM waves (even in water, EM waves propagate at 2.25×10^8 m/s); as these speeds are much greater than anything else generally encountered in nature, the limitations on information capacity are entirely on other physical processes (such as the speed of electrical signals in wire).

However, any significant benefits of electromagnetic waves are negated by the energy loss of EM waves in water. EM energy loss in water is very difficult to model mathematically; however, it can be understood intuitively much more easily, and modeling it mathematically does not add very much more beneficial information. Experimentally, EM absorption in water has been discussed in great depth and more can be found on it as a compilation of sources, rather than a single source [2]. Certainly, EM absorption spectrum has regions where transmission can be made without great energy loss, most notably the visible light spectrum (this is almost certainly a property of evolution developing light receptors on the wavelength that can transmit through water). In water with impurity, much of the light that is transmitted through the liquid is then absorbed by the impurities. Muddy water is an excellent example of water with impurities that does not allow for the transmission of visible light. Much lower frequency EM waves can propagate through water without significant attenuation, however, EM antennae need to be much larger at lower frequencies, which is not convenient for individual divers to carry on themselves. This is why our group did not consider EM waves as a possible method of communication in an underwater environment.

Intuitively, it is clear that communication is very possible through the acoustic realm, as both humans and animals are able to communicate great distances in certain environments using acoustics.

Attenuation

When a signal is received, it is of lower intensity than when it was transmitted. This is primarily due to two phenomena, the inverse square law and energy attenuation.

The inverse square law arises from energy spreading over a larger area as distance increases. Given a constant amount of energy over an increasing wavefront, the energy density must decrease. Assuming spherical spreading, the area increases with the square of distance, and so the intensity follows an inverse square relationship.

Energy loss is a difficult process to model and takes into account a lot of frequency dependent resonant effects of water, however it can be shown to follow $V = V_0 - \alpha d$ in both air and water, for V as a measure of the volume, which is defined as $V = 10\log_{10}(I/10)$ (measured in dB), for initial volume V_0, distance from the signal source d, and constant α, which depends on a variety of factors, such as frequency, temperature, pressure, and other characterizations of impurities in the environment (e.g. humidity in air, or salinity in water) [3, 4].

From this, the intensity at a distance as a function of both the inverse square law and energy loss

is given by $I = 10^{(V_0 - \alpha d)/10} \cdot \frac{(d_0)^2}{d^2}$. The volume is then $V = V_0 - \alpha d + 20 \log_{10}\left(\frac{d_0}{d}\right)$.

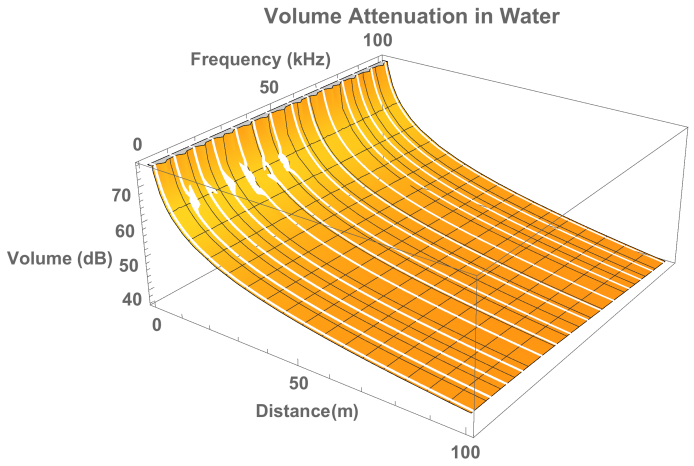


Figure 2: Plot of Attenuation of a 100 dB sound as a function of distance and frequency

Due to the resources available to our group, we did not conduct experiments in an underwater environment. We concluded that the advantages of conducting experiments underwater were too small to warrant the time, money, and effort designing the experiment. Additionally, it is clear that water conducts sound waves better than air waves, allowing us to conduct experiments in air knowing that performance in air would likely be worse than in water.

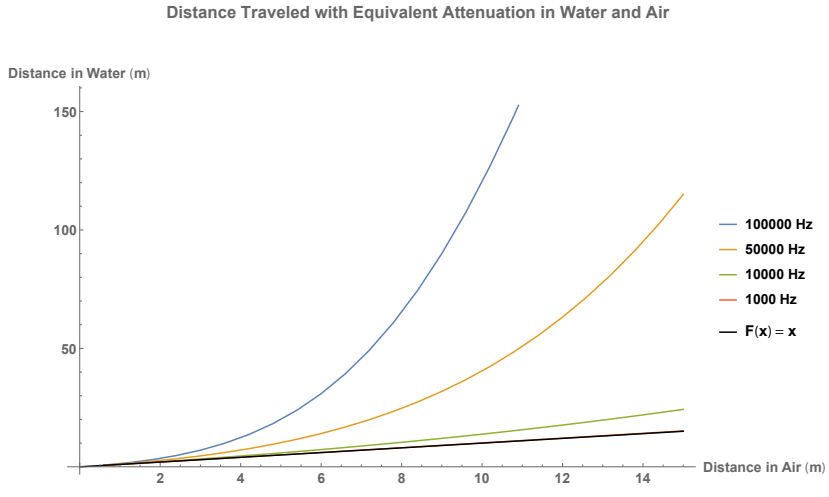


Figure 3: Plot of distances in Air vs distances in Water for Equivalent Acoustic Attenuation at various frequencies. See Appendix A for α values used

As seen in Figure 3, it is clear that attenuation is a frequency dependent effect, and that at higher frequencies water transmits sound at a much better rate than in air. However, it is important not to be misled - the rate of attenuation increases as frequency increases both in air and water, it simply increases faster in air. It is also important to note that when detecting frequencies, the sound

attenuation does not play a significant effect, up until the point where the microphone cannot pick up the signal. Sound attenuation is important to note if information is encoded across the amplitude domain - in which case, confusion could occur between the difference in a quiet signal close and a loud signal far away. To account for this, under the assumption that the distance between users does not vary greatly during a message, any amplitude modulation needs to be calculated relative to itself and not empirically. Additionally, any messages that are transmitted using amplitude modulation must be purposefully diverse with a minimum of two different (though likely more) amplitudes, so that this relative amplitude can be calculated.

Noise

Acoustic environments, particularly in an underwater setting, have generally two types of noise to consider: ambient noise, as well as random bursts of sound. Underwater environments produce background noise which can be characterized by a Gaussian distribution but has decaying power across the frequency domain while the power spectral density of ambient noise attenuates at a rate of approximately 18 dB/decade [1].

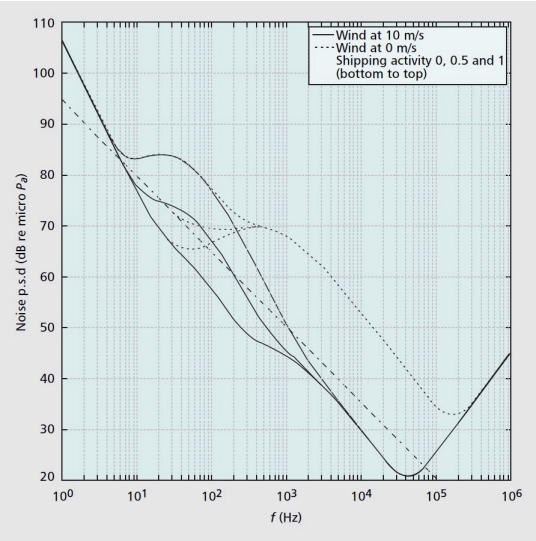


Figure 4: Simulation of Ambient Noise as a function of frequency, taken from Milica Stojanovic and James Preisig's work [1]

Ambient noise decreases approximately linearly with frequency, and attenuation increases approximately linearly with frequency, so it is important to note that the bandwidth of possible signals changes across different distances [1].

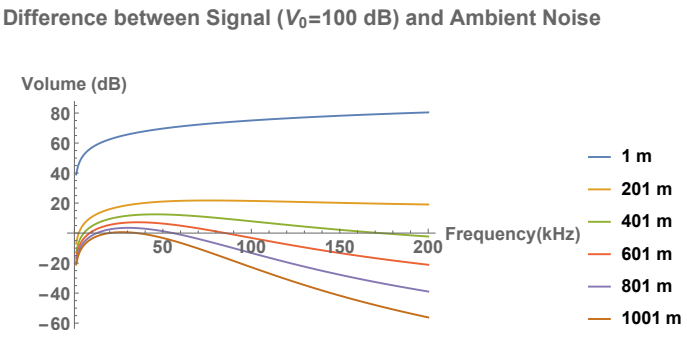


Figure 5: Simulation of difference between attenuated signal and ambient noise as a function of frequency

Nearby humans, animals, machinery, and geologic events are unpredictable and unpreventable. The only real solution to sporadic, loud noises is to encode the message robustly such that an individual noise may cause a few errors in the bits, but will not ruin the entire message's meaning.

Doppler Shifting

If the speed of the source and receiver are comparable to the speed of propagation (in the same direction), Doppler shifting can create a discrepancy between the frequencies emitted and the frequencies received.

Taking wave speed c, source speed v_s, and receiver speed v_r, the Doppler shift of an frequency f_0 emitted at the source and observed at the receiver is given by $f = \dfrac{(c + v_r)}{(c + v_s)} f_0$.

To get a general understanding of the order of magnitude of Doppler effects: some of the absolutely fastest ocean currents travel at around 2.0 m/s, and from anecdotal evidence, divers generally do not swim faster than 0.5 m/s. Taking a max net speed of 4 m/s per diver, the total Doppler shifting would contribute a frequency shift of $\sim 0.5\%$ of the total frequency. For a signal sent at 40 kHz, this can cause a shift of approximately 200 Hz.

Multipath Interference

Multipath interference is the phenomenon where the same signal from a source takes multiple paths to a single receiver and produces a blurring effect for the receiver. It is caused by one of two physical phenomena: reflection off a surface, and refraction through a medium. Multipath interference becomes more important as transmission distance increase.

Refraction occurs due to a variance in sound speed across the medium. Sound speed through water depends on a variety of parameters like temperature, salinity, and pressure, all of which depend on depth and location.

Multipathing can produce a time-smearing effect as a signal travels across a medium to produce a similar, but in fact fundamentally different, effect to Doppler shifting. Doppler shifting can stretch a series of signals in time to change the perceived length of the signals for the receiver, but the ordering of the signals is entirely preserved. Multipathing can also produce a shift in the time domain (though it can only lengthen), but is not required to preserve the ordering of the signals.

Ideally, the transmitted signal would propagate in a direct path to the intended user; noting this, the only one difference between that intended signal and the multipathed signals is the relative volumes of the signals. As noted in the Acoustic Attenuation section, acoustic attenuation increases with path length following $V = V_0 - \alpha d + 20 \log_{10}\left(\frac{d_0}{d}\right)$. Additionally, Any reflection of a signal off the ocean surface or floor, decreases the strength of the signal itself (the exact amount of decreasing depends on the type of surface, i.e. whether it is hard or soft) [1].

In Appendix B we discuss a possible method to protect against some problems arising from multipathing.

Calculating Usable Domains

We have discussed certain limitations of transmission in the acoustic domain. Given a few parameters it is possible to calculate the usable frequency and amplitude subdomains for transmitting signals. The attenuation effects in water dictate the minimum amplitudes used for transmissions to distance d, given some minimum volume detectable by the microphone. The ambient noise dictates the usable frequency bands for transmissions to distance d, taking some initial volume. The negative effects of multipathing are difficult to characterize, and while we present some speculations in the appendix, we were unable to seriously present a serious analysis of preventing the negative effects of multipathing. The effects of Doppler shifting are more complicated in their limiting effect on usable domains; problems and possible solutions arising from Doppler shifting are discussed in later sections.

III MODULATION

Modulation is the technique of encoding a message inside of a carrier signal that can be transmitted across a physical channel; for our project, this entails encoding digital bits into acoustic sound waves. In practice, messages are modulated into the time, frequency, amplitude, and phase domains.

Orthogonal Frequencies

We will often casually refer to two frequencies being orthogonal, so it is important to clarify what we mean by such a statement.

In the set $\mathcal{L}^2([0,T])$ of square integrable functions on the interval $[0,T]$ we define an inner product $\langle f,g\rangle_{[0,T]} = \int_0^T f(t)\overline{g(t)}dt$. This forms a vector space[1].

Definition. *We say frequencies f_1 and f_2 are orthogonal frequencies provided that $\langle \cos(2\pi f_1 t), \cos(2\pi f_2 t + \phi)\rangle = 0$ for all ϕ.*

For integers $m \neq n$, the frequencies m/T and n/T are orthogonal in the vector space described above. As a result, it is common practice to use carrier waves of different frequencies, as they provide a natural set of orthogonal vectors.

Proposition. *Frequencies f_1 and f_2 are orthogonal if and only if $\langle \cos(2\pi f_1 t), \cos(2\pi f_2 t)\rangle = \langle \cos(2\pi f_1 t), \sin(2\pi f_2 t)\rangle = 0$*

Proof. The forward direction follows trivially from the previous definition.

Suppose $\langle \cos(2\pi f_1 t), \cos(2\pi f_2 t)\rangle = \langle \cos(2\pi f_1 t), \sin(2\pi f_2 t)\rangle = 0$ and let $\phi \in \mathbb{R}$. Then,

$$\begin{aligned}
\langle \cos(2\pi f_1 t), \cos(2\pi f_2 t + \phi)\rangle &= \langle \cos(2\pi f_1 t), \cos(2\pi f_2 t)\cos(\phi) - \sin(2\pi f_2 t)\sin(\phi)\rangle \\
&= \langle \cos(2\pi f_1 t), \cos(2\pi f_2 t)\cos(\phi)\rangle - \langle \cos(2\pi f_1 t), \sin(2\pi f_2 t)\sin(\phi)\rangle \\
&= \cos(\phi)\langle \cos(2\pi f_1 t), \cos(2\pi f_2 t)\rangle - \sin(\phi)\langle \cos(2\pi f_1 t), \sin(2\pi f_2 t)\rangle \\
&= \cos(\phi)\cdot 0 - \sin(\phi)\cdot 0 \\
&= 0
\end{aligned}$$

□

Digital Modulation

Digital modulation (or keying) is used when a digital signal is to be represented across a physical medium. There is a discrete number of symbols that must be communicated across a continuous channel; digital modulation is the concept of sending a piece-wise combination of analog signals, each taken from a finite set.

The most basic forms keying using a sinusoidal carrier wave are Amplitude Shift Keying (ASK), Phase Shift Keying (PSK), and Frequency Shift Keying (FSK). Each of these dictates what part of the wave takes on discrete values: the amplitude, the phase, or the frequency. For instance, in FSK, playing a tone of a certain frequency will signify a certain value (such as a 1), and a different tone will signify a different value (such as a 0).

We focused much of our work on two natural extensions of these forms of keying: Orthogonal Frequency Division Multiplexing (OFDM) and Quadrature Amplitude Modulation (QAM).

[1] If it is clear which interval I we are talking about we will simply write $\langle , \rangle$ rather than $\langle , \rangle_I$

Recovering a Signal

To encode a digital message into a vector, the desired information is encoded into the coefficients of a sum of linearly independent vectors. Such that, to encode the message "1,0", one could send a 2-dimensional vector of $1 * [1, 0] + 0 * [0, 1] = [1, 0]$.

Suppose we would like to send the signal $[a_1, ..., a_N]$. We can use the digital signal $f(t) = \sum_{i=1}^{N} a_i v_i(t)$ for coefficients a_i and linearly independent carrier signals $v_i(t)$ over a time T. It is trivial to show that

$$\langle f, v_j \rangle = \sum_{i=1}^{N} a_i \langle v_i, v_j \rangle$$

and so, by solving the linear system of equations generated by this equation for the N values of v_j, each of the N coefficients a_i can be solved for. If the carrier signals $\{v_i(t)\}_{i=1}^{N}$ are orthogonal, then $\langle v_i, v_j \rangle = \delta_{i,j}$. Thus,

$$a_i = \langle f, v_i \rangle / \langle v_i, v_i \rangle = \langle f, v_i \rangle / \left\| v_i \right\|^2$$

which means we can recover the constants a_i by calculating N inner products. For this reason we generally select $\{v_i\}$ which are orthogonal.

Sampling

Physical objects and physical signals are defined continuously across the entire time domain, while computers operate on a clock cycle and values are defined only at discrete points in time. A computer can only be as exact as it is preset to be, while a physical wave is in essence infinitely precise. Engineers do not deal with continuous signals - rather, continuous signals are sampled at discrete points in time.

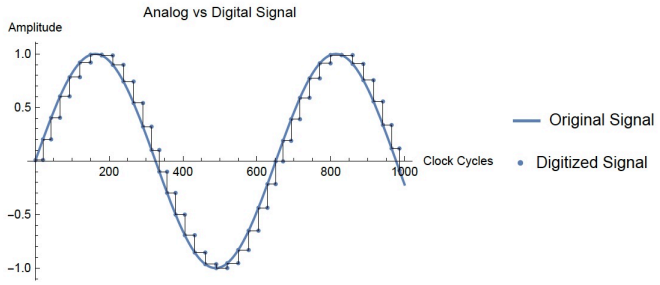


Figure 6: Plot of an analog signal against a digitized interpretation of the signal

Taking N samples at a rate of r samples per second, on the range of $t = 0$ to $t = (N-1)/r$. These samples can be thought of as a vector in $\mathbb{C}^N$.

We will not show this explicitly, but it seems intuitive that as the sum of sinusoidal functions can form an orthogonal basis for $\mathcal{L}^2[0, N/r]$, the functions $\left\{\cos\left(2\pi \frac{n}{T} t\right) + i \sin\left(2\pi \frac{n}{T} t\right)\right\}_{n=0}^{N-1} = \left\{e^{2\pi i \frac{n}{T} t}\right\}_{n=0}^{N-1}$, sampled at every $t \in \{n/r\}_{n=0}^{N-1}$ form an orthogonal basis for $\mathbb{C}^N$.

Therefore, some subset of the frequencies $\mathbb{N}/T$ is often used to transmit on.

In practice, the Discrete Fourier Transform (DFT) is often used. The DFT is an invertible linear transformation from $\mathbb{C}^N \to \mathbb{C}^N$, and can be thought of as a change of basis, from the standard

basis, where each basis vector corresponds to an amplitude at some time, to the frequency basis described above.

However, not all the frequencies contained in the basis can be used. Given some sampling rate r, any sinusoidal function with frequency greater than $r/2$ will have an alias. In other words, there is another sinusoidal function with frequency less than $r/2$ which produces the same set of points. The frequency $r/2$ is called the Nyquist Frequency, and we avoid transmitting on frequencies greater than the Nyquist Frequency altogether.

Multiple Frequency Shift Keying (M-FSK)

To modulate a signal using M-FSK to encode N-bits of information, some set $S = \{f_1, ..., f_{2^N}\}$ of 2^N orthogonal frequencies over an interval $[0, T_s]$ is chosen[2]. In a single time step T_s, N bits of data are sent by transmitting a single signal of set S, such that one frequency corresponds to one symbol.

For instance, suppose we want to transmit an N-bit digital signal $d = [a_1, ..., a_N]$, where $a_i \in \{0, 1\}$. The transmitted signal is then,

$$f(t) = \cos(2\pi f_k t), \text{ for } t \in [0, T_s], \text{ where } k = \sum_{i=1}^{N} a_i 2^{i-1}.$$

Each of the 2^N frequencies are uniquely identified with one of the 2^N possible N-bit symbols, where k gives the bijection between frequencies and symbols.

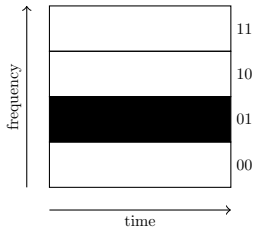
Orthogonal Frequency Division Modulation (OFDM)

In OFDM, to communicate N-bits of information, we choose some set $S = \{f_1, ..., f_N\}$ of N orthogonal frequencies over an interval $[0, T_s]$. In each symbol time T_s, N bits of data are transmitted by either send or not sending a sinusoidal wave on each of these frequencies.

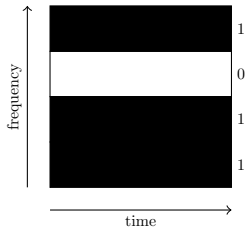
To transmit an N-bit digital signal $d = [a_1, ..., a_N]$, where $a_i \in \{0, 1\}$. The transmitted signal is,

$$f(t) = \sum_{n=1}^{N} a_n \cos(2\pi f_n t) \text{ for } t \in [0, T_s].$$

.



(a) M-FSK signal: 01



(b) OFDM signal: 1011

Figure 7: Visual representations of M-FSK and OFDM

[2]In M-FSK and OFDM, it is actually not required that the frequencies be orthogonal, or that there are 2^N of them (in the case of M-FSK), however doing so has major practical advantages and is therefore extremely common.

Figure 7b shows a visual representation of OFDM over one symbol time. A black rectangle indicates that the corresponding frequency is transmitted, while a white block indicates that the corresponding frequency is not transmitted. The figure shows 4 orthogonal frequencies, and so 4 bits are transmitted every symbol time. **maybe put in caption?**

This can be compared to M-FSK, in which exactly one frequency is played at any time. Given N orthogonal frequencies, OFDM has the advantage of being able to transmit N bits of information per symbol time, in comparison to the $\log_2(N)$ bits per symbol time possible with M-FSK. The increased data rate comes at the cost of the amplitude each band can be transmitted at. Assuming some maximum transmission power OFDM requires the power to be distributed between at most N frequencies, while M-FSK allows the entire power to be used on a single frequency.

Relation Between Maximum Bit Rate and Bandwidth

Ideally, frequencies used to send information could be arbitrarily dense and each block of information could be of arbitrarily short length. However, due to the fact that the signal received cannot be continuous, maximum bit rate cannot be arbitrarily high. Intuitively, the max bandwidth is limited by bandwidth, sampling frequency and symbol rate. We will show in this section that:

Proposition. *The maximum bit rate of a frequency domain modulated by orthogonal frequencies is equal to the bit rate, if the sampling rate is at greater than than twice the maximum frequency of the bandwidth. If a signal is modulated using orthogonal frequencies, with a sampling rate higher than twice of the maximum frequency, the maximum bit rate across the frequency domain is given by the bandwidth.*

Proof. For the sake of convenience, we first let F_s, B and T_s be the sampling rate, bandwidth and symbol time respectively.

Ideally, we are able to locate exact position of each symbol and apply a DFT on all samples. The length of the DFT[3] is $N = F_s \cdot T_s$, the number of samples per symbol time. Note that the DFT is a transformation from $\mathbb{C}^N$ to $\mathbb{C}^N$. However, the samples we process are all real valued, which means only half of the data is meaningful[4]. In other words, a DFT on real values can be regarded as a transformation from $\mathbb{R}^N$ to $\mathbb{C}^{N/2}$. On the other hand, by Nyquist theorem, we can only identify frequencies up to $\frac{F_s}{2}$. So, $\frac{N}{2}$ discrete bins are shared among $\frac{F_s}{2}$ frequencies. The resolution of the DFT is $\frac{F_s}{N}$, where $N = F_s \cdot T_s$ from previous statement. After simplification, we find that the resolution of the DFT is equivalent to the symbol rate regardless of the sampling frequency (for sufficiently large sample frequencies).

Resolution of DFTs limits the density of frequencies used to encode the data. According to Nyquist theorem, With a sampling rate greater than twice the maximum frequency of the bandwidth, any frequency in the bandwidth can be identified. So, the maximum number of frequency bands we can use is BT_s at a rate of $1/T_s$ symbols per second. So, *the maximum bit rate is equal to the bandwidth.* □

A Note on Constellations

The *I-Q* is a practical way to visualize a digitally modulated signal. All forms of keying can be pictured in terms of what is referred to as the *I-Q* plane. For any real-valued signal, at any

[3]FFT (Fast Fourier Transform) is an algorithm commonly used to replace DFT to reduce the computational time from $O(n^2)$ to $O(nlog(n))$, where n is the length of transformation. We use an FFT algorithm to calculate the DFT.

[4]This is seen in the fact the result of the DFT on real valued samples is symmetric

frequency we can decompose it into its sine and cosine components and consider the coefficients of such a decomposition. Alternatively, if we decompose into the standard basis $\{e^{2\pi i/N}\mathbf{t}_N\}_{n=0}^{N-1}$, we can consider the real and imaginary parts of the corresponding coefficient.

In other words, a sinusoidal signal of frequency f can be expressed as the sum of cosine and sine at frequency f. That is, $r\cos(2\pi ft-\phi) = r\cos(2\pi ft)\cos(\phi) + r\sin(2\pi ft)\sin(\phi) = a\cos(2\pi ft) + b\sin(2\pi ft)$ for constants $a = r\cos(\phi)$ and $b = r\sin(\phi)$ for all r, ϕ.

For any signal and chosen frequency we can plot these two parts. By convention, the cosine part (real) is plotted on the horizontal axis, denoted I, while the sine part (imaginary) is plotted on the vertical axis, denoted Q.

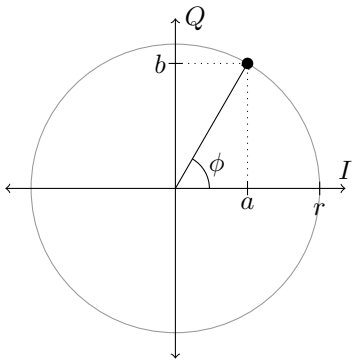


Figure 8: Function $r\cos(2\pi ft-\phi)$ on the I-Q plane (the dot)

Figure 8 shows the function $r\cos(2\pi f(t-\phi))$ on the I-Q plane corresponding to frequency f. As is apparent, the distance from the origin is given by the amplitude of the original sinusoidal wave, while the angle from the x-axis is given by the phase of the original wave.

For instance, any signals lying on a given circle about the origin all have amplitude given by the radius of the circle received on a frequency corresponding the the constellation.

Then, M-FSK or OFDM correspond to modulating the signal by whether or not the received point is at the origin, or anywhere outside the origin, for some finite number of I-Q planes

PSK corresponds to modulating the signal by the angle of a received point.

QAM corresponds to modulating the signal by a pattern parametrized by phase and amplitude at a given frequency.

Quadrature Amplitude Modulation (QAM)

Modulation

To modulate N-bits of information using QAM-2^N, we choose some set of $S = \{(I_1, Q_1), ..., (I_{2^N}, Q_{2^N})\}$ of 2^N pairs of real numbers[5].

In each symbol time T_s, N bits of data are sent by transmitting a signal

$$f(t) = I_i \cos\left(\frac{2\pi}{T_s}nt\right) + Q_i \sin\left(\frac{2\pi}{T_s}nt\right)$$

[5]Somewhat confusingly, the convention is to call a modulation that can send N bits of information "QAM-2^N, as there are 2^N different messages transmittable.

for $t \in [0, T_s], n \in \mathbb{N}$ and any one $i \in \{1, ..., 2^N\}$.

Over the interval $[0, T_s]$, $\cos\left(\frac{2\pi}{T_s}nt\right)$ and $\sin\left(\frac{2\pi}{T_s}nt\right)$ are orthogonal and so the coefficients I_i and Q_i can be recovered easily.

Notethat the frequencies n/T_s and m/T_s are orthogonal over the interval $[0, T_s]$. Therefore, QAM can be used on multiple frequencies simultaneously.

In practice, multiple signals are strung together one after another. To send the set of K symbols, $\{I_{i_1}, ..., I_{i_K}\}$ we transmit the signal

$$f(t) = I_{i_j} \cos\left(\frac{2\pi}{T_s}nt\right) + Q_{i_j} \sin\left(\frac{2\pi}{T_s}nt\right), \text{ for } t \in [(j-1)T_s, jT_s]$$

for $j \in \{1, ..., K\}$.

Alternatively two piecewise constant functions,

$$I(t) = \begin{cases} I_{i_j} & t \in [(i-1)T_s, iT_s) \\ 0 & \text{otherwise} \end{cases} \qquad Q(t) = \begin{cases} Q_{i_j} & t \in [(i-1)T_s, iT_s) \\ 0 & \text{otherwise} \end{cases}$$

can be defined for $j \in \{1, ..., K\}$.

We can then write the signal as

$$f(t) = I(t) \cos\left(\frac{2\pi}{T_s}nt\right) + Q(t) \sin\left(\frac{2\pi}{T_s}nt\right).$$

Any kind of phase modulation naturally leads to the possibility of discontinuities at times of phase shifts. In theory such discontinuities should not matter. However, in the physical channel, the speakers which transmit the signal cannot change position instantaneously, a large discontinuity results in a rapid change in the position of the speaker, causing a large spike in amplitude, as well as other mechanical inaccuracies.

To remove such discontinuities, a pulse shaping filter is applied to each $I(t)$ and $Q(t)$. If these functions are made to be continuous or differentiable, then the signal $f(t)$ will be continuous or differentiable. There are a variety of commonly used filters, such as raised cosine filters and root raised cosine filters.

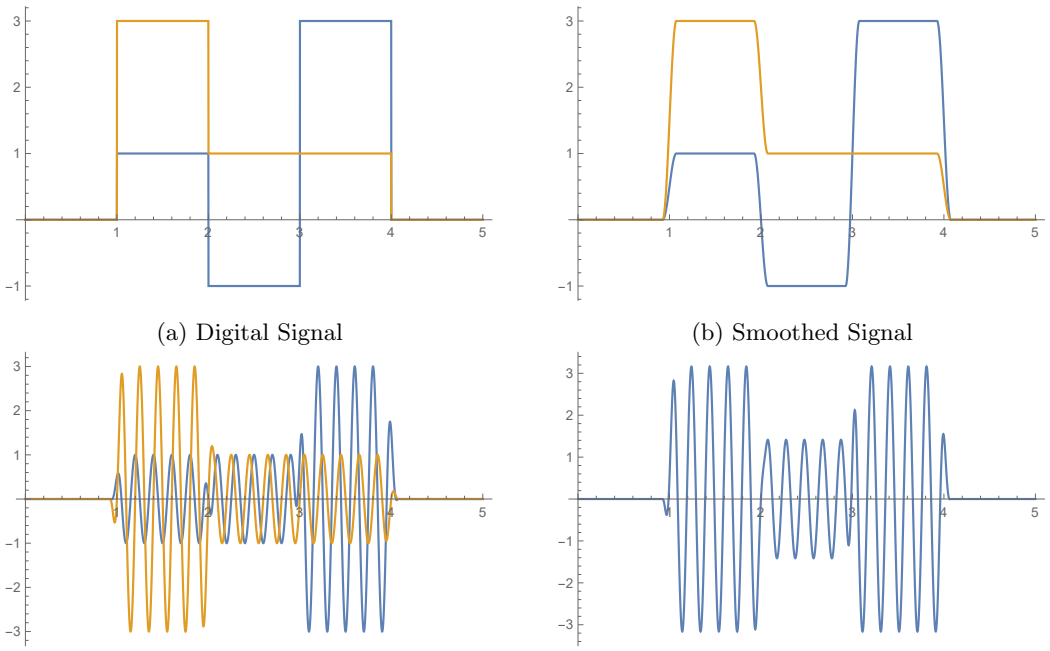
After pulse shaping, the duration for which the functions $I(t)$ and $Q(t)$ are constant is decreased. Thus, the time spent per symbol must be increased so that they are constant for at least a time T_s in order to maintain orthogonality between neighboring frequencies.

Figure 9 shows the modulation process. Figure 9a shows the digital signals, $I(t)$ and $Q(t)$ in blue and orange respectively. Figure 9b shows the same digital signal with a pulse shaping filter applied. As a result, each function becomes continuous and differentiable. Figure 9d shows the total signal, which is the sum of the two sinusoidal signals shown in Figure 9c.

Demodulation

For a received QAM signal $f(t)$, both $I(t)$ and $Q(t)$ must be recovered to determine the digital signal sent. Since $\cos\left(\frac{2\pi}{T_s}nt\right)$ and $\sin\left(\frac{2\pi}{T_s}nt\right)$ are orthogonal over an interval of length T_s, then

$$I_{j-1} = \left\langle f(t), \cos\left(\frac{2\pi}{T_s}nt\right)\right\rangle_I \qquad Q_{j-1} = \left\langle f(t), \sin\left(\frac{2\pi}{T_s}nt\right)\right\rangle_I$$



(a) Digital Signal

(b) Smoothed Signal

(c) Sinusoidal Components

(d) Modulated Signal

Figure 9: QAM modulation

where I is some interval of length T_s for which $I(t)$ and $Q(t)$ are constant.

Figure 10a shows the product of $f(t)$ with $\cos\left(\frac{2\pi}{T_s}nt\right)$ and $\sin\left(\frac{2\pi}{T_s}nt\right)$ in blue and orange respectively. The integral of each of these functions over an integral of length T_s gives the points (I_{i_j}, Q_{i_j}) pictured in Figure 10c.

It is often difficult to determine when a signal starts. This is effectively seen as a phase shift in $f(t)$. This offset will result in the graphs seen in Figures 10b and 10c, which are analogous to their neighboring figures. As expected, this results in a rotation of the points by the phase shift.

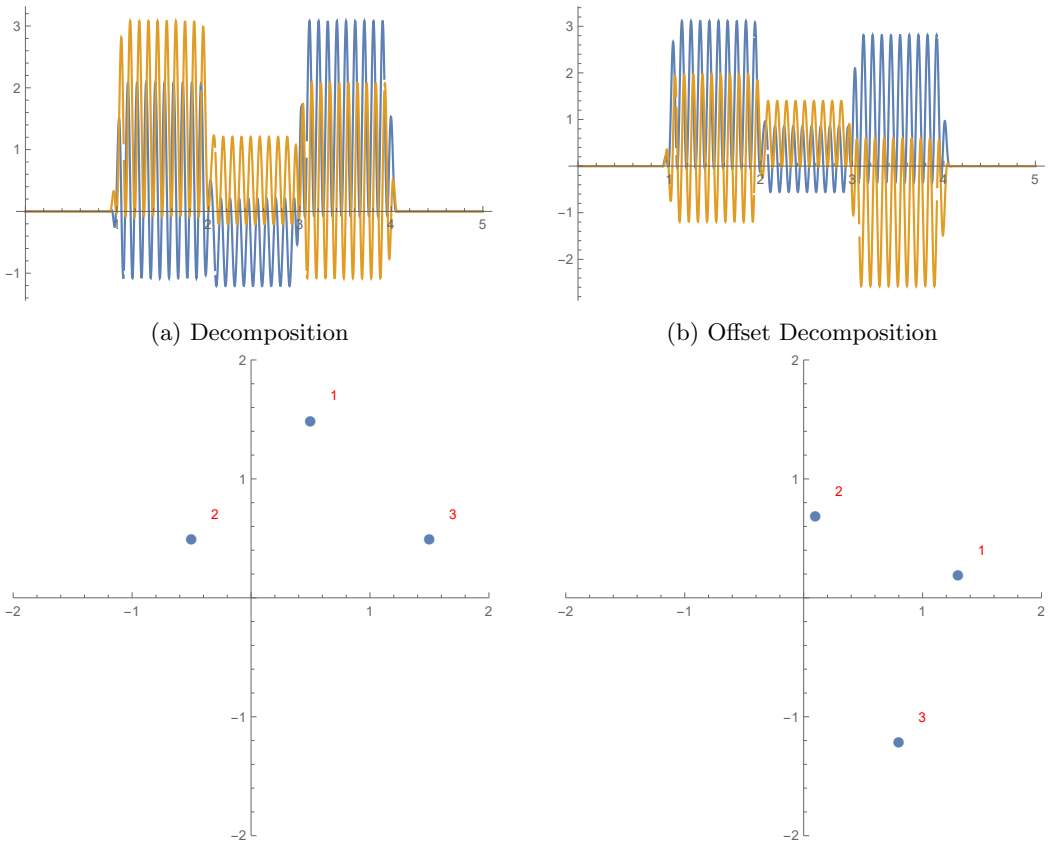
Since multiple points in our original constellation may have the same distance from the origin, it is possible for a rotation to change the message demodulated from the received signal; to counteract this, synchronization bits are used. A synchronization bit is simply a bit with a known position in the signal, and a known value. Using these bits, we calculate the difference between the received phase and the intended phase, and the signal can be rotated back to the correct position.

Multiple Domain Hybridization

QAM and OFDM Together

Having discussed modulation in both the frequency domain as well as in the phase-amplitude domain, an immediate urge is to create a hybridization of both kinds of modulation, to play on the advantages of both kinds of modulations to add in messages. One way to consider this is to take OFDM modulation, and then on any frequency bands that are being played, to utilize QAM to further encode a signal.

Another way to consider this, is that while QAM 16 is created by modulating between 4 amplitudes



(a) Decomposition

(b) Offset Decomposition

(c) Original Constellation

(d) Rotated Constellation

Figure 10: QAM demodulation

for sine and 4 amplitudes for cosine waves, one could consider that the center point of the standard rectangular 16 point constellation, where there is amplitude 0 for both sine and cosine, to be an additional point. When considering it in this fashion, it becomes clear that the benefits are there, but not hugely significant - such that QAM 16 produces 4 bits, this hybridization, otherwise known as QAM 17 produces $\log_2(16+1)$ bits.

The cost is that the density of the points near the origin increases significantly, making distinguishing each much harder. In fact, the rectangular QAM 16 we have discussed is already suboptimal in terms of spacing between constellation points and simply adding an additional point at the origin is clearly not ideal either. We discuss possible better constellations in E.

Figure 11 shows a standard rectangular QAM 16 constellation as well as the hybrid constellation discussed above. The two axes indicate the real and imaginary part of the signal received at some frequency. The light grey circle indicates the maximum power transmitted. Within such a circle, the distance between constellation points should be maximized in order to reduce the chance of being unable to distinguish two points.

(a) QAM 16 constellation (b) QAM 16 + 1 constellation (hybrid)

Figure 11: QAM constellations

Orthogonality of Time Shifted signals.

For a signal on a given frequency, if we play a single sinusoidal tone on an orthogonal frequency, the demodulation of our signal is unaffected. However, signals on a given frequency are not pure sinusoidal signals, but rather piecewise sinusoidal signals. This means that if a second signal from a different source is played on a different frequency, there is the possibility that such a signal will interfere with the first, if the start and stop of each symbol cannot be synchronized. This will result in signals no longer being orthogonal, even though the frequencies of the sinusoidal parts are orthogonal.

It is therefore important to characterize this non-orthogonality as it will be a factor in how many frequencies we can use simultaneously, and on which frequencies we can transmit.

The first property we examine is the maximum possible interference on orthogonal frequencies. We first select a signal time $T_s > 0$ and the corresponding natural choice of set of orthogonal frequencies $S = \{n/T_s\}_{n=1}^{\infty}$.

Definition. *Let $s_f(t)$ be a piecewise sinusoidal signal, with exactly one phase and amplitude change occurring at $t = T_s$. Discretize $s_f(t)$ and define*

$$\mathcal{N}_{f,n} = \left\{ \left\langle s_f(t), \cos\left(2\pi \frac{n}{T_s} t\right) \right\rangle_I^2 + \left\langle s_f(t), \sin\left(2\pi \frac{n}{T_s} t\right) \right\rangle_I^2 \right\}_{I \subseteq [0, 2T_s]}$$

for all n and all possible intervals I of length T_s contained in $[0, 2T_s]$.

Since we have discretized our signal, there are finitely many such I and so $\mathcal{N}$ is finite.

Note that if we had not discretized the signal then $\mathcal{N}_{f,n}$ would invariant under any phase change of the demodulating signals, $\cos\left(2\pi \frac{n}{T_s} t\right)$ and $\sin\left(2\pi \frac{n}{T_s} t\right)$; for a discretized signal this is no longer the case, except for shifts equal to integer multiples of the sample time. However, given a sufficiently large number of sample points, it is essentially invariant. Simulations were tested with various randomly chosen phase shifts, and the results were numerically equivalent.

Definition. *Assuming such values exist, define,*

$$\Lambda_n = \max_{f \in \mathbb{N}/T} \mathcal{N}_{f,n} \qquad\qquad \lambda_n = \min_{n \in \mathbb{N}/T} \mathcal{N}_{f,n}$$

Then $\Lambda_n = 0$ when $s_f(t)$ is orthogonal to these demodulating sinusoidal waves. Moreover, Λ_n increases as the two channels become "less orthogonal". We will use Λ_n as a measure of "worst

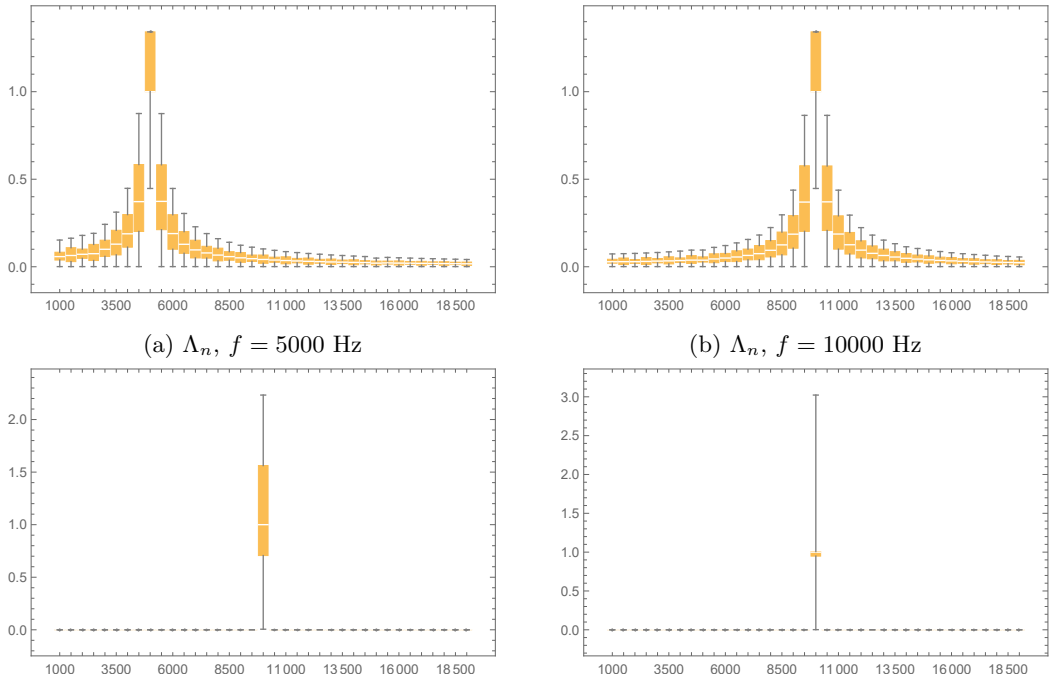
case orthogonality". If an interfering QAM signal arrives, it may be at any phase shift relative to the signal we wish to observe. Thus it is important to understand how the maximum possible interference is dependent on the frequencies of the two signals.

Similarly, λ_n will be zero if the two channels are orthogonal for any interval I. We will use λ_n as a measure of "best case orthogonality".

QAM

We first explore the most general case, QAM. As discussed in A Note on Constellations, all the forms of modulation discussed in this paper can be thought of in terms of QAM.

Figure 12 shows box and whisker plots for such a simulation repeated $N = 19$ times with randomly selected 2 symbol QAM 16 transmissions. A random phase for demodulation was chosen. Each symbol was transmitted for $1.5T_s$ to ensure there was some interval of length T_s which entirely contained each symbol. Two frequencies, $f = 5$ kHz and $f = 10$ kHz, were selected for analysis. The results have normalized to the median of the value Λ_n or λ_n for frequencies $n = 0.5, 1.0, 1.5, ..., 19$ kHz. Note that a signal corresponding to no phase shift was appended to each trial. As expected, this results in the minimum value for Λ_n being (numerically) zero for all $n \neq f$. We use a sampling rate of 384 kHz.



(a) Λ_n, $f = 5000$ Hz

(b) Λ_n, $f = 10000$ Hz

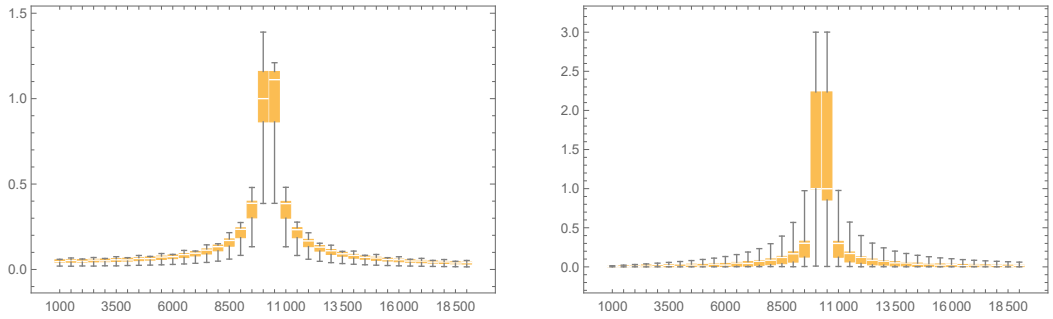
(c) λ_n, $f = 5000$ Hz

(d) λ_n, $f = 10000$ Hz

Figure 12: Normalized Γ_n, γ_n for QAM signal, $n = 0.5, 1, 1.5, ..., 19$ kHz

These plots exhibit a maximum at the analyzed frequency; the closer the frequencies are to the center frequency, the larger Λ_n is. It appears that if $|n/T - f|$ is sufficiently large, Λ_n can be made arbitrarily small.

As expected, λ_n is numerically zero for all $n/T \neq f$. This means, that there are some intervals I for which the channels are orthogonal, even with the phase shift.



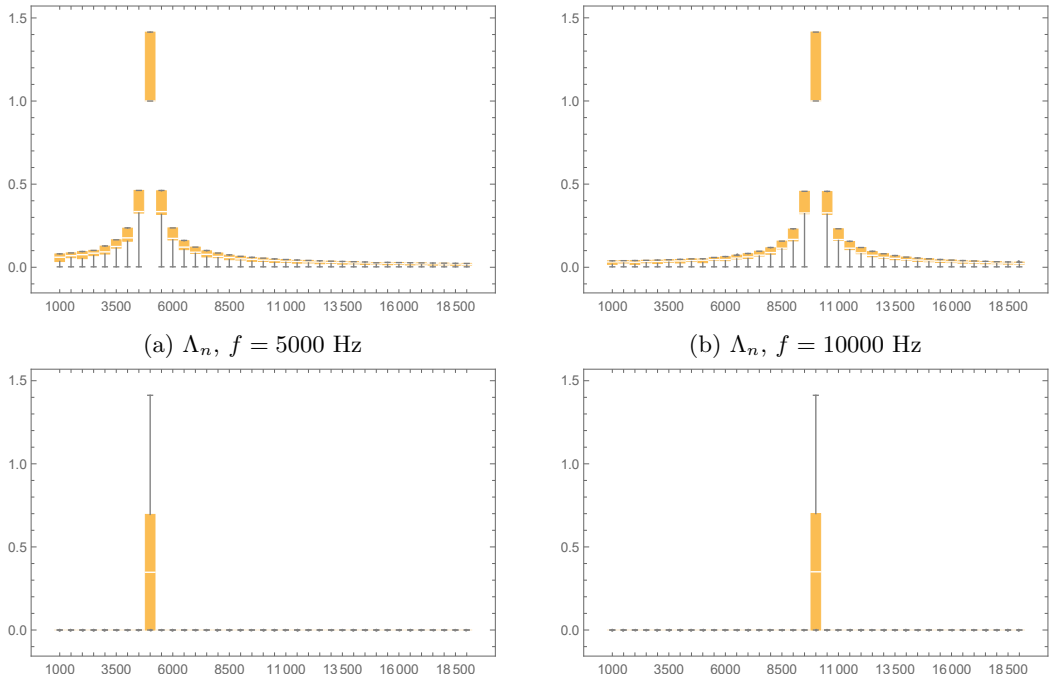
(a) Λ_n, $f = 10250$ Hz (b) λ_n, $f = 10250$ Hz

Figure 13: Normalized Λ_n, λ_n for QAM signal, $n = 0.5, 1, 1.5, ..., 19$ kHz

Figure 13 shows the results of Λ_n and λ_n when the center frequency is not orthogonal to our other frequencies. Again we see a drop off in λ_n as $|n/T - f|$ increases. As expected, λ_n is not zero for most $n/T \neq f$.

M-FSK and OFDM

As we mentioned briefly in the section "A Note on Constellations", all forms of modulation we have mentioned can be thought of in terms of QAM. This means that we can perform a similar analysis for M-FSK and OFDM by limiting the possible transitions from those considered in QAM to those used in M-FSK and OFDM.



(a) Λ_n, $f = 5000$ Hz (b) Λ_n, $f = 10000$ Hz

(c) λ_n, $f = 5000$ Hz (d) λ_n, $f = 10000$ Hz

Figure 14: Normalized Γ_n, γ_n for M-FSK/OFDM signal, $n = 0.5, 1, 1.5, ..., 19$ kHz

Figure 14 shows a similar set of experiments was performed taking the four possible transitions for given frequency. As seen, the general shape of the two plots is the same. We do however see that the values of Γ_n fall off quicker as $|n/T - f|$ increases.

Accounting for Doppler Shifting in Modulation and Demodulation

A Doppler shift of a signal $f(t) = \sum_{i=1}^{N} a_i v_i(t)$ can be expressed as

$$f'(t) = f(k_D t) = \sum_{i=1}^{N} a_i v_i(k_D t) = \sum_{i=1}^{N} a_i v_i'(t)$$

over a time T/k_D, where $v_i'(t) = v_i(k_D t)$, and k_D is a constant governing the Doppler shift.

As above, we can recover each a_i by solving the N equations generated by

$$\langle f', v_j' \rangle = \sum_{i=1}^{N} a_i \langle v_i', v_j' \rangle$$

as long as $\{v_i'\}$ is linearly independent.

Over the time T/k_D these vectors remain orthogonal, and so we can recover the signal without issue.

This also applies in the discrete case provided k_D is of the form $k_D = \frac{N-1}{K}$ for some $K \in \mathbb{N}$.

Even if k_D is not of this form, in practice, given that k_D we considered is within 1 percent, the standard basis used in the DFT remain linearly independent, and so coefficients a_i can be recovered by recalculating each each shifted vector v_i' and solving a system of N equations.

"Nearest" Standard Basis

Given that functions to compute the DFT of a discrete sample are common, ideally we would be able to compute the DFT of a Doppler shifted basis. In reality, the Doppler shifted v_i may not align with any of the standard basis vectors for the DFT in any dimension.

Appendix D gives insight into how close the nearest standard basis is to our Doppler shifted basis. Evidence is provided to support the claim that that given given either a desired frequency band, or a desired number of DFT samples, there is a standard orthogonal basis which matches our frequencies very closely.

The tradeoff between the method outlined in Appendix D and solving a linear system for the coefficients is between accuracy and computational time, although this analysis suggests that in most cases just using a DFT on the number of samples closest to the expected shift is "good enough".

In practice, the Doppler shift is not known, and so it must be estimated. The algorithm used to determine the Doppler shift and number of points to use in the DFT is shown in Algorithm 1.

Algorithm 1 Determining Optimal Sample Number to account for Doppler shift

1: **procedure** FINDSHIFTEDSAMPLENUMBER(raw_sample)
2: $N = \{$possible lengths of raw_sample after shift$\}$
3: **for** $n \in N$ **do**
4: $f =$ index nearest to Doppler shifted frequency
5: $sample =$ first n points of raw_sample
6: $a_i = f^{\text{th}}$ coefficient of DFT($sample$)
7: **return** n such that $x_n = \max_{i \in N} |a_i|$

IV MULTIPLE ACCESS

Given N nodes there are $N(N-1)$ directed edges between these nodes. Therefore, given N users, there are $N(N-1)$ possible pathways required for all users to be able to communicate simultaneously. Figure 15 shows a visualization of this fact.

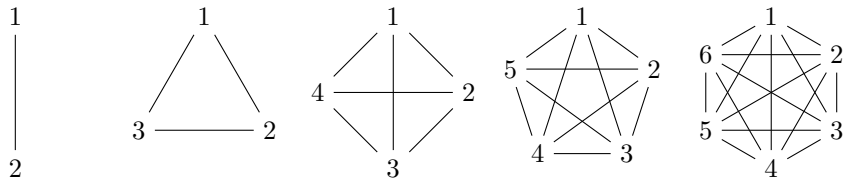


Figure 15: Complete (undirected) graphs for $n = 2, 3, ..., 6$

There are two main ways to specify addressing. The simplest ways designate subdomains in the total domain beforehand. When a user wants to communicate to another user, they use the pathway that is designated for the sender-receiver pair. Though, when two users are not communicating, the pathway between them is not being used.

Alternatively, rather than encoding the addressing information in the specific subdomain, one can encode the intended destination inside of the information of the message. For example, for N users, there could be N bandwidths in the frequency domain, such that each sender broadcasts on their corresponding bandwidth, and encodes to whom they are messaging in the first $\log_2 N$ bits.

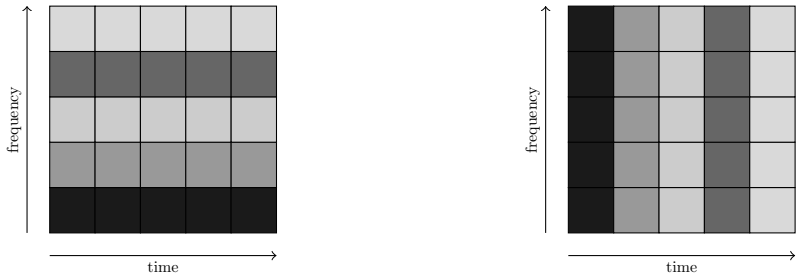
It is possible to only create a pathway between users when needed, allowing those users who are communicating to use more pathways (more bandwidth). A protocol which determines how channels are allocated is referred to as a Medium Access Control (MAC) protocol. Given the comparatively long propagation times of acoustic waves in water compared to the speed of light, MAC protocols used in above ground networks are unlikely to be efficient.

Basic Multi-Access Methods

FDMA/TDMA

The most simple methods of Channel Access Control are Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA). These protocols assign either a frequency band, or set of times to each of the $N(N-1)$ pathways between N users. When a user wants to communicate with another user, they transmit in the appropriate subdomain in the time-frequency domain. Figure 16 shows representations of FDMA and TDMA with 5 pathways. Each color represents a distinct pathway. One or more modulated symbols are placed within each square shown in this figure.

Unless timing can be precisely determined TDMA is not practical. Since the distances between users may vary in our networks, there is no realistic way to simply implement TDMA. FDMA could be used, however there are some considerations, such as narrow band interference. If there is environmental noise at a specific frequency it is possible that the entire frequency would be unintelligible.



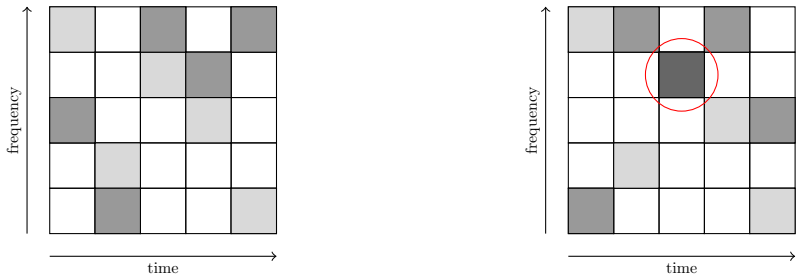
(a) FDMA (b) TDMA

Figure 16: FDMA and TDMA with 5 distinct pathways

Frequency Hopping

Rather than transmitting at a fixed frequency, each user switches frequencies in a known pattern. Figure 17 (a) shows two such patterns. In the presence of narrow band interference, only a portion of each users transmission is lost, rather than all transmissions of some users.

The main disadvantage of FHSS is the possibility of two patterns overlapping. Figure 17 (b) shows the same patterns as in (a), however the darker pattern has been shifted left by one time step. In this case the two patterns overlap, and the information in the intersecting block is likely lost. When two patterns intersect it is called a hit. While it is impossible to eliminate all hits when hopping across all frequencies, patterns can be chosen to minimize the total number of hits.



(a) Two Patterns (b) Time Shifted Patterns

Figure 17: Two hopping patterns with different time shifts

Medium Access Control Protocols

Current Research

Recently, there has been active interest in research about MAC protocols for underwater acoustic sensor networks (UAWSN). Research has focused on sensor networks over longer distances than we are concerned with. These networks often require data packets to hop through multiple intermediate channels before reaching their destination [5, 6, 7, 8].

Much of the current research into UAWSN, has looked at the throughput as a function of offered load, with little consideration to the average propagation time of a signal. Some such protocols stop communication between two nodes entirely until other node have finished communication [5].

Suggestions

If the primary goal of communication between humans, or other devices requires timely and frequent communications, current MAC protocols for underwater environments are suboptimal. The networks we are interested in are of much shorter range, often under 100m. The propagation delay we considered is at least one order of magnitude lower than considered in most current research. While the delay is still much greater than EM waves, it is likely that investigating MAC protocol designed specifically for short range and low delay times could produce a significantly more efficient communication network.

Due to time limitations we were only able to begin to read the literature on the topic. There are many software packages which are used to simulate networks, notably ns3 which contains a module for the underwater acoustic channel. If knowing how a MAC protocol compares to known protocols is not important, then it would likely be easier to write your own simulation code.

V INFORMATION ACCURACY

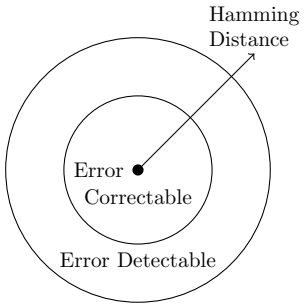
Linear Codes

There will be errors that occur in the transmission of bits across a physical domain. Linear codes allow us to add redundancy into the message be sent, such that if some are lost, the message's informational content is preserved. Generally, the goal of linear codes is to take the information of a single bit, and to spread the information across the other bits that are being sent. If the informational capacity is not changed, then re-arranging the distribution of where information is stored will not increase the robustness of the message - so a linear code must also increase the dimension of the message (by increasing the number of bits). The implementation of different kinds of linear codes is exactly the balancing of effectively adding redundancy without excessively lengthening the message. Linear codes are rooted in deep mathematical concepts and understanding the mathematics can be very rewarding; however, for our engineering tasks a physical understand can provide all the context necessary their implementation. We investigated two linear codes: Hamming Codes and Reed-Solomon Codes.

Hamming Codes

Hamming codes add redundancy to the word in a uniform fashion, such that removing any some number of bits is error correctable, and removing more is error-detectable. For example, the word "Bicydle is clearly a misspelling of the word "Bicycle, and the word "Barcdly is clearly a misspelling of some word, though the actual word is impossible to tell.

One intuitive way to think about Hamming codes is that they take a physical arrangement of messages and they space them apart further, so that small enough errors fall within a designated region of a message.



Hamming codes provide good protection against a uniform probability distribution of error, as each word is protected equally.

Reed Solomon Codes

Physically, Reed-Solomon codes take a small number of words and apply a process to encode the words into a larger number of words, such that losing any individual word is recoverable. The idea of Reed-Solomon codes is similar (though not exactly) to how people can understand a mistake by context: the sentence "I live you, is most likely "I love you, which we understand because there is information stored in the group of words, not just the words themselves.

Reed-Solomon codes are robust against burst errors, where a whole word may get entirely corrupted; conversely, any error in a word corrupts the entire word, and so Reed-Solomon codes are not robust against uniform error distributions.

Effective Dictionary Creation

It is always advantageous to minimize error rates, thus there is an incentive to encode more common messages using less error-prone signals. Taking the assumption that there is a non-uniform distribution of the importance of messages that will be sent (such that some messages or users are more important than others), and that some modulations are less error-prone than others. In E, we discuss the creation of a minimally error-prone list of modulations.

This algorithm should decrease significant errors in the communication network:

1. Characterize the total number of modulations that can be sent across all bandwidths, such that D is the total number of modulations.
2. Assign a value to each modulations to inform upon its proneness to error.
3. Construct a dictionary of D words, such that it is the collection of all words length $Log_2 D$ constructed from binary symbols.
4. Rank the D words by their importance in an average set of messages.
5. Linearly assign the least error-prone modulations to the the most frequent messages.

VI REAL IMPLEMENTATIONS

QAM

QAM is implemented in Mathematica. Our implementation allows for an arbitrary constellation to be used, although in general we used a standard QAM-16 rectangular constellation. The symbol time, inter-symbol delay, frequency, start and stop bits, are all configurable. In addition, an arbitrary pulse shaping function is allowed.

The implementation was used primarily for the characterization of interference. It was tested in the acoustic channel by recording a transmission played through headphones directly into a speaker. Such transmissions were generally successful, but in-depth testing was not conducted.

OFDM

OFDM is implemented in Matlab. The main feature of our implementation is multi-access, which allows up to 10 users communicating asynchronously simultaneously. In later subsections, we briefly explain algorithms and techniques involved in the program.

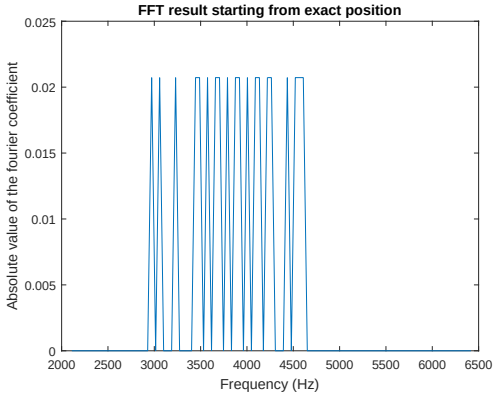
Striving Towards the Bandwidth Limit

Since we are doing experiments in an office environment, with major noises coming from people in the vicinity, the bandwidth is chosen from 3k to 10k Hz. According to the preceding theorem about maximum bit rate, theoretically we can achieve 7k bps. However, we need to make sure our FFT window contains data points corresponding to the same symbol.

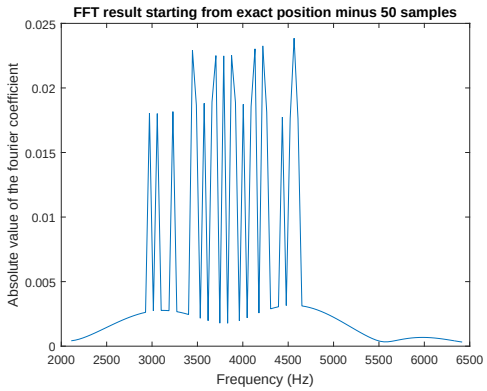
As shown in Figure 18, if the starting position of FFT is incorrectly determined, there will be additional error. In Figure 18c, the coefficient of 4435.8Hz is only about 51% of that of 4565Hz frequency, but they represent the same data value '1'. In Figure **??**, where the correct starting position is used, these coefficients have the same magnitude. For an FFT window of 1024 samples, 10% of inaccuracy in starting position already causes significant errors. Although we developed some algorithms to detect starting position, there is still a modest degree of inaccuracy due to noises. So, we set our default symbol length and FFT window length as 1102 and 1024 respectively. The approximately 7% buffer zone in the default symbol length reduces this inaccuracy by allowing a range of starting points to all produce the correct results.

Another consideration is to add a starting bit channel, in which we put an agreed starting bit pattern. There are several advantages of setting up a starting bit channel. First, we can assign least error-prone frequency to the channel to improve accuracy of locating starting position of the first symbol. Second, we can spare some buffer frequency for this channel with less negative impact on the overall bit rate than using a buffer on every frequency band. Most importantly, with this buffer zone, Doppler shifting can be detected by searching for the best fit shifted frequency. In the implementation, we assign the highest frequency for the starting bit. We first calculate the maximum possible Doppler shift we need to account for, which is 0.5%. So, we set up four empty buffer frequency channels to isolate the starting bit channel so that non-orthogonality of shifted frequencies does not have significant impact on starting bit. When decoding, we locate the maximum convolution value between possible frequencies and the sound file. With almost no interference from other frequency, we can get the approximated Doppler shifting ratio and adjust length of FFT window accordingly[6]. In our experiment, we use 160 different frequencies to

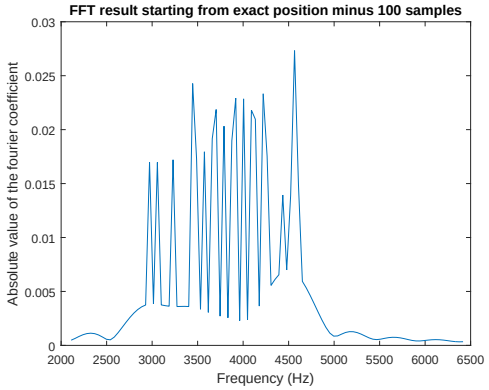
[6]This has been explained more in depth in the section "Accounting for Doppler Shifting in Modulation and Demodulation"



(a) Perfect FFT starting position



(b) Starting at 50 samples before



(c) Starting at 100 samples before

Figure 18: How does inaccuracy of FFT starting position cause error

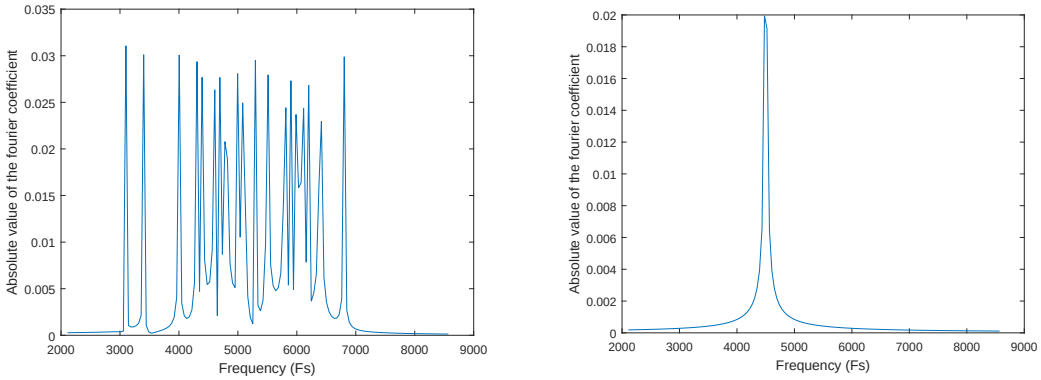
transmit our data and 5 more frequencies for the starting bit. The efficiency is 97%.

From the previous section on the effects of Doppler shifting on orthogonality, we notice the importance of having orthogonal base vectors for easily decoding a signal. Compare figure18a and figure19a, the performance of frequencies selected exactly on FFT vectors is more accurate if the basis vectors are assumed to be orthogonal. Note that the n^{th} entry after performing an FFT corresponds to the frequency $n \cdot F_s/N$, where F_s is the sampling frequency and N is the length of each FFT window. Therefore, frequencies f_i we used should satisfy $f_i \cdot N/F_s \in \mathbb{N}$. In our final design, we use symbol rate of 40Hz and typical sampling rate of 44100Hz. Therefore our frequencies range from 2972Hz to 10034Hz with a 43.07Hz difference. We are sending at 160 different frequencies at symbol rate of 40Hz so the bit rate is 6400 bps in 7105Hz bandwidth, in other words, 90.08% of bandwidth.

In the following subsections, we will discuss algorithms used in our implementation.

Modulation

The modulation process starts with a bit sequence to be transmitted. First we convert the bit sequence to a information matrix. The number of rows of the matrix is given by the number of frequencies used for modulation, and the number of columns is given by the number of symbols sent.



(a) FFT of non-orthogonal basis

(b) FFT on sinusoidal wave of 4479Hz

Figure 19: FFT performed on non-orthogonal frequency bands

An element-wise multiplication is performed on each row, with a continuous sinusoidal function of the frequency represented by that row. Finally all the sinusoidal functions are summed, resulting in the acoustic wave to be sent. The matrix contains the starting bit information at the row with the highest frequency. For our implementation, the starting bit is two non-zero elements at that row located ahead of the information.

If the multiple access is needed, the frequencies used by each user will be divided by the number of users. Additionally, an additional five columns will be added to indicate which user is sending the information. Figure 20 is an example of an information matrix for 4-user multi-access, with the current user occupying the second frequency band.

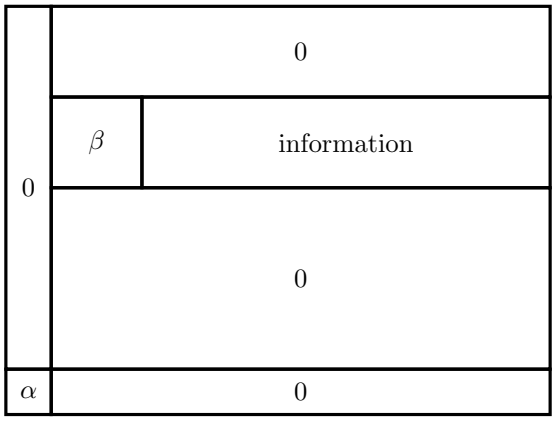


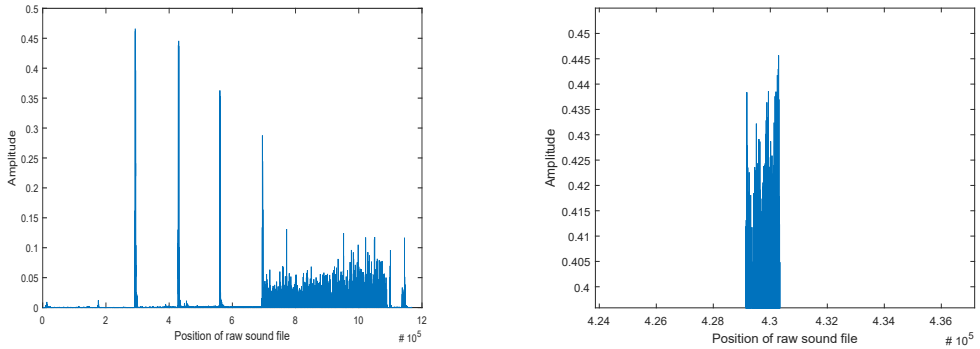
Figure 20: Information matrix for multi-access, α is the starting bit of the information matrix, β is a matrix to indicate the frequency band used, 0 means zero matrix

Demodulation: Synchronization

In any communication protocol, synchronization between the transmitter and receiver is necessary in order to decode the signal. Especially in our case, the performance of FFT depends on whether the FFT window is contained within a block of data. If a single FFT window covers two different blocks of data, it is difficult, if not impossible to correctly convert the information to frequency

domain.

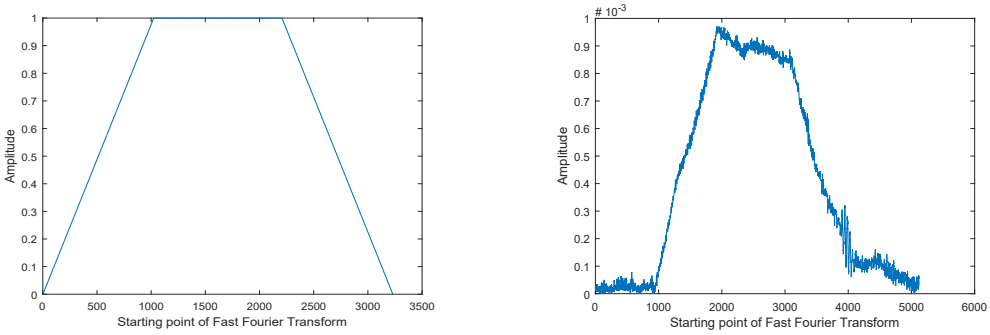
To determine the starting point of each transmission, two starting bits are transmitted on a frequency band designated as the start bit frequency band.



(a) Full transmission (b) Magnification of second peak

Figure 21: FFT coefficient of start bit frequency vs. time

In order to find the correct starting point, there are two major steps. First, an approximation of each starting bit needs to be found. We convolve the received acoustic signal with a pure sine function at the starting bit frequency, which gives us the approximation of the starting bits as shown in Figure 21a. While it initially appears that such spikes are sufficient for finding the starting points, Figure 21b shows that this is not the case. By zooming into a peak, we see that the peaks actually occur over some amount of time, and therefore we cannot simply take the time which maximizes the magnitude.



(a) Theoretical Pattern (b) Experimental Pattern

Figure 22: Patterns used to find true starting position

After acquired the approximation of starting bits, the exact starting point of each starting bit needs to be obtained. The maximum of the starting point peak has been chosen to initial the searching for the exact starting point. FFT has been performed on each sample starting from three times of the length of the FFT window ahead of the maximum and ending at two times of the length of the FFT window behind the maximum. After that, we extract the absolute value of the Fourier coefficient of the starting bit frequency, which gives us the plot Figure 22b. An ideal FFT pattern of the starting bit frequency has been given in Figure 22a, the pattern is an isosceles trapezoid, which has a top of two symbol size minus a length of a FFT and a bottom of two symbol size plus

a length of FFT. To obtain a complete FFT pattern, theoretically, the FFT needs to start at the position ahead of the starting point for one FFT window.

The ideal FFT pattern is used to convolve with the extracted value to match those two pattern and acquire the exact starting position.

Demodulation: FFT

The received acoustic wave needs to be converted back to the information matrix we constructed during modulation, so the information can be successfully decoded. After getting the exact starting position p as in the previous steps, we perform FFT on $[p + (i-1)N + 1, p + iN]$ of received signal for $i = 1, 2, ...$ Then we match the corresponding frequency with the FFT bins. In our implementation, we send 200 symbols in each package. Therefore, the reading will end at $i = 200$.

The data points read from FFT bins are still coefficients of the frequencies at that period of time. This means attenuation, noise, and other inaccuracies caused when transmitting messages result in a matrix with entries in $[0, 1]$ rather than $\{0, 1\}$. Therefore, more work must be done to classify entries as zeroes and ones. Considering the sensitivity of different frequencies is different, we divide each row of data representing the same frequency channel by the maximum of that row. Then, we get similar histogram plot of each channel as Figure 23. Instead of blindly dividing the data at 0.5, we apply the Expectation-Maximization algorithm to cluster the data. Since in most histogram results, data points representing zeros and ones distribute like a normal distribution. The drawback is the computational time. In real implementation underwater, computational power may be limited, so simpler clustering methods such as 50% constant cutting should also be considered.

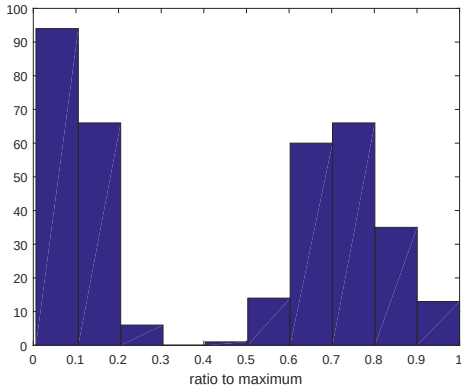
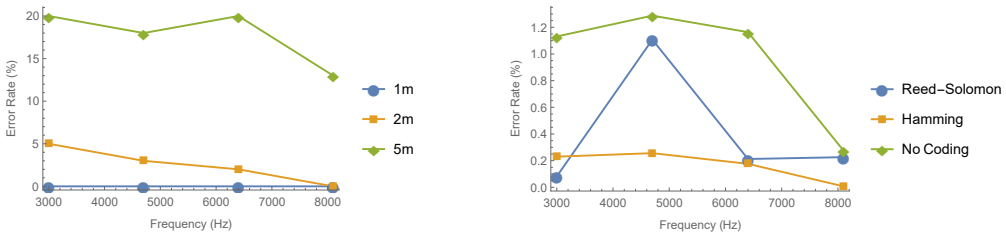


Figure 23: Histogram of FFT coefficient read from a real recording

Experimental Data

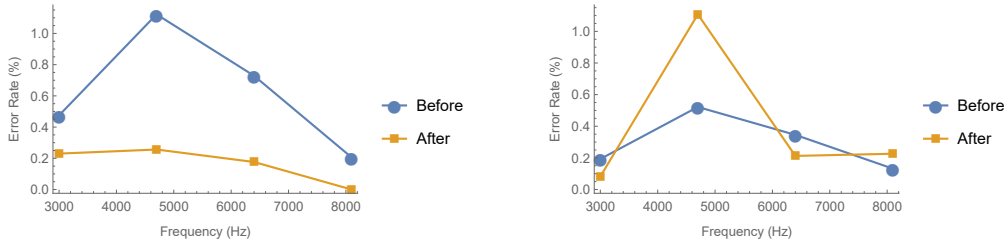
Transmission error rates were measured for our implementation using a standard computer speaker as a transmitter, and an iPhone 6 as a receiver. Figure 25 show the relationship between various frequency bands of bandwidth 1.7kHz starting at the given frequency, and then error rate. Figure 25a shows this relationship at varying distances between the transmitter and receiver. Figure 25b shows this relationship using varying error correcting codes.

Comparison of Reed-Solomon code and Hamming code based on two major criteria: the extra coding space needed and the percentage of error can be corrected. Reed-Solomon code needs to be



(a) Using Hamming Codes (b) At 1 meter

Figure 24: Transmission Error Rate vs. Frequency (1.7 kHz bandwidth, at 1 meter)



(a) Using Hamming Codes (b) Using Reed-Solomon code

Figure 25: Transmission Error Rate vs. Frequency (1.7 kHz bandwidth)

2.5 times coding space comparing with the original code yet hamming code only need 7/4 times coding space. For the correction of errors, Figure 25 show us that hamming code is better regarding the performance of error correcting.

Conclusions from Data

Given our experiments three conclusions can be drawn. First, muti-access protocol is developed successfully, though the starting bits are intolerant of noise. Second, Hamming code outperforms Reed-Solomon code. However, both linear codes fail to correct most errors when the uncorrected error rate is higher than 10%. Finally, the error rate is generally lower in higher frequency channels. Developing dictionary entropy encoding is a good choice.

REFERENCES

[1] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Communications Magazine*, vol. XX, 2009.

[2] "Optical absorption of water compendium." Web.

[3] "Underlying physics and mechanisms for the absorption of sound in seawater." Web. `http://resource.npl.co.uk/acoustics/techguides/seaabsorption/physics.html`.

[4] "Npl acoustics: Calculation of absorption of sound by the atmosphere." Web. `http://resource.npl.co.uk/acoustics/techguides/absorption/`.

[5] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *IEEE Communications Letters*, vol. 11, 2007.

[6] H. Ramezani, F. Fazel, M. Stojanovic, and G. Leus, "Collision tolerant and colission free packet scheduling for underwater acoustic localization," *IEEE Transactions on Wireless Communication*, vol. XX, 2015.

[7] M. Molins and M. Stojanovic, "Slotted fama: a mac protocol for underwater acoustic networks," *IEEE Oceans*, 2006.

[8] W. Lin and K. Chen, "Mhm: A multiple handshaking mac protocol for underwater acoustic sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2016, 2016.

VII APPENDICES

A Frequency Dependent α Values

Frequency (Hz)	α in Air (dB/m)	α in Water (dB/m)
1000	0.005005	0.000052
10000	0.241868	0.000733
20000	0.597649	0.002428
30000	0.859229	0.005131
40000	1.068015	0.008686
50000	1.266241	0.012904
60000	1.474783	0.017593
70000	1.703539	0.022568
80000	1.957449	0.027673
90000	2.239145	0.032779
100000	2.550116	0.037791

Table 1: To show a frequency dependence in the α values, we arbitrarily chose parameter values to be near average values
Air: Temperature = 20° C, Pressure = 101.3 kPa, Relative Humidity = 30%
Water: Temperature= 20° C, Salinity = 35 ppt, Depth = 5 m, pH = 8

B Possible Minimization of Problems arising from Multipathing

The negative effects of multipathing arise from a single symbol signal interfering with a multipathed signal, for clarity, let us call them the 'main signal and the 'interfering signal respectively.

When two signals interfere, it is possible that they can contribute to a nearly entirely destroyed signal, if there a shift of half a wavelength; we do not know how to account for the multipathing effects that could contribute to nearly destructive interference.

Clearly, if the interfering signal only overlaps with the main signal slightly, the main signal will not get corrupted, and clearly if the signals overlap entirely, the main signal will get corrupted. There must be some δt such that the interfering signal can overlap with the main signal without truly corrupting the signal, however, we do not propose any good method to compute δt.

The algorithm we implemented strived to achieve the informational channel maximum by sending an individual symbol at a rate of 1/40 s, and then increasing the bits per symbol. Our messages were sent over 5 second intervals, so we will now consider effects on these time scales. To do any sort of estimates on the effects of multipathing, we assume that the varying effects governing multipathing do not change on the order of magnitude of the time to transmit a symbol (tenths of a second), but only on the order of seconds, minutes, or maybe longer. Let us take δt to be half the symbol time; this means we only need to consider multipathing for time larger than 1/80 s. With this assumption, we consider the attenuation effects of the difference in time for times greater than 1/80 s and 5 s, which is a difference in path length of at least $\sim$19 m.

Because the rate of attenuation is frequency and environmentally dependent, it is difficult to produce a formula to characterize the difference between the intended signal and the multipathed signals; however, to build an intuition, we simulated a plot of difference in volume between a signal and the same 19 meter later.

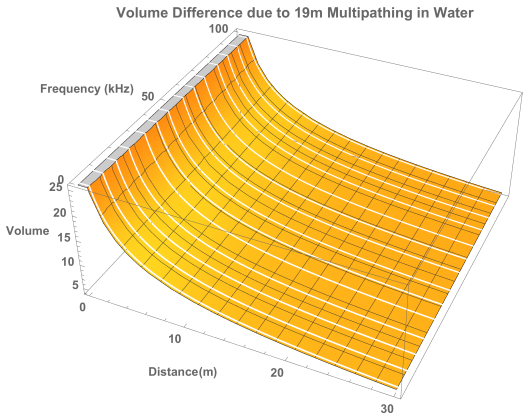


Figure 26: Plot of the difference in volume of signal $V_0 = 100$ dB with itself 19 meters further. Although the rate of attenuation increases with frequency, the difference in volume at a distance only varies by a maximum of 1.25 dB across the 1 MHz bandwidth. A table of these values is in Appendix C

To transmit a message to a destination 31 meters away, the minimum difference between the intended message and the multipathed message is 8.1 dB. This means that to conduct any sort of amplitude modulation without fear of any significant multipathing effects, all amplitude modulation must be done in an 8dB band, otherwise it will be more complicated to understand the difference between an intended signal and a multipathed signal. To use a larger range of amplitudes for modulation, it is necessary to shorten the distance across which communication is intended to be perfect. Though, increasing the amplitude range should not cause huge problems; it is difficult to predict how much multipathing will truly interfere without experimenting in an underwater environment.

For this specific context, the length in time of a single message dictates the negative effects of multipathing, and provides a band of amplitudes within which amplitude modulation is perhaps less not corruptible by multipathing (this shows an advantage of sending more information per message, less often).

C Multipath Attenuation Values

	1kHz	11kHz	21kHz	31kHz	41kHz	51kHz	61kHz	71kHz	81kHz	91kHz
1 m	31.7	31.7	31.8	31.9	32.	32.2	32.4	32.6	32.8	33.
6 m	17.2	17.2	17.3	17.4	17.5	17.7	17.9	18.1	18.3	18.5
11 m	12.9	12.9	13.	13.1	13.2	13.4	13.6	13.8	13.9	14.1
16 m	10.5	10.5	10.6	10.7	10.8	11.	11.2	11.3	11.5	11.7
21 m	8.9	8.9	9.	9.1	9.2	9.4	9.6	9.8	10.	10.1
26 m	7.8	7.8	7.9	8.	8.1	8.3	8.4	8.6	8.8	9.
31 m	6.9	6.9	7.	7.1	7.2	7.4	7.6	7.8	7.9	8.1

Table 2: Table of the difference in volume of signal $V_0 = 100dB$ with itself 19 meters further. All values are in dB.

D Nearest Standard Basis

Definition. *Let $K, r \in \mathbb{N}$. Define $\mathbf{t}_K = [0, 1/r, 2/r, ..., (K-1)/r] \in \mathbb{C}^K$ and define $\mathbf{t}_{K,i}$ to be the i^{th} component of $\mathbf{t}_K$. Denote the vector obtained by evaluating f at each of the K points in $\mathbf{t}_K$ by $f(\mathbf{t}_K)$.*

Definition. *Let S be a set of vectors in $\mathbb{C}^N$ and let $k_s > 0$. For some metric $F : \mathbb{C}^N \times \mathbb{C}^N \to \mathbb{C}$ and $f \in S$ define $\gamma(S, k_s, F, f) = F(f, f) / \max_{s \neq f \in S} F(f, s)$.*[7]

The measure γ will allow us to explore how a specific function f compares to other functions in S. Assuming we take F to be the absolute value of the inner product of two vectors in S, then the larger the values of γ, the "more orthogonal" S.

For our analysis of Doppler shifting we assume k_s is known and then take S to be some subset of $\{f_n^K(\mathbf{t}_{K+1})\}_{n=0}^K$, where K is the natural number closest to $k_s(N-1)$, with the Doppler shifted function $f_m^T(k_s\mathbf{t}_{K+1})$ appended. By choosing K as close to $k_s(N-1)$ as possible, we are selecting a vector space $\mathbb{C}^K$ with a basis containing an element with frequency as close to nk_s as possible. In particular, this element is $f_M^T(\mathbf{t}_{K+1})$ where M is the natural number closest to nk_s. We use the standard inner product as F.

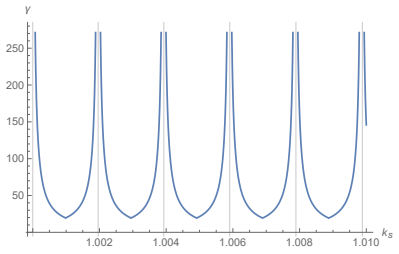
Note that if $K = k_s(N-1)$ then $f_m^T(k_s\mathbf{t}_K)$ is already contained in the set $\{f_n^T(\mathbf{t}_{K+1})\}_{n=0}^K$ for all n and T. Moreover, $\gamma(S, k_s, F, f_m^T(k_s\mathbf{t}_K)) = \infty$ as expected. What we are interested in is the behavior of γ when k_s is between these points at various values of n and T. This will give us insight into how selecting the sampling rate and frequencies used will impact the effect Doppler shifting has on obtaining a DFT decomposition.

Figure 27 shows the response of γ to changes in k_s at two values of n. The vertical grid lines are placed at $k_s = \frac{N-1}{K}$ for integer K. As expected, $\gamma = \infty$ at these points. Moreover, γ is bounded below by some positive constant.

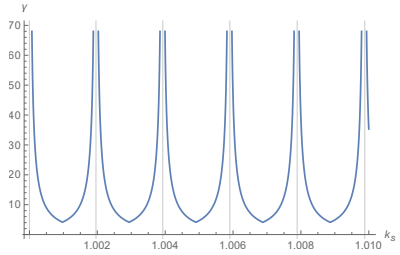
Figure 28 shows the response of γ to changes in n at two values of k_2. In both cases we a monotonically decreasing function for $n < N/2$. As expected, $\gamma = \infty$ when there is no Doppler shift ($k_s = 1$). Thus if we bound n, there is a positive minimum value of γ.

Figure 29 shows the response of γ to changes in the number of points used to calculate the DFT.

[7] As you might have noticed, γ is not well defined if f if $F(f, s) = 0$ for all $s \in S$, but as expected we put $\gamma = \infty$ if this is the case.

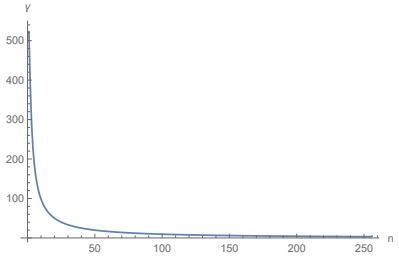


(a) $n = 50$

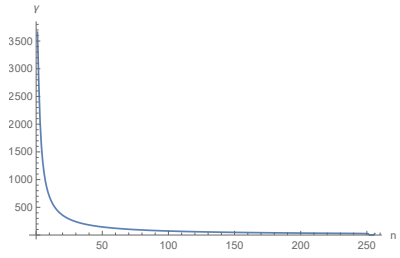


(b) $n = 200$

Figure 27: $T = N = 512,\ k_s \in [1, 1.01]$



(a) $k_s = 1.001$



(b) $k_s = 1.01$

Figure 28: $T = N = 512,\ n \in \{1, 2, ..., 256\}$

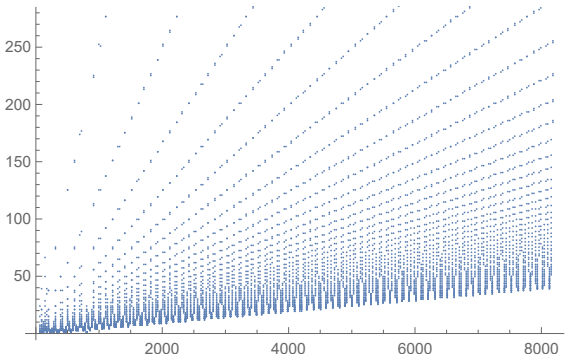


Figure 29: $(T = N) \in \{128, ..., 8192\}, n = 400, k_s = 1.01$

It appears that for any value of γ there is a number K such that a DFT on a sample containing at least K samples will result in at least this value of γ.

Together these three graphs suggest that given some finite set of frequencies, we can make γ arbitrarily large. That is, given either a desired frequency band, or a desired number of DFT samples, it is possible to obtain any chosen minimum value of γ.

E A Consideration on the Effective Usage of Constellations

In any type of modulation that can be decomposed into an IQ plot, as with QAM, it is possible to characterize the cost of sending a message in some domain of the IQ plot, versus the discernability

of different subdomains. In theory it is possible to find the *most effective* usage of the IQ domain to design a QAM constellation.

Taking some maximum acceptable amount of error to be given, there are two questions that can be asked to address the problem: given a maximum power output (distance from the origin on the IQ plot), what is the maximum number of bits that can be encoded into the domain; and given N-bits, what is the smallest convex polygon that can encompass all N constellation points (such that 'smallest' is characterized roughly by the standard deviation from the origin). This is, at its root, a packing problem.

Considering the latter problem may be slightly easier to visualize: what is the densest arrangement of N-points on the IQ plane, such that the average 'cost' of each point in relation to others is decided upon and fixed. In other words, this is a soft body packing problem. An analogous problem to this is the soft body packing problem, of how will a set of electrons arrange themselves with a force pushing them together. This is different from the hard body problem where the cost function of placing a particle by another particle follows a step function: it is an infinite cost to put a marble inside another marble, but they sit next to one another at no cost.

The cost can be calculated of an arrangement can be calculated, though some approximations must be made: the closer nodes are (the less discernible the message is), the higher the error, $error \propto \frac{1}{\langle \Delta d \rangle}$; additionally, the farther from the center, the higher induced error on other channels (discussed in , $error \propto r$.

We did not solve for various possible arrangements under these conditions, however, we present this for consideration as it is an important thing to consider in the creation of the list of modulations to be used. Solving for an ideal assignment of modulations is analogous to solving for a realization of a Wigner crystal, which is a solid, crystalline phase of electrons. For this reason, we suggest that it may be solved for using common mathematical physics techniques, through calculus of variations.

In practice, this is taking a function that will map an arbitrary arrangement to a real-valued cost, then varying the arrangement by small amounts to find local minimums in the cost of possible arrangements. Much research has gone into this, and we suggest it may be most rewarding to capitalize on the work that others have done by adapting a physics engine to solve this problem.

F Github

Our project is located at `https://github.com/RoyRin/Epropulsion_Code`