Ultrasonic Ranging and Indoor Localization for Mobile Devices

Submitted in partial fulfillment of the requirements for
the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Patrick J. E. Lazik

B.S., Electrical Engineering, Boston University
M.S., Electrical and Computer Engineering, Carnegie Mellon University

Carnegie Mellon University
Pittsburgh, PA

August, 2017

To my parents

Abstract

Location tracking on mobile devices like smartphones has already begun to revolutionize personal navigation. Unfortunately, these services perform poorly indoors when GPS signals are no longer available. Highly accurate indoor location tracking would enhance a wide variety of applications including: building navigation (malls, factories, airports), augmented reality, location-aware pervasive computing, targeted advertising, social networking, participatory sensing and could even support next generation beam forming MIMO wireless networks. Current indoor localization systems for smartphones often use RF signal strength from WiFi access points or Bluetooth Low Energy (BLE) beacons to fingerprint indoor locations. Such systems are sensitive to environmental changes and obstructions, require extensive training procedures and are limited in both absolute as well as semantic localization accuracy.

We propose using audio signals in the ultrasound spectrum, just above the human hearing range, to provide ranging and localization for many off-the-shelf mobile devices that are equipped with microphones. Ultrasonic ranging provides several advantages over RF-based ranging and fingerprinting approaches, which make it attractive for indoor localization. A relatively low propagation speed and carrier frequency allow for precise propagation time measurements in software using commodity hardware. Acoustic signals also have a low penetration depth, which confines them to target areas for accurate semantic localization. In this dissertation we address several challenges related to acoustic localization, including system scalability, ranging and localization accuracy, energy efficiency, robustness to noise, elimination of human perceivable audio artifacts, efficient use of limited acoustic bandwidth and rapid deployment strategies.

Acknowledgments

I would like to extend my sincerest gratitude and thanks to my advisor and thesis committee chair, Anthony Rowe, for the support and guidance he has given me over the years of pursuing a masters and PhD degree at Carnegie Mellon University. I was incredibly fortunate to be offered a research position with Anthony during my first week of classes as a MS student, which evolved into him taking me on as his PhD student a year later. I could not have hoped for a better, more devoted, mentor and role model, who lead me to find my passion and showed me how to navigate the world of academia and research in electrical and computer engineering. Thank you Anthony for leading me to where I stand now.

I would also like to thank my doctoral committee members Dan Siewiorek, who had expressed interest in my research at an early stage and served on my qualifier exam committee, Pei Zhang who first helped me to explore the space of indoor localization in one of his classes and Charles Shelton, who has been one of my strongest ties to the Bosch Research and Technology Center in Pittsburgh over the years and has been elemental to promoting my research there.

I am very grateful to my supervisors and co-workers at the Bosch Research and Technology Centers in Pittsburgh and Chicago: Chris Martin, Jeff Donne, Elizabeth Latronico, Joseph Szurley, Billy Li and Nick Wilkerson, all of whom have helped me push the envelope on indoor localization and building technology related systems. Thank you for believing in me and my research to this day. Furthermore I want to acknowledge the Robert Bosch LLC for their financial support and for providing a welcoming and engaging work environment.

I was very fortunate to have been part of the Terraswarm Research Center over the past five years, which has allowed me to collaborate with leading universities on cutting edge research and demonstrations. Thank you to all of the members and its director, Edward Lee, for the long term financial support, the frictionless teamwork and all of the inspiring projects.

Only a fraction of the work in this dissertation would have been possible without my colleagues at CMU's WiSE Lab. I would especially like to acknowledge Niranjini Rajagopal for devoting truly countless hours, at day and at night, to our vision of indoor localization. It has been a privilege to work with you and I cannot thank you enough for all of the insight, ideas, sweat and tears you have poured into our research over the years. I would also like to recognize Oliver Shih and Sindhura Chayapathy for all of the expertise and hard work they have contributed. Max Buevich, Craig Hesling, Artur Balanuta, Adwait Dongare, Chris Palmer, Luis Pinto, Khushboo Bhatia and Toni Fox, thank you all for making our lab the fantastic place it is.

Pursuing a PhD was a long and arduous journey, and it would not have been possible without having my best friend, Dana Shukovsky, by my side. Thank you Dana, for being there with me for so many years, believing in me and giving me your unwavering support and companionship.

Finally, I would like to express my deepest thanks to my parents, Erhard and Edith Lazik, for the boundless support, encouragement, inspiration and unconditional love they have been giving me every day of my life. Thank you for prioritizing my education, for nurturing my interests and for enabling me to reach for the stars. This dissertation is dedicated to you.

The work in this dissertation was supported in part by the Bosch Research and Technology Center in Pittsburgh, TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA, and the National Science Foundation under award# 1534114.

Contents

1 Introduction **1**
1.1 Background 1
1.2 Motivation 2
1.3 Problem Statement 5
1.4 The Acoustic Location Processing System (ALPS) 6
1.5 Thesis Statement 9
1.6 Contributions 9

2 Related Work **11**
2.1 Overview 11
2.2 Range-based Approaches 11
2.2.1 Time Of Flight (TOF) 12
2.2.2 Time Difference of Arrival (TDOA) 13
2.2.3 Round Trip Time Of Flight (RTOF) 15
2.3 Range-free Approaches 16
2.3.1 Angle of Arrival (AOA) 17
2.3.2 Received Signal Strength Indicator (RSSI) 18
2.4 Ranging Technologies 20
2.4.1 Radio Detection and Ranging (RADAR) 20
2.4.2 Sound Navigation and Ranging (SONAR) 21
2.4.3 Light Detection and Ranging (LIDAR) 21
2.4.4 Ultra Wide Band (UWB) Ranging 22
2.5 Localization Systems 24
2.5.1 Global Positioning System (GPS) 24
2.5.2 Cricket Location-Support System 25
2.5.3 Dolphin 26
2.5.4 Microsoft RADAR 27
2.5.5 Motetrack 28
2.5.6 Active Badge 28
2.5.7 Active Bat 29
2.5.8 BeepBeep 29
2.5.9 Enhanced 911 (E911) Mobile Phone Localization 31
2.6 Time Synchronization 31
2.6.1 The Network Time Protocol (NTP) 32

2.6.2 The Global Positioning System (GPS) 33
2.6.3 Cellular Timing Service 34
2.6.4 Flooding Time Synchronization Protocol (FTSP) 34
2.6.5 Glossy Time Synchronization Protocol 36

3 ALPS System Architecture and Platform 37
3.1 Overview 37
3.2 ALPS Platform Hardware Design 39
3.3 Beacon and Network Master Firmware 42
3.4 iOS Application 43
3.5 Future Hardware Improvements 44
3.6 Transmission Protocol 44
3.7 Speaker and Microphone Directionality 46
3.7.1 Experimental Setup 46
3.7.2 Omnidirectional Ultrasonic Speaker Horn 47
3.7.3 Quad Sectored Speakers 51
3.7.4 iPhone Microphone 53
3.8 Power Consumption 55

4 Ultrasound Signal Modulation and Demodulation 58
4.1 Overview 58
4.2 Pulse Compression 60
4.3 Reducing Audible Artifacts 62
4.3.1 Audio User Study 64
4.3.2 Human Health Concerns 67
4.3.3 Animal Exposure 67
4.4 Modulation Scheme With Data Transmission 68
4.4.1 Modulation 69
4.4.2 Multiple Access 71
4.4.3 Demodulation 72
4.4.4 Evaluation 74
4.5 Modulation Scheme without Data Transmission 80
4.5.1 Modulation 81
4.5.2 Multiple Access and Tiling 81
4.5.3 Demodulation 82
4.5.4 Non-Line of Sight Signal Multipath Detection 83
4.5.5 Evaluation 85
4.5.6 Sectored Speaker Array Modulation 87

5 Time Synchronization 89
5.1 Overview 89
5.2 Time Synchronization Background 90
5.3 iPhone Network Time Synchronization Benchmarks 91
5.3.1 Clock Granularity 91

5.3.2 ADC Timing Performance . . . 92
5.3.3 OS Timing Performance . . . 93
5.3.4 NTP Timing Performance . . . 94
5.4 Clock Recovery via TDOA Based Localization . . . 95
5.4.1 iOS App Design . . . 97
5.4.2 Evaluation . . . 99
5.4.3 Modulation Scheme . . . 99
5.4.4 Deployment and Geometry Error . . . 101
5.4.5 Synchronization and Ranging Performance . . . 103
5.4.6 Limitations . . . 105
5.5 BLE Time Synchronization . . . 106
5.6 802.15.4 Time Synchronization . . . 108
5.7 LoRaWAN Time Synchronization . . . 109

6 Location Calculation **111**
6.1 Trilateration . . . 111
6.2 Multilateration . . . 113
6.3 Geometric Dilution of Precision (GDOP) . . . 114
6.4 Solver Software Architecture . . . 116
6.5 Intertial Sensor Fusion . . . 117
6.5.1 Process Model . . . 117
6.5.2 Measurement Model . . . 119
6.6 Rapid User-Assisted Setup . . . 120
6.6.1 Procedure . . . 120
6.6.2 Algorithm . . . 121
6.6.3 Evaluation . . . 122
6.7 Localization Performance Evaluation . . . 124
6.7.1 Stationary Receiver . . . 125
6.7.2 Mobile Receiver and IMU Sensor Fusion . . . 127

7 Conclusions **130**
7.1 Future Work . . . 132
7.1.1 Tag Localization and Tracking . . . 132
7.1.2 Angle of Arrival (AOA) Measurement . . . 133
7.1.3 Automatic Beacon Setup . . . 134
7.1.4 Advanced Location Engine . . . 134
7.1.5 Augmented/Virtual Reality Integration . . . 135

A ALPS Platform Evolution **137**
A.1 Generation 1 . . . 137
A.1.1 Hardware . . . 137
A.1.2 Software . . . 138
A.1.3 Generation Summary . . . 139
A.2 Generation 2a . . . 140

A.2.1 Hardware 140
A.2.2 Software 141
A.2.3 Summary and Possible Improvements 143
A.3 Generation 2b 144
A.3.1 Hardware 144
A.3.2 Software 145
A.3.3 Summary and Possible Improvements 146
A.4 Generation 2c 146
A.4.1 Hardware 146
A.4.2 Software 147
A.4.3 Summary and Possible Improvements 149
A.5 Generation 3a 149
A.5.1 Hardware 149
A.5.2 Software 151
A.5.3 Summary and Possible Improvements 151
A.6 Generation 3b 153

Bibliography **157**

List of Figures

1.1 Mobile device and infrastructure complexity of indoor localization systems vs. absolute localization error . . . 4
1.2 ALPS architecture overview . . . 8

2.1 Time of Flight ranging system and transaction diagrams . . . 12
2.2 Time Difference of Arrival pseudo-ranging system and transaction diagrams . . . 14
2.3 Round Trip Time Of Flight ranging system and transaction diagrams . . . 16
2.4 Angle of Arrival measurement system and transaction diagrams . . . 17
2.5 Received Signal Strength Indicator measurement system and transaction diagrams 19
2.6 Time Domain PulsON440 and Decawave DWM1000 Ultras Wide Band ranging modules . . . 23

3.1 ALPS system architecture . . . 38
3.2 ALPS beacon architecture . . . 40
3.3 ALPS ceiling mounted beacon with solar energy harvesting hardware . . . 41
3.4 ALPS network master/plug forwarder architecture . . . 42
3.5 Generation 3b ALPS network master/plug forwarder hardware . . . 43
3.6 Generation 3b ALPS transmission protocol . . . 45
3.7 Beam pattern measurement setup with pan-tilt mount and measurement microphone in anechoic chamber . . . 48
3.8 Omnidirectional ultrasound horn CAD model . . . 49
3.9 Ultrasonic beam patterns of piezo bullet speaker, ribbon speaker and piezo bullet speaker with omnidirectional ultrasonic horn . . . 50
3.10 Omnidirectional ultrasound horn design evaluation . . . 51
3.11 ALPS beacon Generation 3b, quad speaker array horizontal beam pattern . . . 52
3.12 iPhone 3GS microphone beam pattern . . . 53
3.13 iPhone and iPad microphone frequency response . . . 54
3.14 Power consumption of a Generation 3b ALPS beacon during two TDMA cycles . 57

4.1 Pulse Compression illustration before and after matched filtering . . . 61
4.2 Jitter performance of tones and chirps of various lengths and bandwidths . . . 63
4.3 Chirp components for the modulation scheme with data transmission . . . 64
4.4 Audio perception user study results . . . 66
4.5 Example of four symbols encoded with Rate-Adaptive Chirp Spread Spectrum . . 69
4.6 Beacon ID packet structure using Rate-Adaptive Chirp Spread Spectrum . . . 70

4.7 Experimental receiver setup for evaluating ALPS modulation scheme with data transmission . . . 75
4.8 Impact of fade in/out on BER in ALPS modulation scheme with data transmission 76
4.9 Impact of chirp length on BER in ALPS modulation scheme with data transmission 77
4.10 Impact of concurrent symbols on BER in ALPS modulation scheme with data transmission . . . 78
4.11 Spectrogram of a single beacon transmitting ultrasound packets modulated according to the ALPS modulation scheme with data transmission . . . 79
4.12 Spectrogram of four beacons transmitting ultrasound packets modulated according to the ALPS modulation scheme with data transmission . . . 79
4.13 Distance vs SNR using a Generation 1 ALPS beacon transmitting at $48dB(A)@1m$ 80
4.14 TDMA coloring schemes for acoustic tiling . . . 82
4.15 Ultrasound ranging error of generation 2c ALPS beacons in free space and corridor environments . . . 86

5.1 Delay in consecutive iOS system time calls . . . 92
5.2 Audio buffer OS time-stamping jitter . . . 94
5.3 NTP second tick deviation from UTC second tick . . . 95
5.4 Estimation of start of ultrasound transmission cycle timing . . . 96
5.5 iOS app flowchart for clock recovery via TDOA based localization . . . 98
5.6 Ultrasound packet structure of modulation scheme used to evaluate clock recovery via TDOA based localization . . . 100
5.7 Performance of modulation scheme used to evaluate clock recovery via TDOA based localization . . . 100
5.8 Effects of beacon localization error and geometry on clock recovery via TDOA based localization . . . 101
5.9 Experimental setup for evaluating TOF based localization error when using clock recovery via TDOA based localization . . . 103
5.10 Acoustic time synchronization accuracy and localization error using clock recovery via TDOA localization . . . 104
5.11 BLE synchronization transaction diagram . . . 106
5.12 BLE advertisement packet reception latency . . . 107
5.13 Beacon ultrasound transmission jitter using 802.15.4 time synchronization . . . 109
5.14 LoRaWAN time synchronization jitter . . . 110

6.1 Trilateration example diagram . . . 112
6.2 Multilateration example diagram . . . 114
6.3 Geometric Dilution of Precision (GDOP) example diagram . . . 115
6.4 Panorama of automatically set up kitchen area using three beacons . . . 122
6.5 Kitchen area beacon mapping output using user-assisted rapid setup . . . 123
6.6 Lab area beacon mapping output using user-assisted rapid setup . . . 124
6.7 Evaluation setup for localization performance of generation 1 ALPS beacons in atrium environment . . . 124

6.8 Ultrasound impulse response measurements in atrium and small room environments . . . 125
6.9 Localization error of generation 1 ALPS beacons in atrium and small room environment . . . 126
6.10 Localization performance of generation 2c ALPS beacons in lab environment with IMU sensor fusion . . . 127
6.11 Localization performance of generation 2c ALPS beacons in kitchen environment with IMU sensor fusion . . . 128
6.12 Localization performance of generation 2c ALPS beacons in kitchen environment with IMU sensor fusion and one blocked beacon . . . 128

A.1 Generation 1 ALPS beacon architecture . . . 138
A.2 Generation 1 ALPS Beacon hardware . . . 139
A.3 Generation 2a ALPS beacon architecture . . . 140
A.4 Generation 2a ALPS beaon hardware . . . 141
A.5 Generation 2a, 2b, 2c, 3a ALPS network master architecture . . . 142
A.6 Generation 2a, 2b, 2c, 3a ALPS network master hardware . . . 143
A.7 Generation 2b ALPS beacon architecture . . . 144
A.8 Generation 2b ALPS beacon hardware . . . 145
A.9 Generation 2c ALPS beacon architecture . . . 147
A.10 Generation 2c ALPS beacon hardware . . . 148
A.11 Generation 3a ALPS beacon architecture . . . 150
A.12 Generation 3a ALPS beacon hardware . . . 152
A.13 Generation 3b ALPS beacon with tile ceiling clip mount . . . 153
A.14 Generation 3b ALPS beacon with magnetic mount . . . 154
A.15 Generation 3b ALPS beacon with magnetic mount top view . . . 155

List of Tables

3.1 Power output of different types and sizes of solar cells at a $5cm$ distance from a 1600 lumen CFL bulb 55

4.1 Identification accuracy of NLOS signals with machine learning approach using multiple features 85

4.2 Impact of amount of training samples on accuracy of NLOS signal detection using machine learning approach 87

5.1 Audio ADC sampling jitter on iPhone 4 and 5S 93

6.1 Beacon mapping error using user-assisted rapid setup 123

A.1 ALPS platform generations and features 156

Chapter 1

Introduction

1.1 Background

Since the dawn of humanity, navigation has been an invaluable tool that allows people to traverse the world from an origin to a destination in a controlled manner. From the earliest recorded use of celestial based navigation in Homer's Odyssey in the 8th century BC [45], to modern satellite navigation systems using the Global Positioning System (GPS), navigation has been an indispensable catalyst to civilization. Despite the thousands of years of progress separating ancient navigation methods to current ones, the core approach has still remained the same:

1. Locate yourself in relation to your destination.
2. Determine a path to your destination and follow it.

This dissertation addresses the first step of this process in indoor environments, where current solutions like GPS are not available.

1.2 Motivation

Localization systems have already revolutionized how smartphones interact with the surrounding world. Up to this point, location-aware applications have been limited to outdoor environments where devices are in reception range of satellite systems such as GPS. Unfortunately, satellite-based approaches do not function indoors since their weak signals do not easily penetrate through building walls, leaving the places people spend the majority of their time in as blind spots. Although many technical principles of GPS can be applied to indoor localization systems, the challenges of indoor and outdoor localization are vastly different. While GPS provides global outdoor coverage using 31 satellites [79], the currently most prevalent indoor systems using WiFi and Bluetooth Low Energy (BLE) beacons like [20] would require an order of magnitude or two more beacons/access points to provide comparable localization precision inside an average sized airport. The pervasiveness of obstacles inside buildings and the dynamic nature of these environments make localization difficult as they block or add significant noise to the phenomena that one can measure to determine one's location like WiFi signal strength, the earth's magnetic field, the Time of Flight (TOF) of acoustic or RF signals and so on.

It is also interesting to perform a rough coverage cost comparison between GPS and an indoor system. The US Department of Defense estimates that GPS has cost an estimated total of $14,089$ million USD since its inception until 2016 (including development and deployment) in 1995 USD ($22,941$ million USD in 2017 USD) [82]. While these figures seem daunting, GPS provides approximately $510,072,000km^2$ of coverage, or $0.0222km^2$/USD. If we imagine covering the same area with WiFi access points, assuming a $100m$ coverage radius ($0.0314km^2$) and a cost of 50 USD per unit, this would equate to a global coverage cost of $811,805$ million USD, or around 35x the cost of GPS, not accounting for installation or maintenance costs. It should also be noted that a hypothetical WiFi based localization system such as this would provide poor performance since the coverage areas of the access points do not overlap.

If these challenges were overcome, the applications of indoor location-based services would enhance a wide variety of applications including: augmented reality, pervasive computing, advertising, social networking and asset management. Various techniques have been proposed and developed for indoor localization such as broadcast-based technologies (WLAN, RFID etc.) and motion-based technologies, such as inertial sensing-based localization. Many technology companies have been racing to develop Active Badge [103] type indoor location solutions like iBeacon [20] and Gimbal [15] that provide proximity-based services. RF signal strength based solutions are notoriously unable to provide accurate localization due to multipath caused by indoor obstructions. They also suffer from the barrier problem, where a node placed close to a wall often incorrectly localizes a user in the neighboring room. Although this may only present a low-error in terms of absolute distance, it equates to a large error in terms of context. In order to compensate for poor ranging technologies, the number of beacons can be increased and the transmit power reduced to improve spatial resolution. However, this higher density of beacons increases setup time along with hardware and management costs.

Ultrasound-based localization systems have been shown to outperform many RF-based systems in terms of accuracy. For example, the MIT cricket system [87] can range to within $5cm$, has a boundary detection accuracy of $1cm$ and can compute 3D location to within $10cm$ and orientation within $3°$. Ultrasound is comparatively much slower than electromagnetic waves, which makes it easy to measure signal Time Of Arrival (TOA) and perform TOF, Time Difference Of Arrival (TDOA) or Round Trip Time Of Flight (RTOF) measurements. It also does not suffer from the barrier problem since ultrasound does not penetrate walls nearly as easily as RF. The main drawback to ultrasonic approaches is that they require custom hardware and infrastructure. Our goal is to bring many of the benefits of ultrasonic localization to existing mobile devices like smartphones and tablets.

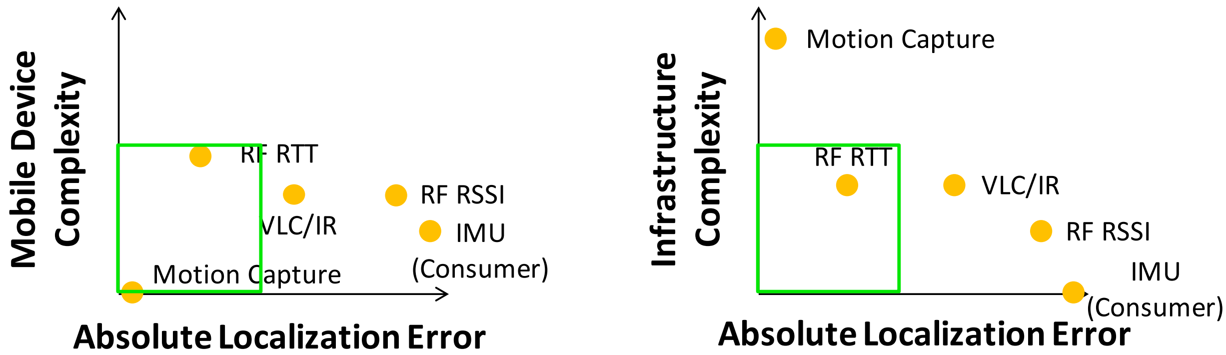


Figure 1.1: Mobile device and infrastructure complexity of indoor localization systems vs. absolute localization error

Figure 1.1 shows a comparison of current localization technologies in terms of mobile device complexity and infrastructure complexity versus location error. Current systems like [58, 59, 77, 87] require custom hardware (high mobile device and infrastructure complexity) in order to achieve low positing error or use WiFi signal strength based approaches for coarser-grained localization [38]. Inertial Measurement Unit (IMU) based systems require several inertial sensors, are only able to provide relative positioning and accumulate error over time due to drift. Visual Light Communication (VLC) and IR based systems such as [65, 90] work with unmodified smartphones while other systems like [103] require custom receiver hardware and are affected by ambient lighting conditions. RF RTOF systems like [9, 30] utilize Ultra Wide Band (UWB) ranging technology to obtain precise location information at high update rates, however, require custom receiver and transmitter hardware. We aim to provide highly accurate localization with minimal mobile device complexity, while also keeping the amount of transmitters and their complexity to a minimum. In this dissertation we describe the development of an ultrasound-based, indoor localization system ALPS, which utilizes off-the-shelf mobile devices as receivers and cheap transmitters to fit into the green squares in Figure 1.1.

1.3 Problem Statement

We propose to develop a method and system for providing sub-meter accurate indoor ranging and location data to off-the-shelf mobile devices using ultrasound signals. By off-the-shelf mobile devices we refer to any unmodified mobile computing device, which is equipped with digital sound recording hardware that has a frequency response reaching above the human hearing range. Current typical devices such as smart-phones and tablets are equipped with audio hardware sensitive up to a frequency of $24kHz$.

Given the increased processing capabilities of mobile devices, it is now possible to demodulate data and ranging information in real-time. In Chapter 4 we show the feasibility of detecting and measuring the propagation time of ultrasonic transmissions. We believe that this communications primitive can be used to develop transmitters that act as a fine-grained indoor localization infrastructure.

One can think of a mobile device as acting like a software-defined radio (SDR) for acoustic signals. This flexibility allows us to perform relatively complex signal processing, but also brings up many research challenges including: How should we construct the signal? How many signals can coexist in a single space (channel capacity)? Since timing and localization are so closely related, what are the limits in terms of time synchronization and ranging? This requires a new architecture that lies at the intersection of acoustic communication, timing, localization and mobile computing.

We aim to answer the following research questions:

1. How well can the limited ultrasonic audio bandwidth of commodity mobile devices be used for ranging and localization applications? What parameters impact performance in

terms of ranging accuracy, reliability, robustness, energy consumption and scale?

   (a) How can we make acoustic ranging transmissions imperceptible to humans, but still detectable using commodity mobile devices?

   (b) How can we allow for multiple-access between transmitters and receivers?

   (c) What is the capacity of this communication channel and how robust is it to external noise?

   (d) How can we efficiently map received signals to the beacons that sent them?

2. What are the trade-offs and limitations performing time synchronization through a mobile operating system's stack that would impact applications with tight timing requirements like ranging and localization?

3. What are the critical design components required of a transmission infrastructure to make an acoustic localization system economical and easy to deploy?

4. How can we maximize the signals coverage of ultrasound beacons to decrease the amount of beacons needed for an installation?

5. How does the proposed system compare to state-of-the-art technologies in terms of localization accuracy and update rate?

1.4 The Acoustic Location Processing System (ALPS)

The core concepts of this thesis are demonstrated through a system called the Acoustic Location Processing System (ALPS) which addresses many of the challenges found in indoor localization. ALPS is an ultrasound-based indoor localization system that uses the small amount of bandwidth just above the human hearing range to localize off-the-shelf mobile devices such as smartphones and tablets. As shown in Figure 1.2, a typical installation consists of two or more beacons deployed in the target area at known locations. The beacons transmit an ultrasonic ranging signal at pre-defined release intervals, i.e. Time Division Multiple Access (TDMA). The beacons are time synchronized using 802.15.4 radios that listen to periodic transmissions from a network master. The mobile device(s) to be localized are time synchronized to the beacon infrastructure using BLE and record audio at a high-sampling rate ($48kHz$) for a period of time to collect available transmissions. The mobile device(s) then demodulate any received signals, determine which beacon each signal originated from based on its TOA and then calculate the TDOA of the signals. The location of the receiver can then be calculated using multilateration based on the TDOA of the signals and the location of the beacons. ALPS is capable of time synchronizing tightly with the ultrasound transmissions by recovering the network master's clock using multilateration. While synchronized it can perform TOF ranging (see Section 5.4), which requires fewer ultrasound signals to be received to localize the device compared to TDOA pseudo-ranging. The location calculations are performed on a cloud based location engine, which communicates with the mobile devices via a WiFi or cellular data connection.

The ultrasound signals emitted by ALPS beacons are designed to be inaudible to humans, receivable by mobile devices and are able to provide high resolution ranging information. Specifically, we use acoustic linear chirp signals in the $20kHz - 21.5kHz$ frequency band, which is just outside the human hearing range ($20Hz - 20kHz$), but detectable using common MEMS microphones with $48kHz$ ADCs on mobile devices. The ultrasound modulation scheme is described in Chapter 4.

In a range-based localization system, signal timing accuracy translates directly into ranging and therefore localization accuracy. ALPS is able to determine the TOA of received ultrasound signals to a very high accuracy in part because we designed these signals to exhibit a property well known in the RADAR community called Pulse Compression (see Section 4.2). This dramatically increases the Signal to Noise Ratio (SNR) of the signal at the receiver over conventional

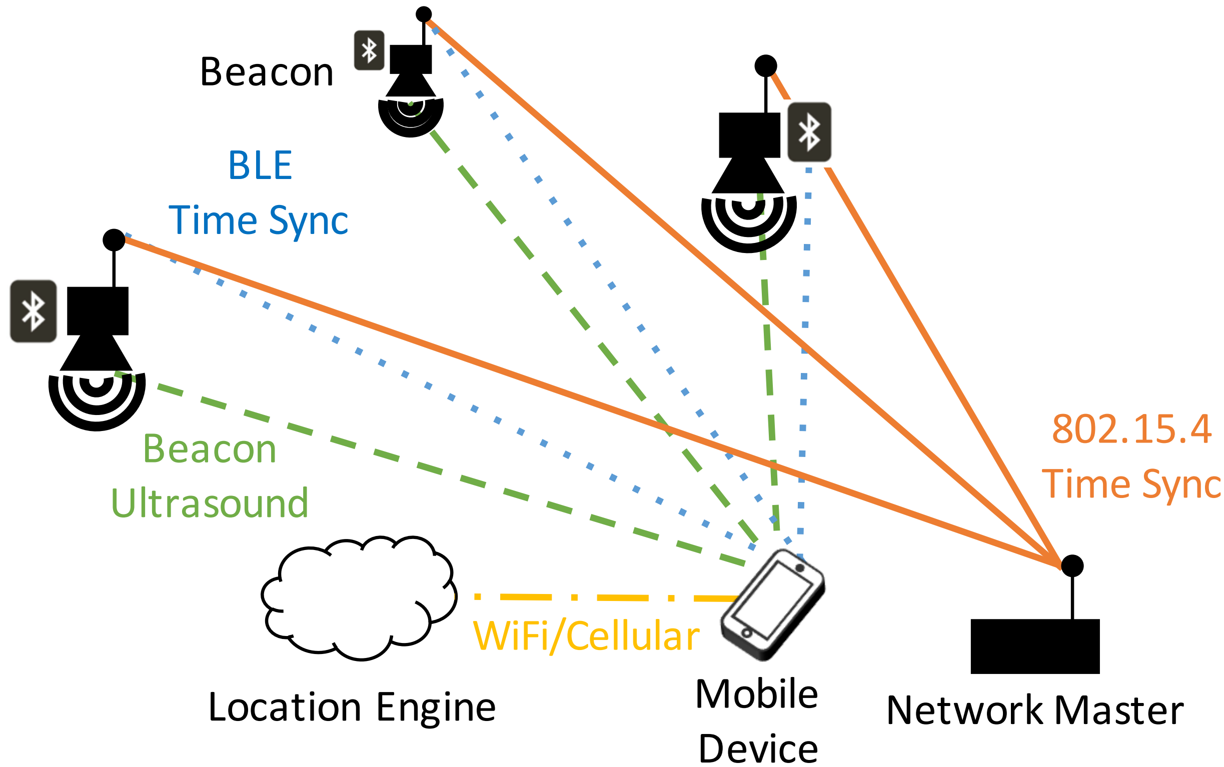


Figure 1.2: ALPS architecture overview

systems that do not employ Pulse Compression.

Despite the ultrasound signals being outside of the human hearing range, one of the main challenges associated with near sonic modulation over standard audio speakers (such as those used in ALPS beacons) is avoiding humanly perceivable artifacts. Since speakers are mechanical systems, they cannot instantly transition between gain settings without creating clicking noises. To alleviate these problems, our chirp signals require slow amplitude fade-in and fade-out changes, slow frequency changes and all adjustments are only made during zero-crossing points in the signal. We conducted a user study and extensive range and timing jitter testing in Section 4.3.1 to verify that our acoustic signal design fulfills all of our design goals.

Once a recording of several ultrasound signals is completed, it is demodulated in software on the mobile device and the TDOA or TOF (if tightly time synchronized via clock recovery) of the signals is measured based on their TOA and the time the signals were transmitted. Next the received signals need to be associated with their source beacons and in turn the known locations of the beacons in order to calculate the position of the mobile device. Since the beacons transmit ultrasound according to a fixed schedule, time synchronizing each mobile device to this schedule allows them to determine the source beacon of each ultrasound signal. To do this, our beacons and network master nodes transmit periodic BLE packets that contain a counter value indicating the time offset from the broadcast of the BLE packet to the beginning of the ultrasound transmission cycle. Due to the indeterministic timing jitter of delivering a BLE packet to the application running on the mobile device, our time synchronization precision is only to the nearest time slot (see Section 5.5). Once the TOA values are measured and mapped to the corresponding beacons, the ranges or pseudo-ranges from the mobile device to these beacons are calculated and sent to a cloud based solver, which calculates the location of the mobile device by multilateration or trilateration respectively, and sends the result back to the device.

1.5 Thesis Statement

The measurement of time synchronized ultrasonic signal propagation can be used to precisely range to and localize mobile devices.

1.6 Contributions

The work as part of this thesis will provide the the following contributions. The chapters of this dissertation which address each contribution are noted below.

1. The design and evaluation of an ultrasound modulation scheme (Chapter 4):
   (a) Which can provide accurate ranging information.
   (b) Which can provide multiple access for transmitters and receivers.

(c) Which is imperceptible to humans, but detectable by commodity mobile devices.

(d) Which is able to allow the receiver to map individual ranging signals to the transmitters that sent them.

2. Design and evaluate a method to time synchronize mobile devices to the transmission infrastructure which allows them to perform both pseudo-ranging (TDOA) as well as direct ranging (TOF) (Chapter 5).

3. Design and evaluate an embedded beacon platform (Chapter 3 and Appendix A)

   (a) Which can support the modulation scheme from 1.

   (b) Which can provide localization coverage to large areas with a relatively low amount of beacons by dispersing the transmitted signals in an omni-directional fashion.

   (c) Which is sufficiently energy efficient to allow for battery or energy harvesting operation.

4. Design and evaluate a location engine (Chapter 6)

   (a) Which can perform multilateration and trilateration to determine the location of mobile devices.

   (b) Which is scalable in terms of the amount of mobile devices being localized.

   (c) Which integrates with standard communication protocols for interfacing with mobile devices and location services.

   (d) Which provides IMU sensor fusion for improving localization accuracy and update rate.

   (e) Which supports the rapid localization of beacons by non-technical users for easy system deployment.

Chapter 2

Related Work

2.1 Overview

This dissertation touches upon topics related to ultrasonic communication, ranging, localization and time synchronization. All of these areas have large bodies of related research, so our discussion will focus primarily on closely related work in the mobile computing space.

At the core of any localization system lies a ranging or measurement technique which can be classified as a range-based approach or a range-free approach, as are described in Section 2.2 and Section 2.3 respectively. In Section 2.4 we detail current ranging technologies which incorporate these techniques and then describe localization systems that are based on these technologies in Section 2.5. Finally we address the related work on time synchronization that is critical to many localization systems in Section 2.6.

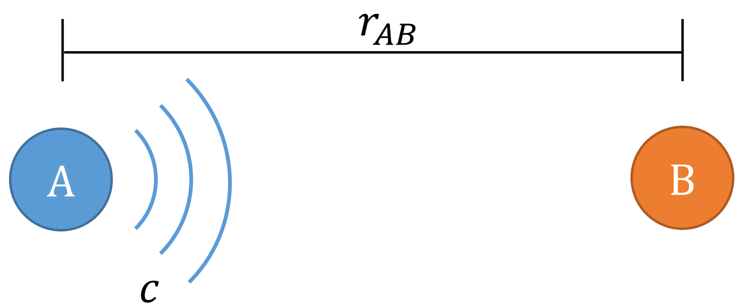
2.2 Range-based Approaches

Range-based approaches are used to directly or indirectly measure distances between a target that is being localized and another object or anchor point. This typically involves measuring the propagation time of a signal or the difference in propagation times. Systems such as [42, 59, 83, 87], make use of these techniques.

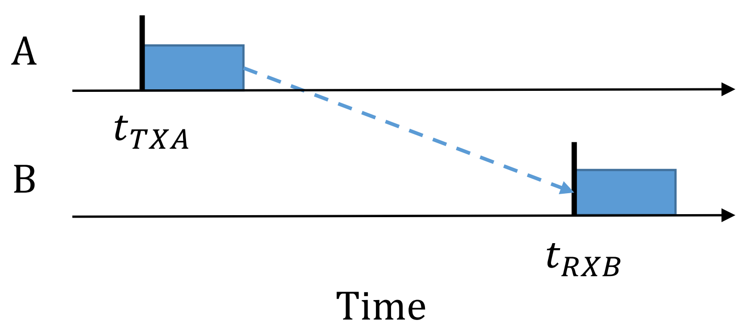
2.2.1 Time Of Flight (TOF)

TOF is a ranging method that directly measures the signal propagation time between a transmitter and a receiver, for example as seen between transmitter A and receiver B in Figure 2.1. TOF relies on knowing the time of transmission t_{TXA}, the time of arrival of the signal at the receiver t_{RXB} and the signal propagation speed c. From these values, the range r_{AB} can by calculated using Equation 2.1.

$$r_{AB} = c(t_{RXB} - t_{TXA}) \tag{2.1}$$



(a) TOF System diagram



(b) TOF Transaction diagram

Figure 2.1: Time of Flight ranging

There are several ranging and localization systems which employ variations of TOF ranging, such as many Radio Detection and Ranging (RADAR) (Section 2.4.1) and SONAR systems (Section 2.4.2), as well as Cricket [87] (Section 2.5.2) and ALPS. The primary challenge of many TOF based systems, is to time synchronize the transmitter and receiver to a precision high enough to allow the measurement of the propagation time of the signal being sent. ALPS uses a clock recovery method (see Section 5.4), similar to that of common GPS receivers to time synchronize with the beacon infrastructure. Common RADAR and SONAR systems do not require time synchronization since the transmitter measures the TOF of the signal traveling to its target and its reflection returning to the same transmitter.

2.2.2 Time Difference of Arrival (TDOA)

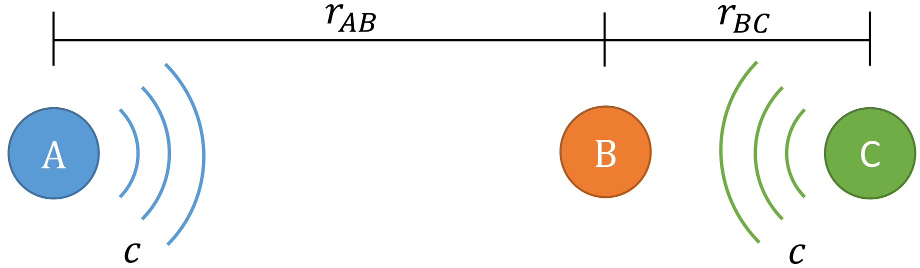
TDOA is a pseudo-ranging method that measures the Time Difference Of Arrival of two or more signals sent from transmitters to a receiver, for example as seen between transmitters A and C sending out signals of propagation speed c to a receiver B in Figure 2.2. Pseudo-ranging, as opposed to direct ranging like TOF, does not directly calculate the ranges r_{AB} and r_{BC} since the times of transmission t_{TXA} and t_{TXC} are not known. Instead, it assumes that the transmissions of the transmitters are concurrent or at known offsets and that otherwise only the times of arrival t_{RXB0} and t_{RXB1} are known. From this the difference of the ranges $r_{AB} - r_{BC}$ can be calculated using equation 2.1. Starting with the TOF equations 2.2 and 2.3:

$$r_{AB} = c(t_{RXB1} - t_{TXA}) \tag{2.2}$$

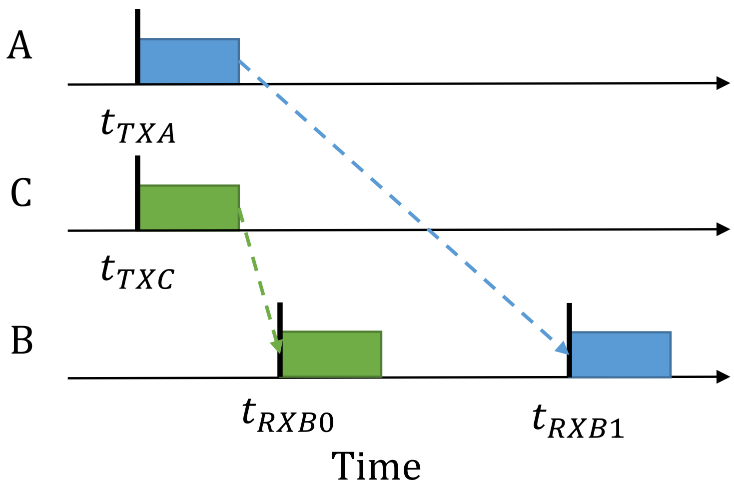
$$r_{BC} = c(t_{RXB0} - t_{TXB}) \tag{2.3}$$

Taking the difference between these ranges

$$r_{AB} - r_{BC} = c(t_{RXB1} - t_{TXA} - t_{RXB0} + t_{TXB}) \tag{2.4}$$



(a) TDOA System diagram



(b) TDOA Transaction diagram

Figure 2.2: Time Difference of Arrival pseudo-ranging

Since we assume a concurrent signal transmission, $t_{TXA} = t_{TXB}$. Therefore:

$$r_{AB} - r_{BC} = c(t_{RXB1} - t_{RXB0}) \tag{2.5}$$

If the range between the transmitter A and C is known $r_{AC} = r_{AB} + r_{BC}$, we can substitute this into equation 2.5, from which the individual ranges r_{AB} and r_{BC} can be calculated using equations 2.6 and 2.7 respectively.

$$r_{AB} = \frac{c(t_{RXB1} - t_{RXB0}) + r_{AC}}{2} \tag{2.6}$$

$$r_{BC} = \frac{-c(t_{RXB1} - t_{RXB0}) + r_{AC}}{2} \tag{2.7}$$

The main advantage of TDOA based systems is that only the transmitters (or in an inverse system the receivers) need to be time synchronized instead of both the transmitters and the receivers. This allows for virtually unlimited receivers in systems like the GPS (see Section 2.5.1) and enables systems where the latter synchronization is not possible, like shooter localization systems such as [95]. Many systems such as ALPS [66, 67, 68] incorporate TDOA for localization. The main disadvantage of TDOA over TOF lies in the requirement for one additional transmitter per measured dimension (i.e. two in the above example for 1D, three for 2D and so on).

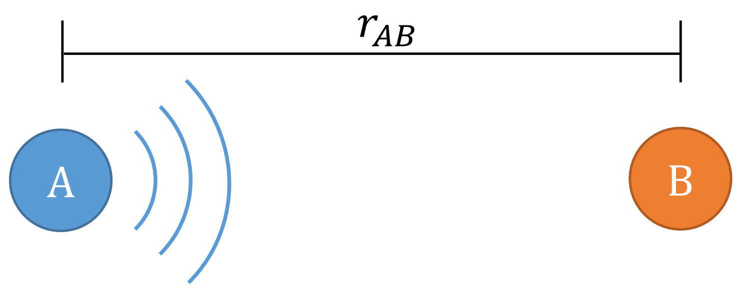
2.2.3 Round Trip Time Of Flight (RTOF)

RTOF is a ranging method that measures the time of flight of one signal sent from a node A to a node B, which then replies with another signal sent back to the A after a known time delay as seen in Figure 2.3. To calculate the range r_{AB}, node A sends a signal with propagation speed c to node B at time t_{TXA}, which is timestamped by its local clock. Node B receives the packet at time t_{RXB} and replies with another packet after a known delay of $t_{TXB} - t_{RXB}$. Node A receives the reply at time t_{RXA} and records this timestamp. The RTOF t_{RTOF} can then be calculated by equation 2.8.

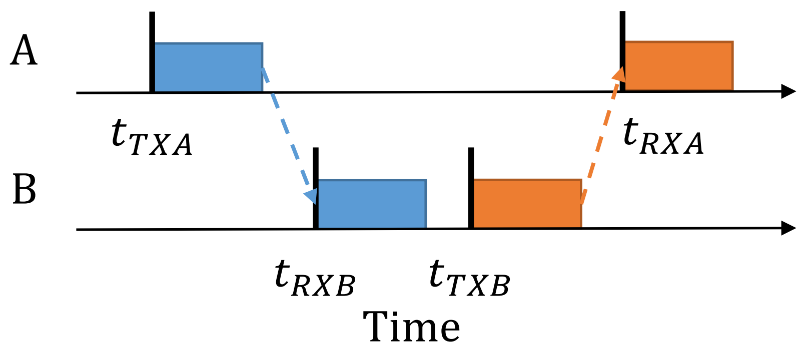
$$t_{RTOF} = t_{RXA} - t_{TXA} - (t_{TXB} - t_{RXB}) \tag{2.8}$$

Since t_{RTOF} is twice the time of a one-way trip, the range r_{AB} can be calculated using equation 2.9.

$$r_{AB} = \frac{c\, t_{RTOF}}{2} \tag{2.9}$$



(a) RTOF System diagram



(b) RTOF Transaction diagram

Figure 2.3: Round Trip Time Of Flight ranging

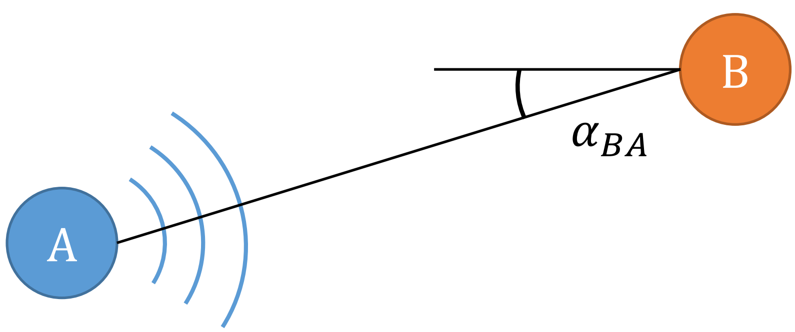
The major advantage of RTOF over schemes like TOF and TDOA is that no time synchronization is required between the nodes, which is often difficult to achieve to a viable precision in RF based systems. The scheme, however, is at a major disadvantage in terms of scalability since it requires an exchange of packets to take place, rather than only having one-way communication between nodes like in TOF and TDOA. There are several systems that employ RTOF for ranging, such as the majority of Decawave UWB systems [9] (Section 2.4.4), Nanotron systems [26], as well as the Beep Beep system [84] (Section 2.5.8).

2.3 Range-free Approaches

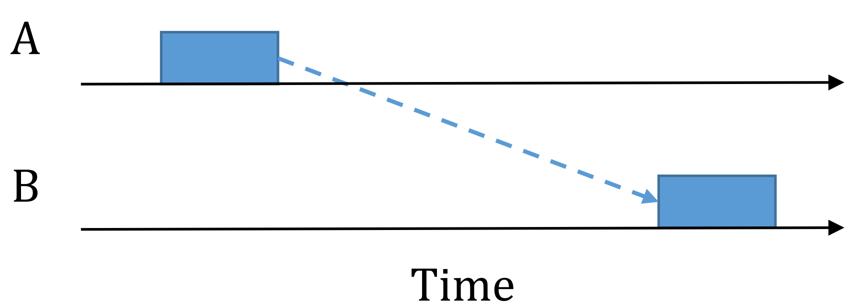
Range-free approaches do not rely on direct range measurements to determine distances, but instead rely on other metrics that can be used to localize devices such as the angle a signal arrived at with respect to the receiver or the signal strength of artificial or natural phenomena which may be mapped to a location or be used to calculate ranges. Systems such as [38, 72, 76, 99, 103], make use of these techniques.

2.3.1 Angle of Arrival (AOA)

Angle Of Arrival (AOA) is a method used to determine the angle of arrival α of a signal sent from a node A to a node B. As opposed to the ranging methods discussed previously (TOF, TDOA, RTOF), this method does not determine any ranges or pseudo-ranges, but only the angle that a signal propagated from with respect to the receiver. Systems commonly use arrays of receiving elements (antennas for RF and microphones for acoustic systems) to determine the AOA of a signal. The angle can be determined using the relative TOA of the same signal at different receiving elements on the same unit, which are at known locations or using the relative Received Signal Strength Indicator (RSSI) of the signal at the different receivers.



(a) AOA System diagram



(b) AOA Transaction diagram

Figure 2.4: Angle of Arrival measurement

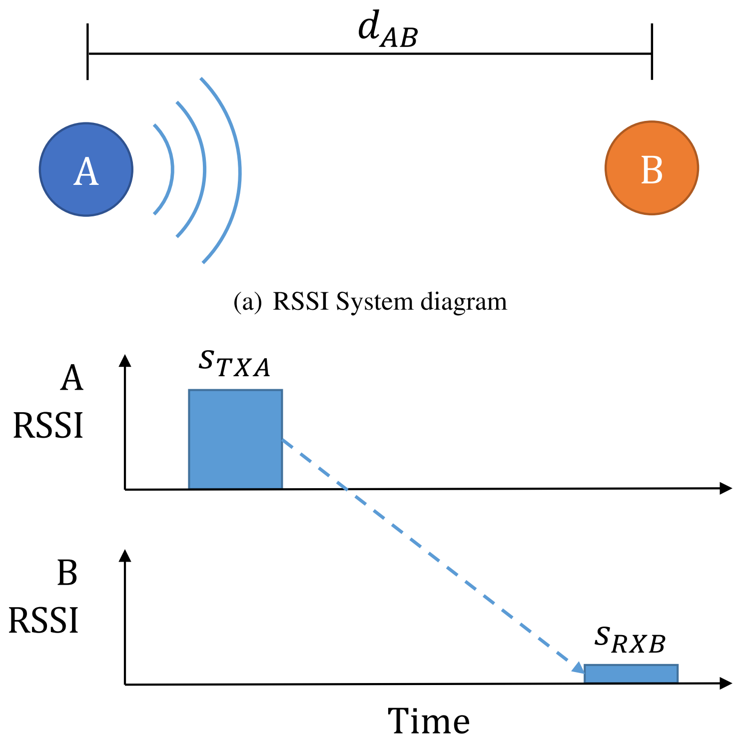
Although limited AOA measurements are already possible with only two receiving elements, practical systems such as Cisco's Hyperlocation system [19] use upward of 30 antennas for higher accuracy. Several AOA measurements from multiple transmitters can be used to determine the location of a receiver using triangulation. AOA measurements are often combined with range measurements using TOF, TDOA or RTOF for better localization in systems like [102]. The advantages of AOA lie in not requiring time synchronization or two-way communication between nodes, however, the additional hardware costs of having multiple receiving elements (or transmitting elements in an inverse system) is a major drawback, especially for mobile and power constrained devices.

2.3.2 Received Signal Strength Indicator (RSSI)

RSSI based ranging and localization are techniques that can be regarded to be range-based or range-free. These systems measure the range or location of a node B based on one or more nodes A that transmit packets of a known signal strength S_{TXA} and are received by B at a reduced signal strength S_{RXB} due to the loss in power associated with free space propagation and absorption by obstacles. From the difference in s_{TXA} and s_{RXB} an approximate range can be calculated by using a path loss model [96] such as the free space path loss formula 2.10, where s_{TXA} and s_{RXB} are in dB, f is the frequency of the signal and c is the propagation speed of the signal.

$$s_{TXA} - s_{RXB} = 20\log_{10}(\frac{4\pi r_{AB} f}{c}) \tag{2.10}$$

Different models can be formulated to different environments, propagation mediums and signal types. Alternately, the signal strength can be mapped to a location experimentally, which would be classified as a range-less localization technique. In general RSSI, especially for RF



(a) RSSI System diagram

(b) RSSI Transaction diagram

Figure 2.5: Received Signal Strength Indicator measurement

based systems, is a very noisy metric, which is highly dependent upon the environment, receiver and transmitter orientation, the directionality of the signals emitted by the transmitters, obstacles and multipath propagation, which leads to high positioning errors, usually of several meters. Despite this, it is currently the most prevalent method used in indoor localization products such as [1, 12, 15, 17, 20, 34, 97] and many more. Its popularity is primarily due to it working with WiFi signals from commodity access points, which are ubiquitous in most indoor environments and RSSI being an easily accessible metric on modern smartphones, tablets and laptops. One of the first systems which pioneered indoor localization using WiFi RSSI is Microsoft's RADAR system [38], which is described in Section 2.5.4.

2.4 Ranging Technologies

This section examines several ranging technologies which utilize distance estimation techniques from Section 2.2 and form the basis of many range-based localization systems.

2.4.1 Radio Detection and Ranging (RADAR)

RADAR has its roots as a military technology, originally developed before and during World War II for aircraft detection. It uses radio waves to measure range, velocity, angle and even the shape of objects. There are a plethora of commercial RADAR systems for a wide array of applications, ranging from military missile defense systems, to weather RADAR, to RADAR sensors in autonomous vehicles for obstacle detection. The architecture of a common RADAR system consists of a transceiver transmitting an RF signal at an object. The signal is reflected by the object, back to the transceiver and the signal's TOF (see Section 2.2.1) is measured. This information can be used to calculate the range of the object from the transceiver by using the classic RADAR equation 2.11 [91], where R is the range between the RADAR transceiver and the target object, P_S is the transmitted power of the signal, G is the antenna gain, λ is the wavelength of the signal, σ is the RADAR cross section and P_E is the received power. Other metrics such as the orientation of the RADAR antenna at the time of detection or the Doppler shift in the received signal may be used to calculate the location and velocity of the object.

$$R = \sqrt[4]{\frac{P_S G^2 \lambda^2 \sigma}{P_E (4\pi)^3}} \tag{2.11}$$

The RADAR community has been elemental in developing several signaling and ranging methods that are elemental to practically all range-based localization systems, including ALPS. One of these methods is Pulse Compression, which is described in Section 4.2.

2.4.2 Sound Navigation and Ranging (SONAR)

Active Sound Navigation and Ranging (SONAR) in principle is similar to RADAR (see Section 2.4.1), except that it uses sound waves as signals instead of radio waves. This makes SONAR particularly well suited for underwater applications, where RF signals attenuate faster than sound. In fact, marine animals have been using the principles behind SONAR for navigation for millions of years. Although SONAR is capable of functioning using air as a transmission medium instead of water or other dense materials, RADAR and Light Detection and Ranging (LIDAR) are generally favored for these applications due to their superior range and sampling rate in air. Man-made SONAR, like RADAR has most of its roots in military applications, although the first underwater echo ranging device was developed as a response to the Titanic disaster in 1913 [105]. SONAR systems have many applications, primarily in the maritime sector, like underwater mapping, military submarine and vessel detection and accident investigation for sunken ships and planes.

There are two types of SONAR systems: active, which like RADAR relies on an transceiver to send out a signal which reflects off of an object and is again received by the transceiver, and passive, which simply listens to signal sources that are in the environment. Both active and passive SONAR can be used to determine the range and location of an object using TOF and TDOA respectively. AOA, RSSI and Doppler shift may also be used to determine the location, range and velocity of an object.

Many signaling and ranging methods that are used in SONAR are also employed in RADAR, such as Pulse Compression (see Section 4.2), which is a technique also used by ALPS.

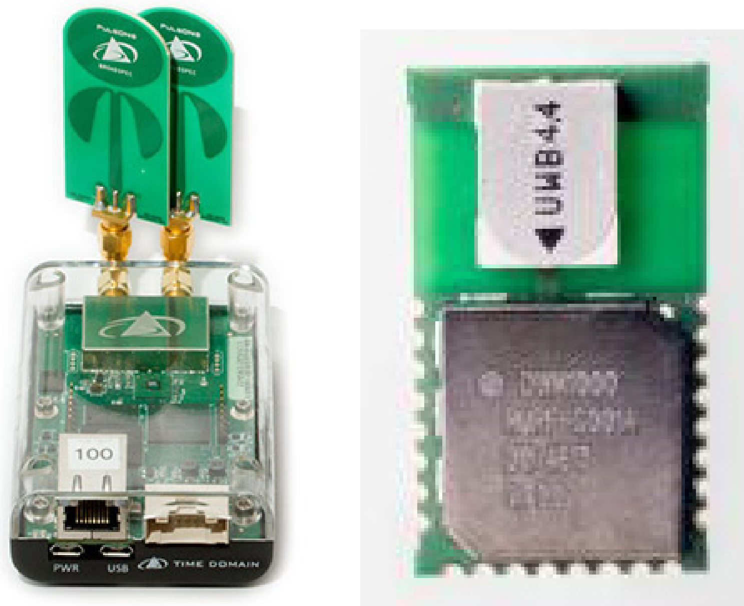
2.4.3 Light Detection and Ranging (LIDAR)

LIDAR is the most recent of the ranging technologies described so far, originating in the 1960's and is similar to RADAR and active SONAR in that it commonly measures the TOF of a signal being sent to and reflected off of an object. Instead of radio signals like RADAR or acoustic signals like SONAR, LIDAR uses light (ultraviolet, visible or infrared), typically emitted by a laser for ranging. Like RADAR, it is also generally used in air or space, but has the advantage of being able to capture higher resolution data due to the shorter wavelength of the emitted signal. This makes it ideal for applications such as high resolution obstacle detection in autonomous cars, 3D laser scanning , mapping, and atmospheric sensing. Generally LIDAR systems have less range and are more expensive than similar RADAR systems, but are becoming increasingly popular due their high resolution and the high amount of interest in autonomous cars and robots.

In terms of accuracy applied towards indoor localization systems, LIDAR based systems are regarded as the most accurate of all current technologies [7], however, their high cost and power requirements are prohibitive for many indoor localization applications, especially for mobile devices.

2.4.4 Ultra Wide Band (UWB) Ranging

UWB based systems use RF signals, usually in the sub-$1GHz$, $2-5GHz$ and $6-10GHz$ frequency ranges, over wide bandwidths for ranging. The wide bandwidth allows for precise timestamping of received signals. These systems commonly employ RTOF (see Section 2.2.3) ranging schemes, however, TDOA (see Section 2.2.2) is also possible with tight time synchronization [32] and TOF (see Section 2.2.1) for example when acting as RADAR (see Section 2.4.1) systems. UWB ranging has been an active area of research since the early 2000s [46, 62, 70], and has lead to the development of the IEEE 802.15.4a standard [35], which specifies two physical layer standards using UWB and Chirp Spread Spectrum (CSS) for precise ranging in short range networks. To our knowledge, there are currently only two companies who design IEEE 802.15.4a compliant chipsets specifically for UWB ranging: Time Domain [30] and more recently Decawave [9]. Time Domain enables sub-$2cm$ ranging accuracy at an update rate of up to $125Hz$ over up to $1.1km$ (dependent upon the environment, antenna and government RF power regulations) with its PulsON 440 module shown in Figure 2.6(a). This module is also capable of acting as a RADAR and performs TOF ranging by bouncing signals off of objects and receiving their reflections.



(a) Time Domain PulsON 440 module with second antenna for RADAR operation [30]

(b) Decawave DWM1000 module [9]

Figure 2.6: UWB ranging modules

Decawave's DWM1000 module, shown in Figure 2.6(b), integrates its DW1000 UWB ranging chipset, which claims a ranging precision of $10cm$, a maximum update rate of more than $100Hz$ and a range of up to $280m$ (although this requires a power level that is not compliant with FCC regulations). At a cost of 15.19 USD in single quantities, the DW1000 chipset is reasonably cheap and only requires few external components, which combined with its high accuracy, high update rate, small size, long range and ability to pass through a limited amount of walls have made it a clear choice for the majority of cutting edge UWB indoor localization systems such as [13, 29, 63]. Due to commonly employing an RTOF ranging scheme, these are however not as scalable as many competing systems. UWB chipsets also have not made their way into any smartphones, tablets or laptops yet, and it is questionable whether this will ever happen due to power and space constraints, as well as the need for a beacon infrastructure. Current ALPS beacons and network masters/and plug forwarders (see Section 3.2) also use this chipset for inter-beacon ranging, as well as future tag tracking.

2.5 Localization Systems

Localization systems are used to determine the location of a target, such as a mobile device in an environment. They may incorporate one or more of the techniques discussed in Section 2.2 and Section 2.3, and may be based upon one of the technologies described in Section 2.4.

2.5.1 Global Positioning System (GPS)

GPS is probably the most well known localization system to date. It is a type of Global Navigation Satellite System (GNSS) (other examples include Europe's Galileo, Russia's GLONASS and China's BeiDou-2) consisting currently of 32 satellites (31 active) and provides global outdoor localization coverage with an accuracy of around $5m$. It was developed by the US Department of Defense starting in 1973, and became fully operational in 1995 with a constellation of 24 satellites in medium earth orbit places [83].

GPS functions by transmitting RF signals from its satellites, which are tightly time synchronized using atomic clocks and are orbiting at precisely known locations in orbits around the globe. In its current constellation, about nine satellites are visible from anywhere on earth (that is not obstructed by obstacles like buildings) at any point in time. Each satellite transmits a signal that encodes a pseudo-random code that is known to the receiver, along with the time of transmission of a designated point in the pseudo-random code (known as an epoch), as well as the satellite's position at that time. A GPS receiver receives these signals from multiple satellites simultaneously and cross correlates the known pseudo-random code with the received signals to find their times of arrival. Since GPS receivers are typically not tightly time synchronized with the satellites, a TDOA (see Section 2.2.2) ranging scheme is used to determine a set of pseudo-ranges from the receiver to the satellites. The location of the receiver can then be solved for using multilateration. The speed of the receiver as well as highly precise Coordinated Universal Time (UTC) can also be calculated from the received signals. GPS receivers are commonly also able to perform TOF ranging (see Section 2.2.1) once they are tightly time synchronized.

GPS receivers are cheap and ubiquitous nowadays and have found their way into many systems from military missiles, to smartphones, to automotive navigation systems, to cameras that geotag photos. Unfortunately, GNSS signals do not penetrate walls, and are therefore not accessible in buildings so they cannot be used for indoor localization systems.

2.5.2 Cricket Location-Support System

MIT's Cricket Location-Support System [87] is an ultrasound and RF based localization system that measures the TOF (see Section 2.2.1) of ultrasonic pulses transmitted by beacons to mobile receivers, which are time synchronized to the beacons via RF. It is able to achieve a ranging accuracy of $5cm$, a localization accuracy of $10cm$ and an orientation accuracy of $3°$.

Cricket uses what is often called a thunder/lightning approach for ranging by first sending out an RF message that travels at the speed of light (lightning) to time synchronize nearby receivers to a beacon's transmission and also send the beacon's location, followed by an ultrasonic signal (thunder) at a known time offset from the RF signal. Since the RF signal's propagation time is negligible to that of the ultrasonic signal, the TOF of the ultrasound can be precisely and easily measured. The location of the receiver is determined by collecting range samples from multiple beacons and then performing trilateration.

Cricket uses custom beacons and receivers with standard $40kHz$ piezo ultrasound transmitting/receiving elements and $418MHz$ radios. $40kHz$ ultrasound transceivers are highly directional and in the Cricket system have a beam width of about $30° + / - 3dB$ in both the horizontal and vertical planes. This results in a high required beacon density for localization, however, also reduces interference between beacons. The system handles multiple access by randomizing transmissions in time instead of coordinating them, which reduces complexity and power consumption. Orientation of the receiver is calculated using AOA techniques and an array of ultrasound receiver elements on each receiver.

ALPS also faces the speaker directionality problem, although since it uses a lower frequency band, it is not as prominent as that of Cricket. Previous ALPS beacon generations used an omni-directional ultrasonic speaker horn (see Section 3.7.2), for increasing ultrasound signal coverage, while the latest beacons use a four speaker array for even better coverage and range (Section 3.7.3).

2.5.3 Dolphin

Dolphin [58, 59] is an acoustic based system that adopts a TDOA pseudo-ranging approach using ultrasonic modulation. Dolphin uses a $50kHz$ carrier for ultrasound signal transmission that is phase modulated by 511 bit Gold codes using Direct Sequence Spread Spectrum (DSSS) for multiple access. This allows all of their time synchronized beacons to transmit simultaneously and their receiver to disaggregate the signals in software by cross correlation or matched filtering to determine their TOA and which transmitter they originated from.

Dolphin uses broadband $50kHz$ ultrasound transmitters, which in their test setup are deployed in a very dense manner with four transmitters placed at the vertices of every $1.2m$x$1.2m$ square of area. The system provides a localization accuracy of $< 5cm$ for 95% of the measured points, and a high update rate due to the concurrent transmissions of ultrasound ranging signals. These transmitters also have a highly directional beam pattern, similar to MIT's Cricket (Section 2.5.2), therefore requiring a high transmitter density.

2.5.4 Microsoft RADAR

Microsoft RADAR [38] was the first WiFi based indoor localization systems and uses RSSI measurements (see Section 2.3.2) from multiple access points to determine the position of multiple laptop receivers. It is able to determine the location of a receiver to a median accuracy of $2.94m$ over a roughly $1000m^2$ large office space area using only three WiFi access points.

RADAR used an empirical and a propagation model based method for mapping WiFi RSSI data to locations. The empirical method consisted of walking through the target environment and taking RSSI measurements at regular distance intervals at multiple receiver orientations. To determine a receiver's location, RSSI samples were taken and then correlated with the training data set by using a k-nearest neighbour pattern matching algorithm. The propagation model based method consisted of modeling the WiFi signal propagation characteristics based on the target environment's floor plan and the Wall Attenuation Factor (WAF) of the walls. Overall the empirical model performed better with a median localization accuracy of $2.94m$, while the propagation model based method provided a median resolution of $4.3m$. While the empirical model performed better, it is generally more difficult to train and changes rapidly depending on how many people there are in rooms, if furniture is moved, etc. The propagation model method is somewhat more easy to train if the floor plan and the WAF of all walls are known, however, this model does not account for differences between receiver and transmitter hardware and dynamic obstacles such as moving people.

Microsoft RADAR laid the foundation for several systems such as [1, 15, 17, 20, 34, 40, 53, 71] and many more. WiFi RSSI based localization has come a long way, however, the inherent noisiness of the RSSI metric still make it lag behind range-based systems as can be seen in the results of Microsoft's annual localization competitions [22, 23, 24, 25].

2.5.5 Motetrack

The MoteTrack [72] system uses a similar approach to Microsoft's RADAR system (see Section 2.5.4), with an emphasis on distributed operation in a sensor network. MoteTrack is able to achieve a median and 80^{th} percentile location-tracking accuracy of $2m$ and $3m$ respectively over an office space area of $1742m^2$ using 20 beacons.

MoteTrack works by measuring the RSSI (see Section 2.3.2) of RF signals transmitted by several beacons within range of a receiver. Instead of WiFi signals, the system relies on custom hardware with $433/916MHz$ radios. MoteTrack varies the transmission power level of each RF transmission and transmits the used power level in its radio messages to provide more diverse data samples with different multipath, range and signal strength characteristics depending on the transmission power level. Like in Microsoft's RADAR system, RSSI and transmitter ID data is mapped to locations throughout the target space manually in a training phase. MoteTrack is designed to be fault-tolerant and can function with up to 60% of the beacons failing thanks to a decentralized location estimation protocol.

2.5.6 Active Badge

Active Badge was a pioneering indoor localization system which dates back to 1990 and uses IR transmitters and receivers to localize users via a simple RSSI (see Section 2.3.2) based localization scheme. The system consists of IR transmitters carried by users (active badges), which transmit unique ID codes once every $10s$ to IR receivers placed throughout the target environment. If a receiver receives an IR packet from a badge, it will signal a location server that the user associated with the received ID is present near the receiver. This provides room level accuracy and does not suffer from the barrier problem (incorrectly localizing transmitters to an adjacent room), since IR does not penetrate through walls. The badges lasted up to 16 months on batteries, however, the receivers needed to be hardwired in a network, which carries a high installation cost. There were also scaling limitations in terms of the number of transmitters as there was no multiple access scheme.

2.5.7 Active Bat

The Active Bat system [57, 104] is an ultrasound and RF based TOF (see Section 2.2.1) indoor localization system, which similar to MIT's Cricket (see Section 2.5.2) uses the thunder/lighting principle to measure the TOF of ultrasound signals. It achieves a localization accuracy of $14cm$ for 95% of location samples.

Active Bat system consists of mobile ultrasound and RF transmitters which send RF messages with the transmitter's unique ID to a network of receivers that are deployed in the environment. The TOF of the ultrasound signal emitted by the transmitter is measured by first transmitting an RF message to time synchronize nearby receivers, followed by an ultrasound signal after a fixed time interval. The receivers are all networked and attached to a central location server, which performs trilateration on the received TOF and ID values from the networked receivers. The orientation of transmitters can also be calculated based on the transmitter's location and the location of the receivers due to the directional ultrasound signal only being received by receivers in its narrow beam. Multiple access is handled by the location server coordinating the transmissions of the transmitters via a TDMA scheme.

2.5.8 BeepBeep

BeepBeep [84] is one of the first acoustic ranging systems that was implemented on off-the-shelf mobile phones. It uses an acoustic based RTOF ranging scheme (see Section 2.2.3) to determine the range between two mobile phones. BeepBeep uses the free running clock of the audio subsystem on the phone as a time reference for timestamping the transmission and receipt of the audio signals it sends. Its system setup is like that shown in Figure 2.3(a) and its ranging scheme works as follows:

1. Both devices A and B start recording audio.
2. Device A transmits an acoustic signal to B and records it using its own microphone which is co-located with its speaker.
3. Device B receives the signal and replies by sending another acoustic signal to A and records its transmission with its microphone.
4. Device A receives and records the acoustic signal from B.
5. Both devices examine their audio recordings to determine the time when they transmitted and received the acoustic signals (i.e. times t_{TXA}, t_{TXB}, t_{RXA} and t_{RXB} in Figure 2.3(b)).
6. The devices exchange these timestamps and can then calculate the distance between them using equations 2.8 and 2.9.

Two important aspects to note about the above process are the use of the free running audio clocks to timestamp the transmission and reception of the acoustic signals on both devices and that both devices record their own transmissions, which they timestamp later on when processing their recordings. When attempting to timestamp transmissions as they are submitted to the audio driver for transmission, there is a non-deterministic delay before playback starts, which can significantly impact the range measurement. By having the devices record their own transmission using their microphone, which is co-located with its speaker, this delay is eliminated and replaced by a constant, known time offset corresponding to the TOF of the acoustic signal from the microphone to the speaker. The use of the free-running audio clock instead of the phone's system clock also eliminates any clock adjustment that the OS may apply while the ranging operation is running. ALPS also makes use of the free running audio clock as is described in Section 5.3 and Section 5.4.

The BeepBeep system uses linear chirp audio signals for ranging instead of pure sine waves.

These signals exhibit Pulse Compression when being correlated at the receiver, which dramatically increases the SNR in the recording. Pulse Compression (see Section 4.2) is a signaling technique originally developed by the RADAR community, which is also used in ALPS for increased range and improved range resolution.

2.5.9 Enhanced 911 (E911) Mobile Phone Localization

The US FCC mandates several features for mobile phones, which allows them to be located in emergency situations [11]. E911 Phase 2 required that 95% of a network operators phones must be location capable by the end of 2015. This means that an emergency responder must be able to localize a phone to within $300m$. This requirement is expected to decrease over the next few years to within $10m$.

Most current smartphones provide their own location services based on a combination of GPS, WiFi, BLE and cell tower based localization, which can be forwarded to a 911 responder. If a phone does not provide location services, it may still be localized based on the AOA (see Section 2.3.1) between multiple antennas on multiple cell towers, the TDOA (see Section 2.2.2) of the phone's signals to at least three towers, or using fingerprinting of certain channel characteristics such as multipath fading which is mapped to locations within a cell. None of the latter methods provide highly accurate location data due to the Line of Sight (LOS) requirement for AOA and TDOA and the noisy nature of the fingerprinting method.

2.6 Time Synchronization

Time synchronization is tightly coupled to the performance of TOF and TDOA based localization systems such as ALPS. In TOF systems, time synchronization is required between transmitters and receivers, while in TDOA systems only the transmitters (or receivers in an inverse system) need to be time synchronized. This section provides and overview of time synchronization technologies that are commonly used in mobile and embedded devices such as smartphones, tablets and sensor network nodes.

2.6.1 The Network Time Protocol (NTP)

The Network Time Protocol (NTP) [75, 78] is the most common time synchronization protocol for computer systems in use today. It synchronizes systems commonly to within less then $1ms$ on wired local area networks and 10s of ms over the Internet of UTC. NTP employs a hierarchical structure of time sources, where each level is called a ”stratum”. Stratum layers are numbered starting at 0 at the top, and incremented by one for every additional layer. The stratum 0 layer is comprised of reference clocks that are highly accurate, such as GPS based and atomic clocks. Stratum 1 devices are time servers that are time synchronized to within several microseconds of stratum 0 devices. Each additional stratum layer time synchronizes with the one above itself at a gradually decreasing precision from the reference stratum 0 layer.

NTP's clock synchronization algorithm calculates the round trip-delay time associated with sending a packet back and forth between the client and the time server. This is virtually identical to the RTOF ranging method (Section 2.2.3), except that propagation time is not converted into a range. The packets may take different routes outbound and inbound, resulting in an asymmetric time delay, which can be filtered over multiple transmissions. Typically clients measure the round trip time-delay to at least three time servers, filter outliers and perform additional statistical analysis to estimate the time offset to UTC.

NTP is used in iOS and Android operating systems (among other technologies) to time synchronize the system's clock over a WiFi or cellular data connection. As we show in Section 5.3.4, there is a significant timing jitter associated with running NTP in a userspace application due to the often asymmetric latency of wireless data connections and the non-deterministic delay caused by timing packets at the application layer. ALPS employs BLE time synchronization and clock recovery methods to time synchronize to the beacon infrastructure as these methods produce less timing jitter (see Section 5).

2.6.2 The Global Positioning System (GPS)

As a result of providing high accuracy localization, GPS also provides a free, high precision time source, which can be distributed to receivers within an error of approximately $30ns$ [79]. GPS satellites are time synchronized to a reference time source at the US Naval Observatory and carry atomic clocks that are extremely stable to keep synchronization between themselves. Based on the TDOA pseudo ranging technique described in Section 2.2.2, we can see that in order to determine the location of a GPS receiver in three dimensions, we need to solve four simultaneous equations that include pseudo-ranges to four satellites. Given a position, the equations can be solved for GPS time. This is a clock recovery method similar to what ALPS uses as described in Section 5.4.

GPS receivers commonly can output a 1 Pulse Per Second (PPS) signal, which transitions on the second edge of GPS time, which is on the same scale as UTC. This transition is generated by the receiver's local clock based on the last GPS time synchronization, so it will contain clock drift and jitter errors caused by the receiver's clock. Frequency recovery of the $1Hz$ signal is typically done using a Frequency Locked Loop (FLL) or a Phase Locked Loop (PLL), which reduces these errors.

In [68] we advocated time synchronizing all network master nodes to GPS time to provide global time synchronization across large installations. Current generation 3b ALPS beacons are time synchronized to plug forwarders via 802.15.4, which in turn are time synchronized to a central network master via a Long Range Wide Area Network (LoRaWAN), which provides microsecond level synchronization precision across large installations (see Section 5.7). This eliminated the need for GPS receivers, which require LOS to the satellites and therefore reduced the installation flexibility of the system.

2.6.3 Cellular Timing Service

Several cellular networking standards feature or even rely upon tight time synchronization. Synchronous Code Division Multiplexing (CDMA) based networks such as CDMA2000 require sub-microsecond time synchronization between base stations and cell phones to coordinate simultaneous transmissions such that the signals are mutually orthogonal to each other in order to be decodable [98]. Long Term Evolution - Time Division Duplex (LTE-TDD) is a TDMA based protocol that relies upon time synchronization to cell phones within $1.5\mu s$ [81] to time multiplex its transmissions. Network Identity and Time Zone (NITZ) is a widespread protocol used to provide local time and time-zone information to cell phones. It's required accuracy is only the order of minutes [51], although in practice it usually achieves sub-second levels.

Although there are several cellular communications protocols that require tight time synchronization, it is unclear which smartphones, if any, synchronize their system clocks according to them (aside from NITZ) and to what precision. Protocols like CDMA2000 and LTE-TDD are usually handled by the baseband processor, which may not support synchronization of the system clock. Beacons could potentially be synchronized via these protocols, but the required hardware would be cost prohibitive in comparison to the 802.15.4 and LoRaWAN time synchronization ALPS uses (see Section 3.6).

2.6.4 Flooding Time Synchronization Protocol (FTSP)

The Flooding Time Synchronization Protocol (FTSP) [73] is a well known time synchronization protocol for synchronizing large multi-hop networks. The protocol consists of the following steps:

- A root node is elected based on the unique node IDs existing in the network.
- The root node periodically timestamps synchronization packets according to its global time and immediately broadcasts them as well as its node ID to all nodes withing range.
- Each receiving node immediately timestamps the received synchronization packets according to its own local time. The combination of the global and local timestamps that are referring to the same time instant is referred to as a reference point.
  - If the root node receives a packet from a node with a lower node ID, it gives up root status.
  - If a listening node does not receive a packet within a set time interval, it declares itself the new root node.
- Every receiving node estimates the clock offset between its local time and the global time of the root node based on the packet it received. Once it has received enough consistent synchronization packets it can correct its local clock skew and becomes synchronized.
- Synchronized nodes broadcast synchronization packets to the nodes within their broadcast radius to synchronize nodes further away from the root node. This allows reference points to be indirectly distributed across the network and gradually synchronizes nodes across multiple hops.
- This cycle repeats continuously to maintain synchronization.

FTSP is designed to work with sensor nodes which have low level access to interrupt routines for timestamping the synchronization packets. This reduces timestamping latency and jitter. The jitter is further reduced by calculating a node's clock skew in relation to global time over several packet transmissions. FTSP achieves a $1.5\mu s$ synchronization precision over a single hop with an average precision of $0.5\mu s$ per hop using Mica and Mica2 sensor nodes [74]. The protocol therefore provides more than enough precision for synchronizing nodes in ALPS installations, but falls short on power consumption as compared to the simpler protocol ALPS uses as described in Section 3.6.

2.6.5 Glossy Time Synchronization Protocol

The Glossy time synchronization protocol is a novel method to time synchronize sensor nodes which achieves an average time synchronization error of below $1\mu s$ [52]. Glossy uses a rapid flooding scheme to propagate time synchronization packets from a root node across an 802.15.4 wireless network through concurrent transmissions. It exploits the constructive interference of simultaneous time synchronization packets arriving at nodes to maximize their packet reception rate and is able to time synchronize large sensor networks of 94 nodes in less than $2.28ms$ with a reliability of greater than 99.98%. The protocol maximizes the likelihood of multiple packets achieving constructive interference at nodes by tightly bounding the latency from receiving a packet to transmitting another over the next hop. Range differences of up to $150m$ for simultaneously transmitted packets are allowable before destructive interference occurs at the receiver. Due to the tight timing constraints, the implementation of Glossy is radio specific, but it is theoretically possible to implement for most 802.15.4 radios.

Chapter 3

ALPS System Architecture and Platform

3.1 Overview

An ALPS setup consists of four main components as seen in Figure 3.1:

- Beacons
- A network master/plug forwarders
- Mobile devices
- A software location engine

The beacons are small embedded devices that are deployed throughout the target environment at known locations responsible for signaling to mobile devices. The beacon infrastructure is time synchronized using 802.15.4 by a central network master. Each beacon periodically transmits time multiplexed ultrasound raging signals to mobile devices that are to be localized within the environment. In order to improve coverage in larges spaces, plug forwarders can distribute messages across the network. The beacons send BLE packets to the mobile devices to time synchronize them with respect to the ultrasound transmission cycle in order for them to map received ultrasound signals to the locations of the beacons that transmitted them. A software

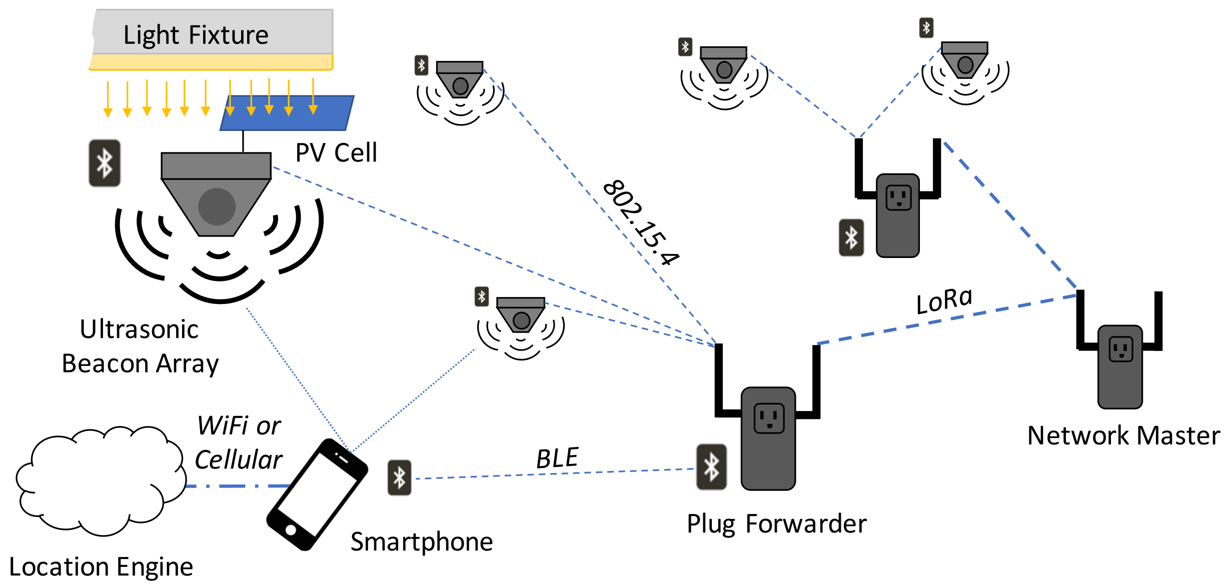


Figure 3.1: ALPS system architecture

location engine, which may be run as a cloud-based service or on the mobile devices, calculates the location of the mobile devices based on the time of arrival of the ultrasound signals they received and the location of the beacons.

The basic method to localize a device is as follows:

1. Time synchronize beacons to a central network master via 802.15.4.
2. Transmit periodic time multiplexed ultrasound ranging signals and BLE time synchronization packets from the beacons.
3. Time synchronize mobile device via BLE to the ultrasound transmission cycle.
4. Record ultrasound signals with mobile device and demodulate to measure TOA of signals.
5. Map TOA values to corresponding beacon locations based on transmission schedule.
6. Send TOA values and beacon mapping to location engine and solve for the location of the mobile device.
7. Send calculated location back to mobile device or process/store as needed by application.

In the following sections, we will describe the hardware and software design of the ALPS platform (Sections 3.2-3.5), provide a detailed explanation of the platform's transmission protocol (Section 3.6), describe and evaluate acoustic challenges and design decisions made in regard to the directionality and frequency response of the system's speakers and microphones (Sections 3.7-3.7.4) and evaluate the power consumption and energy harvesting capabilities of the beacons (Section 3.8).

3.2 ALPS Platform Hardware Design

ALPS beacons are custom embedded devices, which consist of the primary components shown in Figure 3.2 and Figure 3.3(c). A TI CC2650 System on Chip (SOC) with a 32-bit ARM Cortex M3 is the main processing unit of the system, which also contains on-board 802.15.4 and BLE radios. A stereo audio codec running at a sampling rate of $48kHz$ is used to generate the ultrasound signals that are amplified by the class-D piezo speaker amplifiers and then transmitted by the low cost piezo ultrasound speakers (< 1 USD). The piezo speaker amplifier contains an on-board DC-DC boost converter which supplies up to 19VP-P to the amplifier to better drive the piezo speaker, which improves the range of the system and also features a more advanced, lower power modulation scheme that increases the transmission efficiency. Four solid state relays are used to route ultrasound transmissions to individual speakers and for disabling speakers that are not in use. Although there are four speakers, only two unique ultrasound signals can be played back simultaneously since the audio codec only has two DACs. Beacon configuration settings and over-the-air firmware updates are stored on an 8MBit flash chip and a MEMs microphone connected to the audio codec can be used for acoustic beacon-beacon ranging. A Decawave DWM1000 UWB radio (see Section 2.4.4) can be used for longer range beacon-beacon ranging and tag tracking. An energy harvesting controller manages system power and charges three rechargeable batteries using an external solar cell. As seen in Figure 3.3, the hardware is housed in a ceiling mountable enclosure with a 3D printed top that contains the speakers and is attached to an off-the-shelf base containing the PCB and batteries.

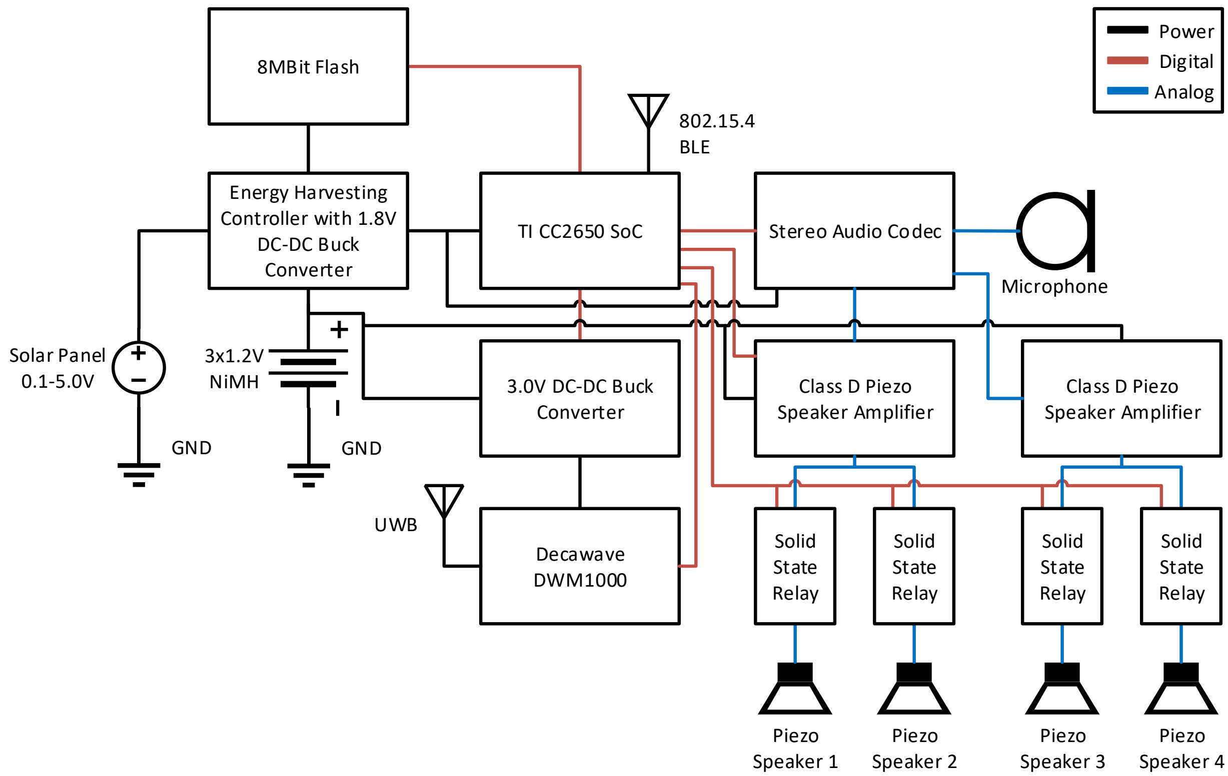
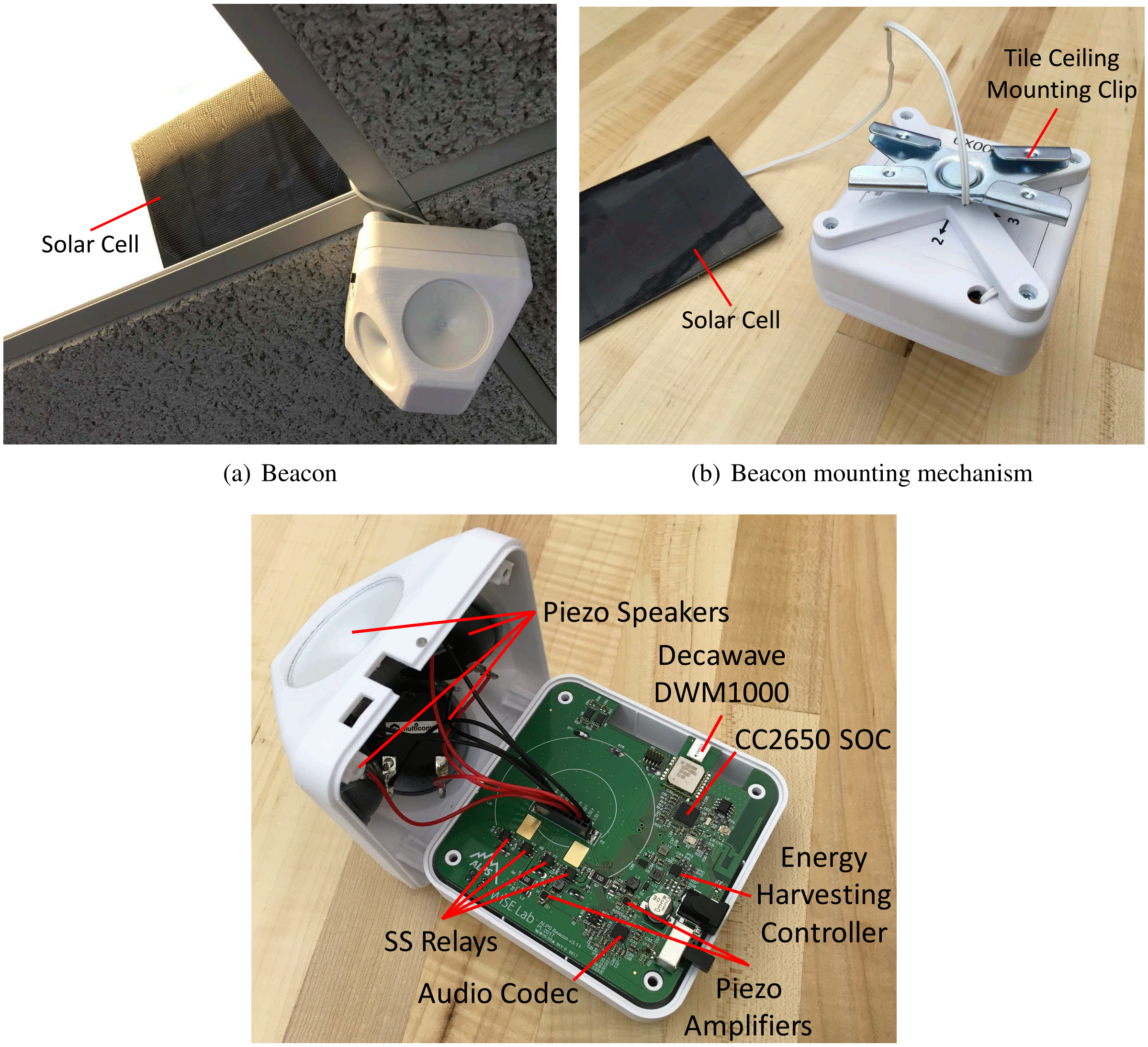


Figure 3.2: ALPS beacon architecture

The platform is designed to have a low enough power consumption so that it can be powered using a small solar cell, harvesting energy from artificial or natural light sources (see Section 3.8). This allows for a flexible installation at a low cost, since the beacons do not need to be connected to AC wall power, which is often difficult to access at ceiling mounting locations. The harvested energy is buffered in three ultra low self discharge NiMH batteries with $2000mAh$ each, which have a high cycle lifetime of 2000 cycles and retain 70% of their charge after ten years.

The network master/plug forwarder hardware is based on the same TI CC2650 SOC used in the beacons (see Figure 3.4 and Figure 3.5(b)) and shares a significant amount of their code base. It also features a LoRaWAN radio for providing a single hop, long range data link and time synchronization (see Section 5.7) to plug forwarders (see Figure 3.1) in large deployments.



(a) Beacon (b) Beacon mounting mechanism

(c) Beacon PCB

Figure 3.3: ALPS beacon hardware

A Decawave DW1000 UWB radio (see Section 2.4.4) may be used for ranging to beacons and future tag tracking. All of this is integrated into a plug design, which directly plugs into electrical outlets as can be seen in Figure 3.5(a). The network master/plug forwarder has a USB port to connect to a computer for sending commands to the ALPS deployment, as well as for distributing over-the-air firmware updates.

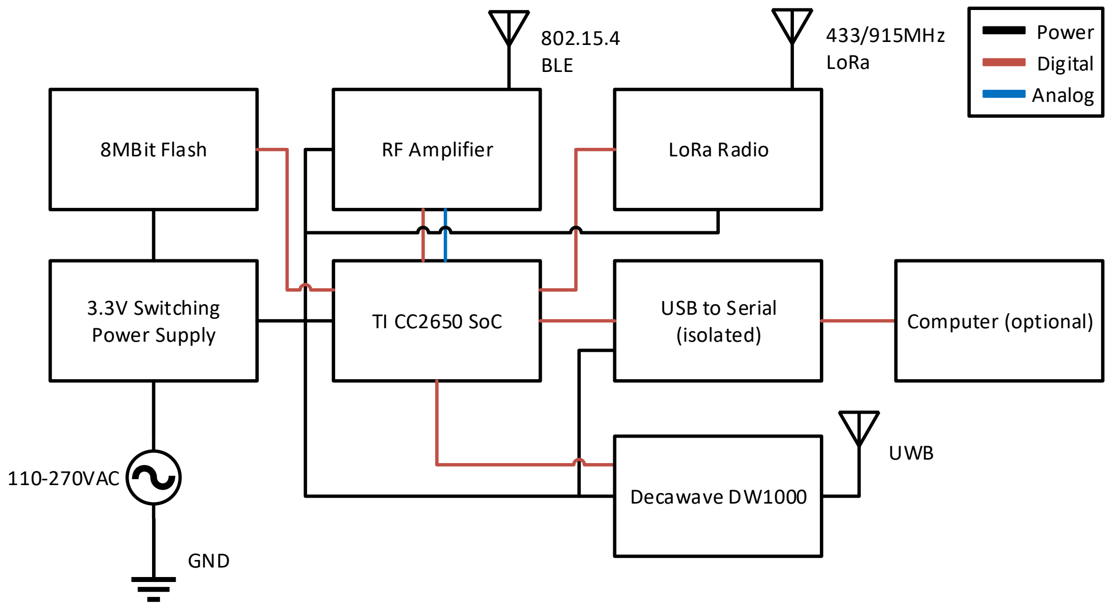
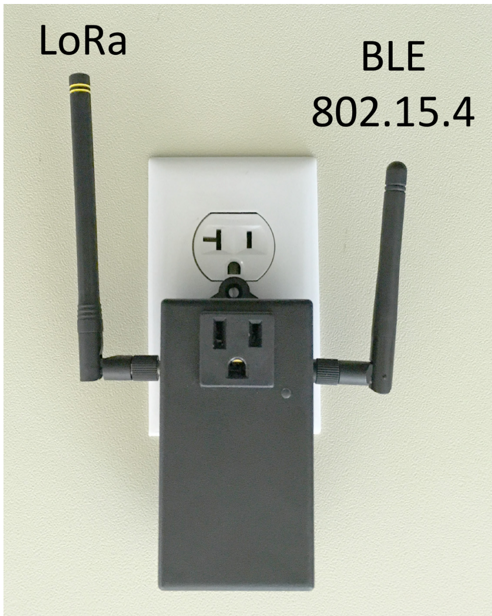


Figure 3.4: ALPS network master/plug forwarder architecture

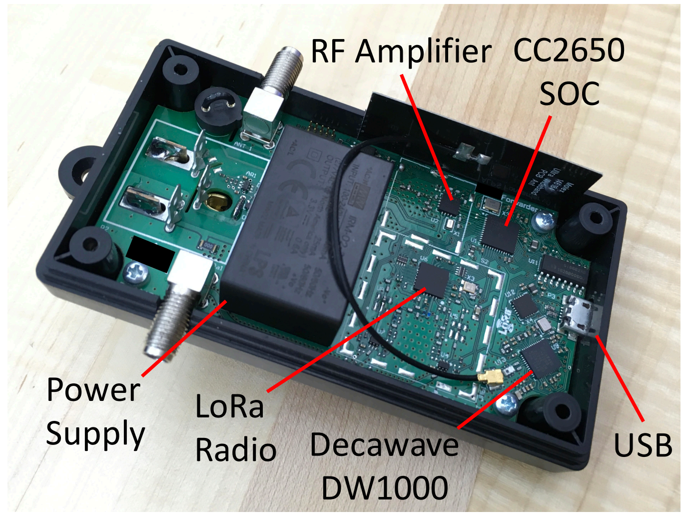
3.3 Beacon and Network Master Firmware

ALPS beacons and network master nodes run TI-RTOS [31], which is a preemptive multitasking microkernel based Real Time Operating System (RTOS) that also provides device drivers for most peripherals on the CC2650. The firmware is segmented into multiple tasks (generally one per peripheral e.g. radio, audio, flash memory, etc.), which run at different priority levels and can be preempted. Since the CC2650 features a Direct Memory Access (DMA) controller, the CPU can run other tasks during SPI transactions and when audio is being played back. The on-chip radio is controlled by an ARM Cortex M0 core that has shared memory access with the M3 core and can therefore transmit and receive packets without help from the CPU. Ultrasound waveforms are generated on the fly using ARM's CMSIS DSP library [3].

The firmware running on the network master/plug forwarder is a variant of the beacon firmware reusing several tasks like accessing the external flash memory and radio. The forwarders are responsible for running the LoRa radio and for sending out time synchronization packets. It also provides a serial interface which can be used to control the system.



(a) Network master/plug forwarder



(b) Network master/plug forwarder PCB

Figure 3.5: Generation 3b ALPS network master/plug forwarder hardware

3.4 iOS Application

The software running on the phone consists of an audio recording routine, a demodulator, a BLE synchronization routine and the necessary networking code to communicate with the cloud based location engine.

Upon starting the application, the BLE routine first attempts to time synchronize with the ultrasound transmission cycle of the beacons. It does this by listening for iBeacon advertisement packets sent by the beacons that contain a counter value indicating the time elapsed since the start of the current transmission cycle (see Section 5.5). The application timestamps the receipt of these packets and calculates its corresponding system time in relation to the beacons' transmission cycle. Once the application is synchronized via BLE, the audio recording routine is started and runs continuously in the background. It fires a callback whenever an audio buffer is filled, which is then passed to the demodulator (see Section 4.5.3). The demodulator extracts the TOA values of the received ultrasound signals and the corresponding beacon IDs based on the

BLE time synchronization. These values are then sent to the location engine over a websocket connection, or may be processed internally if the location engine is running locally.

The iOS application is written in objective C and the demodulator is written in C from code generated by MATLAB's C coder.

3.5 Future Hardware Improvements

The current ALPS design uses multiple speakers for improved ultrasound signal coverage. The mismatch between two audio channels and four speakers limits us to transmitting from only two speakers at a time to prevent interference, which results in a longer transmission time (see Section 4.5.6). Based on the speaker beam pattern seen in Figure 3.11 and the linear increase of power associated with adding multiple speakers to an amplifier, the increase in coverage should have been directly proportional to the increase in the power requirement. However, due to the overhead in the idle power consumption of the piezo speaker amplifiers, the addition of an another amplifier disproportionately increased the power requirement. Despite this, it is still possible to power each beacon perpetually using solar energy harvesting as evaluated in Section 3.8. A future beacon generation would likely support four channel audio to eliminate the need to time multiplex transmissions between the speakers and would use a four channel piezo amplifier with a single DC-DC boost converter to reduce the power overhead of multiple piezo amplifiers with multiple boost converters.

3.6 Transmission Protocol

This section presents the protocol used to coordinate RF and ultrasound transmissions across a network of beacons. Figure 3.1 shows a typical setup with an overview of the transmission protocol as referred to below shown in Figure 3.6. ALPS utilizes a mixture of LoRaWAN,

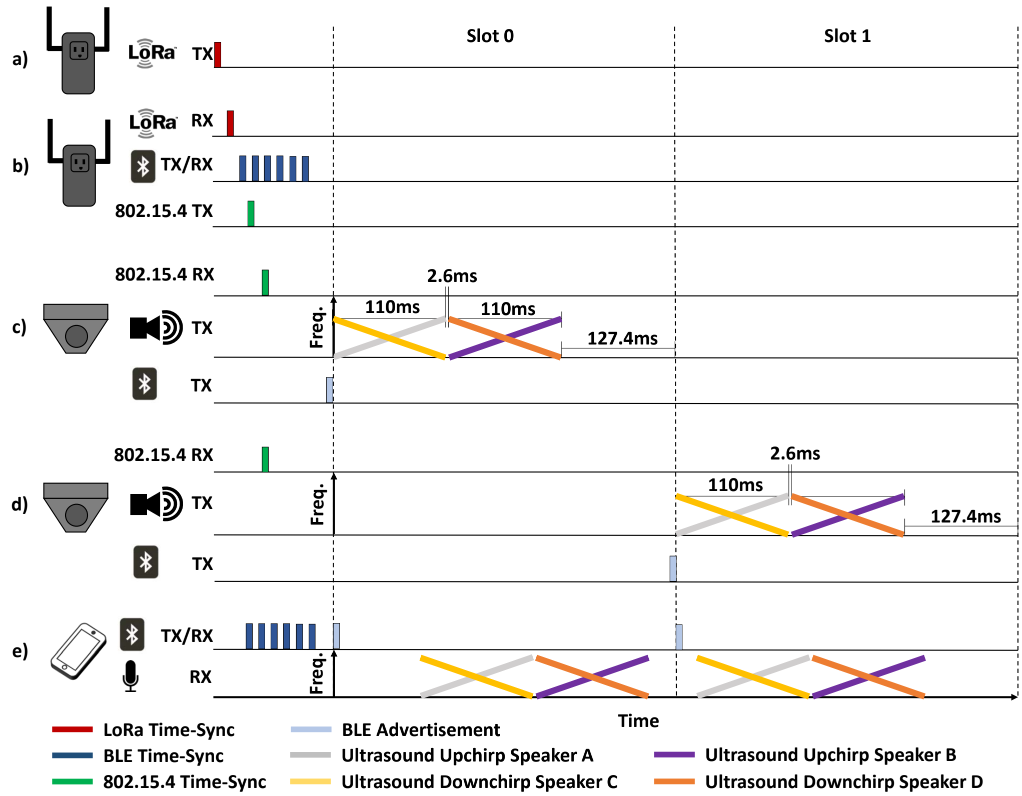


Figure 3.6: Generation 3b ALPS transmission protocol

802.15.4 and BLE communication. 802.15.4 remains in use instead of solely BLE due to its increased range, additional features like automatic acknowledgements, more available channels (as opposed to BLE advertisement channels) and existing legacy code. LoRaWAN is used to both transmit time synchronization signals across large deployments in a single hop from a central network master as shown in (a), as well sending system control commands. The same hardware used for the network master is re-purposed as an RF forwarding device (b), which acts as a bridge between the LoRa network and the beacons, which operate on BLE and 802.15.4. The forwarders are synchronized by the network master using LoRaWAN (Section 5.7) and then synchronize the beacons (c) and (d) via 802.15.4 (Section 5.6) and the smartphones (e) via BLE (Section 5.5), as seen in the time span before slot 0. Synchronization is performed periodically so that the clocks of the devices don't drift too far apart.

The ultrasound modulation detailed in Section 4.5.6 is used for ultrasound ranging. Up- and down-chirps are transmitted simultaneously from two speakers as they are orthogonal to each other and the beacons only support two audio channels (see Section 3.2). This transmission can be seen as an "X" in (c-e). The second "X" is transmitted from the remaining two speakers after the first. Each ultrasound chirp in our system is sized to be $100ms$ in length (plus an additional $5ms$ each for fade in/out) to provide adequate range, with 2.6ms between successive chirps to switch the audio channels via the solid state relays (Section 3.2). Each "XX" transmission is followed by a period of silence for $127.4ms$ to let the ultrasound transmissions travel across the room and dissipate, resulting in a total slot length of $350ms$. The length of the chirps and the period of silence may be adjusted according to room size.

3.7 Speaker and Microphone Directionality

This section details the constraints imposed upon ALPS due to the directionality of the speakers in the beacons and the microphones in the mobile device receivers. We present measurements of the beam patterns of our beacons, sample mobile devices and show two ways how we were able to achieve an omnidirectional beam pattern with our beacons: First describe the design of an omnidirectional ultrasonic speaker horn, which can disperse ultrasound signals from a directional speaker. Secondly we show an omnidirectional sectored speaker design, which is a more costly solution compared to the horn, but can support AOA measurements between the beacon and the mobile device and has an increased transmission range.

3.7.1 Experimental Setup

To measure the beam pattern of speakers, we designed an automated measurement system consisting of a serially controlled, motorized pan tilt mechanism onto which the speaker under test would be mounted, as can be seen in Figure 3.7. We would then place the setup into an anechoic chamber and position an Audix TM-1 measurement microphone at a fixed distance in front of the mechanism. A Motu ultralite mk3 audio interface running at a sampling rate of $192kHz$ would generate and record the test sounds and an Onkyo HT-R540 amplifier was used to drive the speaker. Linear chirp signals, which sweep a sine wave signal between two frequency bounds at a constant rate, would be played back across our target frequency range of $19kHz$-$24kHz$, which accommodates many modern smartphone and tablet models as can be seen in Section 3.7.4 and [69]. An ideal speaker would emit every frequency in this range at the same power level, however, real speakers have non-flat frequency responses so the transmit power level may vary greatly with respect to frequency. After several samples were recorded, we would rotate the speaker a fixed interval in the horizontal plane and again record several chirp signals. This process would continue over at least $150°$ in the horizontal and vertical planes. The resulting data can be used to produce beam patterns that show the intensity of the received signal in relation to the angle of the speaker. Ideally a beam pattern should be omnidirectional across all tested frequency bands, which would mean that it disperses the signal equally in all directions.

3.7.2 Omnidirectional Ultrasonic Speaker Horn

In certain installations, it may be sufficient to have a single omni-directional speaker that supports a lower-cost deployment. In an early generation of ALPS (Generation 2b - see Section A.3, and Generation 2c - see Section A.4), beacons featured an omnidirectional ultrasonic speaker horn, which was used to disperse the sound of their speaker. In a typical loudspeaker, as the audio frequency increases, the spatial spread of the signal decreases, eventually forming a narrow beam. In our system, we ideally want an omni-directional speaker that has a flat frequency response across the $19-24kHz$ frequency band, that can uniformly deliver ranging signals without distortion. Since no such speaker was commercially available, we designed a custom transducer based on a multi-sector omni-directional horn design shown in Figure 3.8, Figure A.8(a) and Fig-

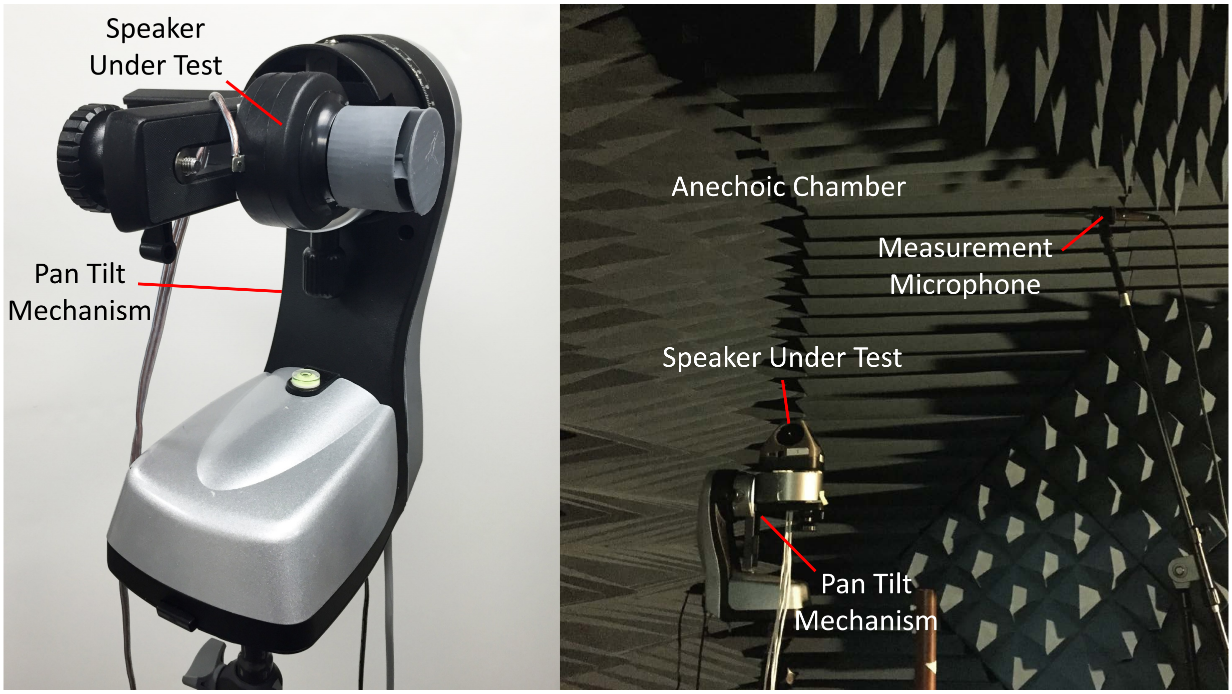


Figure 3.7: Beam pattern measurement setup with pan-tilt mount and measurement microphone in anechoic chamber

ure A.10(a). This turned out to be a non-trivial effort that required significant experimentation.

We initially evaluated multiple commercial speakers in order to determine suitable driver components and geometries. In terms of frequency response, we found that ribbon tweeters (as used in Generation 2a beacons - see Section A.2) had an excellent frequency response and horizontal dispersion pattern. Unfortunately, they require large magnets that are both heavy and expensive ($>$ 70 USD). They also have a narrow vertical beam pattern. In certain scenarios, they could be an ideal transducer, but are too expensive for general purpose indoor localization applications. We also evaluated piezo bullet speakers since they are low-cost ($<$ 2 USD) and have a reasonably linear frequency response. Unfortunately, without a horn to guide the signal, they are quite directional. The top two rows in Figure 3.9 show a comparison of the vertical and horizontal beam patterns of a ribbon tweeter and piezo bullet speaker.

The acoustic literature has many models that describe a wide variety of speaker designs [39]. Most of the common designs tend to be for audible frequencies and exhibit confined beam

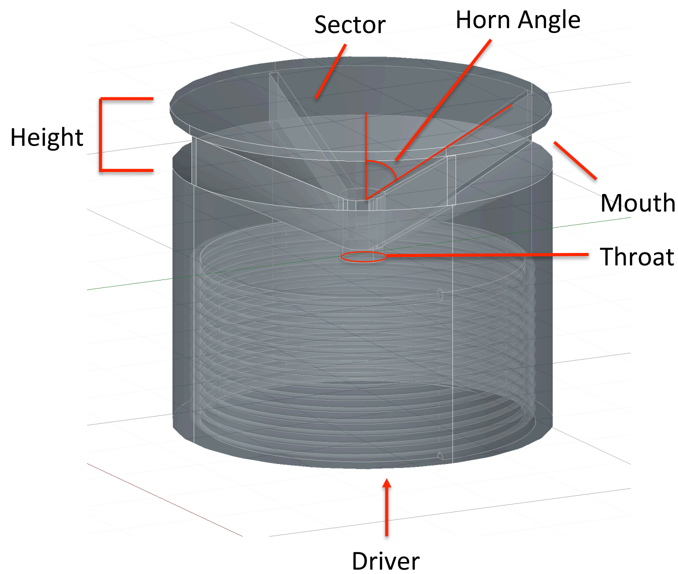


Figure 3.8: Omnidirectional ultrasound horn CAD model

patterns. In order to design a custom horn, we initially modeled a cone based on standard horn equations. These models specify the width of the horn's mouth to be $4.76mm$ in diameter to support frequencies above $20kHz$. The resonant chamber needs to be at least one wavelength, or $1.6cm$ in length. The horn throat then needs to be sized in order to reduce distortion while having sufficient amplification. A point source (pin-hole speaker) would be ideal, except that the volume would be insufficient. Figure 3.8 shows the basic geometry of our omni-directional horn. In order to evaluate performance, we varied the horn angle, the height of the top of the horn and experimented with different numbers of internal sectors. Each horn variant was printed on an SLA 3D printer and then tested using a pan-tilt mechanism that allowed automatic frequency response measurements to be taken at different angles. We tested 12 different horn designs generating a vertical and horizontal frequency response plot by using the method detailed in Section 3.7.1.

We define two metrics to compare different speaker configurations. These metrics are computed from the gain values at different frequencies and directions, as seen in Figure 3.9. To

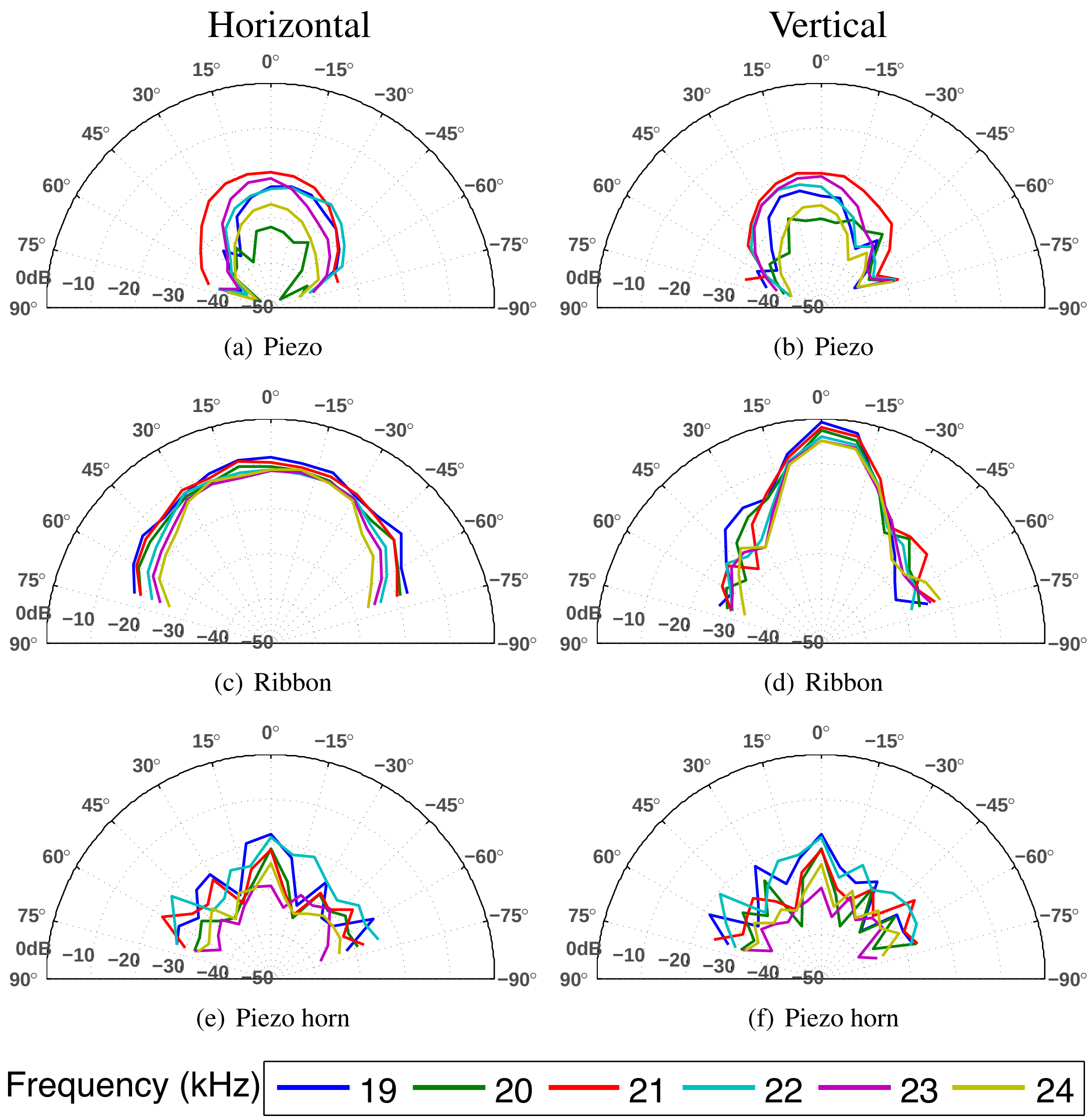


Figure 3.9: Ultrasonic beam patterns

measure the flatness of frequency response, we compute the frequency distortion. The frequency distortion of a speaker in a particular direction is the difference between the maximum and minimum gain in the frequency band of interest. We average this metric across all directions to compute the frequency distortion (lower plots in Figure 3.10(a-d)). To measure the deviation from omni-directionality for a speaker, we first find the gain in a particular direction by averaging the gain across the frequency band. We then compute the average deviation from the mean gain across all directions to arrive at the directional distortion (upper plots in Figure 3.10(a-d)).

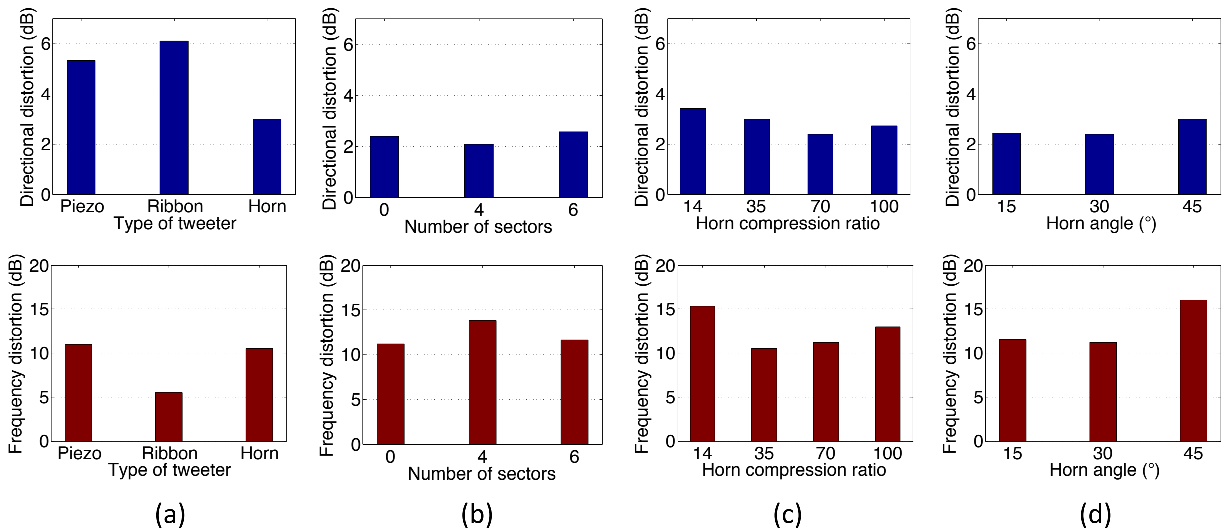


Figure 3.10: Omnidirectional ultrasound horn design evaluation

Both these metrics are averaged across the horizontal and vertical orientations for each speaker.

Frequency distortion as well as directional distortion both directly impact the SNR at the receiver. Frequency distortion will create a mismatch between the recorded signal and the template used during matched filtering, while directional distortion will vary the signal level with respect to the angle between the beacon and the receiver. A decrease in SNR increases timing jitter when determining the TOA of the received ultrasound transmissions, which in turn negatively impacts ranging and localization performance.

3.7.3 Quad Sectored Speakers

Although the speaker horn from Section 3.7.2 provided an omnidirectional beam pattern, we decided to use a different speaker for the current ALPS beacons (see Section 3.2) due to reliability problems and the large size of the previously employed piezo bullet speaker. Since the horn from previous generations was not compatible with the new speaker, we decided to develop a four speaker design, which also provides omnidirectional dispersion, but also has the possibility of obtaining AOA measurements (see Section 2.3.1) for potential single beacon deployments in the future.

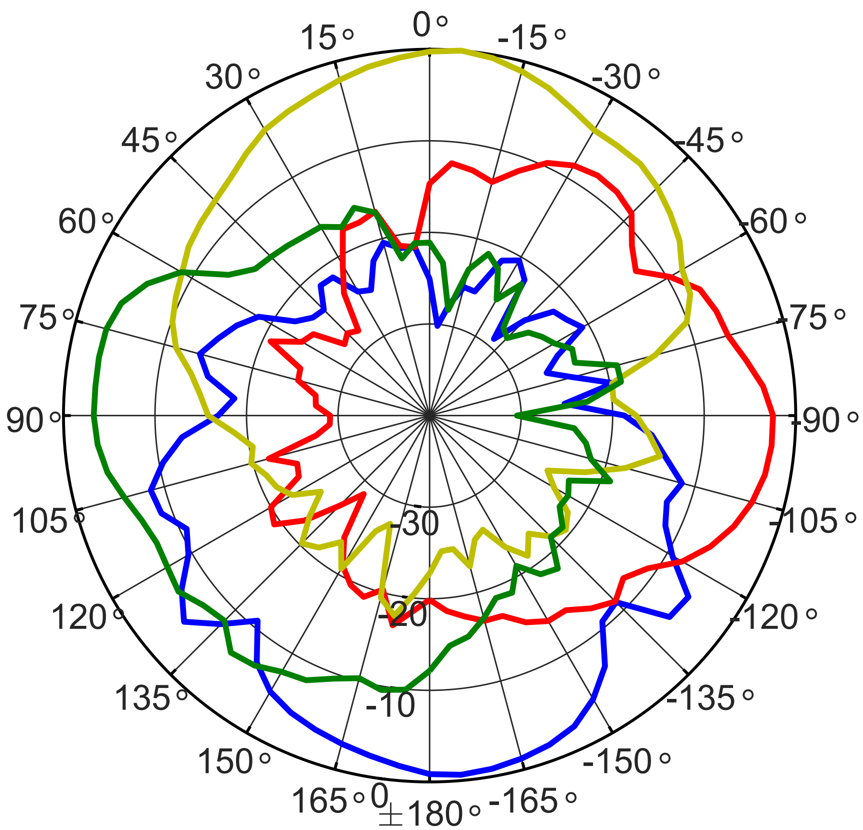


Figure 3.11: ALPS beacon Generation 3b, quad speaker array horizontal beam pattern

The horizontal beam pattern of a Generation 3b beacon can be seen in Figure 3.11, where each color represents the pattern of one speaker. The vertical pattern is largely the same as the speakers are symmetric about both planes. Note that the curves on this pattern measure the received signal strength after being cross correlated with the transmitted chirp signal, rather than the received signal strength at different frequencies as shown in Figure 3.9. While the information about signal strength at specific frequencies is lost in this representation, it is the signal strength over the entire frequency band which determines the performance of the transmitter and is more clearly represented in this way. When the patterns of all speakers are combined, the dispersion of the signal is very omnidirectional, with a difference of $12dB$ between the maximum and minimum points on the pattern, and has a longer range of a $35m$ radius compared to the $20m$ radius range of previous generation beacons using the horn.

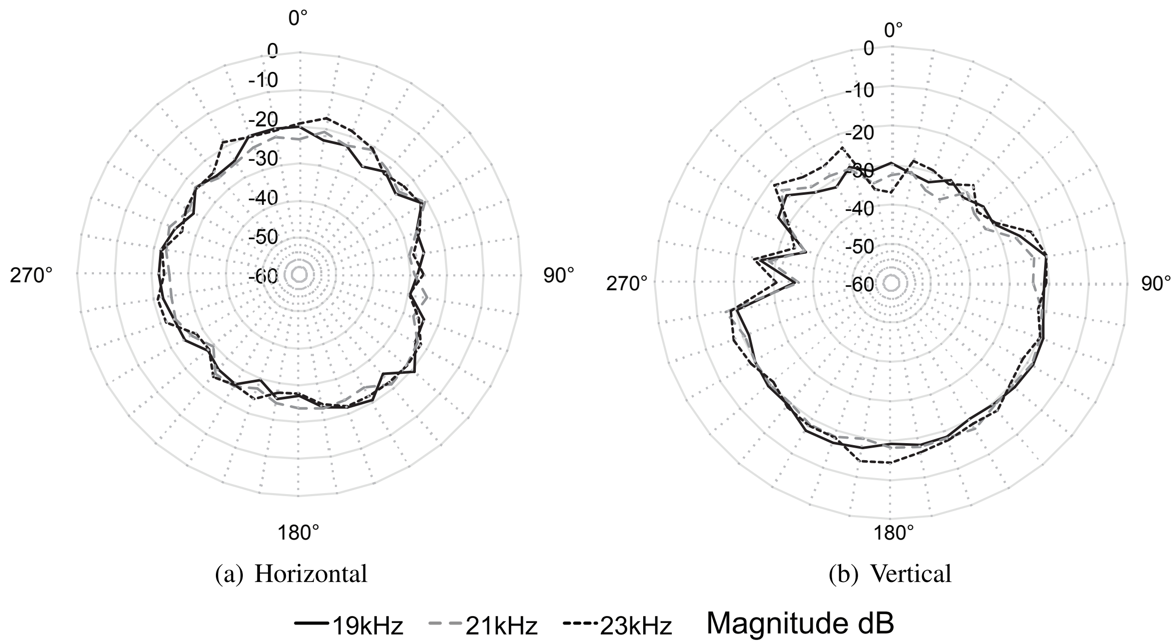


Figure 3.12: iPhone 3GS microphone beam pattern

3.7.4 iPhone Microphone

We measured the beam pattern of the microphone in an iPhone 3GS to validate that the ultrasound signals can be adequately received in most orientations. To do this we inverted our test setup from Section 3.7.1 by placing the phone on the pan tilt mount and replacing the measurement microphone with a speaker, from which we transmitted our test signals. Other smart phone models are expected to exhibit similar beam patterns since the MEMS microphones inside current smartphones are similar to that of the 3GS and are generally highly omnidirectional. Figure 3.12 shows the beam pattern of the iPhone 3GS, where the front of the microphone is pointing towards the $180°$ marker in the vertical pattern, and the screen of the phone is pointing towards the $180°$ marker in the horizontal pattern. While the horizontal pattern is reasonably omnidirectional at all frequencies in the $19 - 23kHz$ range, the vertical pattern shows significant attenuation from $70°$ to $280°$ due to the back side of the microphone being blocked by the phone's chassis. Although signals arriving at that angle are significantly attenuated, they can often still be detected. Many newer phone models contain an additional microphone on their top side, which could potentially

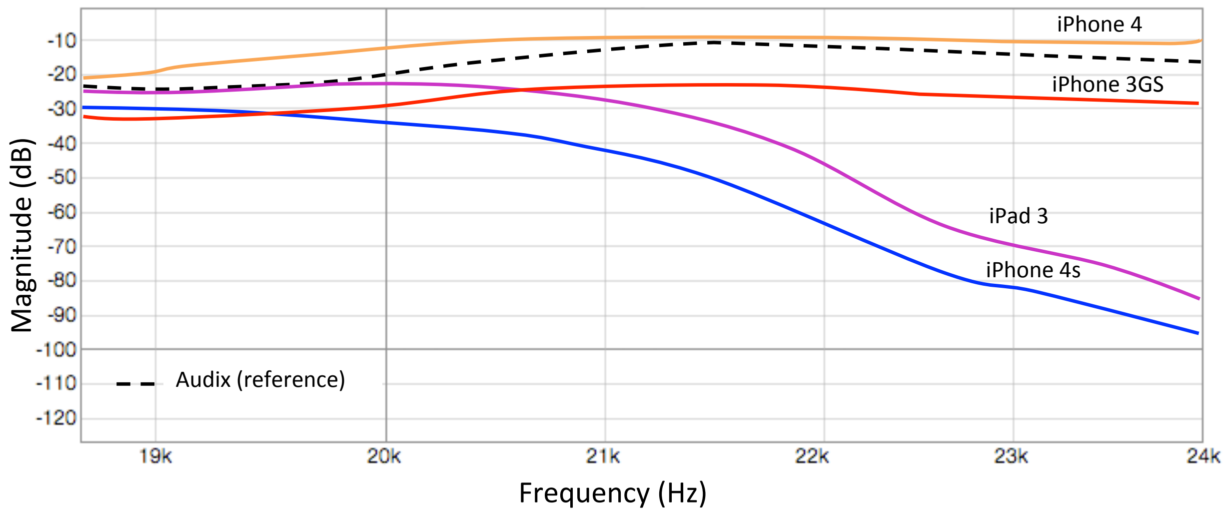


Figure 3.13: iPhone and iPad microphone frequency response

be used in combination with the main microphone to achieve greater omnidirectionality.

The frequency response of the microphone of past iPhone and iPad models compared to the Audix measurement microphone (dashed line) is shown in Figure 3.13. The non-straight shape of the measurement microphone reference line is due to the frequency response of our test speakers. Since the microphone is calibrated to have a nearly flat frequency response, any line that follows its shape should share a similar characteristic. It can be seen that both the iPhone 3GS and iPhone 4, which represent some of the best models in terms of frequency response, exhibit very flat frequency responses up until the Nyquist limit of their $48kHz$ audio codec. Most newer iPhone and iPad models, with the exception of current iPad Pros that exhibit responses similar to the old iPhones, have a frequency response which rapidly drops off past $21kHz$. This diminished bandwidth has led us to change our modulation scheme through the generations as detailed in Section 4.5. Several Android smartphones are evaluated in [69] and their frequency response ranges are shown to be similar to that of current iPhones.

3.8 Power Consumption

By far the largest cost of installing a beacon-based localization system such as ALPS is the installation cost, which usually includes running power wires to each beacon. In typical ALPS installations, an ultrasound transmission typically lasts for $220ms$ and the transmission power increases exponentially with range. Ultrasound transmission is by far the most power consuming process of the beacons since they spend most of their time in low-power modes. The low processing and RF communication overhead are negligible in comparison.

ALPS beacons are capable of running perpetually off of solar energy harvested from artificial and natural lighting in typical installations. We tested the power output of several solar cells when placed at a $5cm$ distance of a 100W equivalent CFL bulb rated at 1600 lumens, the results of which are shown in Table 3.1. The data was collected by connecting the solar cell to the input of a Generation 3b ALPS beacon to engage the Maximum Power Point (MPP) tracking feature of the energy harvesting controller. This modulates the input impedance of the energy harvester to lower the output voltage of the solar panel to 80% of its open circuit voltage, which results in maximum power conversion. Note that long rectangular panels performed slightly worse than shorter ones since the bulb is cylindrical and concentrates its light onto a point rather than more broadly like tube CFLs.

Panel Type	**Panel Size (cm)**	**Panel Area (cm^2)**	**Power at MPP (mW)**
Monocrystalline	7.0x5.5	38.5	31.43
Monocrystalline	10.0x8.0	80.0	47.97
Monocrystalline	13.7x8.1	111.0	55.81
Monocrystalline	18.0x8.0	144.0	50.10
Monocrystalline	11.6x16.0	185.6	92.85
Monocrystalline	13.8x16.0	220.8	91.63
Polycrystalline flexible	6.5x14.5	94.25	38.73
Polycrystalline flexible	16.5x7.8	128.7	48.60

Table 3.1: Power output of different types and sizes of solar cells at a $5cm$ distance from a 1600 lumen CFL bulb

The power consumption of a Generation 3b beacon over two $1.5s$ long TDMA cycles is shown in Figure 3.14. The power trace was collected using a Keysight N6705C DC power analyzer with a highly precise N6784A source measurement unit. Over the period of a communication cycle the beacon is responsible for receiving time synchronization messages using the CC2650 SOC and then periodically transmitting ultrasonic chirps and BLE advertisement packets (see Section 3.6). A cycle starts at (1) with the beacon waking up to receive an 802.15.4 time-sync packet and then immediately going to sleep at (2) to wait for its ultrasonic transmission slot. Initially, if the node has not yet been synchronized, it will perform low-power listening on the 802.15.4 channel until it receives a time-sync packet, after this it only needs to synchronize every few minutes. At (3) the beacon wakes up and turns on its audio codec, which requires about $40ms$ to transition to an active state. Next (4), the beacon turns on its piezo speaker amplifiers, which require $8ms$ to become active and also transmits a BLE advertisement packet for detection (BLE proximity) and transmission tiling as described in Section 4.5.2. The beacon then transmits an up-chirp and a down-chirp over speakers A and C at (5), switches its solid state relays to speakers B and D during a $2.6ms$ period of silence between transmissions (6) and then transmits another up-chirp and down-chirp. After this the beacon puts all of its systems into sleep mode and waits for the next transmission cycle (7). Note that it does not need to receive another 802.15.4 time-sync packet to transmit in the next cycle because it is still synchronized. At (8) the beacon wakes up to turn its audio codec on and steps (4-7) repeat through (9-12).

The average power consumption of the beacon over the $1.5s$ cycle is $64.6mW$ and achieves a $30m$ range radius. When factoring in a 90% efficiency of our energy harvesting controller that charges the batteries, the solar cell of size $185.6cm^2$ or larger (see Table 3.1) provides more than enough energy for perpetual operation. It should be noted that many applications do not require perpetual operation of the system. For example airport and retail installations would only be active during business hours and could harvest energy at night if the lights are left on.

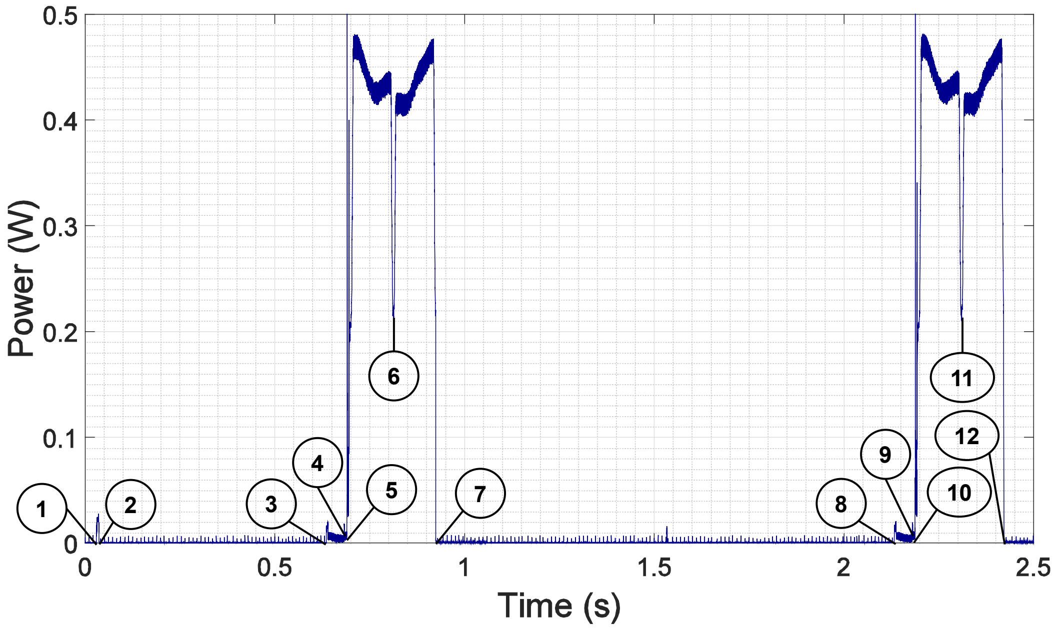


Figure 3.14: Power consumption of a Generation 3b ALPS beacon during two TDMA cycles

Chapter 4

Ultrasound Signal Modulation and Demodulation

4.1 Overview

ALPS uses ultrasound signals just above the human hearing range range (commonly cited as $20Hz$-$20kHz$ [85]), yet still within the sensitivity range of modern mobile devices for ranging and data transmission. There are several advantages for using ultrasound instead of RF signals for this, namely:

- They propagate relatively slowly, at about $340.29m/s$, so timing their propagation for using TOF, TDOA and RTOF ranging methods is easily possible to a high precision using commodity hardware.
- They do not penetrate walls and therefore the system does not suffer from the barrier problem, commonly experienced by RF based systems, where a device may be erroneously localized in an adjacent room since RF signals penetrate walls easily while ultrasound signals are absorbed.

- Off-the-shelf mobile devices can receive them using their microphone and digitize them with their audio codec.
- The received signals can be demodulated and processed entirely in software by the mobile device.

Despite these advantages, there are also many challenges in designing a localization system using ultrasound signals:

- There is very limited ultrasound bandwidth available on most mobile devices that commonly use $48kHz$ audio codecs (typically only $1.5kHz$), with which we need to achieve:
  - Precise ranging
  - Multiple access
  - Adequate latency by minimizing transmission length
  - Data transmission (for mapping received ultrasound signals to the beacons that transmitted them)
- Although theoretically inaudible by humans, the playback through loudspeakers may introduce audible artifacts
- Multipath signals lead to inaccurate range measurements and need to be detected

The sections below describe solutions to these challenges. Section 4.2 examines how we designed our ultrasound signals to exhibit pulse compression for increasing the SNR at the receiver and improving ranging resolution. Section 4.3.1 details a user study we performed to determine how to shape the ultrasound signals to be inaudible to humans. Section 4.4 describes the design and evaluation of a rate-adaptive Chirp Spread Spectrum ultrasound modulation scheme for ranging and data transmission, including how multiple access for simultaneous transmissions is handled. Section 4.5 describes and evaluates a time multiplexed Chirp Spread Spectrum modulation scheme, which is used in current ALPS systems and features shorter ultrasound packets, multipath signal detection, beacon tiling for large installations and sectored speaker array support for improved ultrasound signal coverage.

4.2 Pulse Compression

Pulse Compression is a technique used in RADAR systems (see Section 2.4.1) to increase range resolution through an increase of SNR at the receiver. When performing ranging using a standard sinusoidal pulse of constant frequency as a signal, the range resolution improves inversely proportional to the length of the pulse. After correlating the reflected signal with the original tone waveform, a signal with a broad base similar to that in Figure 4.1(a) can be seen. The magnitude as well as the breadth of this signal increases proportionally to the length of the pulse, therefore increasing the received signal magnitude, but decreasing its range resolution. Pulse Compression on the other hand employs chirp waveforms that linearly increase (or decrease) in frequency as ranging signals. Now when the received signal is correlated with the original chirp, the width of the intercorrelated signals is smaller than what you would see from a standard sinusoidal pulse. Figure 4.1 shows an example of a tone and chirp before and after filtering. The peak value after filtering is identical, but the chirp appears to be compressed (hence the name Pulse Compression). This compression makes the signal simpler to detect as it effectively increases its SNR, which leads to lower amounts of timing jitter, hence improving the range resolution. The gain in SNR and the improvement in range resolution is given by the compression ratio of the chirp, which is equal to its time bandwidth product.

We now briefly summarize some of the key theoretical properties of Pulse Compression. A linear frequency modulation is described by the following equation:

$$s(t) = sin(2\pi(f_c + \frac{k}{2}t)t) \tag{4.1}$$

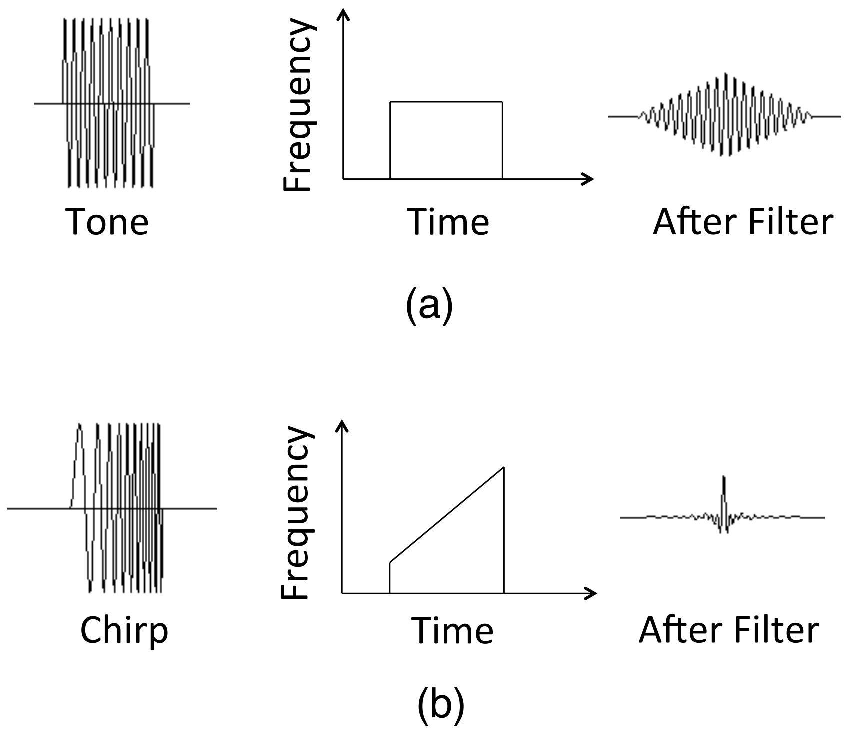


Figure 4.1: Pulse Compression illustration before and after matched filtering

For $0 \leq t \leq \tau$ where τ is the pulse duration, k is the rate of frequency change, f_c is the starting frequency and t is time. The bandwidth Δf can be computed as:

$$\Delta f = k\tau^2 \tag{4.2}$$

The range resolution ρ can be computed as:

$$\rho = \frac{c}{2\Delta f} \tag{4.3}$$

where c is the propagation speed of the medium (in this case sound which is about $340ms/s$). Given the $2kHz$ of bandwidth available on mobile devices, the best range resolution we can expect in practice is $8.5cm$.

We evaluated the impact of various modulation parameters on timing jitter, which provides insight into ranging accuracy. Wide jitter distributions would result in poor distance estimates. During these experiments, we transmitted equally spaced chirps and then measured the distance in time between adjacent chirps. The jitter value is simply the difference between the detected chirp spacing and the transmitted chirp spacing. Our experimental setup consisted of a generation 1 ALPS beacon (Section A.1), which consisted of a piezo speaker connected to a PC controlled audio interface via a home theater amplifier. The beacon transmitted ultrasound signals to an iPhone 3GS and an Audix TM1 measurement microphone, which were both mounted on a tripod at a $2m$ distance from the beacon's speaker. In Figure 4.2 we show plots for chirps recorded by the Audix and iPhone, both recording at a $48kHz$ sampling rate. The first histogram shows the performance of a fixed tone. We clearly see that without Pulse Compression the ranging resolution is quite poor (on the order of $2 - 4ms$, which corresponds to $6 - 14$ meters). Next, we use a $20ms$ tone and adjust the chirp bandwidth. We clearly see that additional bandwidth reduces the jitter. In the right-most histogram we then increase the length of the chirp from $20ms$ to $100ms$. By increasing the chirp length we also see the jitter slightly reduce. These graphs verify the properties of Pulse Compression on chirps and show that our fade-in and rate adaptation is not introducing a significant degradation in quality.

4.3 Reducing Audible Artifacts

One of the main challenges associated with near sonic modulation over standard audio speakers is avoiding humanly perceivable artifacts. Since speakers are mechanical systems, they cannot instantly transition between gain settings without creating clicking noises. To alleviate these problems, our chirp signals require slow amplitude fade-in and fade-out changes, slow frequency changes and all adjustments are only made during zero-crossing points in the signal. Figure 4.3 shows the overall layout and various parameters associated with a chirp symbol used in our mod-

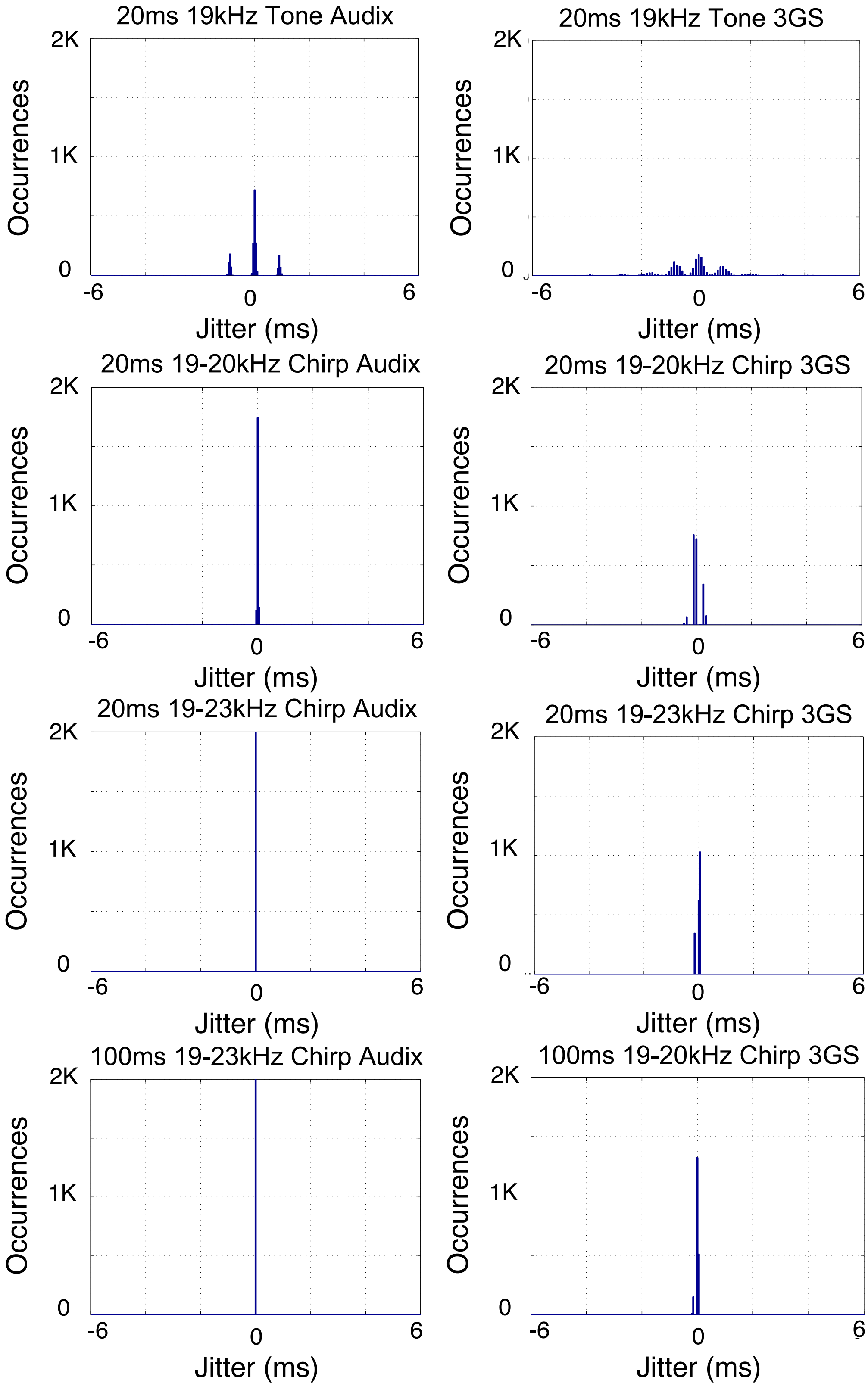


Figure 4.2: Jitter performance of tones and chirps of various lengths and bandwidths

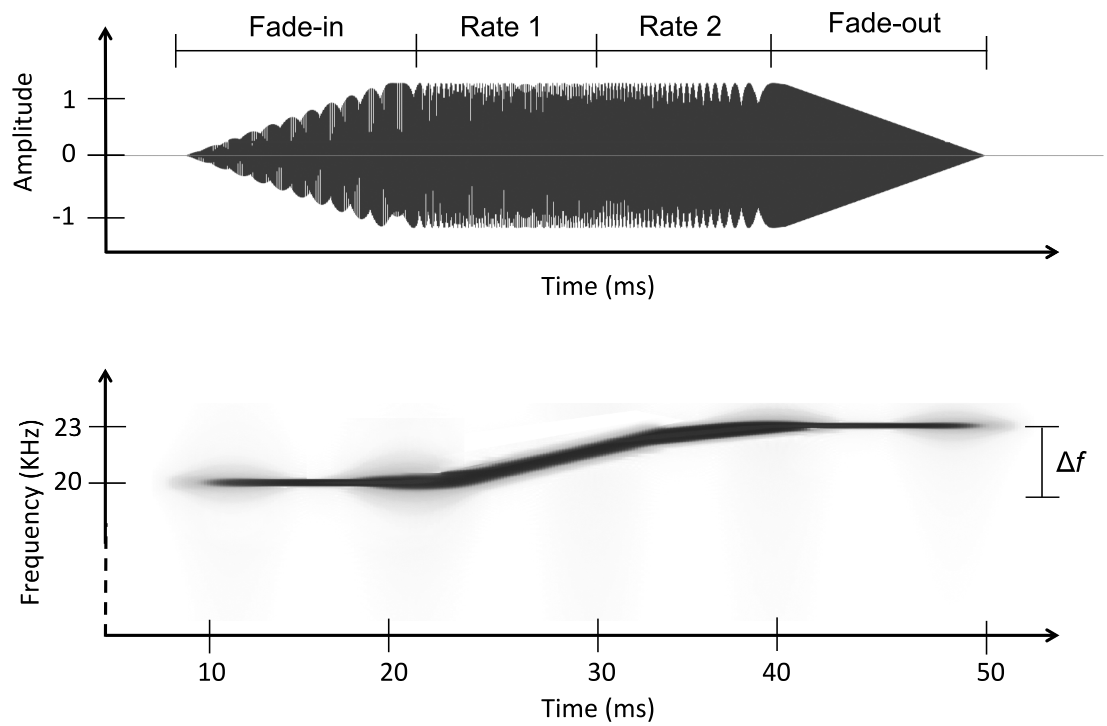


Figure 4.3: Chirp components for the modulation scheme with data transmission

ulation scheme with data transmission (Section 4.4). The chirp symbols used in our modulation scheme without data transmission (Section 4.5) only use a single rate, but are otherwise identical. The spectrogram in the lower portion of the figure shows that the fades occur at a constant frequency followed by the two chirp rates, and then a fade-out at the highest fixed frequency. The chirp waveform that is used for correlation does not include the fade-in and fade-out periods since they interfere with the Pulse Compression.

4.3.1 Audio User Study

In order to better understand the perceived effect of these attributes we conducted a user study where participants ranked the perceived loudness of different waveform configurations. The test was designed to evaluate the perceived audio artifacts associated with: (1) size of frequency jumps, (2) length of fade-in and fade-out, (3) linear versus exponential fades, (4) chirp duration, and (5) multiplexing of chirp rates. Participants were asked to watch a video that presented them with an intense click ($70dB(A)$) as a loud reference and a soft click ($10dB(A)$) as a low-level reference, followed by 35 different test sequences. Each test sequence was played three times per title slide in the video. All of the tests were randomized and some of the title slides contained no sound as a placebo value. Users would rank the intensity of the sound on a scale from 0 to 5 where 0 was considered silence and 5 was a loud sound. The experiments were conducted using high-end audio headphones connected to an external DAC at a fixed volume level. The histograms in Figure 4.4 summarize the results obtained from 35 users between the ages of 18 and 35. The dotted line at the bottom represents the average rating for the silent sequences. The error bars represent 1 standard deviation. Some of the tests included fixed tone with long fade-in sequences at different frequencies to act as a crude approximation of the user's frequency range. Our data showed a rapid fall-off around $19kHz$, which agrees with the standard hearing literature.

The first set of experiments was designed to determine if users could detect large frequency jumps. The test waveforms started off with a slow $20ms$ fade-in to a fixed frequency tone between 19 and $23kHz$. The tone would run for $20ms$ and then jump 1,2, or $3kHz$ to a higher fixed frequency followed by a slow fade-out. The slow fade lengths had earlier been selected since they were relatively unnoticeable. The idea was to simulate the types of frequency changes that would be apparent during Pseudo-random Noise (PN) DSSS modulation schemes. As can be seen in Figure 4.4 (a), even at $1kHz$ the artifacts were quite noticeable with an average level above 2. This indicates that just using PN modulation would in fact be quite noticeable as compared to the chirps. Note, this histogram is on a scale of 1 to 5 while the remaining histograms are on a scale from 0-1.

The next sequence of tests compared linear and exponential fading lengths. Each fade-in and fade-out was added to front and back of a $20ms$ chirp between $19kHz$ and $23kHz$. The fade

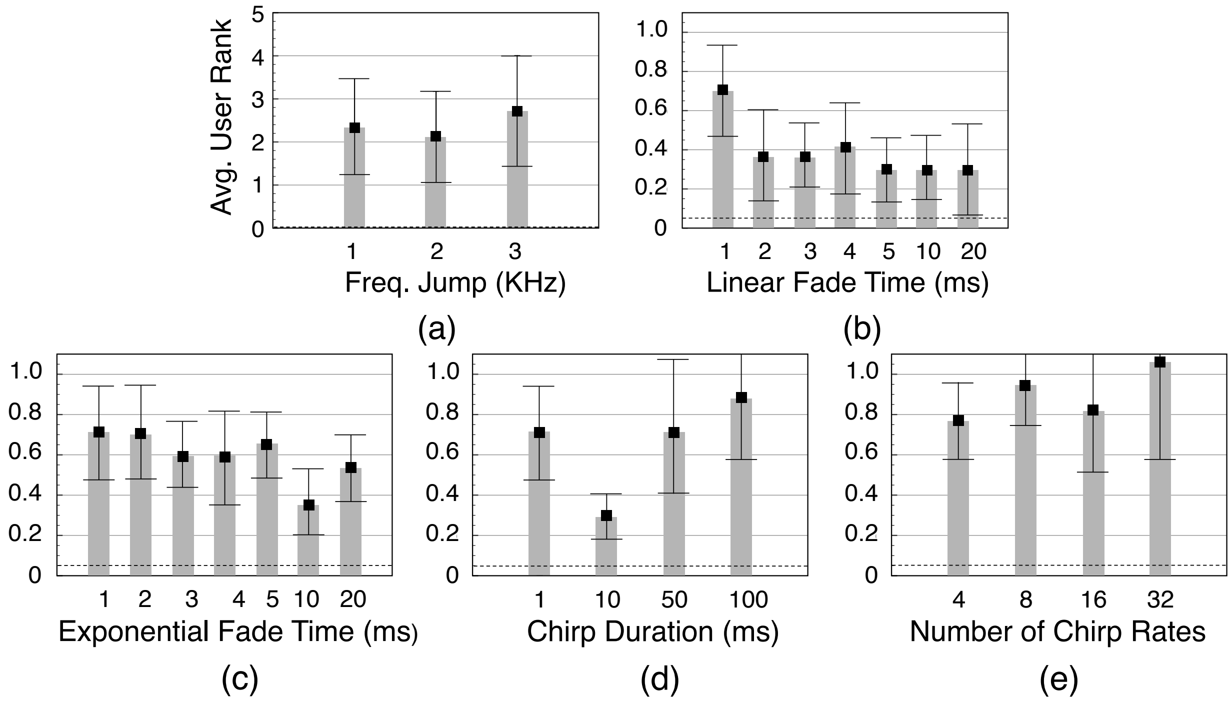


Figure 4.4: Audio perception user study results

periods were on fixed tones and hence do not remove any amplitude from the main chirp. To our surprise it appears that exponential fade approaches tend to be significantly more noticeable than linear fading. Linear fading tends to decrease and then flatten-off at around $5ms$. To minimize transmission time, we chose a final fade value of $5ms$.

Using a long fading length of greater than $20ms$, we then test if the duration of the chirp has any impact on its perceptibility. All of the chirps swept between $19kHz$ and $23kHz$ with a rate configured by the desired test duration. As shown in Figure 4.4 (d), users could easily perceive very short chirps since they are quite similar to frequency jumps. Interestingly, as the length of the chirp increased, users began to notice a ”swooshing” sound. For this reason, we sized our chirps to be at least $20ms$ and no longer than $200ms$. In practice, the chirp should be sized to the excess delay of the channel which is usually around $100ms$ at reasonable power output levels.

The final set of experiments evaluated the impact of applying rate adaptation to the chirps. In these tests, a worst-case chirp was generated for each chirp rate size where users were given the slowest rate followed by the fastest rate. As the second rate increases one would expect the user to perceive the rapid frequency changes. As can be seen in Figure 4.4 (e), there was a slight increase in perception due to increasing the number of possible rates (which leads to a higher rate for one of the chirp sections).

4.3.2 Human Health Concerns

Extensive studies have been conducted to quantify safe volumes and exposure limits of ultrasound on humans. The Health Protection Branch of Health Canada published a report that summarizes multiple studies related to ultrasound [44]. This report suggests that for frequencies above $20kHz$, the level should be kept below $110dB$ to prevent undesirable subjective effects of ultrasound. These effects include fullness in the ear, fatigue, headache and malaise. For reference, in the audio range, $110dB$ is approximately the loudness of a power saw from three feet away. Hearing damage can occur at above $95dB$. The study also indicates that subjects are more susceptible to fixed frequency tonal sounds. For this reason, we believe that chirp pulses can be considered safe under prolonged exposure when kept at a reasonable volume. To achieve the maximum range of $35m$ an ALPS beacon transmits at a power level of approximately $80dB$, which is well below the suggested limits. Furthermore there are several widely adopted products in pubic spaces that generate fairly high-powered near-ultrasound signals, such as ultrasonic motion detectors and emergency public announcement systems like [43]. The PA systems periodically (or often continuously) play back $20kHz$ pilot tones for self-testing system components and are imperceptible to humans.

4.3.3 Animal Exposure

Animals are known to have a significantly greater hearing range than humans [92]. At the extreme, mice, bats, whales and porpoises can hear frequencies as high as $90 - 150kHz$. However, one would be more concerned by the hearing range of household pets and service animals in real-world ALPS deployments. Dogs and cats can hear frequencies as high as $45kHz$ and $64kHz$ respectively. It is difficult to ascertain the full extent of hearing attenuation at higher frequencies or if the sounds have a negative effect on the animals. While not extensively tested, we did play sample tones in a home with two cats. Initially, it was unclear if the cats could hear the tones since they exhibited no noticeable response. We then played a sample tone before feeding each cat for a few consecutive days. It then became apparent that the cats could in fact hear the tone based on their reaction once a food association was established. We have also witnessed a guide dog moving about in a space where an ALPS installation was active, which did not seem to affect the dog's behaviour. Significant further testing would be required to draw any real conclusions, but while it appears that animals do hear the sound, it seems like limited exposure does not cause an (immediate) adverse reaction.

4.4 Modulation Scheme With Data Transmission

In order for a mobile device to localize itself using ALPS, we would like the ability to map received ultrasound transmissions to the corresponding transmitting beacons and with that, their locations. Early generations of ALPS without BLE time synchronization functionality (see Section 5.5), relied on modulating data onto the ultrasound carrier to encode beacon IDs, which can then be mapped to their locations to perform trilateration or multilateration. This section presents our modulation scheme for encoding these IDs into the ultrasound ranging signals and the demodulation process.

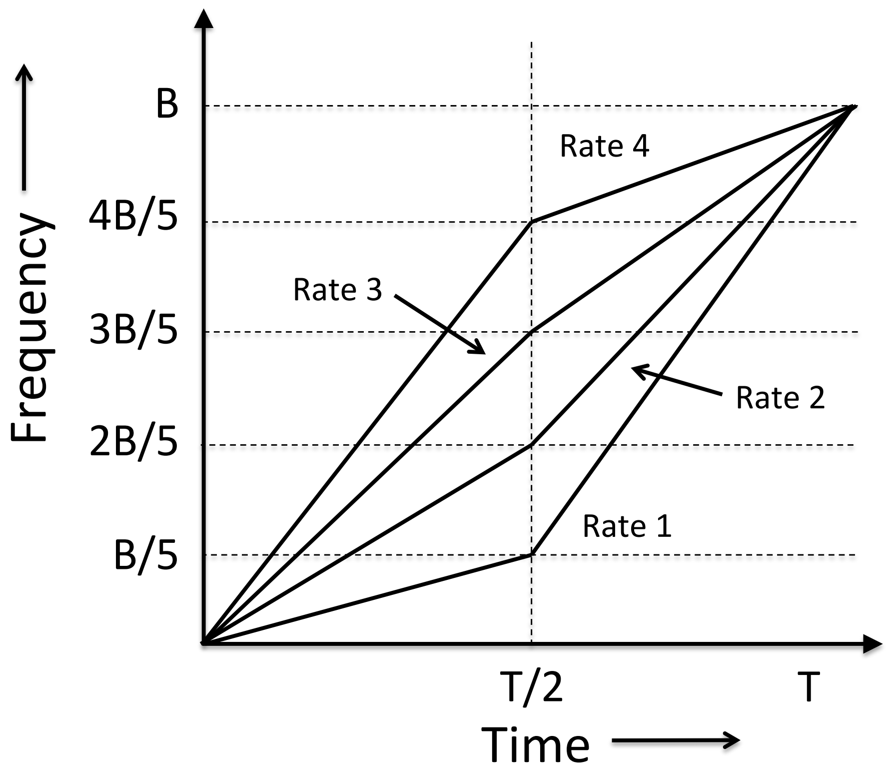


Figure 4.5: Example of four symbols encoded with Rate-Adaptive Chirp Spread Spectrum

4.4.1 Modulation

In Section 4.2 we presented the advantages of using linear chirp signals as ranging signals. In order to encode data using a chirp as a symbol, common techniques such as On Off Keying (OOK) could be applied. OOK would, however result in long packet lengths since only two symbols are used. [94] introduces the use of chirp-rates as a mechanism to create multiple chirp symbols. This approach decomposes each chirp into two interconnected chirps with different frequency rates that change at the half-way point of the symbol. Figure 4.5 illustrates a scheme that supports four unique symbols across a shared bandwidth. Each rate represents a different signal waveform that is correlated with the received signal to extract the embedded sequences of data. In this scheme we provide each beacon with a unique ID, which is encoded as a series of up-chirps, each representing two bits. It is worth noting that [94] was based entirely on simulation.

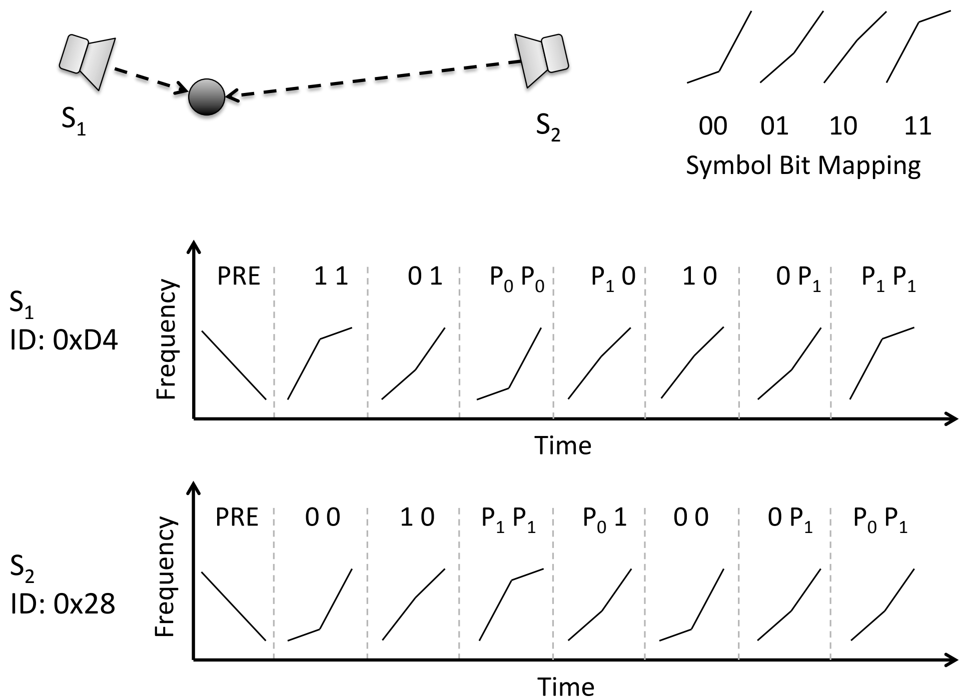


Figure 4.6: Beacon ID packet structure using Rate-Adaptive Chirp Spread Spectrum

We validate that such rate adaptation works in practice in Section 4.4.4.

Figure 4.6 shows a diagram where two beacons are using our chirp modulation scheme along with chirp rate adaptation. Each beacon ID is encoded as a sequence of two (7, 4) Hamming codes, allowing us to transmit 256 unique IDs by using seven two-bit symbols. The error coding allows us to correct up to two single-bit errors and detect all single-bit, as well as two-bit errors. Furthermore, as a mobile device moves through a space, a map can be used to identify which beacons are likely to be in range, allowing out-of-range IDs that were erroneously decoded to be discarded. Each data symbol is represented as a rate adapted up-chirp, and is prefixed by a preamble encoded as a constant-rate down-chirp. The preambles are used to mark the beginnings of data sequences and to measure high resolution TOA information from. The modulation scheme can be easily adapted to larger installations with more than 256 beacons by employing a (15, 11) Hamming code and/or tiling beacons.

4.4.2 Multiple Access

Communication systems with multiple transmitters like ALPS usually require a way to share a common transmission medium to prevent packet collisions, known as Multiple Access Control (MAC) protocols/methods. There are multiple ways to facilitate multiple-access transmissions using ultrasonic chirps including TDMA, Frequency Division Multiplexing (FDMA), CDMA and CSS. While TDMA suffers from scalability, configuration and latency issues since all ultrasound packets need to be successively scheduled in a collision-free manner, it completely eliminates interference between transmissions, imposes no additional bandwidth requirements like FDMA and CDMA and does not require power control for optimal operation like CDMA. Larger deployments may be segmented into multiple zones with separate TDMA cycles as described in Section 4.5.2 to keep cycle lengths to a minimum. Using frequency diversity isn't ideal since a chirp's timing resolution is directly related to the frequency bandwidth that it operates on. Ideally, you would like each chirp to cover the maximum bandwidth to achieve the highest ranging resolution. Chirp Spread Spectrum typically uses chipping codes composed of up-chirps and down-chirps to represent patterns of 1's and 0's. While promising, this approach can require long transmission times depending on the number of bits used in each code. Code division multiplexing commonly requires significant bandwidth (or long codes) as well as power control so that overlapping transmissions do not overpower each other, all of which are not feasible for ALPS.

In generation 1 ALPS employed a rate adaptive CSS technique as discussed in Section 4.4.1 with concurrent transmissions from beacons. As shown in Section 4.4.4, this scheme worked well under controlled conditions, however, we quickly realized that like with CDMA schemes, it was also dependant upon the ultrasound signals being received at similar power levels so that they don't overpower each other. This drawback was further exacerbated by iPhone models after the iPhone 4 having a drastically decreased audio frequency response (Figure 3.13), which cut our usable bandwidth down from $4kHz$ to $1.5kHz$. In early platform generations without BLE functionality, ALPS employed a TDMA scheme with data transmission as described in this section. More recent generations that featured BLE time synchronization used the scheme described in Section 4.5.

4.4.3 Demodulation

The demodulation of the received signal is performed completely in software, in part by a process known as matched filtering. In matched filtering the incoming signal is convolved with a conjugated, time-reversed version of a signature signal that is expected to be contained within the received signal. This results in a distribution showing the similarity of both signals as they are slid across each other. Peaks of high magnitude denote a high cross correlation between both signals, therefore making it likely that an instance of the signature signal is located at the same location as the peak in the received signal. Therefore, by applying a matched filter for each rate adjusted chirp and the preamble, we are able to determine the starting locations of the signals as well as the time difference between them.

There are multiple ways of computing the matched filtering output. Running a matched filter in the time domain requires $O(n \cdot m)$ operations, however when performed in the frequency domain, can be processed in $O(n \cdot log \cdot n)$ time. In the frequency domain, the FFT of the modulated signal is multiplied with a frequency domain representation of the signature signal and is then converted back into the time domain.

Since we are using sequences of multiple chirp rates as symbols to uniquely identify beacons, the symbols need to be as orthogonal as possible (i.e. have low cross correlation properties between each other) in order to be differentiable after matched filtering. Rate adjusted chirps generally fulfill this requirement if the number of chirps within a set is kept to a reasonable number (in our case 4 - see Figure 4.5), but in practice we have found that the superposition of chirps, of different rates that are staggered in time can become very difficult to detect, which is often the case when multiple beacons are transmitting simultaneously in a generation 1 system. For this reason we prefix each data string with clearly identifiable preambles represented by down-chirps, which are highly orthogonal with respect to up-chirps. The preambles mark the beginning of the data sequences, which allows us to bound the region of signal that we perform matched-filtering on, and act as markers for receivers to synchronize to a beacon's broadcasts. Since each beacon broadcasts identical sequences of data periodically, the ID of incoming data sequences can be predicted based on their arrival time with respect to a previous sequence. Therefore once a receiver is synchronized to a particular beacon, TDOA pseudo-ranging can be performed on each detected symbol, before the entire corresponding data sequence is decoded. This allows for significantly higher ranging update rates (but is not required). The time required to send an 8-bit ID is $240ms$, but new ranging information can be obtained at up to every $30ms$ if successive broadcasts are transmitted in a gap-less fashion. The preambles also provide us with an estimate of the amplitude of the following data sequence. Since we know the location of the data symbols with respect to the preamble, we can now filter overlapping symbols according to their magnitude and position in time. In combination with the forward error correction, and discarding erroneously decoded transmitters that are likely to be out of range, we can achieve high packet reception rates.

Another technique that can be applied to help separate overlapping data sequences is Successive Interference Cancellation as described in [101]. Here the modulated signals of successfully decoded data sequences are reconstructed and then subtracted from the received signal in order of descending amplitude before any further decoding is performed. Furthermore the incorporation of a Rake Receiver as described in [86], or an Adaptive Matched Filter could improve robustness against multi-path interference.

Generation 2a and 2b systems employ a TDMA multiplexing scheme where the beacons transmit successively rather than simultaneously. This requires the demodulator to process a significantly longer recording, but makes it easier to detect the encoded beacons IDs and TOA values accurately. Techniques such as Successive Interference Cancellation (SIC) would not bring any benefit to this scheme.

4.4.4 Evaluation

In this section we evaluate the performance of the modulation scheme with data transmission described above.

Our experimental setup included a microphone stand, audio DAC/ADC and piezo electric tweeters. The microphone stand shown in Figure 4.7 consisted of an Audix TM1 omnidirectional measurement microphone and a smart-phone holder. In Figure 4.7 we see the Audix microphone on the left and an iPhone in the holder on the right. The Audix microphone was chosen due to its extremely flat frequency response all the way up to $25kHz$. We transmitted all ultrasound from generation 1 ALPS beacons as described in Section A.1.1, which consist of bullet piezo speakers connected to home theater amplifiers and a MOTU Ultralite-mk3 audio interface.

For each test (unless specified otherwise), audio was transmitted from the speakers and recorded by both the Audix microphone using the mk3 ADC and by an iPhone 3GS. The iPhone used a wireless file-sharing program to push the recorded sound clips back to our main computer for processing. Test sequences could be remotely started and stopped using a VNC client on the iPhone. Streamlining this process enabled us to evaluate an extensive set of parameters. The following graphs were generated from over 25 hours of combined recording samples.

Impact of fading functions on BER

First, we determine the impact of fading the signal in and out on the Bit Error Rate (BER) of data transmissions. In order to compute the BER, we transmit a modulated sequence of 1024

Figure 4.7: Experimental receiver setup for evaluating ALPS modulation scheme with data transmission

random bits using a $20ms$ chirp (sweeping from $19kHz$ to $23kHz$) at different transmit powers while varying the length of the fade in and out durations. Each point in the plot represents 20 seconds of samples. The signal-to-noise ratio was computed based on the average intensity of the chirp signal as compared to the average intensity of the noise floor when there is no transmission. In both cases, the signal was high-pass filtered to remove audible noise. In the following tests, the microphone was mounted approximately $2m$ from the speaker. As the transmit power is decreased, the SNR correspondingly decreases. At each bit interval, the receiver must decide if it correctly detects a 1 or 0 bit by correlating an up-chirp or down-chirp at the correct rate across the signal. A BER value of 0.5 corresponds to the expected value if the bits are decided by random chance (the signal is unreadable). In this experiment, all measurements were taken using the Audix microphone so as to determine the general trend. Figure 4.8 shows that the fade lengths have almost no impact on the BER. This makes sense since the correlated input signal does not include the fade-in and fade-out regions.

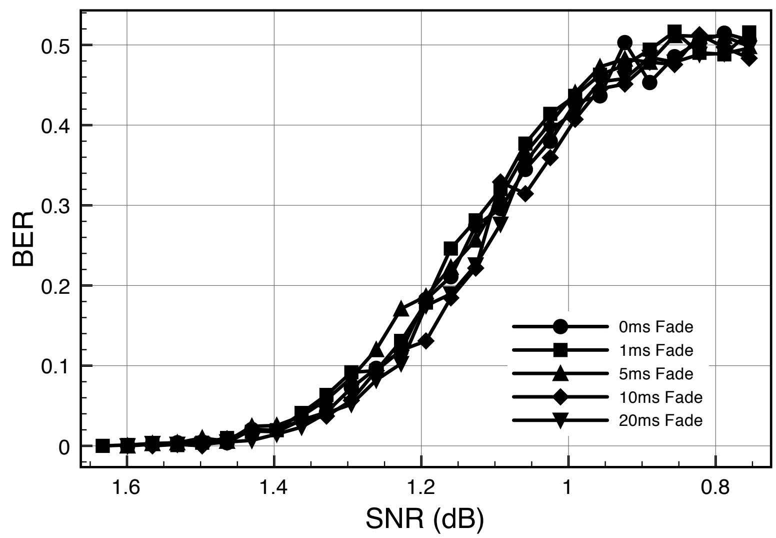


Figure 4.8: Impact of fade in/out on BER

Impact of chirp length on BER

Next, we evaluate how chirp length impacts BER. This test is similar to the one performed in the previous example, except now the fade period per symbol was set constant at $10ms$ ($5ms$ fade in and out) while the chirp length was adjusted. We see that as the chirp length increases, the BER falls off at lower and lower SNR levels. This corresponds to the Pulse Compression equations that indicate that with longer chirps the signal should be distinguishable at lower SNR levels. Based on this performance graph as well as the user study, we select chirp lengths of $20ms$ for use in practice. In general, chirps should be greater than the excess delay of the channel in order to maximize performance under multi-path conditions. The excess delay can be determined by looking at the ultrasonic impulse response of a particular space.

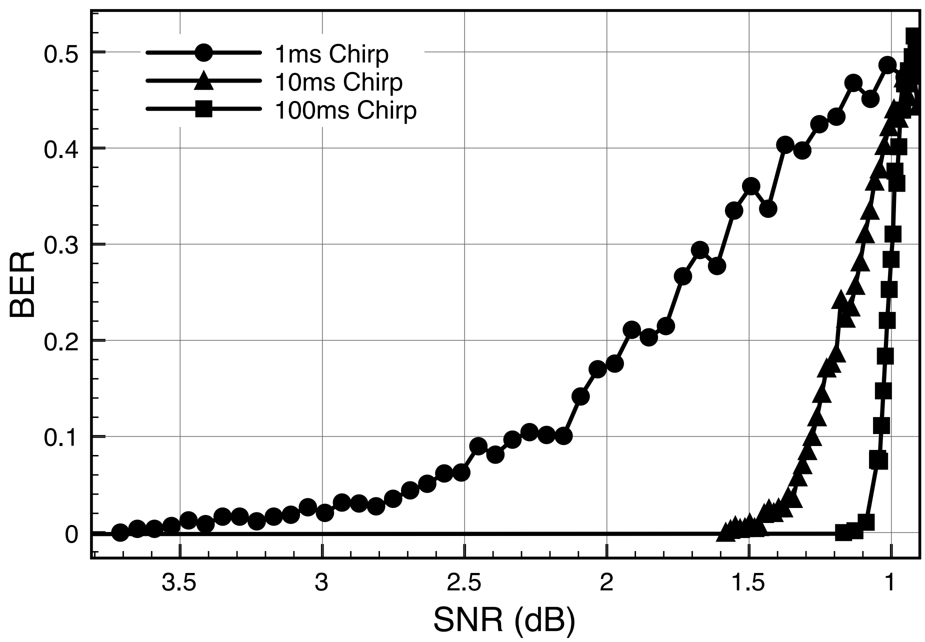


Figure 4.9: Impact of chirp length on BER

Impact of concurrent transmitters on BER

We now evaluate how well the system scales with multiple concurrent transmitters by incrementing the number of rates used by the chirp. This test is only relevant to a generation 1 system as successive generations use a TDMA scheme to multiplex transmissions. For each additional rate, we mix in a signal for all other rate values given a random offset around the signal that we are trying to decode. This corresponds to all other possible transmitters sending data simultaneously. For example, four concurrent transmitters would mean that the chirp can be modulated with four different rates and the three other possible rates are being mixed into the transmitted signal to act as simultaneous transmissions. Figure 4.10 shows the performance of the Audix reference microphone as well as an unequalized iPhone 3GS. First, we see that the Audix and iPhone perform comparably. We also see that the BER remains below 10% up to about 10 concurrent transmitters. In practice, most receivers will not overhear transmissions from all possible chirp rates within close proximity of each other, so this performance is quite pessimistic. Such a situ-

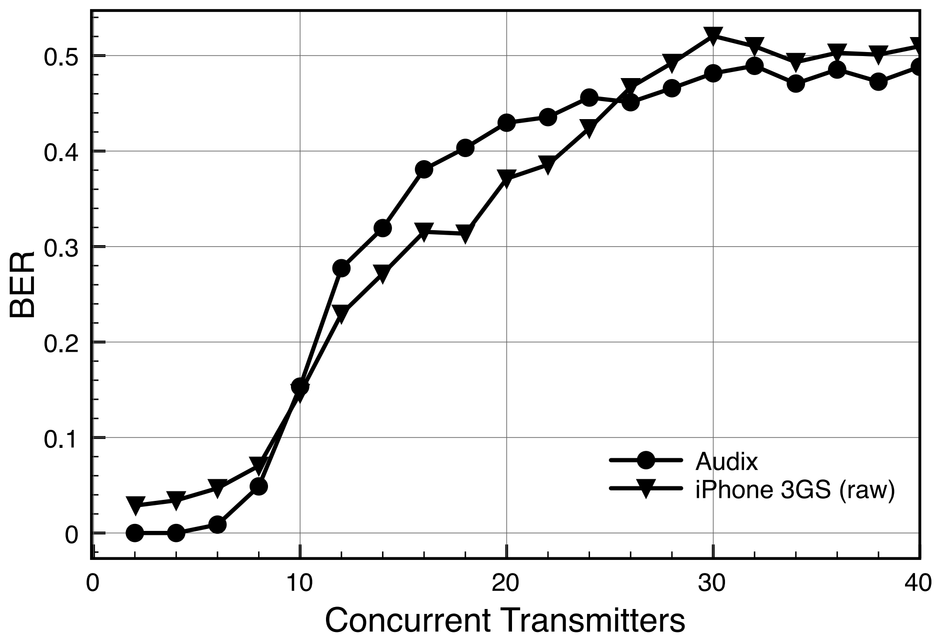


Figure 4.10: Impact of concurrent symbols on BER

ation would only occur if many transmitters were placed in the same location. Above 16 rates, the system's BER begins to severely deteriorate. In order to support additional rates, we would require more bandwidth which is unfortunately limited by the microphone sampling rate. If in the future mobile devices could support higher sampling rates we would be able to support more concurrent transmitters and we would be able to achieve more precise ranging.

Figure 4.11 shows a spectrogram recorded on an iPhone of a 14 bit sequence from a single transmitter. Figure 4.12 shows the spectrogram when many transmitters are broadcasting simultaneously. The spaces between the sequences are still visible, but the other symbols are no longer distinguishable unless the signal is demodulated.

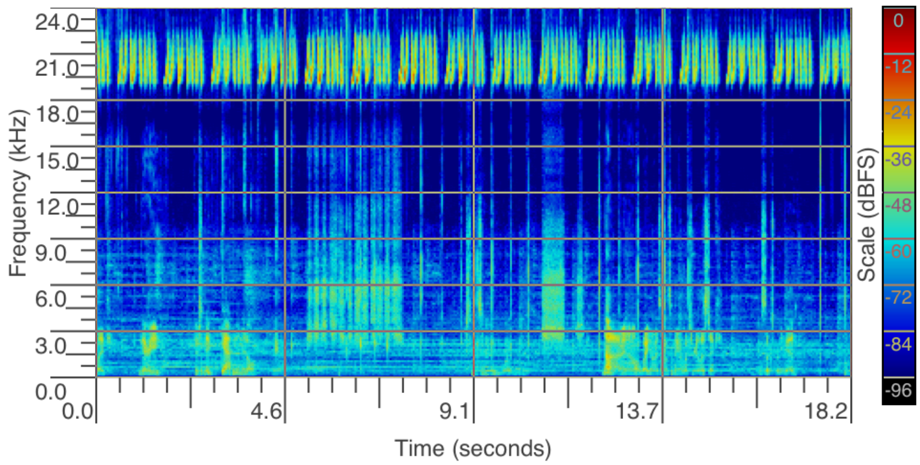


Figure 4.11: Spectrogram with a single beacon transmitting

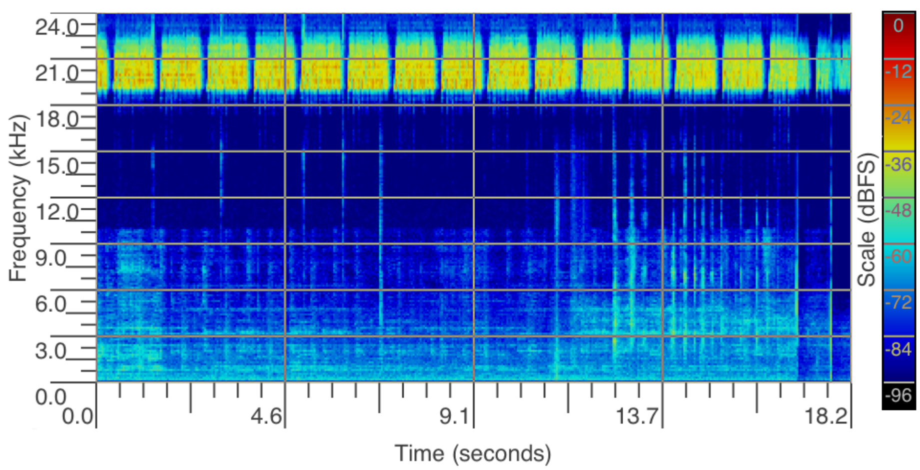


Figure 4.12: Spectrogram with four beacons transmitting

Transmission Range

In order to estimate the transmission range of the system, we measure SNR versus distance as shown in Figure 4.13. For this test, the signal intensity at 1 meter away was measured at a modest $48dB(A)$, which corresponds to a volume level of about 5% of the maximum volume possible on the Onkyo amplifier and equivalent to a sound slightly louder than the humming of a refrigerator. At higher volume levels, we see transmissions as far as $50 - 70$ meters (using generation 1 beacons which are powered by multichannel amplifiers, the embedded beacons of succeeding generations cannot achieve these ranges). Depending on the deployment scenario, the

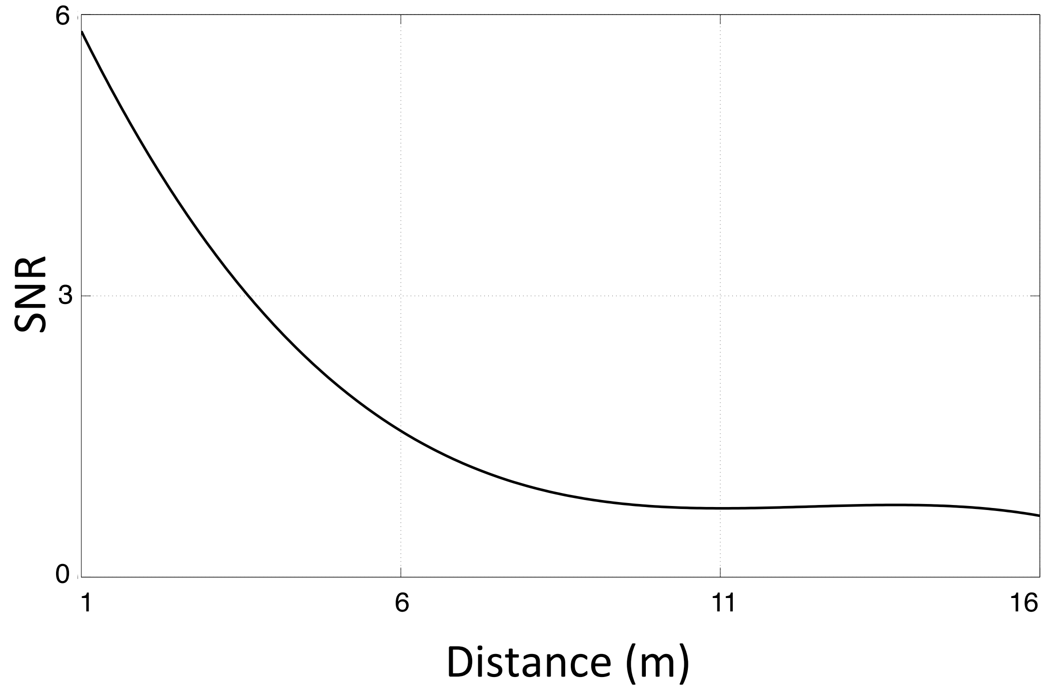


Figure 4.13: Distance vs SNR using a Generation 1 ALPS beacon transmitting at $48dB(A)@1m$

transmit volume can be adjusted one way or another to aid in maximizing coverage and number of concurrent transmitters.

4.5 Modulation Scheme without Data Transmission

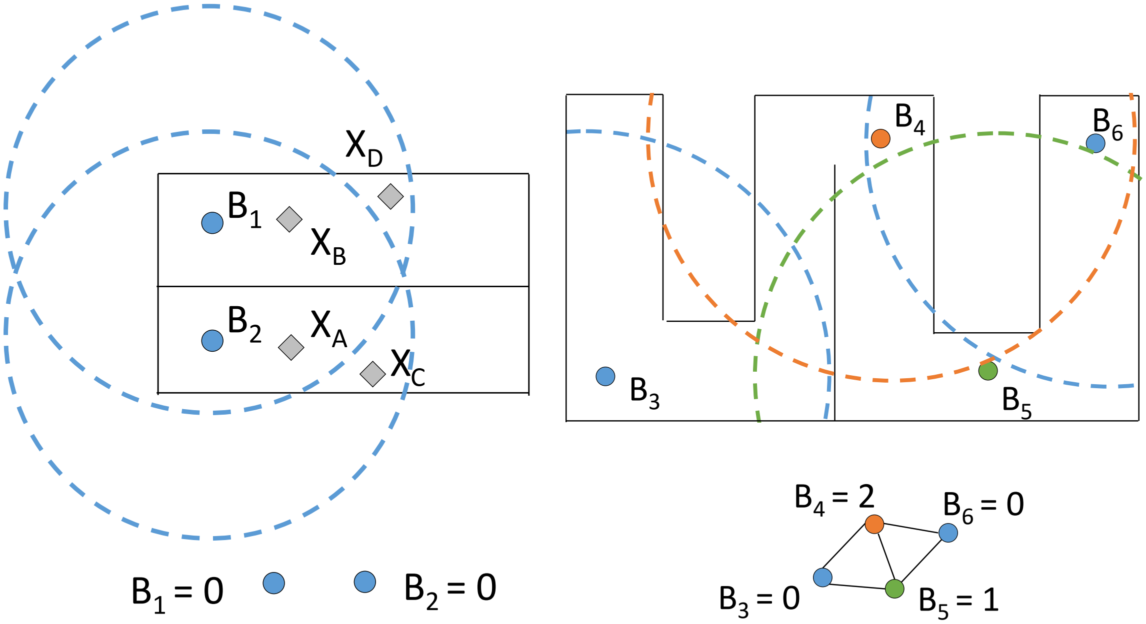
While the modulation scheme with data transmission described in Section 4.4 provided a good way to map received ultrasound transmissions to the beacons which transmitted them, the move to a TDMA based multiplexing scheme prompted the need to reduce the ultrasound packet length from $240ms$ to $60ms$ (including fade in and fade out). This section describes the modulation scheme employed in ALPS generations 2c-3a, which does not transmit beacon IDs over ultrasound, but relies on BLE time synchronization (see Section 5.5) to map received ultrasound signals to the corresponding beacons.

4.5.1 Modulation

In this scheme the only transmitted symbol is a $50ms$ long upchirp from $20.0kHz$-$21.5kHz$, which is prefixed by a $5ms$ fade in function and succeeded by a $5ms$ fade out function. Each beacon simply transmits a single upchirp with these parameters at the beginning of its TDMA slot. The length of $50ms$ was chosen due to the 50% reduction of audio bandwidth available to iPhone models succeeding the iPhone 4 (see Figure 3.13). Since the pulse compression gain of a chirp signal (Section 4.2) is equal to its time-bandwidth product, the bandwidth reduction results in a 50% reduction of SNR at the receiver, which is compensated by the longer chirp length. A $50ms$ chirp length also still performs well in terms of being inaudible to humans as shown in Section 4.3.1.

4.5.2 Multiple Access and Tiling

Multiple access in this modulation scheme is handled by time multiplexing. Each beacon is assigned a time-slot during which it transmits its signal. Since the mobile devices are time synchronized to the TDMA cycle, we can easily map the transmitting beacon to the received ultrasound signal. Unfortunately, giving each beacon a unique time-slot would lead to an extremely long time cycle length which would negatively impact the system's update rate. Since each BLE beacon provides a unique ID, we can increase concurrency by reusing ultrasonic time slots that we know are out of range of each other. The logical approach would be to schedule the ultrasonic transmissions using graph coloring such that acoustic regions never overlap. Unfortunately, as can be seen in Figure 4.14(a) , since acoustic signals alone do not uniquely identify nodes, this can lead to conflicts. In the figure, B_1 and B_2 are in separate rooms so that all areas have unique ultrasonic coverage. The BLE transmission range is shown by the dashed circles. In this example, to uniquely identify a location if B_1 and B_2 shared the same channel we would rely on the BLE to distinguish locations. This would only be possible at points X_C and X_D and not in much



(a) Ultrasonic Schedule (b) Max of RF and Ultrasonic Schedule

Figure 4.14: TDMA coloring schemes for acoustic tiling

of the space confining X_A and X_B. Instead, we need to create edges between beacons if the coverage of beacon overlaps in both acoustic and RF. This would include picking the maximum coverage between BLE and ultrasonic. In our system, BLE range is almost always larger than ultrasonic and it passes through walls. In practice, we can conservatively apply a unit-disc model at the cost of potentially requiring slightly more colors than the minimal requirement. Graph coloring is known to be NP-complete, but we approximate the coloring using a breadth first search greedy approach. Figure 4.14(b) shows an example coloring scheme for four nodes using BLE range.

4.5.3 Demodulation

Demodulation in this modulation scheme is similar to that of Section 4.4. First the approximate beginning of the TDMA cycle in the recording is found by using the BLE time synchronization method described in Section 5.5. A matched filter is used to filter each time slot in the audio recording containing the received ultrasound signals. Next the envelope of the matched filter output is calculated to simplify peak detection. A peak detector with a variable power level threshold is employed to detect the peak values of the envelope, which denote the TOA values of the ultrasound transmissions. The thresholder uses a combination of a threshold calculated to be slightly above the noise floor of the recording, and a variable threshold, which is set to be a fraction of the maximum correlation value found in that time slot. The higher of the two thresholds is set for the peak detector on a slot by slot basis. This setup will allow the detector to find peaks that are below the peak correlation value and account for the often vastly different power levels of each recorded signal. Since the peak correlation does not always correspond to the correct TOA, but may be a multipath copy of the transmission, the first peak within a time slot (rather than the highest peak) satisfying the threshold requirements is selected as the corresponding TOA value. Finally the beacon IDs corresponding to each TOA value are calculated based on the beginning of the transmission cycle and time slot boundaries.

4.5.4 Non-Line of Sight Signal Multipath Detection

A major source of error in TOF ranging systems are incorrect measurements due to multipath Non-Line of Sight (NLOS) signals. Failing to identify the NLOS signals can introduce estimation errors in ranging and therefore seriously affect the localization performance. The identification of LOS/NLOS signals not only facilitates the process of selecting the right measurements, but also helps to further mitigate the ranging bias. Most of the identification techniques deal with the problem based on the range estimates or Channel Pulse Response (CPR), but are often unfeasible in the real world since a large amount of training data is required for characterization. The Cricket system [87] (Section 2.5.2) was one of the first efforts that noticed that the difference between two transmission media could be used to possibly infer NLOS transmissions. In Cricket, the frequency of the ultrasound signals was quite high at $40kHz$, making the transmissions highly directional, which made the correlation between RSSI distance and ultrasonic TOF more obvious. At lower frequencies, with chirp encoding and omni-directional transmitters, the ultrasound diffracts significantly more, making the distinction between LOS and NLOS more difficult.

We developed a machine learning based approach that uses a binary classifier for NLOS detection that is able to learn the characteristics of a space with relatively little training data. This method can supplement the NLOS signal detection that is done by the demodulator described in Section 4.5.3. During the development of the machine learning based approach we collected 3600 samples of LOS data and 1200 samples of NLOS data from arbitrary locations in more then 6 environments. The unbalanced amount of LOS data and NLOS data are designed to model the real world scenario where LOS data is much easier to collect during the installation process. Since the rate of position updates is relatively low, we ideally want to find a set of features that can be extracted from a single measurement. The key insight to our approach is that we are able to detect ultrasonic TOF, ultrasonic RSSI and BLE RSSI, which are different in LOS and NLOS cases.

In Table 4.1 we show classification accuracy with different combinations of features, where F_{us} is the ratio of $RSSI_{us}$ to D_{iB}, F_{iB} is the ratio of $RSSI_{iB}$ to D_{iB}, F_{wav} is the normalized waveform of the received ultrasonic signal, and F_{delay} is the root mean square (RMS) delay spread of the ultrasonic signal. D_{iB} is the distance estimate returned by BLE, $RSSI_{us}$ and $RSSI_{iB}$ are RSSI values from ultrasonic and BLE respectively.

Based on the results in Table 4.1, we selected F_{us} and F_{iB} because they perform best with the least amount of training data. A Support Vector Machine (SVM) classifier is trained with 10-fold cross validation and grid search on selecting the best parameters in order to prevent over-fitting. Other features like the shape of the ultrasonic waveform performed poorly in our experiments.

Feature Set	Accuracy
$\{F_{us}\}$	0.644
$\{F_{iB}\}$	0.925
$\{F_{wave}\}$	0.767
$\{F_{delay}\}$	0.753
$\{F_{iB}, F_{wave}\}$	0.779
$\{F_{us}, F_{iB}\}$	0.965
$\{F_{delay}, F_{wave}\}$	0.787
$\{F_{us}, F_{iB}, F_{delay}\}$	0.959
$\{F_{us}, F_{iB}, F_{delay}, F_{wave}\}$	0.779

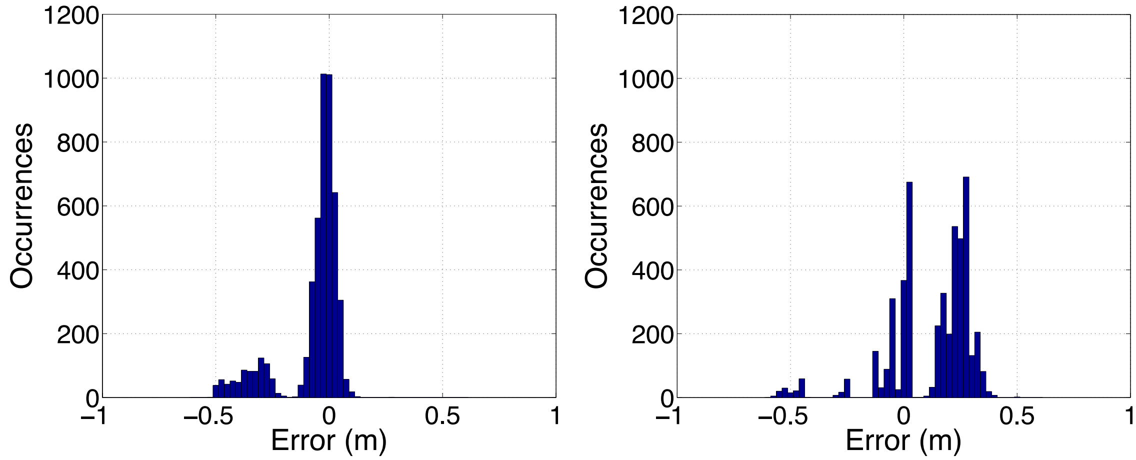
Table 4.1: Identification accuracy of NLOS signals with machine learning approach using multiple features

4.5.5 Evaluation

In this section we evaluate the performance of the modulation scheme without data transmission described above. The fundamental evaluation of chirp signals as described in Section 4.4.4, are also applicable to this modulation scheme since the time-bandwidth product of the $50ms$ $20kHz$-$21.5kHz$ chirps is roughly the same as the $20ms$ $19kHz$-$23kHz$ chirps used before (see Section 4.4).

Ranging and Multipath Performance

In order to evaluate the ranging performance of this modulation scheme and better understand the impact of the environment the system is operating in, we evaluated the ultrasound ranging performance of our generation 2c beacons in two different spaces. Figure 4.15 shows the ranging error in a free space with minimal multipath propagation and in a confined corridor setting, which exhibits high multipath propagation using only the demodulator to filter out NLOS multipath signals. The data was collected by time synchronizing an iPhone 5S to the beacon by holding it directly at the speaker while it was playing evenly spaced $50ms$ chirp signals and then placing it at a known distance away from the beacon. The beacon would then transmit 500 additional periodic chirp signals per sampled distance after a known time delay, for which we calculated



(a) Free space ultrasound ranging error

(b) Corridor ultrasound ranging error

Figure 4.15: Ultrasound ranging error in free space and corridor environments

the measured distance based on the propagation time of the signal. We collected samples at 10 different beacon to receiver distances in both environments. For the free space case using ultrasound a mean absolute ranging error of $8.9cm$ with 95% of the distance samples below $33.5cm$ in error was observed. For the corridor case a mean absolute ranging error of $17.9cm$ with 95% of the distance samples below $34.2cm$ in error was observed.

In Table 4.2 we summarize the identification performance of our machine learning based NLOS detector from data we collected in six environments with different multipath propagation characteristics (kitchen, lab and 4 different office environments). We collected 3600 samples of LOS data and 1200 samples of NLOS data from arbitrary locations in the six environments, of which we used 10% of the LOS data for training and varying the amount of NLOS data for evaluating the detector. We see that even with 1% of the NLOS data used for training, we are able to achieve an 80% classification accuracy. In any one mapping collection cycle, this corresponds to about 300 LOS samples (which are easily captured while holding the phone in the open during the mapping phase) and 12 NLOS samples which the user is instructed to collect. However, we should note that most of classification error results from false negative (FN) instead of false positive (FP) due to the unbalanced data set, which can seriously decrease the performance of

NLOS	Accuracy	FP	FN	Prec.	Recall
1%	0.805	0	0.195	1.00	0.805
4%	0.826	0	0.175	1.00	0.826
7%	0.837	0.007	0.156	0.992	0.843
10%	0.841	0.016	0.143	0.982	0.855

Table 4.2: Impact of training samples on F_{iB} and F_{us} performance

our localization algorithm. With an increased number of NLOS data samples in the training phase, we observe a slight increase in overall accuracy while FN probability greatly decreased as a trade-off with more data collection time.

4.5.6 Sectored Speaker Array Modulation

ALPS beacons feature a sectored four speaker array (see Section 3.2) for improving ultrasound signal coverage and to provide AOA data in the future. The beacons feature a two channel audio codec, which means that they can transmit two different signals simultaneously and can select which speakers to transmit from. Although the speakers are separated by $90°$ in the horizontal plane and $120°$ in the vertical, the signals from adjacent speakers will eventually collide with each other after a certain distance when transmitted simultaneously (see Figure 3.11), which causes interference if they are identical. In order to prevent interference, we only transmit a single upchirp and a single downchirp signal simultaneously from two speakers. Since these signals are orthogonal to each other, they do not interfere and can be easily separated by using a matched filter. This initial transmission is followed by a short $2.6ms$ long period of silence to switch over to the second set of speakers and then a second identical transmission from the remaining two speakers, as can be seen as the ”XX” pattern in Figure 3.6. After this there is a period of silence of $127.4ms$ for the signals to propagate throughout the space. We increased the length of the chirps in this scheme to $100ms$ (plus a fade/in out time of $5ms$ each) to increase the range, however the length of the signals and the length of the period of silence after their transmission can be adjusted based on the size of the space the beacons are deployed in.

The demodulation of this scheme is very similar to the process described in Section 4.5.3, but requires two matched filters to be applied to the data for filtering the up- and downchirps. Since the successive ”Xs” are separated by $112.6ms$ and each speaker always transmits the same chirp at the same time in the time slot, the demodulator can determine which speaker performed the transmission, even if only a single chirp of ”X” is received, since the range of the beacons is limited to $35m$ or $102ms$ of propagation time.

Chapter 5

Time Synchronization

5.1 Overview

Time synchronization is extremely valuable in a wide variety of applications. It enables event ordering, coordinated actuation, energy-efficient communication, low-power duty cycling and the ability to measure distances. Time synchronization and ranging/localization are very tightly knit together in range based systems. Generally more precise synchronization results in more precise ranging/localization. In TOF ranging systems it is used to synchronize receivers with transmitters to measure the propagation time of signals sent between them. In TDOA systems, the transmitters are time synchronized to precisely schedule their transmissions using a common time base, which is key to being able to solve for the receiver's location.

One immediate application of tight time synchronization within the context of localization is the ability for devices to perform direct TOF ranging from beacons instead of TDOA ranging. After a node has heard from four or more nodes within a single area, the mobile device can synchronize with global time and then perform TOF ranging for any successive beacons. In the following sections we present and evaluate a novel time synchronization technique that leverages the continuously free-running audio sampling subsystem of a smartphone to synchronize with global time. Once synchronized, each device can determine an accurate proximity from as little as one beacon using TOF measurements. This significantly decreases the number of beacons required to cover an indoor space and improves performance in the face of obstructions. We show through experiments that this approach outperforms the Network Time Protocol (NTP) on smartphones by an order of magnitude, providing an average $720\mu s$ synchronization accuracy with clock drift rates as low as $2ppm$.

In Section 5.5 we describe and evaluate the BLE based time synchronization method used in current ALPS platforms, which is used to time synchronize receivers to the beacons' TDMA cycle on a time slot level for mapping ultrasound transmissions to their corresponding beacons. While this method does not provide the precision needed to perform TOF ranging, it is employed by ALPS when TDOA ranging is used.

Section 5.6 evaluates the latency and jitter of time synchronizing ALPS beacons via 802.15.4, which has been used as the time synchronization communications medium between network masters/plug forwarders and beacons.

Finally Section 5.7 evaluates the feasibility of using LoRaWAN transmissions for time synchronizing Plug Forwarders to a central Network Master.

5.2 Time Synchronization Background

A significant amount of work from the distributed systems community has focused on time synchronization [47, 56, 64, 78]. The most commonly adopted of these approaches is NTP (see Section 2.6.1) that uses round-trip message delay averaging to set times. In this section we show through our experiments that due to the asymmetric and lossy nature of modern wireless communication channels, it is extremely difficult to reach sub millisecond levels of accuracy. To the best of our knowledge, this is one of the first efforts to explore how tightly smartphones can synchronize with global time at the application level. Celltower and GPS synchronization exist within their own subsystem but are usually isolated from the main system clock or updated at a coarse granularity (seconds). There has been significant work related to message passing based approaches from the sensor networking community [50, 52, 55, 74] that can be applied towards infrastructure timestamping. Eventually these approaches could find their way into mobile phones. Please refer to Section 2.6 for more detailed explanations of related work on time synchronization.

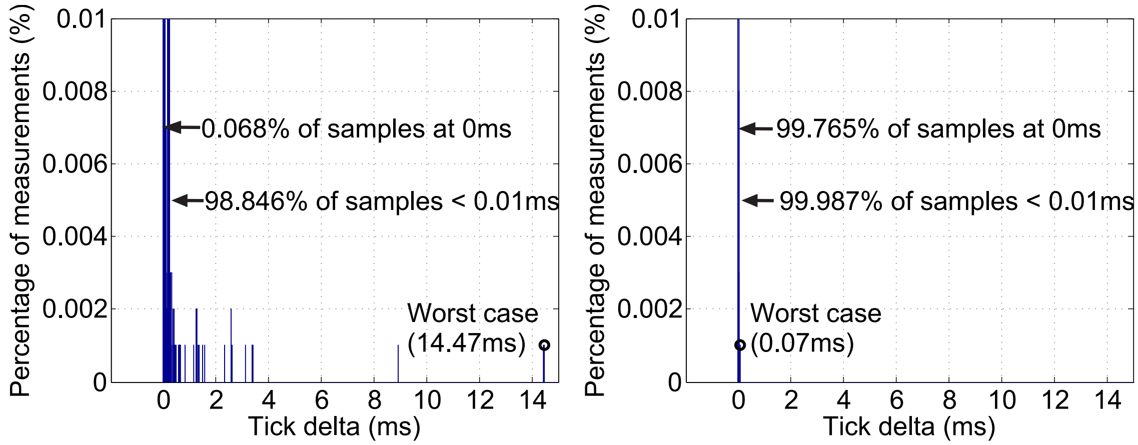
5.3 iPhone Network Time Synchronization Benchmarks

In this section, we explore the limits of time synchronization on iPhone 4 and 5S mobile platforms and propose a new approach that uses the audio recording subsystem to recover the system clock of ALPS beacons. This provides applications with the ability to perform precise time-stamping (especially of audio events) isolated from non-determinism within the operating system and network.

Benchmarking time synchronization accuracy on a smartphone is difficult because the current platforms typically do not expose low-level I/O and the operating systems are optimized for energy-efficiency rather then timing performance. In order to understand the nature of timing and synchronization on smartphones, we ran a set of timing experiments that examine key time-synchronization performance characteristics as described below.

5.3.1 Clock Granularity

We first need to establish the granularity of the clock on our mobile device in order to bound the minimal synchronization accuracy. This can be achieved by calling os_get_time_of_day() continuously in a tight loop and inspecting the tick increments. Figure 5.1 shows the resulting



(a) iPhone 4 (b) iPhone 5S

Figure 5.1: Delay in consecutive iOS system time calls

distribution of tick values for two iOS devices. The histograms show that the system clock has a granularity of 1-microsecond and that jitter associated with OS delays and context swaps is quite low. The faster of the two platforms (iPhone 5S) exhibits almost no jitter from background tasks. A similar experiment on an Android Galaxy Nexus 4 showed high levels of jitter for most reads, on the order of milliseconds and greater due to differences in Android's task scheduler. For this reason, we perform the rest of our experiments using iOS. Also, currently iOS has a lower latency audio subsystem as compared to Android. Round-trip audio times computed by looping test sounds back from input to output using [4] and are typically below $10ms$ on iOS devices.

5.3.2 ADC Timing Performance

In order to compare the smartphone's clock against a reference source, we need to establish a low-latency input or output mechanism. Typical smartphone I/O includes: UART, Bluetooth, WiFi, LTE, the display, audio, external buttons and various sensors like light and acceleration. With the exception of audio and the UART, the interfaces exhibit milliseconds or greater amounts of timing uncertainty. The audio interface is particularly appealing since it contains its own

	Jitter (samples at 48kHz)		
	0	1 ($22\mu s$)	2 ($44\mu s$)
iPhone 4	66%	34%	0%
iPhone 5S	0%	87%	13%

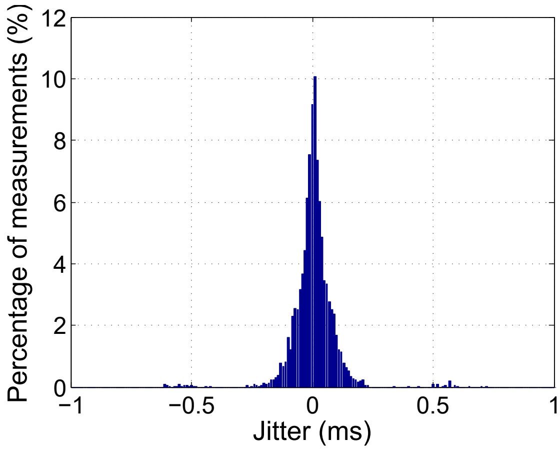
Table 5.1: Audio ADC sampling jitter on iPhone 4 and 5S

continuous sampling clock that can be used for relative time stamping and has the ability to configure buffer sizes and sampling rates.

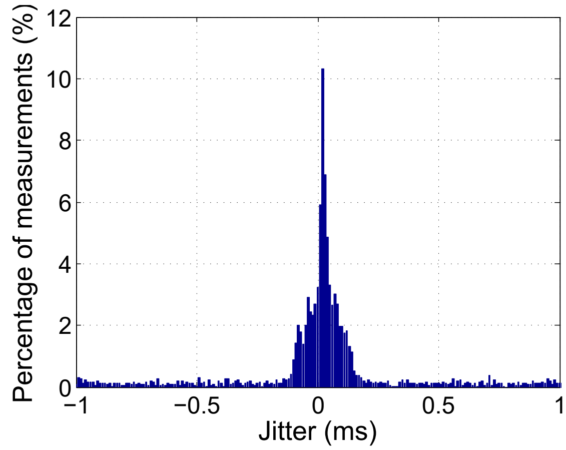
In order to estimate audio latency, resolution and clock drift, we connect the PPS output from a uBlox 7 GPS receiver through a level shifter into the microphone input of a smartphone. The GPS PPS output provides a highly stable clock reference with less then 25 nano-seconds of jitter. In our first experiment, we collect 1 hour of PPS input audio signals. Table 5.1 shows the jitter between the time that the PPS signal transitions as compared to the expected time based on the audio sampling rate. For example, a jitter value of 1 means that the PPS pulse period was 1 audio sample ($22\mu s$) different than expected over that period of time due to sampling jitter or clock drift. From these values we can compute that the clock drift-rate over an extended period on the iPhone 4 and iPhone 5S are $7.17ppm$ and $23.56ppm$ respectively as compared to the GPS reference. We see a worst-case sampling jitter between two PPS edges as 2 audio samples.

5.3.3 OS Timing Performance

In order to measure timing performance based on OS timestamping, we use the same experimental setup from the previous section and also timestamp the arrival of each audio buffer segment using the OS clock ($1\mu s$ granularity) and compute the relative PPS edge in the audio stream. Figure 5.2 shows that the worst-case jitter between OS tick time and a PPS tick is $724\mu s$ with an average jitter of $53.2\mu s$. Since the PPS signal is regular, this allows us to bound the OS timestamping jitter to within $1ms$.



(a) iPhone 4

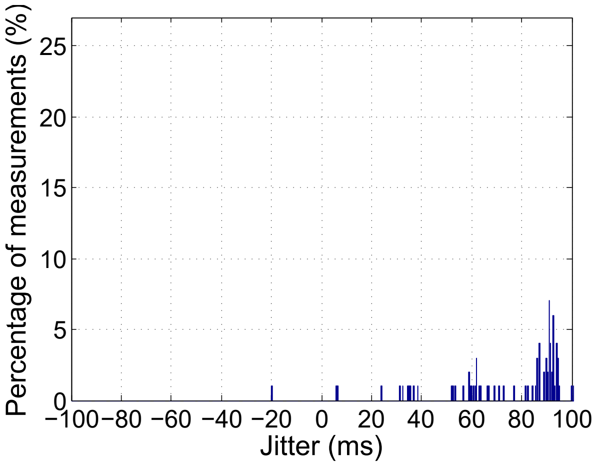


(b) iPhone 5S

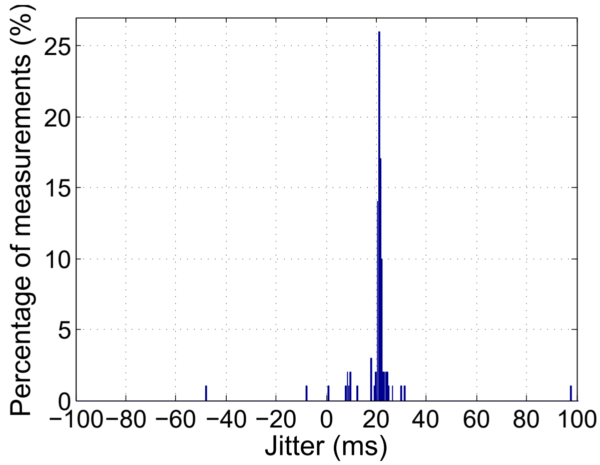
Figure 5.2: Audio buffer OS time-stamping jitter

5.3.4 NTP Timing Performance

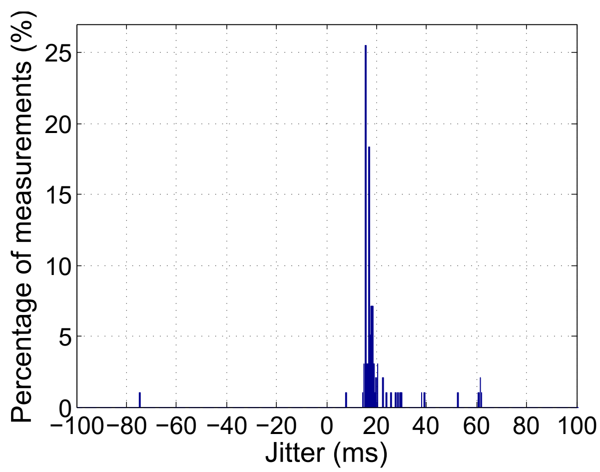
We can now use our OS timestamping bounds to benchmark the performance of NTP running on the phone as compared to the GPS input. Using ios-ntp [21], we capture NTP timestamps along with the system time and our audio PPS input. For each NTP sample, we allow the server to synchronize for 200 seconds before comparing against the OS and PPS time. During this time the NTP process performs clock rate adjustment. Figure 5.3 shows the jitter between the NTP clock and the PPS timestamps over 100 different NTP synchronizations. Since NTP is negatively impacted by jitter and asymmetry in communication channels, we ran experiments using LTE, campus WiFi and an idle WiFi router directly connected to a Stratum 1 NTP server fed by a dedicated GPS clock. We see that NTP using LTE has an average jitter between synchronization attempts of $47ms$ (max $466ms$), while NTP over normal WiFi has an average jitter of $30ms$ (max $326.5ms$) and even in the isolated ideal case, there is an average of $19.3ms$ (max $74ms$) of jitter. For measuring audio TOF, a time of $1ms$ corresponds to a distance of $0.33m$ while the distance equivalent to $47ms$ is more than $15m$. This is not accurate enough to be useful for most indoor localization applications. In the next section, we describe our acoustic TDOA synchronization approach that improves timing accuracy to $720\mu s$ on average in practice.



(a) Remote stratum 1 server over WAN via LTE



(b) Remote stratum 1 server over WAN via WiFi



(c) Local stratum 1 over LAN via WiFi

Figure 5.3: NTP second tick deviation from UTC second tick

5.4 Clock Recovery via TDOA Based Localization

As part of the TDOA calculation, it is possible to estimate the instant when each signal was originally transmitted. We use this approach to synchronize the audio stream with respect to global time, which can then be used as a reference for application-level time-stamping. Time-stamping of audio events based on their position in a buffer completely removes any sources of delay from the operating system or networking stack. Given the relatively small amount of jitter seen when sampling audio, it also stands as a reasonable alternative for synchronizing other events, for example to perform cooperative ranging between two mobile phones.

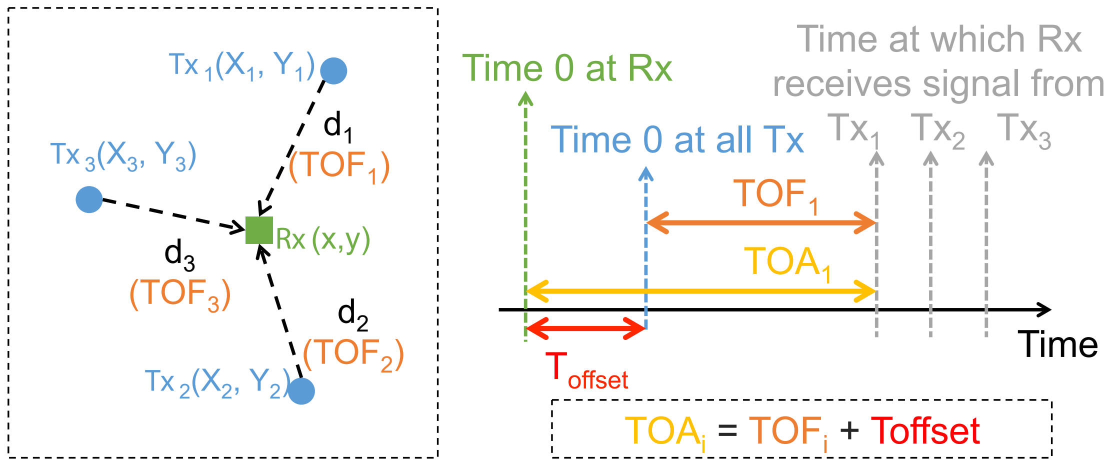


Figure 5.4: Estimation of start of ultrasound transmission cycle timing

Figure 5.4 shows the layout of three transmitters and a receiver in 2-D space, and their corresponding notions of time. We consider the receiver's clock to be offset by T_{offset} from the transmitter's clock. Synchronization is achieved by estimating this offset. Typically this time offset is not estimated since the TDOA equations are used to directly estimate the position of the receiver [61]. However, the time offset can be obtained easily once the position has been estimated, as explained below.

(X_i, Y_i) denotes the position of transmitter i for $i = 1, 2, 3$ and is assumed to be known. The position of the receiver (x, y) is unknown. d_i is the distance between transmitter i and the receiver and is given by Equation 5.1.

$$d_i(x, y) = \sqrt{(X_i - x)^2 + (Y_i - y)^2} \tag{5.1}$$

The TOF of the audio signal from transmitter i to the receiver is given by Equation 5.2.

$$TOF_i = \frac{d_i(x, y)}{V} \tag{5.2}$$

where V is the speed of sound.

The corresponding arrival time of the signal measured by the receiver is the TOA_i, given by

Equations 5.3 and 5.4.

$$TOA_i = TOF_i + T_{offset} \tag{5.3}$$

$$TOA_i = \frac{d_i(x,y)}{V} + T_{offset} \tag{5.4}$$

The receiver needs to estimate T_{offset} given TOA_i and (X_i, Y_i) for $i = 1, 2, 3$. To estimate the T_{offset}, we first estimate the position of the receiver. To estimate (x, y), we use the standard multilateration technique [61] by eliminating T_{offset} and arrive at the TDOA equations. We then find the (x, y) that minimizes the sum of squares of error (ξ) in TDOA using Equations 5.5, 5.6, 5.7 and 5.8.

$$Measured\, TDOA_{ij} = TOA_i - TOA_j \tag{5.5}$$

$$True\, TDOA_{ij} = \frac{d_i(x,y) - d_j(x,y)}{V} \tag{5.6}$$

$$\xi_{TDOA_{ij}}(x,y) = [TOA_i - TOA_j - \frac{d_i(x,y) - d_j(x,y)}{V}]^2 \tag{5.7}$$

$$(\hat{x}, \hat{y}) = \operatorname*{argmin}_{x,y} \sum_{\substack{(i,j) \\ 1 \le j \le N \\ j \ne i}} \xi_{TDOA_{ij}}(x,y) \tag{5.8}$$

We next estimate T_{offset} from $(\hat{x}, \hat{y})$ and the TOA using Equation 5.9.

$$T_{\hat{offset}} = \frac{1}{3}\Bigg(\sum_{i=1}^{3}\Big(TOA_i - \frac{d_i(\hat{x},\hat{y})}{V}\Big)\Bigg) \tag{5.9}$$

5.4.1 iOS App Design

Figure 5.5 shows an overview of the receiver demodulation software that runs on iOS and is used to recover the localization system's clock via TDOA localization. It was prototyped in MATLAB before being ported to Objective-C. The algorithm for demodulating the ultrasonic transmissions and calculating the receiver's position was translated into C code using MATLAB's C coder. The

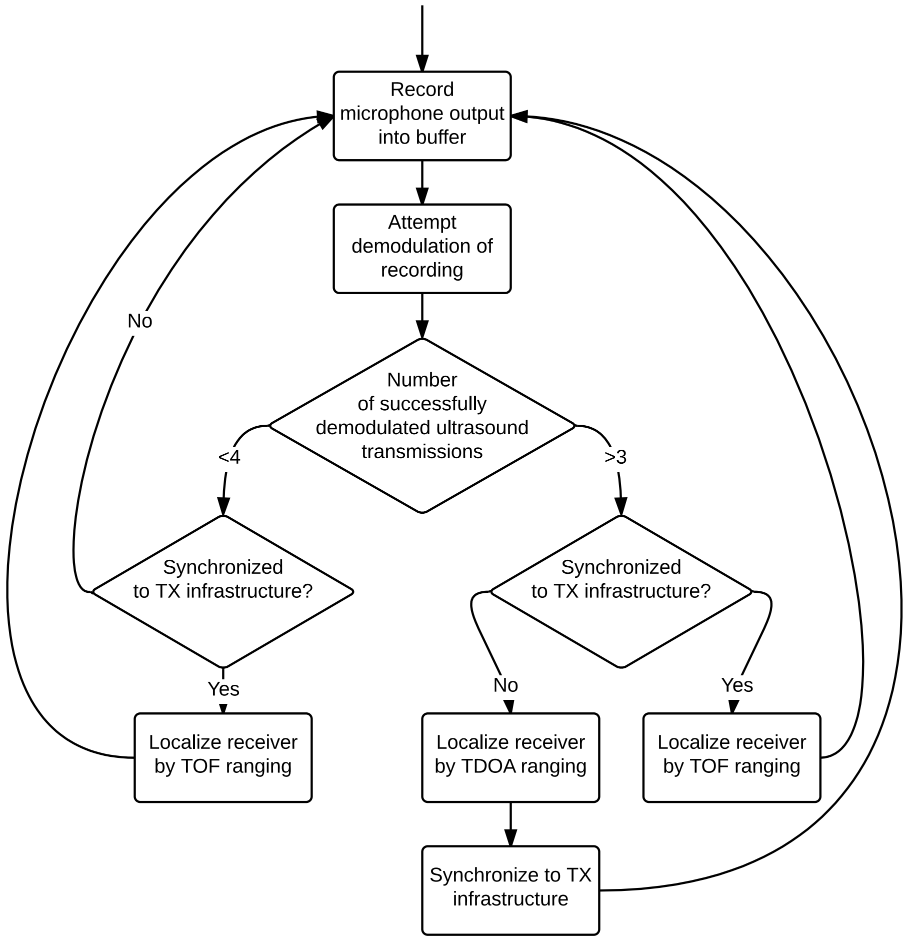


Figure 5.5: iOS app flowchart for clock recovery via TDOA based localization

iOS app continuously listens on the microphone and periodically passes filled audio buffers to the demodulator. Once the phone's position is determined using TDOA ranging and multilateraion, the app computes the TOF of the signals it has just received to determine the start of the previous TDMA cycle relative to the captured audio buffer. The audio sample index s_0 at which the TDMA cycle started is stored in memory and the phone is now synchronized to the transmission infrastructure. A counter keeps track of how many audio samples have been captured since s_0 and calculates successive TOF values of successfully demodulated ultrasonic packets based on their TOA in relation to s_0. Whenever the phone successfully demodulates enough ultrasonic packets to localize itself using TDOA it will resynchronize to the transmitters.

5.4.2 Evaluation

In this section, we evaluate the effect of errors in deployment and measurement of beacon node locations on our ability to synchronize time which is equivalent to estimating the T_{offset} value defined in Section 5.4. We also evaluate the sensitivity of transmitter placement. These two evaluations are performed through simulation given the model described in the section below. We then experimentally evaluate the timing accuracy and ranging capabilities of our platform.

5.4.3 Modulation Scheme

The ultrasound modulation scheme used in the following experiments was adapted from the one described in Section 4.4. Due to the more limited frequency response of the iPhone 5S used in these experiments (same frequency response as the iPhone 4S seen in Figure 3.13), we modified the scheme to employ longer preambles of $150ms$, which are transmitted simultaneously to the encoded data as shown in Figure 5.6. Since chirp signals sweep across a given frequency range, it is possible to partially overlap symbols without causing significant interference. This not only allows us to increase the data rate, but also increase the symbol duration, which increases the SNR of the signal at the receiver while keeping packet duration to a minimum. The preamble is stretched over the entire packet, making it easily discernible after matched filtering and improving the range resolution of the transmission. Data symbols have also been increased in length, with the start of each data symbol being separated by several milliseconds. Instead of using different chirp rates to denote four different data symbols, only an upchirp (frequency increasing over time) and a highly orthogonal down-chirp (frequency decreasing over time) are employed to keep the scheme robust to errors. Each symbol is faded in and out in amplitude over $5ms$ to prevent the loudspeaker from producing audible clicks due to rapid changes in signal power. In order to enable multiple access between ultrasound transmissions, all transmissions are time-multiplexed with only a single transmission taking place per time slot.

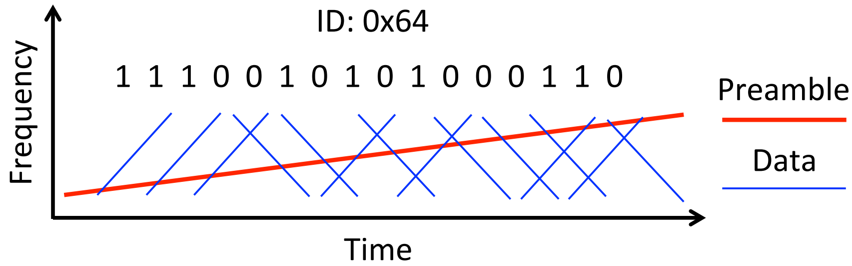


Figure 5.6: Ultrasound packet structure

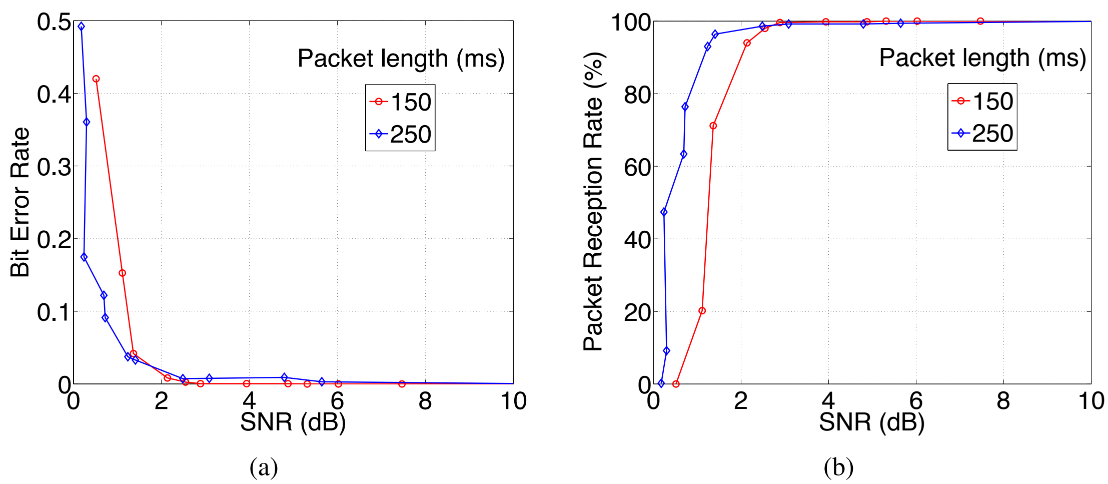
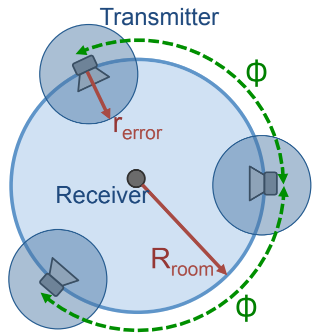
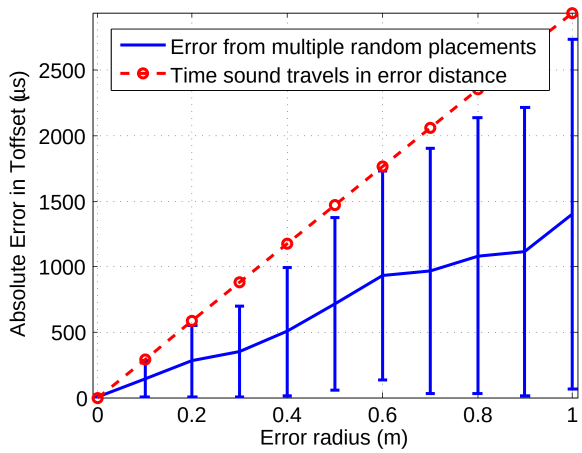


Figure 5.7: Performance of modulation scheme

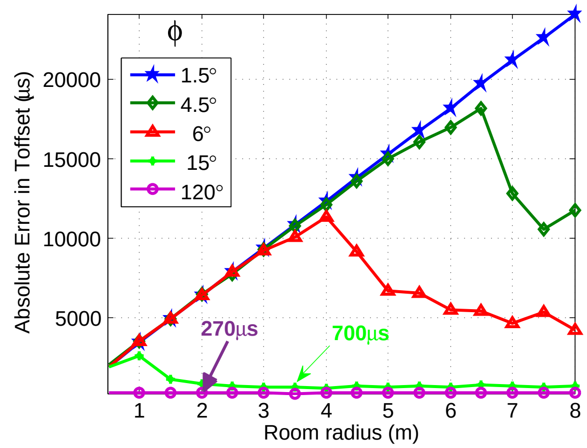
Figure 5.7 shows the performance of our modulation scheme for two different packet lengths of $150ms$ and $250ms$. The $150ms$ packets use a data symbol length of $25ms$, a preamble length of $150ms$ and a symbol separation of $9ms$. The $250ms$ packets use a data symbol length of $30ms$, a preamble length of $250ms$ and a symbol separation of $15ms$. To test the performance of this modulation scheme, we transmitted 500 packets with random data while varying the transmit power to an iPhone 5S, which performed the demodulation of the received data. The $150ms$ long packets perform nearly as well as the $250ms$ long packets in terms of Bit Error Rate (BER) and Packet Reception Rate (PRR), with only a negligible impact on PRR down to an SNR of $2.5dB$. The $250ms$ long packets do not exhibit a significant impact on PRR until an SNR of $1.8dB$.



(a) Model of transmitter and receiver placement in a room



(b) Effect of measurement error in transmitter position



(c) Effect of transmitter placement geometry

Figure 5.8: Deployment and geometry error

5.4.4 Deployment and Geometry Error

We assume the spatial configuration of beacons and receivers in 2-D space as shown in Figure 5.8(a). The room is circular with a radius of R_{room}. For simplicity, the three beacons are assumed to be placed in the periphery of the room. We assume some symmetry in placement as indicated by the separation angle ϕ between two pairs of beacons. The measurement error r_{error} in beacon position is assumed to be uniform in all directions around a transmitter and to be equal at all three beacons. The true position of each transmitter is at a random location on the circle that defines possible deployment error. The receiver is placed in the center of the room and we assume that the speed of sound is constant at 340m/s.

Effect of incorrect measurements in transmitter positions

First, we evaluate the impact of measurement error in the transmitters' positions with R_{room} = 5m and ϕ = $120°$. We then analyze the effect of r_{error} on T_{offset} by sweeping r_{error} from 0 to $1m$ and generating 400 random configurations of the transmitters on their error circles and estimating the T_{offset} for each configuration. The solid blue lines in Figure 5.8(b) show all possible values of errors. We see that the worst-case error for each r_{error} is bounded at the top by the time taken by sound to travel the same distance. Typically, while deploying this system with nominal care, the measurement error can be restricted to below $10cm$, which is equivalent to an error of $270\mu s$ in T_{offset} .

Effect of transmitter placement geometry and room size

Next, we evaluate the effect of room size and geometry of the placement by increasing R_{room} up to $10m$ and varying ϕ from the worst possible placement ($\phi = 0°$) to the best possible placement ($\phi = 120°$) geometry. We assume a value of $10cm$ for r_{error} from the previous section. Figure 5.8(c) shows the worst case error among 400 simulations at each data point. We see that for $\phi = 1.5°$ the error is quite high ($15ms$ for $R_{room} = 5m$) and grows proportionally to the room size. This is because the transmitters are almost at the same location, therefore the three unique TOA equations that we expect from the transmitters are identical, leading to insufficient information. The position of the receiver could be incorrectly estimated anywhere within the room, which is why we see the worst case error in T_{offset} growing linearly with the room size. For the best case geometry of $\phi = 120°$, we see that the error ($270\mu s$) is not dependent on the room size and is determined by r_{error}, except when R_{room} is small and comparable to r_{error}. This is because at $\phi = 120°$ the transmitters are sufficiently separated to provide the timing information required to estimate the receiver position and T_{offset}. For intermediate values of ϕ, we see that as the room size increases, the error increases with room size until the point where the room size

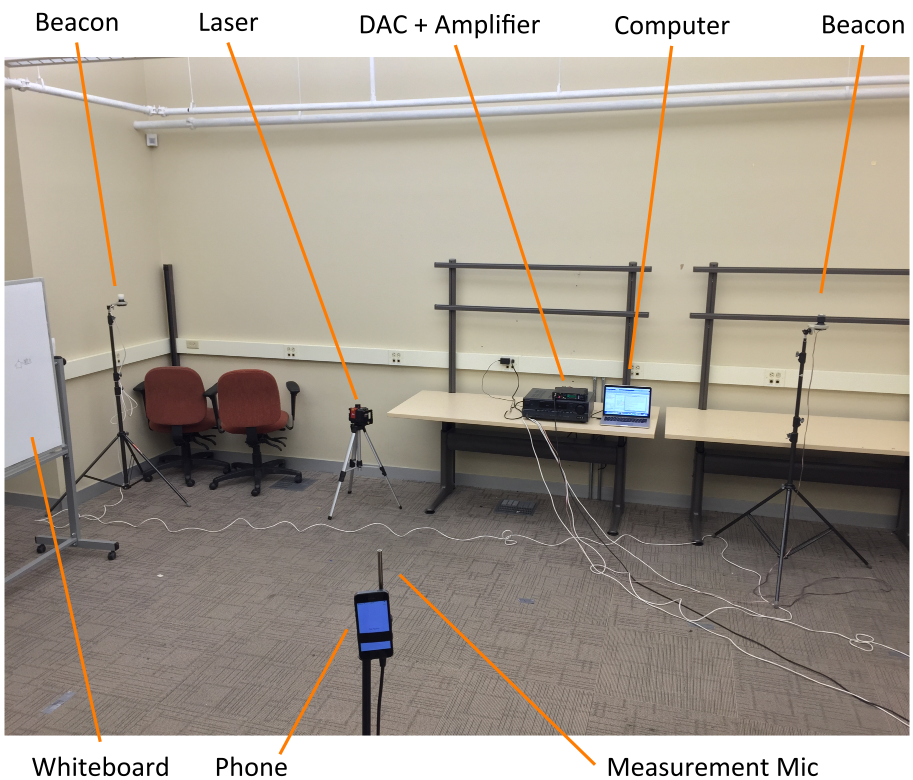
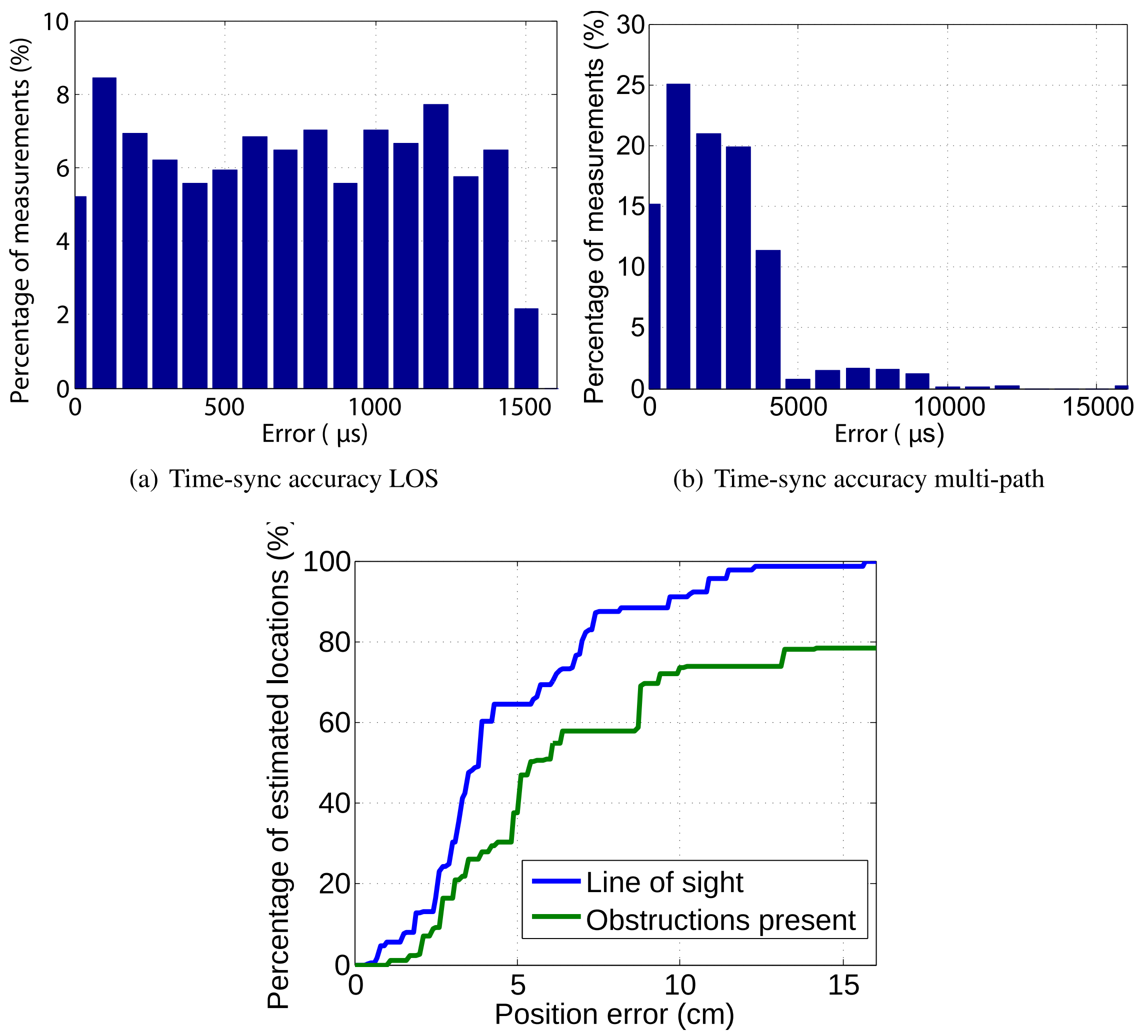


Figure 5.9: Experimental setup

is large enough to provide sufficient spatial separation to the transmitters. This can be seen when each line reaches a maximum and then decreases.

5.4.5 Synchronization and Ranging Performance

In order to evaluate the synchronization error of our method and the resulting positioning error, we set up four ultrasonic beacons in a 4.5x$5.5m$ area. Our ranging signals were generated using a MOTU Ultralite mk3 10 channel audio interface hooked up to an Onkyo HT-R540 amplifier and received by an iPhone 5S and an Audix TM-1 measurement microphone, which were co-located on a microphone stand. The experimental setup can be seen in Figure 5.9. The second recording device allowed us to tightly time synchronize all transmitters as well as the measurement microphone to obtain ground truth TOF and position measurements. The iPhone synchronizes itself to the transmitters using our audio synchronization method and recorded its calculated position and TOF values. We perform measurements at 20 random locations in the room, for which 60 samples were recorded each. A mean absolute time synchronization error of $720\mu s$ with a maximum of $1484\mu s$ was achieved. This resulted in a maximum absolute distance error of $15.6cm$. Figure 5.10(c) shows the Cumulative Distribution Function of positioning error resulting from time synchronization using our method. 98% of the samples exhibit a positioning error of less than $12cm$ and 100% show an error of $16cm$ or less in LOS conditions. This small-scale experiment validates the concept that highly accurate clock synchronization is possible through audio sampling.



(a) Time-sync accuracy LOS

(b) Time-sync accuracy multi-path

(c) Position estimation accuracy

Figure 5.10: Acoustic time synchronization accuracy and position error

Further, we conducted an experiment where one or two transmitters (depending on the position of the receiver) were obstructed by a large white board. In this case, approximately 78% of the calculated position samples were within an absolute distance error of less than $14cm$ (see Figure 5.10(c)) and a maximum error of $4.16m$ was encountered. The large maximum error is due to the phone being unable to measure its position accurately due to multipath or severely attenuated ranging signals, causing it to synchronize erroneously to the transmitting infrastructure when placed in certain locations. Figure 5.10(b) shows that this is reflected in a maximum absolute time synchronization error of $39.1ms$ with a mean of $2.33ms$. These points could be eliminated if we received one good set of samples and then performed TOF ranging. We purposefully disabled this capability to highlight the impact of obstructions on TDOA localization. One could also apply filtering to the recorded signals by thresholding and sampling over multiple TDMA cycles to obtain an accurate lock on the receiver's position. A software-based Phase-Loop Lock (PLL) controller that slowly adjusts the clock based TDOA inputs when available rather than immediately resetting the timing offset on each sample can prevent isolated erroneous position measurements from causing significant time synchronization errors.

5.4.6 Limitations

Although promising, there are a few limitations to our approach for recovering the clock of the localization system via TDOA localization. In 3-D space, our system relies on the receiver being in LOS of at least four transmitters in order to synchronize. However, it is possible that obstructions inside the building or the person holding the mobile device block one or more transmitters. We experimentally studied the effect of this as shown in Figure 5.10(b) and Figure 5.10(c). It is also possible that in certain areas such as long corridors, fewer than three transmitters are present. In these cases, we can utilize the inertial sensors on the phone to track the mobile device using pedestrian dead reckoning [88]. Our method also depends upon measuring the location of the mobile device accurately in order to recover the clock. The IMU sensors may also be used to filter out location errors and to only allow clock recovery when the mobile device is stationary, which generally results in a more accurate location measurement. Another challenge is the need to filter NLOS multipath signals. The presence of a multipath signal in the absence of a direct LOS signal could result in a TOF measurement which is equivalent to the receiver being located at a much larger distance away from the transmitter. We have developed several methods to filter NLOS multipath signals to mitigate this problem as described in Section 4.5.3 and Section 4.5.4.

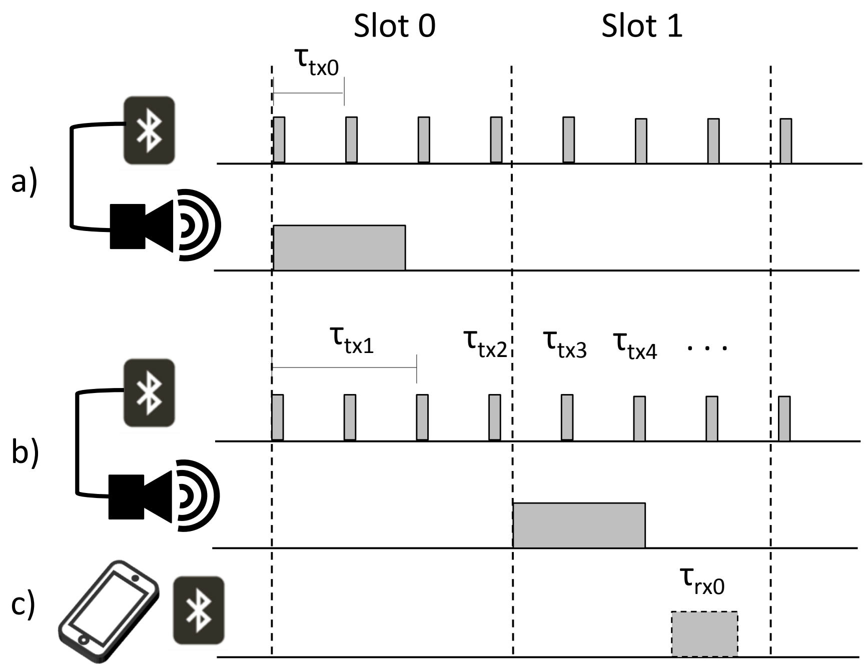
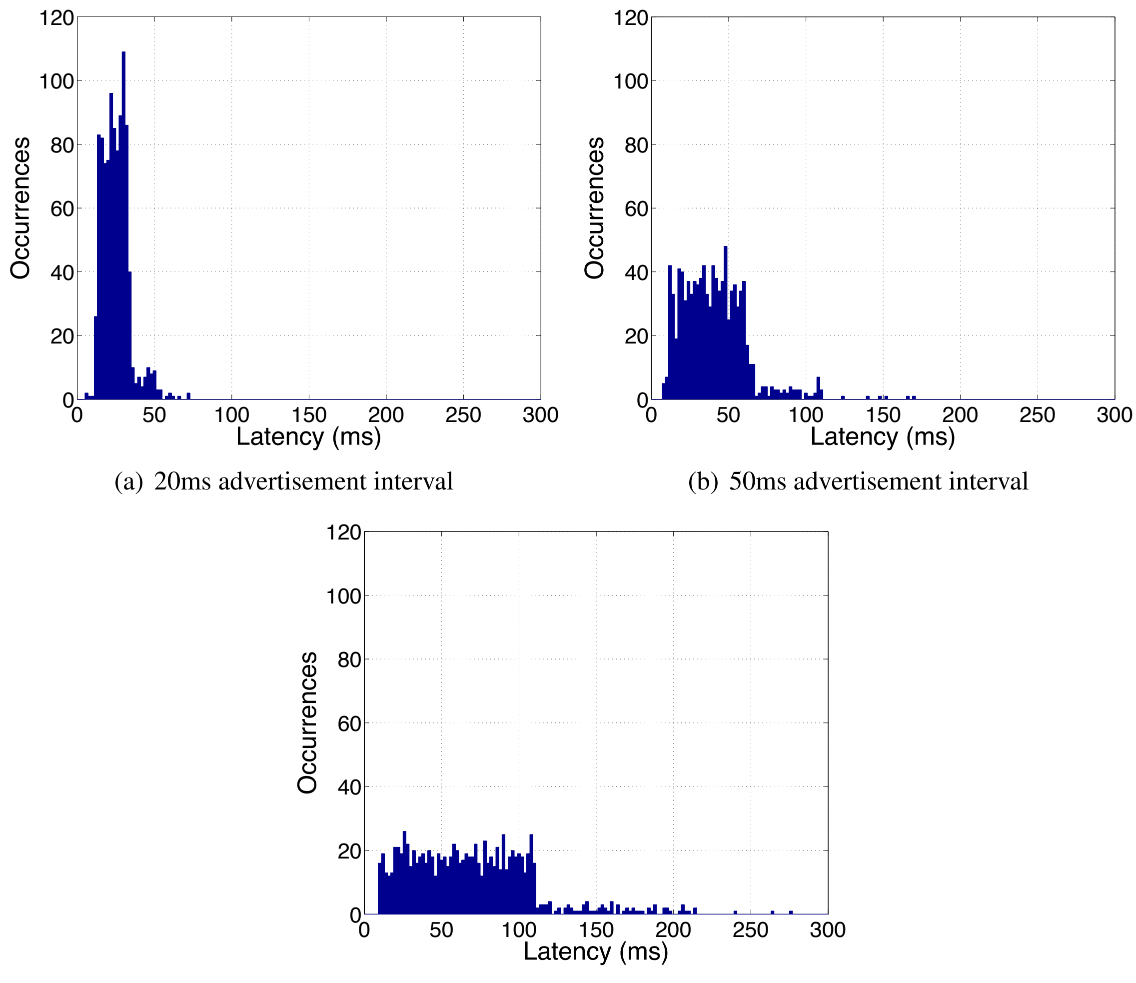


Figure 5.11: BLE synchronization transaction diagram

5.5 BLE Time Synchronization

In order to map received ultrasound transmissions to their respective beacons, ALPS beacons transmit periodic iBeacon BLE advertisement packets that contain a counter value τ_{tx} indicating the time offset from the broadcast of the BLE advertisement packet to the beginning of the TDMA cycle shown in Figure 5.11 (a), (b). Mobile receivers can synchronize to the TDMA cycle by timestamping the BLE packet reception τ_{rx} (Figure 5.11 (c)) and subtracting the received counter value from τ_{rx}. While iBeacon BLE advertisement intervals can be as low as $20ms$, there is a non-deterministic latency associated with receiving them in an application running on a smartphone. Smartphones such as the iPhone 5S do not allow low-level access to their BLE stack for accurate timestamping. On the iPhone 5S, received iBeacon BLE advertisements are passed to the application roughly once a second, but it is unclear how often the phone receives BLE packets and how long it takes before they signal applications.



(a) 20ms advertisement interval

(b) 50ms advertisement interval

(c) 100ms advertisement interval

Figure 5.12: BLE advertisement packet reception latency

In order to evaluate the feasibility of time-synchronizing the mobile device to the TDMA cycle of the broadcasting infrastructure, we measured the latency between BLE advertisement packets and the audio input of an iPhone 5S. We set up a beacon to toggle a GPIO pin that was connected to the phone's microphone input when a new TDMA cycle started and simultaneously started broadcasting BLE advertisement packets containing τ_{tx}. The phone timestamped the reception of each BLE packet in the application and subtracted τ_{tx} to determine when the GPIO pin was toggled in its frame of reference. Simultaneously the phone was recording the GPIO trigger in an audio waveform, which was precisely timestamped to within $1ms$ using the technique detailed in Section 5.3.2. Figure 5.12 shows the BLE advertisement packet reception latency for $20, 50$ and $100ms$ advertisement intervals across 1000 packets. When set to a $20ms$ interval, we measured an average latency of $25.1ms$ with a maximum of $72.4ms$, which is well below our $100ms$ long TDMA slot length, hence allowing slot-accurate time synchronization via BLE. The less frequent intervals provided unacceptable worst-case latency of $169.3ms$ and $275.1ms$ (50 and $100ms$ intervals respectively).

5.6 802.15.4 Time Synchronization

ALPS uses 802.15.4 networking to time synchronize its beacons to a network master or plug forwarder by simply timestamping periodically received packets (see Section 3.6). The synchronization precision of ultrasound transmissions across all beacons is directly related to the localization performance of the system and therefore should be as high as possible. In order to measure the time synchronization performance of the 802.15.4 links we measured the jitter of sending 100 periodic 802.15.4 packets at regular intervals from a generation 2 Network Master (see Section A.2.1) to a generation 2b beacon (see Section A.3.1), which triggered the playback of an ultrasonic chirp signal. The network master and beacon featured an embedded design running firmware bare-metal on an 8-bit Atmel SOC for low latency audio playback and low jitter synchronization. The jitter was measured by connecting an oscilloscope to the line output of the audio codec of the beacon and measuring the jitter between successive ultrasonic signal transmissions. This resulted in a worst case jitter of $20\mu s$ as shown in Figure 5.13, corresponding to an ultrasound ranging error of $6.8mm$, which is more than accurate enough for our purposes.

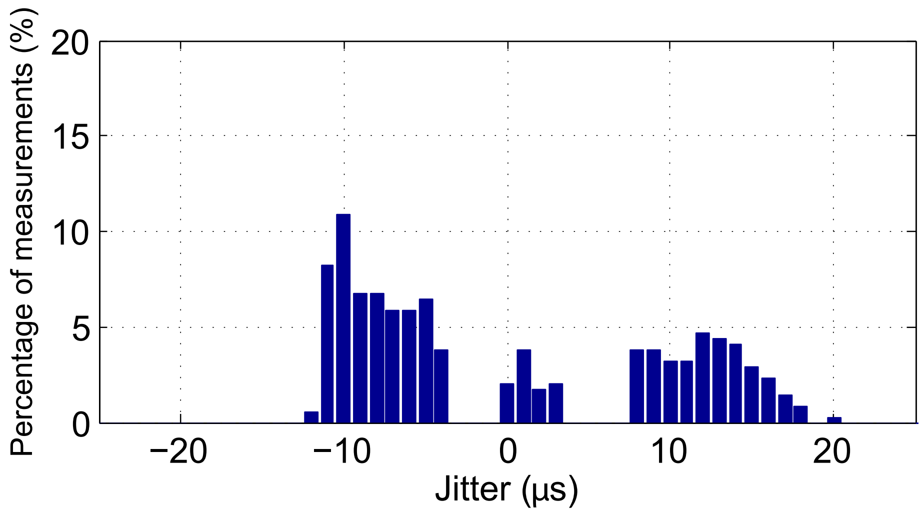
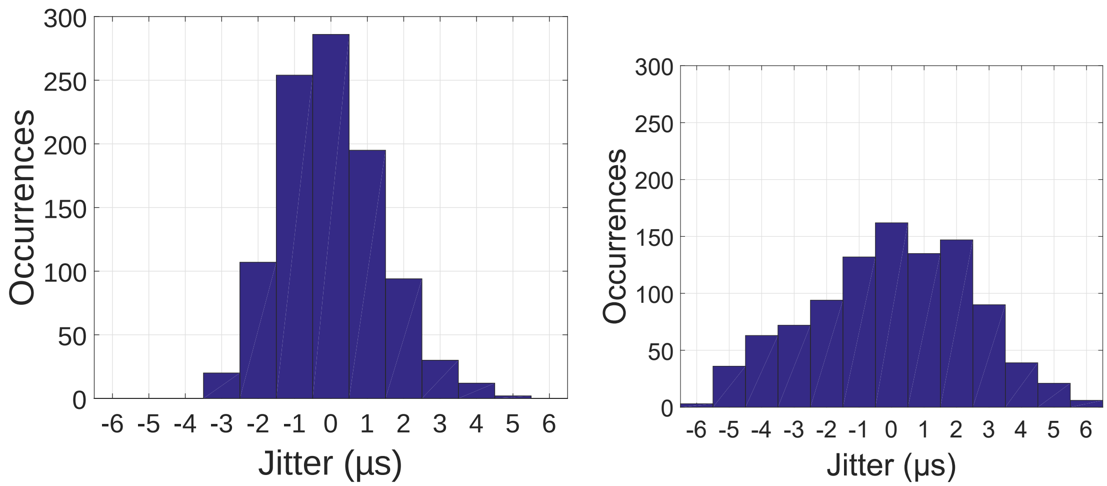


Figure 5.13: Beacon ultrasound transmission jitter using 802.15.4 time synchronization

5.7 LoRaWAN Time Synchronization

The time synchronization between ALPS network masters and plug forwarders (see Section 3.2 is performed using LoRaWAN packets. LoRaWAN provides a license-free and long-range (up to $22km$ LOS, $2km$ NLOS) communication mechanism for synchronizing clocks and collecting data. The long range allows for a single Network Master to reach all Plug Forwarders within a single hop for the vast majority of installations and therefore greatly simplifies the network's complexity. However, since LoRa is narrow-band ($125KHz$) and has extremely long packet sizes (up to $400ms$ limited by FCC regulations), we needed to confirm that it could provide better than our $1ms$ required time synchronization accuracy. Please refer to [93] for a primer on LoRa modulation. Figure 5.14 shows the measured jitter across pairwise LoRaWAN nodes that are toggling a GPIO pin upon completion of a packet. We attenuated the transmitters and exposed them to heavy multipath environments and yet still see a worst-case error of $6\mu s$ at the largest spreading factor (SF11) allowable by the FCC and $4\mu s$ at the smallest, which is more than enough to accurately synchronize our beacons.



(a) Short Range: Spreading Factor 6

(b) Long Range: Spreading Factor 11

Figure 5.14: LoRaWAN time synchronization jitter

Chapter 6

Location Calculation

In order to localize a mobile device based on ranges or pseudo-ranges to beacons at known locations, ALPS performs trilateration (Section 6.1) or multilateration (Section 6.2) respectively. This is done in a cloud based gradient descent solver running in MATLAB. The following chapter will describe the math behind solving for the location of the mobile devices (Sections 6.1 and 6.2), the software infrastructure of the cloud based solver (Section 6.4), a rapid, user-assisted setup procedure for small ALPS deployments (Section 6.6) and how IMU sensor data fusion can improve ultrasound location accuracy and latency (Section 6.6.3).

6.1 Trilateration

Trilateration is a method to determine the coordinates of a point, given the distances from two or more points with known coordinates. The technique is commonly used in TOA (see Section 2.2.1) and RTOF (see Section 2.2.3) based localization systems, where the direct ranges from anchor points to the object being localized are known. Figure 6.1 illustrates an example of a receiver R being localized using trilateration based on the ranges r_A, r_B, r_C between it and three beacons A, B, C at known locations. The circles in the figure show a cross sectional view

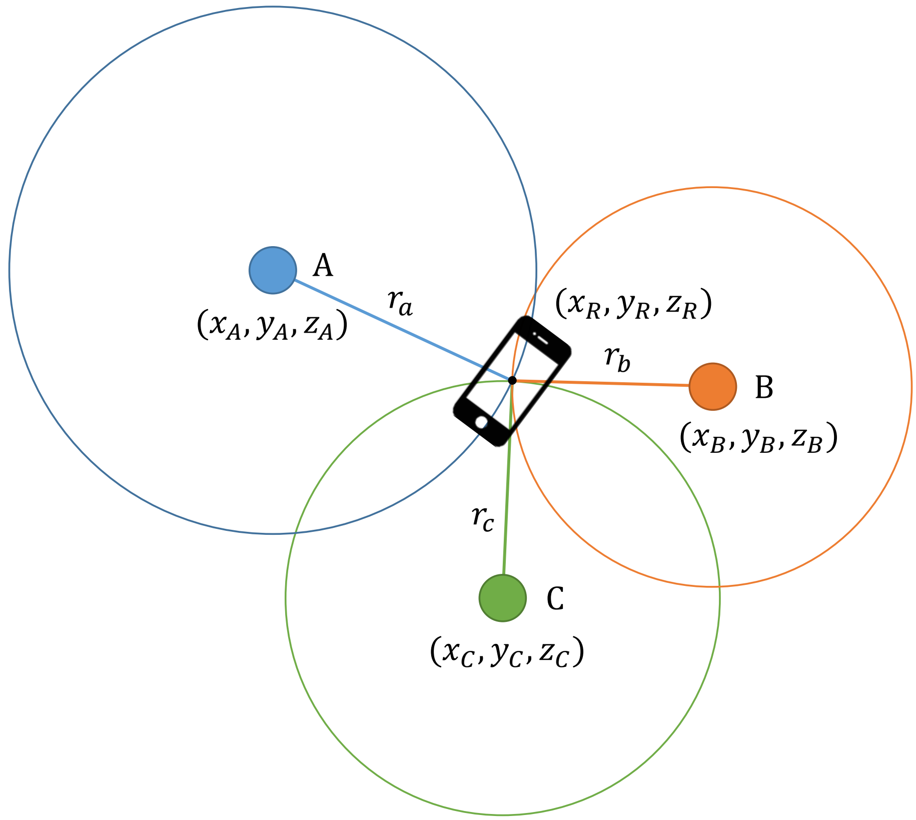


Figure 6.1: Trilateration example diagram

of three spheres of radius r_A, r_B, r_C being projected from the beacons, the intersection of which is the location (x_R, y_R, z_R) of the receiver R. Equations 6.1 [79] are the system of equations to be solved for the location of receiver R, based on the ranges r_A, r_B, r_C and the location of each beacon $(x_A, y_A, z_A), (x_B, y_B, z_B), (x_C, y_C, z_C)$. The ranges can be calculated based on the propagation speed of the signals c that were transmitted by the beacons and their respective propagation times t_{AR}, t_{BR}, t_{CR}. Trilateration requires range measurements from at least n beacons to localize a receiver in n dimensions, however localization accuracy generally increases with the amount of beacons. ALPS uses trilateration for localizing mobile devices once they are synchronized with the network master's clock using the clock recovery method described in Section 5.4.

$$
\begin{aligned}
r_A &= c\,t_{AR} = \sqrt{(x_A - x_R)^2 + (y_A - y_R)^2 + (z_A - z_R)^2} \\
r_B &= c\,t_{BR} = \sqrt{(x_B - x_R)^2 + (y_B - y_R)^2 + (z_B - z_R)^2} \\
r_C &= c\,t_{CR} = \sqrt{(x_C - x_R)^2 + (y_C - y_R)^2 + (z_B - z_R)^2}
\end{aligned}
\tag{6.1}
$$

6.2 Multilateration

Multilateration is a method to determine the coordinates of a point, given the difference in distances, i.e. pseudo-ranges, from two or more points with known coordinates. The technique is commonly used in TDOA (see Section 2.2.2) based localization systems, where the difference in the TOAs t_A, t_B, t_C, t_D of several signals transmitted by beacons A, B, C, D, corresponding to the pseudo-ranges multiplied by the propagation speed c of the signals, are measured. Figure 6.2 illustrates an example of a receiver R being localized using multilateration based on the pseudo-ranges $r_{BA} = r_B - r_A$, $r_{CA} = r_C - r_A$, $r_{DA} = r_D - r_A$ between it and four beacons at known locations. The Equations 6.2 are used to solve for the location of the receiver (x_R, y_R, z_R). The locus of the points having the same pseudo-ranges from a pair of beacons is a hyperbola in two dimensions [79], or a hyperboloid in three dimensions. The figure shows cross sections of hyperboloids for all beacons with reference to beacon A, the intersection of which is the location of the receiver. The resulting localization of this method is highly dependent upon the error of the TOA value from the reference beacon (in this case A). ALPS picks the beacon whose signal was received at the highest SNR as the reference, as it is mostly likely to have the most accurate TOA value. Multilateration requires pseudo-range measurements from at least $n + 1$ beacons to localize a receiver in n dimensions, however, localization accuracy generally increases with the amount of beacons.

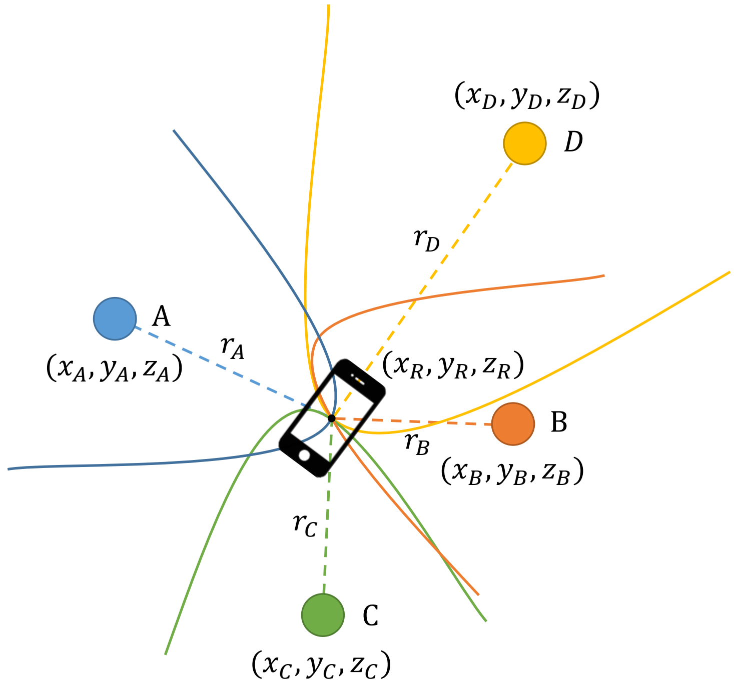


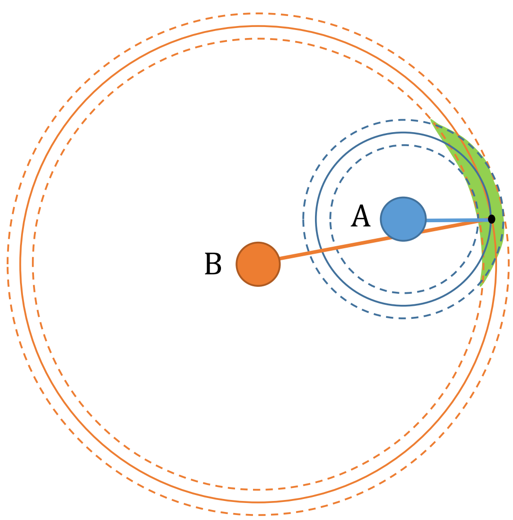
Figure 6.2: Multilateration example diagram

$$
\begin{aligned}
r_{BA} = c\,(t_B - t_A) &= \sqrt{(x_B - x_R)^2 + (y_B - y_R)^2 + (z_B - z_R)^2} \\
&\quad - \sqrt{(x_A - x_R)^2 + (y_A - y_R)^2 + (z_A - z_R)^2} \\
r_{CA} = c\,(t_C - t_A) &= \sqrt{(x_C - x_R)^2 + (y_C - y_R)^2 + (z_C - z_R)^2} \\
&\quad - \sqrt{(x_A - x_R)^2 + (y_A - y_R)^2 + (z_A - z_R)^2} \\
r_{DA} = c\,(t_D - t_A) &= \sqrt{(x_D - x_R)^2 + (y_D - y_R)^2 + (z_D - z_R)^2} \\
&\quad - \sqrt{(x_A - x_R)^2 + (y_A - y_R)^2 + (z_A - z_R)^2}
\end{aligned}
\tag{6.2}
$$

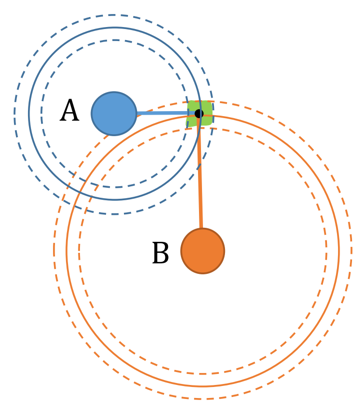
6.3 Geometric Dilution of Precision (GDOP)

In order to achieve high localization precision using trilateration (Section 6.1) or multilateration (Section 6.2), the beacons need to be separated significantly with respect to the location of the receiver in all axes that the receiver is being localized in. This is referred to as the Geometric Dilution of Precision (GDOP), which is a measure of how errors in the range measurements will affect the localization measurement calculated using Equation 6.3 [49]. Generally a $GDOP$ value of < 1 is considered optimal, with increasing values reflecting increasingly less precision. Figure 6.3 shows two nodes ranging to a point with a true range denoted by the solid circles and maximum/minimum ranging errors denoted by the dashed circles. The point may therefore be localized anywhere within the green region, which is significantly larger (i.e. results in a lower localization precision) for a geometry with high (poor) GDOP as shown in Figure 6.3(a) than for a more optimal geometry with a low GDOP in Figure 6.3(b).

$$GDOP = \frac{\Delta(Output\ Location)}{\Delta(Measured\ Data)} \tag{6.3}$$



(a) High GDOP



(b) Low GDOP

Figure 6.3: Geometric Dilution of Precision (GDOP) example diagram

Since most localization applications ALPS is used for only require two dimensional localization, the separation of beacons on the z axis is irrelevant and the beacons can all share the same z coordinate when being mounted on a ceiling. Section 5.4.4 provides insight into how the geometry of beacons affects the localization performance in ALPS.

6.4 Solver Software Architecture

ALPS uses a gradient descent solver running in MATLAB capable of performing trilateration (see Section 6.1) when the mobile device has recovered the network master's clock allowing it to perform TOF (see Section 5.4), as well as multilateration (see Section 6.2) when TDOA is used. MATLAB was chosen as the runtime environment due to its rapid prototyping capabilities and cross-platform support. The functions which perform the gradient descent minimization operation are coded in C to improve performance and are called from MATLAB.

The MATLAB solver provides a websocket interface for mobile devices to connect to. Every connection spawns a new thread, which can be handled by a separate CPU core for improved performance. Once a websocket connection is established, the mobile device asynchronously sends multiple corresponding TOA or TOF, Beacon ID and SNR values for the ultrasound signals that it received and demodulated. The solver then performs the trilateration or multilateration operation based on the received values and the known locations of the beacons and sends back the location result to the mobile device. If floor plan information is available, the solver is capable of detecting localization errors which locate the mobile device outside of the area in range with the beacons it has received signals from (e.g. in a different room, inside a wall, outside the building, etc.). In this case the solver will send back an error to the mobile device.

The solver is also able to publish the location information via XMPP, MQTT or REST to Sensor Andrew [37] or OpenChirp [28] servers for further processing or storage. We envision OpenChirp powering the networking and management backend of future ALPS systems. The lean publish-subscribe architecture is highly scalable and lends itself well to the requirements of ALPS, namely web based system management interfaces, mobile device tracking, tag tracking, map and beacon location storage and a simple MQTT or REST based API for interfacing with the system.

6.5 Intertial Sensor Fusion

IMU based sensor fusion is used in many localization systems such as automotive GPS receivers and aircraft navigation systems to improve location accuracy as well as the update rate of the system. We implemented an Extended Kalman Filter (EKF) in iOS to filter the location estimates of a mobile user by utilizing an iPhone 5S's IMU sensors for tracking. For step count and direction we use the step count from the iPhone's accelerometer and the direction from the compass which already fuses the magnetometer with the rate gyros. The details of our process model and measurement model for the EKF are given below and the performance of our IMU sensor fusion approach is evaluated in Section 6.7.

Our objective is to estimate the 2-D position (x_t, y_t) of the mobile device at time t. Equation 6.4 defines our state vector:

$$X_t = \begin{bmatrix} x_t \\ y_t \end{bmatrix} \sim \mathcal{N}(\mu_t, \Sigma_t) \tag{6.4}$$

where μ_t is the expected value of X_t and Σ_t is the uncertainty in the state. The EKF generates estimates of μ_t and Σ_t based on the prediction from the previous state X_{t-1} and the process model, and then updates this estimate based on measurement Z_t and the measurement model. A time step of $t = 1$ is the time a person takes for one step while walking.

6.5.1 Process Model

The input u_t to this system is given by Equation 6.5:

$$u_t = \begin{bmatrix} \Delta D_t \\ \theta_t \end{bmatrix} \tag{6.5}$$

with noise v_t such that:

$$v_t = \begin{bmatrix} v_t^D \\ v_t^\theta \end{bmatrix} \sim \mathcal{N}(0, M_t)$$
$$M_t = \begin{bmatrix} \sigma_D^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \tag{6.6}$$

ΔD_t is the step length of the mobile device and θ_t is the heading. The step length and heading of the mobile device can be estimated from its IMU sensors and are used as input to the filter. σ_D^2 and σ_θ^2 are the variance in the step length and heading respectively. The focus of our work is not on implementing an accurate step length and heading estimation method, so for our model we conservatively assumed that $2\sigma_D$ is $10cm$ and $2\sigma_\theta$ as $45°$ (For a normal distribution 95.45% of the values lie within 2σ of the mean)

The process model is given by Equation 6.7:

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} + \begin{bmatrix} (\Delta D_t + v_t^D)\cos(\theta_t + v_t^\theta) \\ (\Delta D_t + v_t^D)\sin(\theta_t + v_t^\theta) \end{bmatrix} \tag{6.7}$$

The process model is linearized and μ_t and Σ_t are updated as Equations 6.8 and 6.9 respectively:

$$\mu_t = g(\mu_{t-1}, u_t) \tag{6.8}$$

$$\Sigma_t = G_t \Sigma_{t-1} G_t^T + R_t \tag{6.9}$$

where

$$
\begin{aligned}
g(\mu_{t-1}, u_t) &= G_t \mu_{t-1} + \begin{bmatrix} \Delta D_t \cos(\theta_t) \\ \Delta D_t \sin(\theta_t) \end{bmatrix} \\
G_t &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
R_t &= V_t M_t V_t^T \\
V_t &= \frac{\partial g(\mu_{t-1}, u_t)}{\partial u_t} \\
V_t &= \begin{bmatrix} \cos(\theta_t) & -\Delta D_t \sin(\theta_t) \\ \sin(\theta_t) & \Delta D_t \cos(\theta_t) \end{bmatrix}
\end{aligned} \tag{6.10}
$$

6.5.2 Measurement Model

Though the actual measurements from our system are the TDOA values from the set of visible beacons, these can not be directly used with an EKF due to the linear approximation of the TDOA equations. Instead, we first estimate the position using the TDOA measurements, and use this estimate as our measurement. Our measurement model is given by Equations 6.11 and 6.12:

$$
Z_t = \begin{bmatrix} x_t \\ y_t \end{bmatrix} + w_t \tag{6.11}
$$

$$
w_t \sim \mathcal{N}(0, Q_t) \tag{6.12}
$$

where $Z_t = [\hat{x}_t, \hat{y}_t]^T$ is obtained by multilateration. From Figure 6.11, we observe that 90% of the range errors are less than $30cm$. We assume that the errors in the $\hat{x}_t$ and $\hat{y}_t$ are uncorrelated and assign $Q_t = \sigma_z I$ where $\sigma_z = 30cm$

In case one or more beacons are blocked, or if the phone identifies that one of the signals from the beacons is a NLOS signal, it does not update its measurement Z_t. In this case, we assign $Q_t = \sigma_n I$ where σ_n is a large number, such that the filtering effectively updates the estimate of the location based purely on tracking.

6.6 Rapid User-Assisted Setup

Any beacon-based localization system requires the location of the beacons with respect to the floor plan to provide meaningful location estimates. Most systems assume these beacon positions can be easily determined manually, but in practice this can be quite difficult. Errors in the position of the beacons can cause significant end-to-end localization errors. Measuring beacon positions is a labor-intensive time-consuming process which involves either taking extensive range measurements to walls using laser rangers or employing other equipment like a robotic system with accurate motion control equipped with the ability to sense the signal from the beacons. What makes this process difficult is that the floor plan information itself may not be provided to the installer. We propose a semi-automatic mapping process where the installer deploys the beacons and walks around the room taking a few measurements to aid the mapping process.

The goal of the proposed mapping process is to (a) map the beacons with respect to the floor plan, and (b) generate the floor plan using landmarks such as the corners if it is not already available. This process can be performed by a non-expert user in a few minutes for a single area.

6.6.1 Procedure

The process for mapping three beacons in a single area is given below. The approach can be extended to more beacons in a single area and conceptually also multiple areas. Though not currently implemented, the app could potentially take existing floor plan images and determine anchor points within them. Our mobile app guides the user through these steps:

1. Deploy the three beacons such that they provide good coverage of the area and are in LOS of each other.
2. Hold the phone close to one of the beacons and select the *Sync* option in the app and wait for 10 seconds while the phone synchronizes to the beacons.
3. Identify three points on the floor such that all three beacons are visible from each point. Place the phone at each location, and select the *Floor reference point* option.
4. If the floor plan is not provided, walk around the room and go to each corner and select the *Corner reference point* option. This will compute line segments between the corner points.
5. Specify an origin and the orientation of the $x - y$ coordinate space. One way to do this is to select one of the corners as the origin and an adjacent corner to be on the x or y axis.

6.6.2 Algorithm

The basic principle of the 3-D mapping process is that we make use of the following three types of information to uniquely solve for the beacon positions: (a) ultrasonic-based inter-beacon ranging (b) estimation of $z - plane$ using the three ground measurement points (c) user specified $x - y$ plane origin and orientation. The algorithm for mapping three beacons is as follows:

1. Given inter-node ranges r_{12}, r_{23}, r_{13} between the three beacons B_1, B_2, B_3, define a 3-D coordinate system $\mathbb{R}^3_a$ such that the three beacons are on the $z = 0$ plane, B_1 is the the origin $[0, 0, 0]$, and B_2 is along the x axis $[r_{12}, 0, 0]$. Coordinates of B_3 can be obtained as $[r_{13}\cos(\alpha), r_{13}\sin(\alpha), 0]$, where $\alpha = \arccos(\frac{r_{12}^2+r_{13}^2-r_{23}^2}{2r_{12}r_{13}})$
2. Estimate the coordinates of the three ground measurement points with respect to the beacons in $\mathbb{R}^3_a$.
3. Define a new coordinate system $\mathbb{R}^3_b$ such that the plane that contains the three ground points is the new $z = 0$ plane in $\mathbb{R}^3_b$.

4. The $x - y$ plane of $\mathbb{R}^3_b$ can be defined by its origin and one of the axes. This can be chosen arbitrarily since we would re-assign the $x - y$ plane after generating the floor plan. In our implementation, we did the following: The projection of B_1 on the $x - y$ plane is assigned as the origin $(0, 0, 0)$ of $\mathbb{R}^3_b$. The projection of B_2 on this plane is assigned to lie on the y-axis of the new plane. The x-axis of $\mathbb{R}^3_b$ is found to be normal to the y and z axes.

5. Estimate the location of all the corner points in $\mathbb{R}^3_b$ using trilateration.

6. The $x - y$ coordinates of the required 2-D coordinate system are specified by the user during the calibration process. Either apply an affine transformation on $\mathbb{R}^3_b$ to get the final coordinate system, or for better accuracy, apply non-linear transformations to minimize error across all reference points if more than two reference points are available.

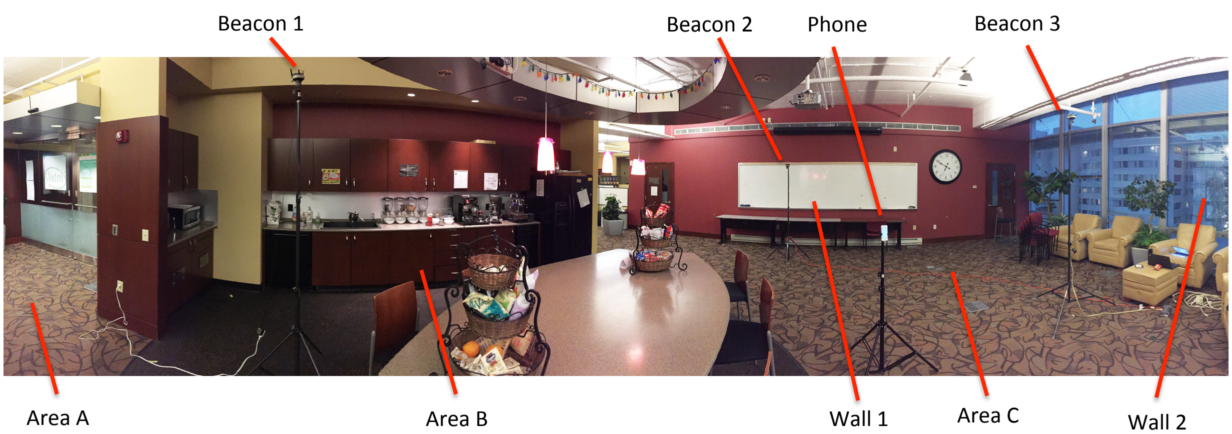
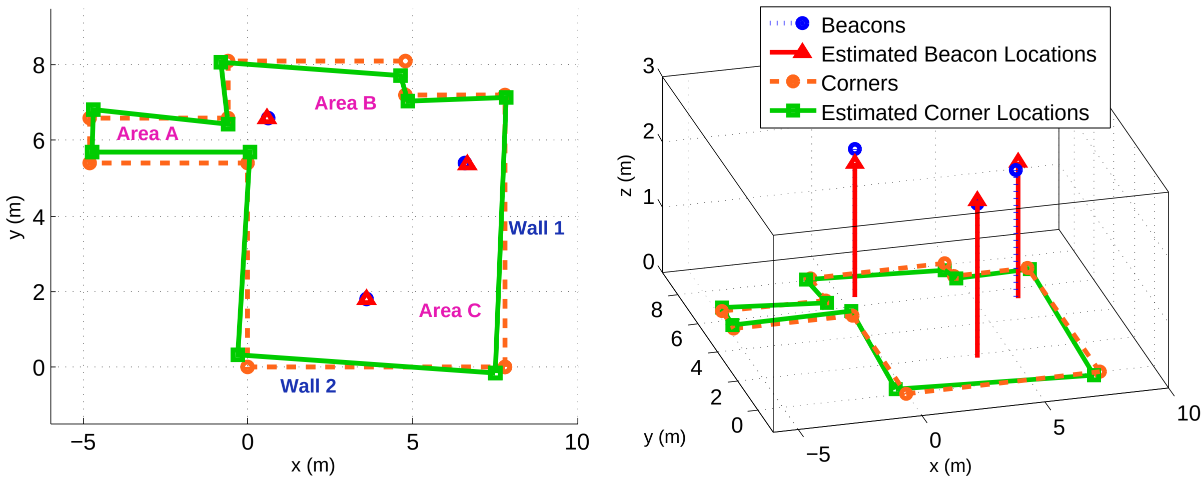


Figure 6.4: Panorama of automatically set up kitchen area using three beacons

6.6.3 Evaluation

We evaluated our mapping process in half a dozen areas: a kitchen and lounge space, a lab and in four office areas. The largest space in terms of area and number of corners was a lounge and kitchen space, as shown in Figure 6.4, with a total area of around $72m^2$. and 10 corners. The generated map is shown in Figure 6.5. A second generated map from the lab setup is shown in



(a) Overhead view

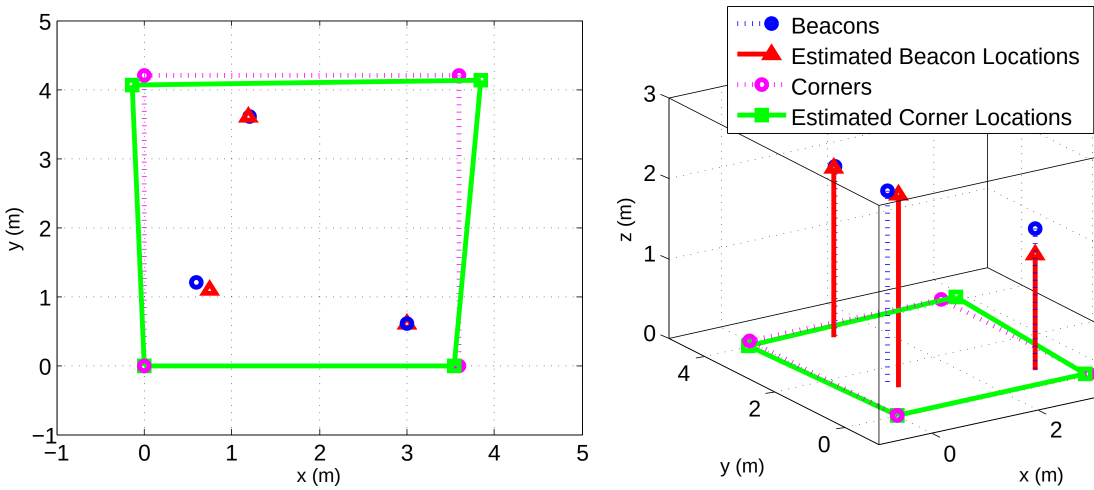
(b) 3D mapping result

Figure 6.5: Kitchen area beacon mapping output

Figure 6.6. Note that this process requires all the corners to be in LOS of the three beacons. Some of the boundaries in Figure 6.5 were not physical walls but were either $1.5m$ tall partitions or were chosen to ensure all corners are in LOS. The results of the mapping process for the kitchen setup and averaged across all six experimental setups are shown in Table 6.1. Our system can determine three-dimensional beacon location with a Euclidean distance error of $16.1cm$ averaged over the three beacons, and can generate maps with room measurements with a two-dimensional Euclidean distance error of $19.8cm$ averaged over all the corners. We observe that while mapping the beacons, the overall error in the height is around $13.5cm$, while the error in the x or y coordinate is less than $4cm$. This is because the heights of the beacons were within $1m$ of each other, whereas they were well separated in the $x - y$ plane.

	Beacon Error (cm)				**Corner Error (cm)**	
Setup	**Avg.**	**x**	**y**	**z**	**Avg.**	**Max**
Kitchen	13.9	2.2	1.4	13.4	26.8	43.6
Lab	18.2	5.4	3.6	13.6	13.0	25.2
Office 1	17.5	4.6	3.5	15.0	10.7	13.9
Office 2	17.2	5.0	1.6	15.1	22.8	34.0
Office 3	15.5	2.3	1.7	11.1	18.9	40.9
Office 4	14.1	3.4	3.1	12.9	26.5	31.4
Overall	16.1	3.8	2.5	13.5	19.8	43.6

Table 6.1: Beacon mapping error using user-assisted rapid setup



(a) Overhead view

(b) 3D mapping result

Figure 6.6: Lab area beacon mapping output

6.7 Localization Performance Evaluation

In this section we present the results of evaluating the localization performance of ALPS using a generation 1 system, with a stationary receiver and precisely measured beacon locations in Section 6.7.1 and with a generation 2c system using automatically mapped beacons and IMU sensor fusion in Section 6.7.2. The generation 1 system consisted of piezo bullet speaker beacons connected to a PC controlled audio interface via a home theater amplifier. The generation 2c system featured an embedded design with BLE time synchronization functionality and an omnidirectional ultrasound speaker horn for improved ultrasound signal coverage.

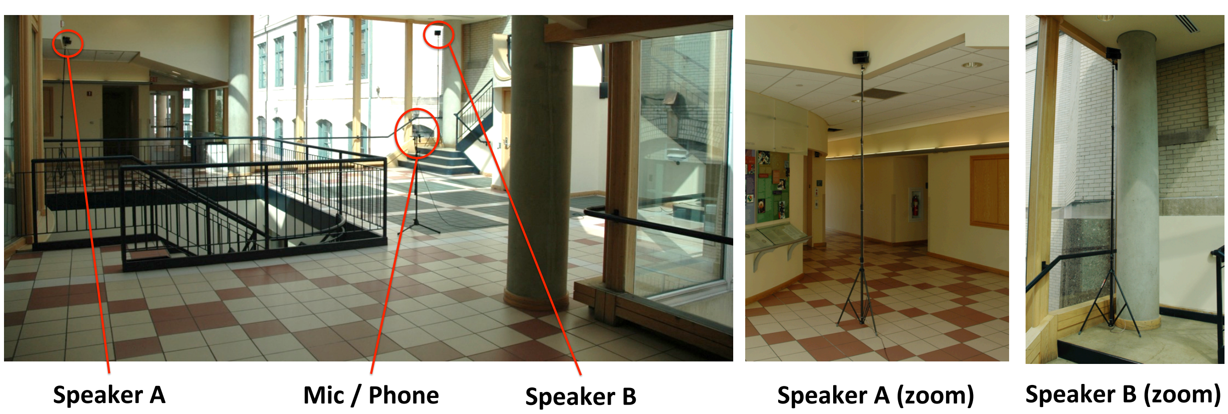
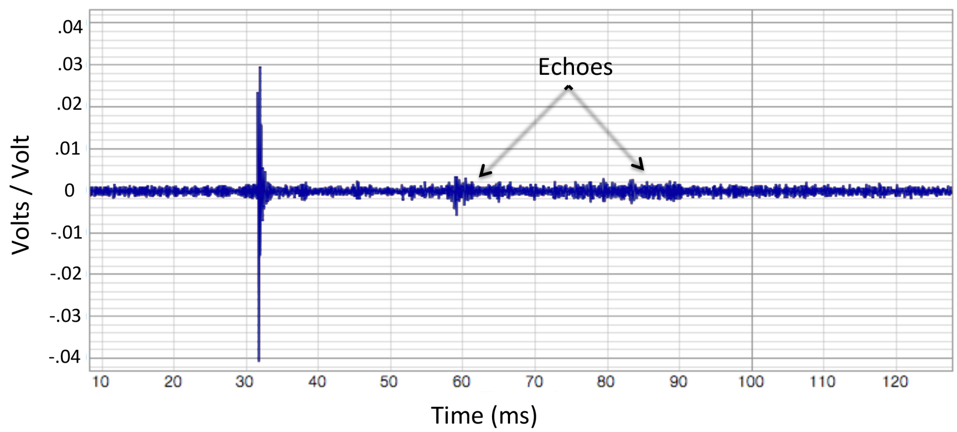
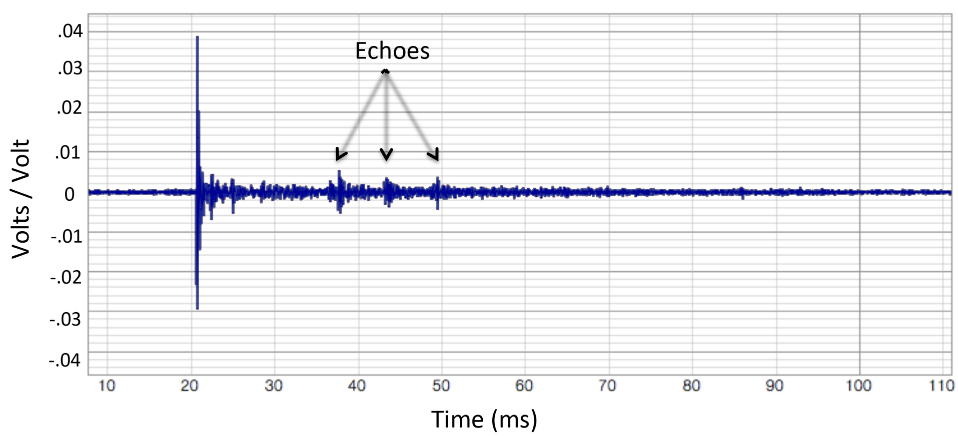


Figure 6.7: Photo of atrium environment (two speakers outside field of view)



(a) Atrium environment

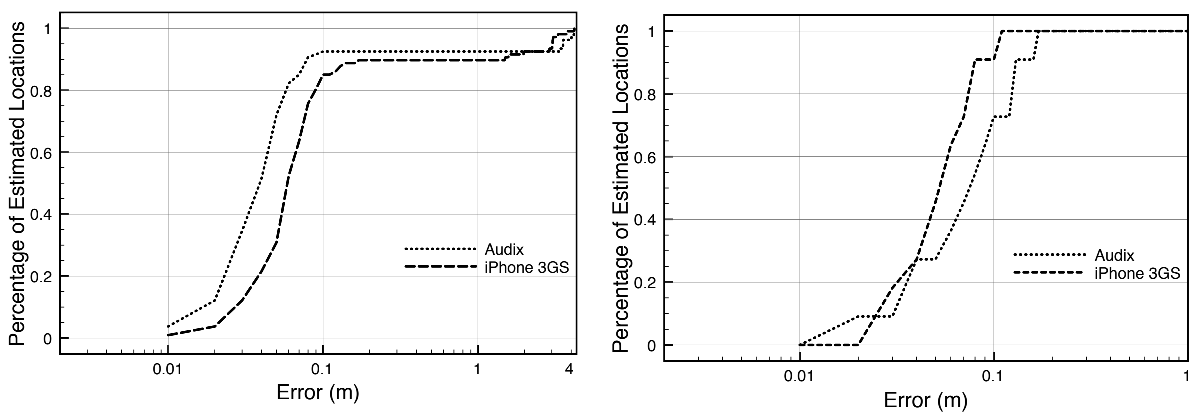


(b) Small room environment

Figure 6.8: Ultrasound impulse response measurements

6.7.1 Stationary Receiver

For this experiment we placed four beacons in the corners of two spaces on campus. The first location is the $20m$ x $20m$ atrium shown in Figure 6.7. We chose the atrium location since it was similar to that of a museum environment with hard walls and tile floors. We then also chose a small $5m$ x $5m$ room with cement walls that exhibits a large amount of multipath fading due to echoing. In order to capture the multipath characteristics of each of these spaces, we record the impulse response at the center of the room. Figure 6.8(a) shows the excess delay in the atrium to be about $90ms$, while Figure 6.8(b) shows an excess channel delay of about $60ms$. One can see the multipath is greater in the small room based on the number of echoes. One can also infer the size of the room by noting the spacing between echoes. We precisely measured the x,y, and z locations of the beacons' speakers using a laser range finder. Using our microphone stand, we moved an iPhone 3GS and an Audix TM-1 measurement microphone to 25 different locations along a grid in the room. At each location we took five audio recordings to compute five position samples. We then computed the ranging accuracy of the system by comparing our measured location as ground truth to each computed location. Figure 6.9(a) shows the distribution of errors across all of the samples.

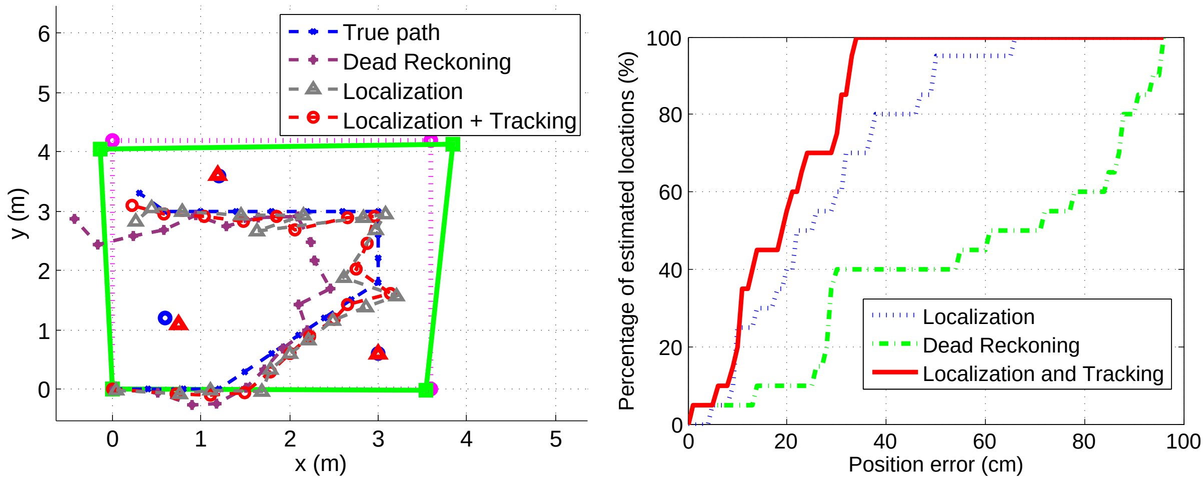


(a) Atrium environment

(b) Small room environment

Figure 6.9: Localization error

We see that 95% of the samples are within $100cm$ of the actual location with a worst-case overall error of $4m$. As shown in Figure 6.9(b), the small room behaves similarly except with a sharper position accuracy fall-off due to the added channel fading. In some cases, the system was only able to detect three out of the four beacons if for example an obstacle blocked one of the speakers. In these cases, our positioning algorithm estimated the 2-dimensional location using the 3 detectable speakers. The step-like shape of the CDFs is attributed to having successfully detected all four beacons in the vast majority of cases, but occasionally choosing the incorrect TOAs for one or more of them when they are blocked or out of range. This results in the vast majority of errors being below 10cm, with only a handful of much larger ones. In practice, adding additional beacons will help alleviate bind spots.



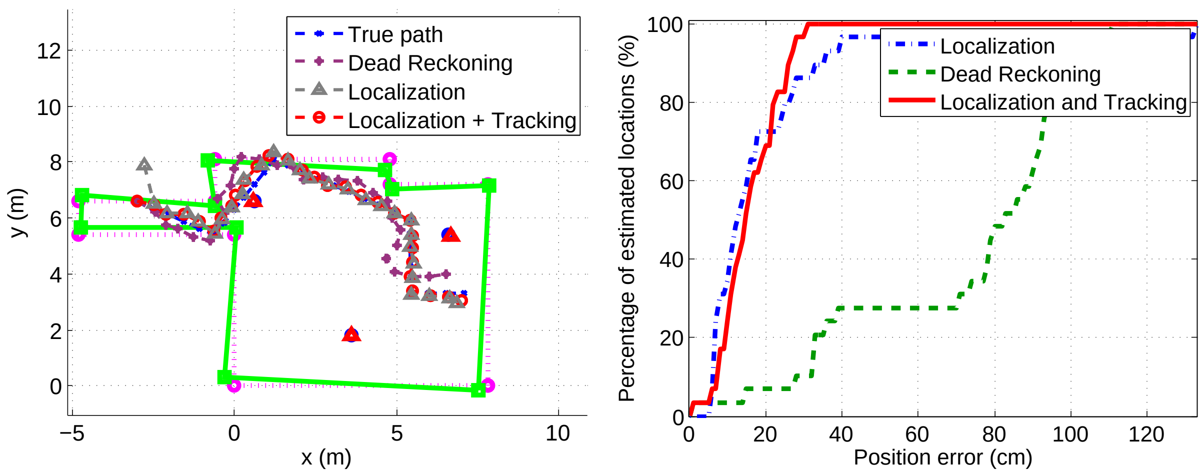
(a) Tracking Path

(b) Localization CDF

Figure 6.10: Localization performance in lab environment

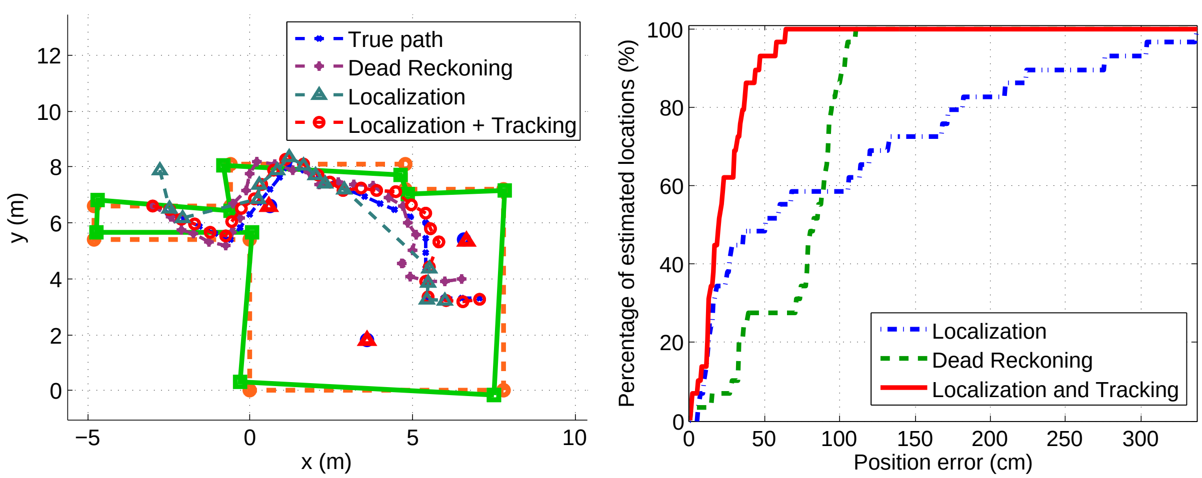
6.7.2 Mobile Receiver and IMU Sensor Fusion

In this section we present the results of evaluating the localization performance of ALPS, in the lab and kitchen environments described in Section 6.6.3. We used the maps that were generated by our rapid user-assisted setup process described in Section 6.6 using generation 2c ALPS beacons (see Section A.4). In each test a user held an iPhone 5S and took approximately 30 steps in the area at a regular walking speed of approximately $1.4m/s$. We collected data from the iOS's heading and pedometer functions. Ultrasonic measurements from the beacons were also collected at every step. We analyzed the data offline using MATLAB. The results from the lab setup are presented in Figure 6.10 and from the kitchen are presented in Figure 6.11. The *Localization* lines refer to position estimates based on only the ranges from the beacons, the *Dead Reckoning* lines refer to position estimates purely based on the IMU sensors and the motion model, the *Localization and Tracking* lines refer to the output of the EKF explained in Section 6.5 and the *True Path* lines denote the ground truth path taken by the person holding the phone. Figure 6.10(b) and Figure 6.11(b) show that tracking does not improve the accuracy significantly as compared to using only localization since the localization is much more accurate than the estimates from the motion model (error less than $30cm$ 90% of the time for the kitchen setup and $47cm$ 90% of the time for the lab setup).



(a) Tracking Path (b) Localization CDF

Figure 6.11: Localization performance in kitchen environment with no blocked beacons



(a) Tracking Path (b) Localization CDF

Figure 6.12: Localization performance in kitchen environment with one blocked beacon

Note that the localization results without tracking are also affected by the errors in determining the beacon's positions using the user-assisted setup procedure (see Table 6.1), as well as the motion of the person holding the phone. We then simulated situations when the user blocks one beacon by removing some of the range measurements from a beacon in the data-set. The *Localization* line in Figure 6.12 shows the localization estimates under this case for the kitchen setup. The location does not update when insufficient measurements are received. We observe that in such cases the system benefits from tracking, as seen in the *Localization+Tracking* line. As can be seen in the CDF in Figure 6.12(b), tracking improved the localization performance from $250cm$ to $50cm$ for the 90^{th} percentile in position error.

Chapter 7

Conclusions

This dissertation examined the use of synchronized ultrasonic signals for precise ranging and localization of mobile devices. ALPS is an ultrasound based indoor localization system for off-the-shelf mobile devices that provides high localization precision and is both scalable and cheap to deploy. It is a beacon based system which uses custom embedded beacons, usually mounted to ceilings, to transmit time synchronized ultrasonic signals to mobile device receivers. The devices receive the ultrasound signals along with timing information via BLE, which allows them to time synchronize with the ultrasound transmission cycle on a time slot level. ALPS employs an ultrasound modulation scheme which allows for precise ranging and is inaudible to humans. A rapid user-assisted setup procedure makes the system easy to deploy and energy harvesting beacons avoid high wiring costs. This dissertation makes the following contributions:

1. **Ultrasound Modulation Scheme**: We designed and implemented two ultrasound modulation schemes that provide precise ranging information to mobile devices. Both schemes allow for multiple access via time multiplexing and are imperceptible by humans. One scheme allows for encoding beacon IDs onto the ultrasound carrier. The demodulator implementation of these schemes filters some multipath interference, which is then further mitigated by a machine learning based approach that exploits the difference in propagation

characteristics between ultrasound and BLE signals for detecting multipath.

2. **Time Synchronization**: We implemented a clock recovery scheme via TDOA based localization on iOS devices that allows them to tightly time synchronize with the ultrasound transmission cycle. This allows for TOF ranging between beacons and mobile devices. We implemented time synchronization protocols for BLE, 802.15.4 and LoRaWAN links for time synchronizing the beacons, plug forwarders and mobile devices to a central network master node.

3. **Beacon and Networking Hardware**: We developed several generations of ultrasonic beacon hardware, which culminated in an energy harvesting beacon design with a switchable beam speaker array as well as a Decawave UWB module for ranging and BLE and 802.15.4 connectivity for data transfer and time synchronization. We evaluated the directionality of several ultrasound speakers and designed an omnidirectional ultrasound speaker horn and a four speaker array for optimal signal coverage. We also designed two networking nodes that allow data transfer and time synchronization with the beacons. The latest network master and plug forwarder networking nodes also provide LoRaWAN connectivity for creating long range networks in large installations.

4. **Location Engine**: We implemented a cloud based location engine which performs the localization computations for the mobile devices. It is able to interface with MQTT and XMPP publish-subscribe protocols for the distribution of location data. We also implemented an IMU sensor fusion based tracking system which improves the ultrasound location accuracy and fills in coverage gaps. Our rapid user-assisted setup procedure simplifies the system's setup by semi-automatically calculating the beacon locations in 3D space.

7.1 Future Work

This section will explore the possibilities for future work on ALPS. Section 7.1.1 covers the possibilities of tag localization and tracking, which would expand the localization capabilities of an ALPS installation to low power tags that are valuable for asset tracking applications. Section 7.1.2 describes the potential for AOA measurements using the sectored speaker array of ALPS beacons for detecting localization errors, reducing the amount of computation for solving for a location by constraining the search space and potential localization using a single beacon. Section 7.1.3 looks at future advanced automatic beacon localization methods, which localize beacons across large installations. Finally, Section 7.1.4 describes improvements to our localization engine by accounting for floor plan data.

7.1.1 Tag Localization and Tracking

The primary use-case for ALPS is the localization of smartphones and tablets, however, there are several applications that would benefit from integrating custom localization hardware such as small tags. For example, asset tracking, worker safety and VR/AR applications often already use custom hardware into which a tag could easily be integrated. Current ALPS beacons have the capability of receiving and sending signals to BLE tags such as iBeacon [20], Gimbal [15] and Eddystone [12], as well as Decawave [9] UWB equipped hardware. While BLE tags don't provide anywhere near the localization precision of ultrasound based localization, they are very cheap, have a low power consumption and can give a general idea as to where the tag is located. UWB tags such as Estimote UWB tag like devices [13] can provide high accuracy and high update rates at a higher cost than BLE tags. ALPS beacons are capable of periodically pinging tags and sending back ranging data to a location server via the 802.15.4 and LoRaWAN backhaul described in Section 3.1.

ALPS meets both the hardware and networking infrastructure requirements to localize and track tags. We envision adding the necessary software for this in the future to make ALPS a multi-technology localization system.

7.1.2 Angle of Arrival (AOA) Measurement

Current ALPS beacons feature an array of four individually addressable speakers with a $90°$ angular separation in the horizontal plane (see Section 3.2). By transmitting uniquely identifiable ultrasound signals using the modulation scheme described in Section 4.5.6, it would be possible to perform AOA measurements at the receiving mobile device with a single microphone. Splitting the beam pattern of the beacon (see Figure 3.11) up into four sectors of $90°$, the sector which the receiving device is located in should be identifiable based on the RSSI of the received ultrasound signals. It may be possible to obtain a more precise AOA measurement based upon the RSSI ratio of multiple ultrasound signals received from different speakers of the same beacon and comparing them to the known beam pattern. This would be challenging because the ratios of the RSSI values change based upon the angle the receiving device has in relation to the vertical plane of the beacon and because RSSI is a notoriously noisy metric.

AOA measurement data could significantly benefit the location solving process in ALPS (see Section 6) by confining the area to solve across based upon the sector the mobile device is located in, hence greatly reducing the amount of computation required compared to solving over the entire space within range of the beacons. This may also be used to reduce errors caused by inaccurate range measurements by checking if the calculated position of the mobile device lies within the region measured by the AOAs of all received ultrasound signals.

Additionally AOA measurements may be used for localization from a single beacon. Once a mobile device is synchronized with the network master using the scheme described in Section 5.4.1 and can perform TOF ranging, the range data can be combined with the AOA data to localize the device to a higher precision. This would be particularly beneficial to smaller rooms where it may be cost prohibitive to install multiple beacons and where the localization error using this method is more tightly bounded.

7.1.3 Automatic Beacon Setup

One of the largest costs of installing a beacon based localization system like ALPS is the cost of surveying the location of the beacons. In Section 6.6 we presented a rapid, user-assisted setup technique for a three beacon system, which calculates the location of the beacons to a high accuracy. This method, however, does not scale well to large installations.

A better approach would be to construct a range graph by utilizing the Decawave UWB ranging modules in current ALPS beacons (Section A.5.1) to perform longer range inter-beacon ranging, rather than by transmitting ultrasound between beacons. Obtaining the geometry of the vertices of the graph based on the edges (distances) between them is a well researched problem in the sensor networking, robotics and math communities [41, 48, 54] known as graph realization. The main challenge in applying this to an ALPS installation is that the graph would often not be fully connected and may even have disjoint vertices. It may be possible for a user to carry around a mobile node with UWB ranging hardware, which can be used to add additional temporary vertices to the graph to obtain a higher level of connectivity. A Simultaneous Localization and Mapping (SLAM) based approach such as Range-Only SLAM [60, 80] or Graph SLAM [100] may also be applicable.

7.1.4 Advanced Location Engine

The current ALPS location engine (see Section 6) only utilizes floor plan data to determine whether a location measurement is valid based on if it lies within the transmission range of the beacons heard by the mobile device. Floor plan information can, however, be used in several more ways to aid localization.

When combined with a ray-tracer, previous location data and IMU sensor fusion, it may be used to detect NLOS signals based on their ranges to the estimated location. NLOS signals may then be discarded and the location can be recomputed using only LOS signals, leading to a better location estimate.

Buildings contain several areas where only one dimensional localization is necessary such as hallways. A more advanced solver could automatically detect these areas based on the floor plan and require less beacons to be placed in these areas.

Floor plan data may also be used to determine the best locations for beacons. [89] describes an automated beacon placement algorithm, which minimizes both the amount of beacons necessary to cover indoor areas and also the resulting GDOP.

The location engine also needs to manage floor plan and beacon location data in a scalable way to support simultaneously solving requests from multiple large installations. We envision using the MQTT based OpenChirp [28] management system for easily storing and retrieving this data. OpenChirp's publish-subscribe architecture lends itself well for tracking applications, where asynchronous multicasting of location data is necessary. Furthermore it provides a simple MQTT and REST API, as well as a flexible web interface for easy app integration and system management.

7.1.5 Augmented/Virtual Reality Integration

With Virtual Reality (VR) [6, 14, 27, 33] and Augmented Reality (AR) [16, 18] systems becoming incredibly popular, there are many applications that would benefit from combining them with highly accurate (sub-meter) indoor localization, for example in the retail and facility maintenance spaces. iOS and Android smartphones already feature software frameworks [2, 8] for VR and AR applications, which may be used in conjunction with ALPS today. We believe that even next generation indoor localization technologies for smartphones like 802.11mc [36], with an anticipated indoor localization accuracy of $1 - 3m$, will likely not be accurate enough for many of these applications like AR based store product finders and machine control/maintenance apps. We will be closely following this space in the future.

Appendix A

ALPS Platform Evolution

This appendix presents the evolution of the ALPS platform over three major and several minor generations. It identifies key features in every generation and describes the hardware, firmware and software architecture. An overview of all platform generations and their features is presented in Table A.1.

A.1 Generation 1

A.1.1 Hardware

The first generation of the ALPS platform served mainly as a proof-of-concept for indoor localization using ultrasound based TDOA pseudo-ranging for mobile devices [66]. The beacons consisted of piezo electric tweeters that were hard wired to multichannel audio amplifiers connected to an audio interface, which generated the ultrasound ranging signals as seen in Figure A.1. In order to create high-quality ranging signals we chose to use a Motu UltraLite-mk3 audio interface. The mk3 provides both a 24bit $192kHz$ ADC and DAC with up to 10 channels of analog output. We connected the mk3 to two Onkyo HT-R540 amplifiers via standard unbalanced analog audio

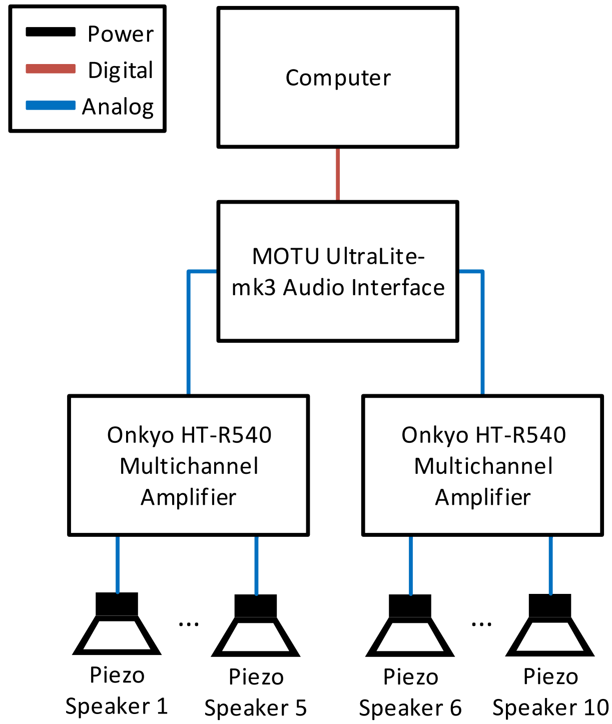


Figure A.1: Generation 1 ALPS beacon architecture

connections. Each Onkyo amplifier provides 7 channels of amplification, so we require two of them to utilize the entire 10 channels of output from the mk3. The HT-R540 has an extended frequency response mode that remains relatively flat up to $100kHz$. Finally, we connect each output channel from the Onkyo to 10 Goldwood GT-1016 Dispersion Piezo Horn Tweeters. These are low-cost ($<$ \$2 each) tweeters that have a frequency response of up to $27kHz$ and were mounted to tripod stands for optimal placement. A photo of the described hardware setup can be seen in Figure A.2.

A.1.2 Software

In generation 1 signal generation, synchronization and playback were handled by a computer running MATLAB, which was connected directly to the audio interface. Reception was performed in iOS on an iPhone 3GS and iPhone 4, after which the audio recording was transferred back to the computer over a WiFi or cellular data collection. The demodulator running in MATLAB

Figure A.2: Generation 1 ALPS Beacon hardware

would then process the recording and the resulting TOA values and beacon IDs were processed by a location solver also running in MATLAB to solve for the location of the smartphones.

A.1.3 Generation Summary

This setup was able to prove the feasibility of transmitting time synchronized ultrasound ranging signals, which were inaudible to humans, to localize iPhone 3GS and iPhone 4 smartphones. It was also used as a testbed to validate the first ultrasound modulation scheme (see Section 4.4) and to determine how to design a signal which is inaudible to humans (see Section 4.3.1). The installation difficulty of the first generation system was high due to its hardwired beacons and need for a computer to drive the audio interface, which called for the development of an embedded platform starting in generation 2a (see Section A.2). During testing the speakers were also found to exhibit directional beam patterns, which limited the ultrasound coverage of each beacon.

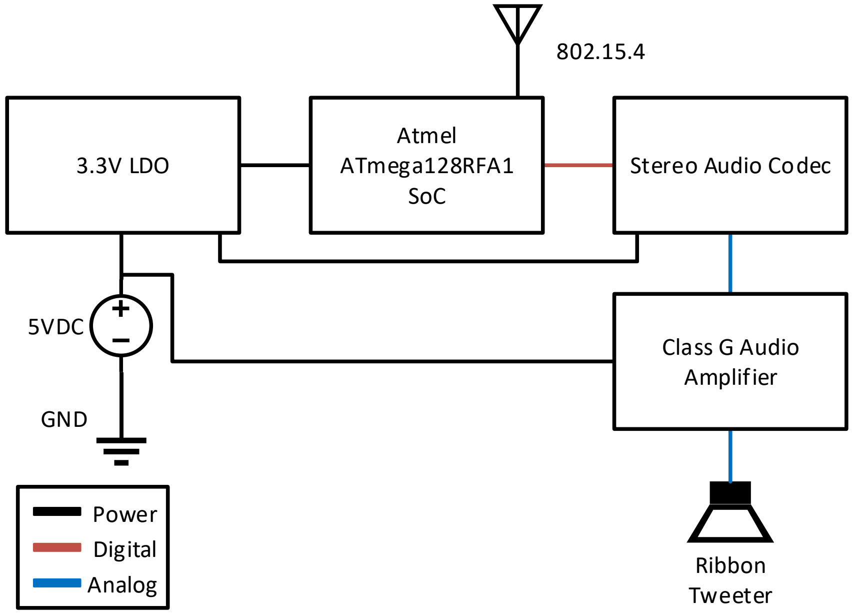
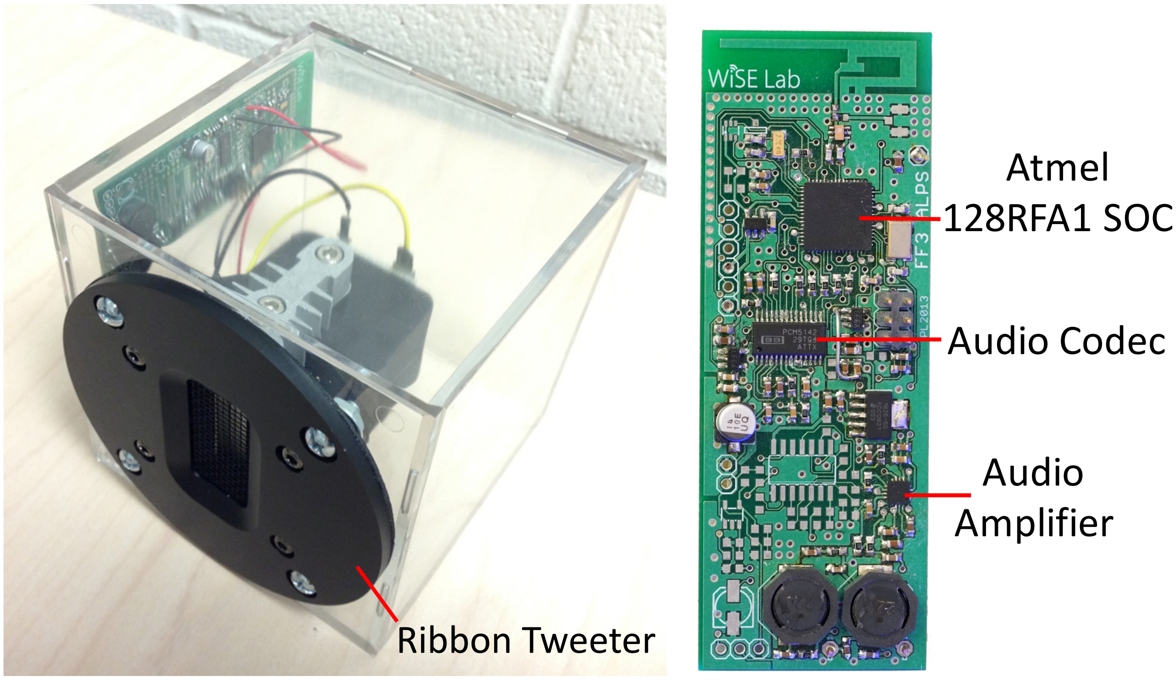


Figure A.3: Generation 2a ALPS beacon architecture

A.2 Generation 2a

A.2.1 Hardware

2a was the first generation to feature embedded beacons (Figure A.4(a)) and a network master node (Figure A.6) to provide wireless time synchronization via 802.15.4. The beacons used an 8-bit Atmel ATMega128RFA1 SOC with an on-board 802.15.4 radio as can be seen in Figure A.3 and Figure A.4(b). The SOC stored the samples of the externally generated ultrasound waveforms in internal flash at a $125kHz$ sampling rate and output them via I^{2}S to an audio codec. Since the ATMega128RFA1 does not have an I^{2}S port, the protocol was implemented using the SPI port and a timer for the word clock. The analog waveform produced by the audio codec was piped into a class G (similar to class AB, but with optimized power consumption) mono audio amplifier and then output to a ribbon tweeter. Although expensive (approximately $\$70$), the ribbon tweeter provided an omnidirectional beam pattern in the horizontal plane as seen in Figure 3.9(e) and therefore allowed for greater signal coverage although the vertical beam pattern



(a) Beacon (b) Beacon PCB

Figure A.4: Generation 2a ALPS beaon hardware

was very directional as seen in Figure 3.9(f). Time synchronization was handled via 802.15.4 packets sent from a network master (Figure A.6) at regular intervals. An interrupt routine which timestamped incoming 802.15.4 packets was sufficient to provide time synchronization on a 10s of microseconds level. The beacons were powered by external $5V$ power supplies and housed in an off-the-shelf cube enclosure.

For the network master (Figure A.6), Carnegie Mellon University's Drone RK Hardware Module [10] was used. The module uses the same SOC as the generation 2a beacons and adds an RF power amplifier for extended 802.15.4 coverage as well as a USB to serial converter (see Figure A.5) to connect to a computer for issuing commands to control an ALPS deployment.

A.2.2 Software

The firmware of generation 2a and 2b beacons was coded in C and ran bare-metal on the Atmel SOCs. Its functionality was simple: It listened on the radio for a time synchronization packet,

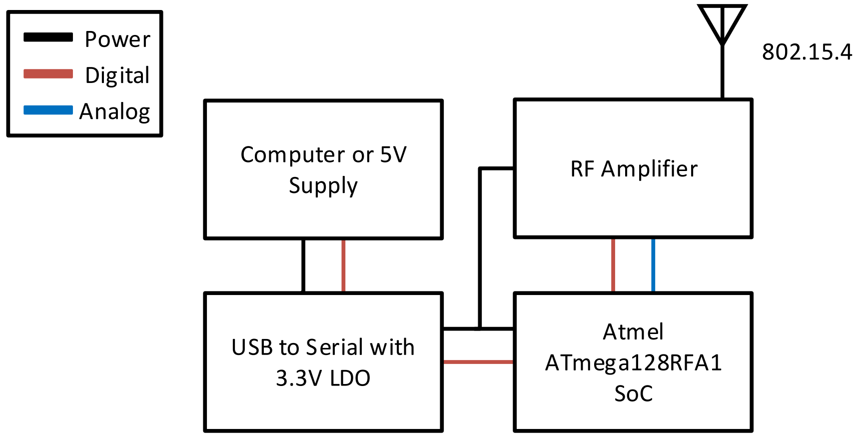


Figure A.5: Generation 2a, 2b, 2c, 3a ALPS network master architecture

on reception waited for its TDMA slot and then started loading the ultrasonic signal data from internal flash onto its emulated I^2S bus and sending it to the audio codec for playback. After this the SOC and audio codec would be put into sleep mode until shortly before the next RF message was expected and then began listening on the radio again. If an RF message was missed, the beacon would continue listening until it received the next time synchronization message. Since the Atmel SOCs lacked a DMA controller, no RF commands could be received while the ultrasound transmission was in progress since the CPU was fully utilized by that transaction. The beacons could also receive other commands via RF for setting their volume and setting the TDMA cycle time.

The firmware of the network master was responsible for transmitting periodic RF packets to the beacons to time synchronize their ultrasound transmissions. This was implemented using a simple timer interrupt routine, which transmitted the RF packet when a timer expired after the set TDMA cycle time. The network master featured a serial terminal that could be opened in a terminal application on a computer it was plugged into. This allowed the user to issue commands to start/stop ultrasound transmissions and to control the beacons (set volume and TDMA cycle time).

The software running on the phone consisted of an audio recording routine which ran con-

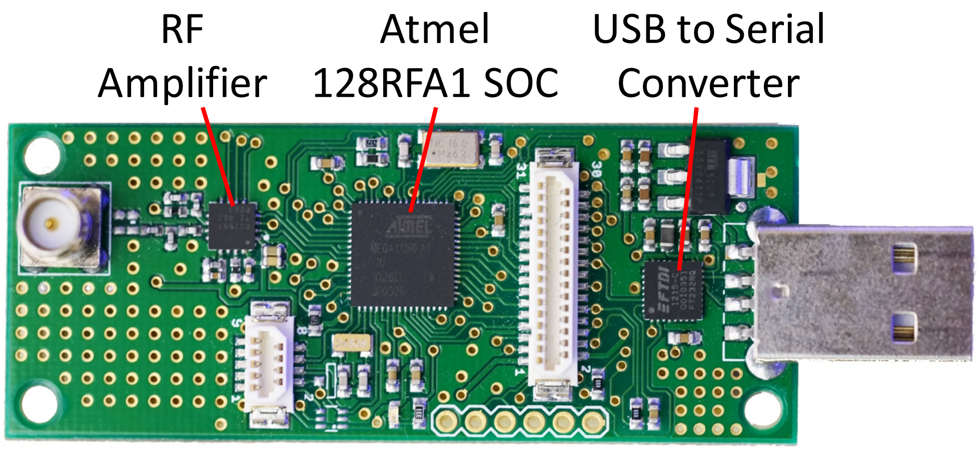


Figure A.6: Generation 2a, 2b, 2c, 3a ALPS network master hardware

tinuously and would fire a callback when an audio buffer was filled. This buffer was then passed to the demodulator (see Section 4.4.3), which would extract the TOA values of the received ultrasound signals and the beacon IDs that were encoded. These values would then be sent to a location solver running in MATLAB on an external computer over the WiFI or cellular data connection in order to solve for the smartphone's location. The demodulator was written in MATLAB and then coded into C using MATLAB's C coder.

A.2.3 Summary and Possible Improvements

This generation validated the use of embedded components and wireless time synchronization for ALPS. The main advantages over the generation 1 system were the significantly simplified beacon deployment since no wires needed to be run from central audio amplifiers to the beacons, although the beacons still needed to be powered by external power supplies, which could be plugged into nearby outlets. The coverage of the beacons was also improved due to the more omnidirectional beam pattern of the ribbon speakers, at the expense of a much higher cost compared to the previous piezo speakers. This generation was placed $4th$ in the Infrastrucure Based Localization category of the 2014 Microsoft Indoor Localization competition [22].

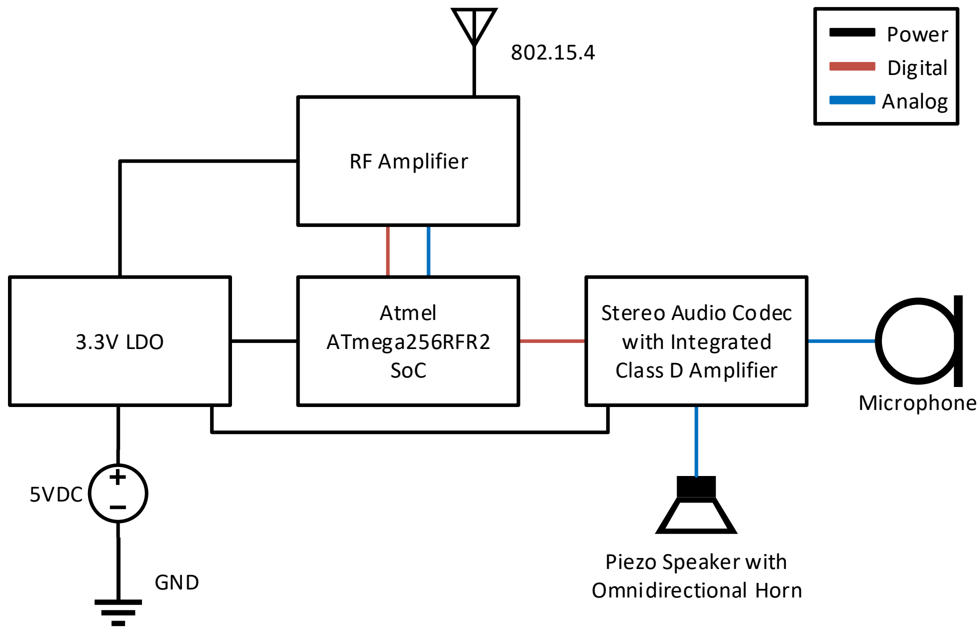
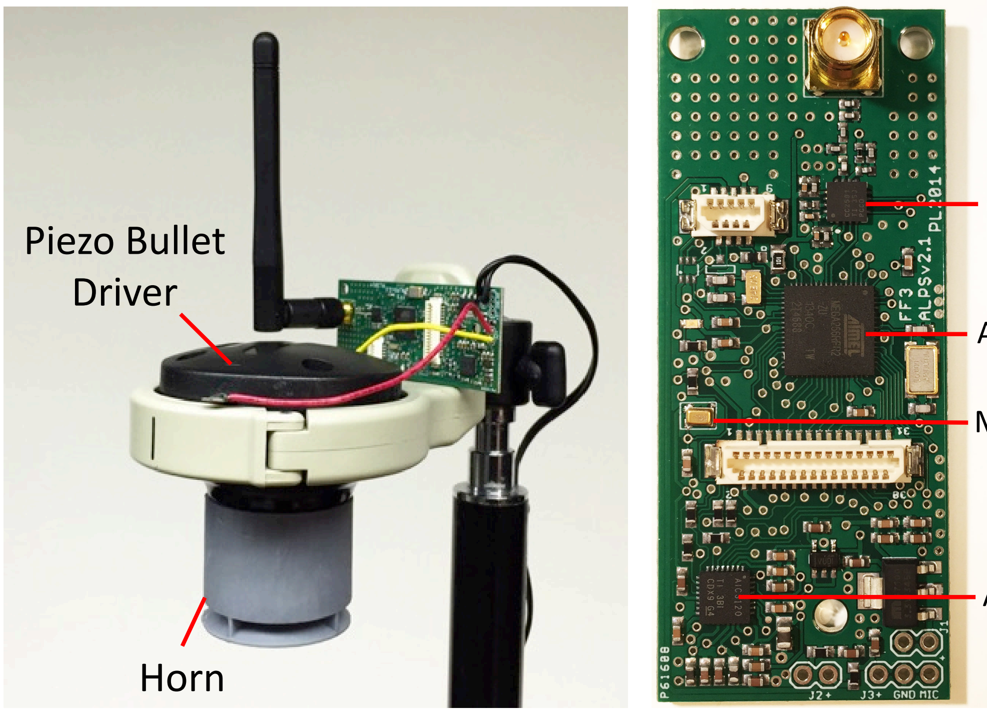


Figure A.7: Generation 2b ALPS beacon architecture

A.3 Generation 2b

A.3.1 Hardware

Generation 2b was the first minor generation increment to the ALPS platform and was focused on the beacon hardware (Figure A.7, Figure A.8(a) and Figure A.8(b)). The new design had improved RF coverage, ultrasound coverage and memory capacity compared to generation 2a. It also featured an integrated audio codec with audio amplifier and a microphone for beacon-to-beacon ranging. The ATMega128RFA1 SOC was replaced by an ATMega256RFR2 chip, which is largely identical, but has twice the flash memory capacity to store longer or multiple ultrasound signals. An RF amplifier was added to the beacons to increase the RF coverage and a Class D audio amplifier integrated into the audio codec was employed to lower the power consumption compared to the Class G amplifier. We developed an omnidirectional ultrasonic speaker horn (see Section 3.7.2), which allowed a cheap piezo bullet speaker (similar to that used in generation 1) to provide omnidirectional ultrasound coverage at a significantly reduced cost to the previously



(a) Beacon (b) Beacon PCB

Figure A.8: Generation 2b ALPS beacon hardware

used ribbon speaker. The beacon now also was tripod mountable, similar to the generation 1 beacons. A MEMS microphone was hooked up to the audio codec to enable beacon-to-beacon ranging for the automatic localization of beacons once they are deployed (see Section 6.6). This generation used the same network master node as generation 2a.

A.3.2 Software

The firmware running on the beacons and the network masters, and the demodulation software on the mobile devices was largely identical to that of generation 2a beacons described in Section A.2.2.

A.3.3 Summary and Possible Improvements

With increased RF and ultrasound coverage, larger deployments were now possible. Multiple deployments were performed using this generation and we determined that powering the beacons via an external power supply drastically limited installation options or made them difficult and costly since the beacons are usually mounted on or close to ceilings, where no electrical outlets are available.

A.4 Generation 2c

A.4.1 Hardware

Generation 2c was introduced in [67] and used mostly the same beacon hardware as 2b (Section A.3), but added support for battery power, a custom 3D printed enclosure (Figure A.10(a)) and BLE via a daughter board (Figure A.7 and Figure A.10(b)) for coarse time synchronization with the mobile device receivers (see Section 5.5). Although the beacons were not power efficient enough for long-term battery power deployments, the installation for testbeds and demonstrations was greatly simplified by not requiring an external power supply. The custom enclosure allowed the beacons to be mounted on tripods, on ceilings and above tile ceilings with the horn sticking through a ceiling tile. The BLE daughter board allowed for coarse time synchronization with the mobile devices so that they could map received ultrasound transmissions to their respective beacon in a TDMA cycle. This eliminated the need for data transmission via ultrasound, which had become difficult due to the reduced ultrasonic bandwidth of newer iPhone models as shown in Figure 3.13.

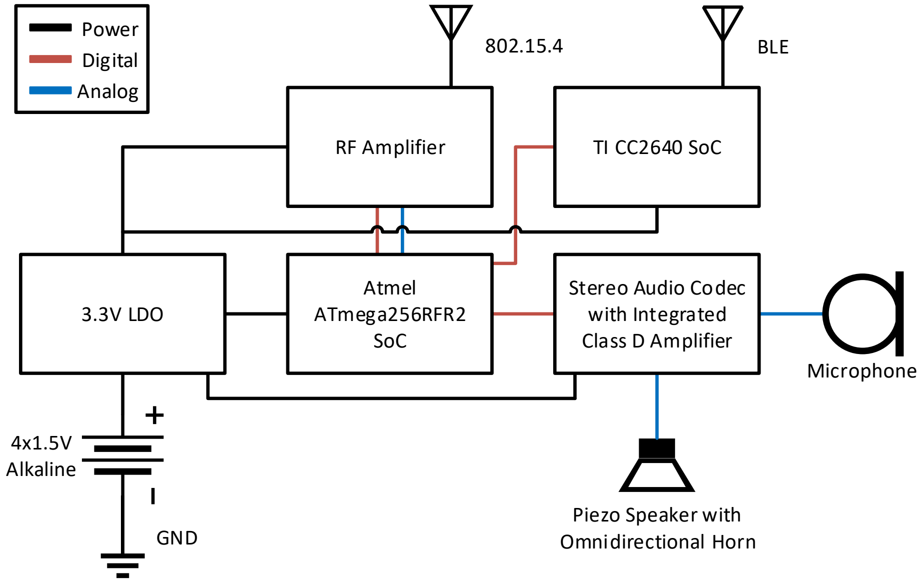


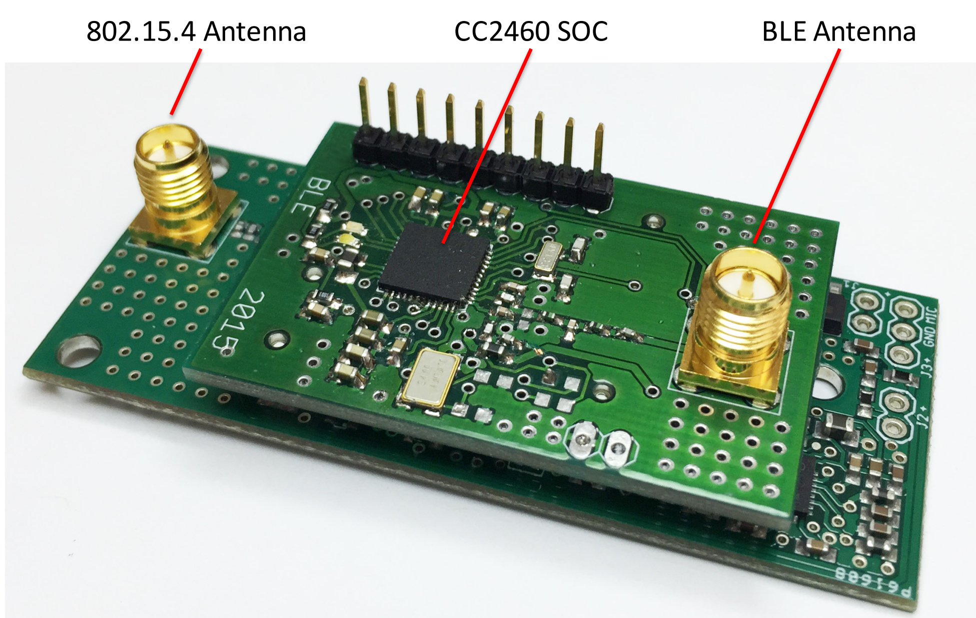
Figure A.9: Generation 2c ALPS beacon architecture

A.4.2 Software

The firmware running on the Atmel SOC was very similar to that running in generations 2a and 2b (see Section A.2.2), with the addition of a GPIO line triggering the CC2640 SOC at the start of every TDMA cycle to provide coarse time synchronization to the mobile device receivers via BLE. The CC2640 ran TI's BLE stack [5], which is a full BLE stack running on TI's RTOS TI-RTOS [31]. TI-RTOS is a preemptive multitasking microkernel based RTOS which also provides device drivers for most peripherals on the CC2640. In generation 2c, the only task of the CC2640 was to transmit BLE advertisement packets containing the time elapsed since the start of the current TDMA cycle at a $20ms$ interval (lowest possible according to the BLE4 specification). See Section 5.5 for more details about this time synchronization method.

The application running on the mobile devices had to be updated starting in this generation to incorporate the BLE time synchronization. The BLE advertisement packets were sent in an iBeacon compatible format, which can be continuously received by iOS. Additionally, software for performing the clock recovery method described in Section 5.4 to subsequently perform TOF ranging was implemented.

(a) Beacon



(b) Beacon PCB

Figure A.10: Generation 2c ALPS beacon hardware

A.4.3 Summary and Possible Improvements

This generation first introduced and validated slot-level time synchronization via BLE between the beacon infrastructure and the mobile devices. This eliminated the previous requirement of encoding transmitter IDs into the ultrasound signal and hence simplified the demodulator and reduced its power consumption on the mobile devices. Power consumption on this generation was still fairly high and it could not be powered off of batteries for long-term installations. The addition of the BLE daughter board added complexity and increased power requirements. With the introduction of the multi-mode 802.15.4 and BLE CC2650 SOC, the Atmel SOC and the CC2640 would be replaced by a single CC2650 SOC in future generations. This generation placed 1st in Infrastructure Based Localization category in the 2015 Microsoft Indoor Localization Competition [23].

A.5 Generation 3a

A.5.1 Hardware

Generation 3a marks a major change to the ALPS beacons, while still being compatible with the network master from the previous generations (Figure A.6). Generation 3a beacons featured an energy harvesting, embedded hardware platform as can be seen in Figure A.11 and Figure A.12(b). The platform was designed to have a low enough power consumption so that it can be powered using a small solar cell, harvesting energy from artificial or natural light sources (see Section 3.8). This allows for a flexible installation at a low cost, since the beacons do not need to be connected to AC wall power, which is often difficult to access at ceiling mounting locations. The harvested energy is buffered in three ultra low self discharge NiMH batteries with $2000mAh$ each, which have a high cycle lifetime of 2000 cycles and retain 70% of their charge after ten years.

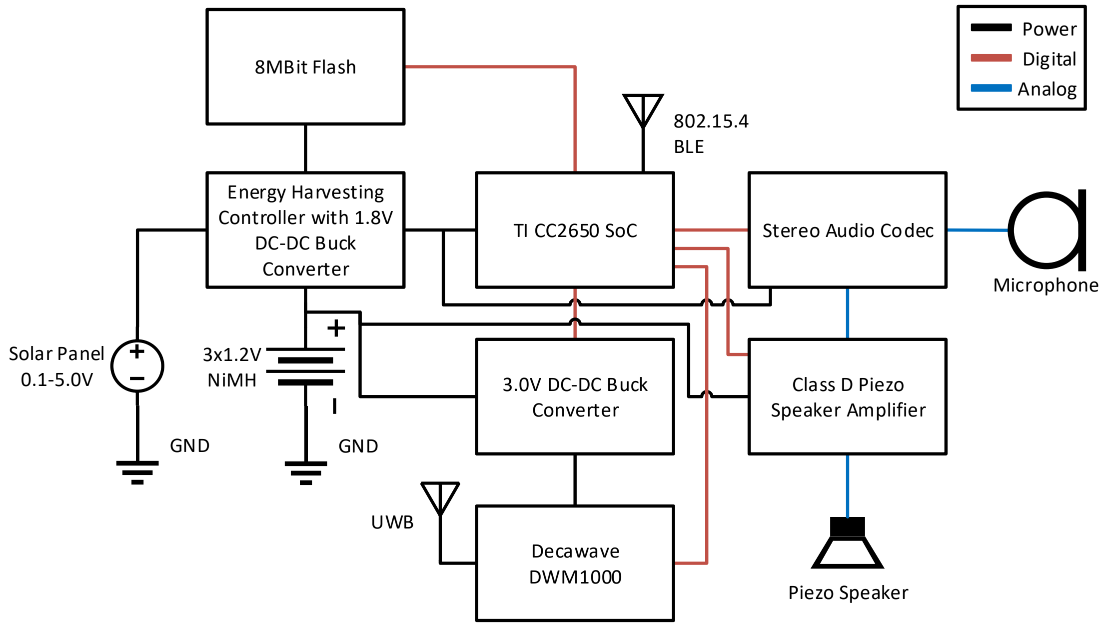


Figure A.11: Generation 3a ALPS beacon architecture

The beacons featured a single PCB design, which uses a TI CC2650 multi-standard BLE and 802.15.4 SOC with a 32-bit ARM Cortex M3 core connected to a $192kHz$ audio codec (running at $48kHz$), a MEMS microphone expansion port and a low cost piezo ultrasound speaker ($<$ $\$1$), connected to a Class D piezo speaker amplifier to receive and transmit ultrasound signals respectively. The 802.15.4 radio mode is used for time synchronization and communication with the network master, while the BLE radio mode is used to time synchronize the mobile devices coarsely to the ultrasound transmission cycle. The piezo speaker amplifier contains an on-board DC-DC boost converter which supplies more voltage than in previous generations to better drive the piezo speaker, which improves the range of the system and also features a more advanced, lower power modulation scheme that increases the transmission efficiency. An 8MBit flash chip stores firmware updates as well as configuration settings. The hardware is housed in an off-the-shelf enclosure (Figure A.12(a)) that is tripod mountable with custom cutouts and an integrated battery compartment. The beacons also feature a Decawave DWM1000 UWB ranging module (see Section 2.4.4) for inter-beacon ranging and future tag tracking.

A.5.2 Software

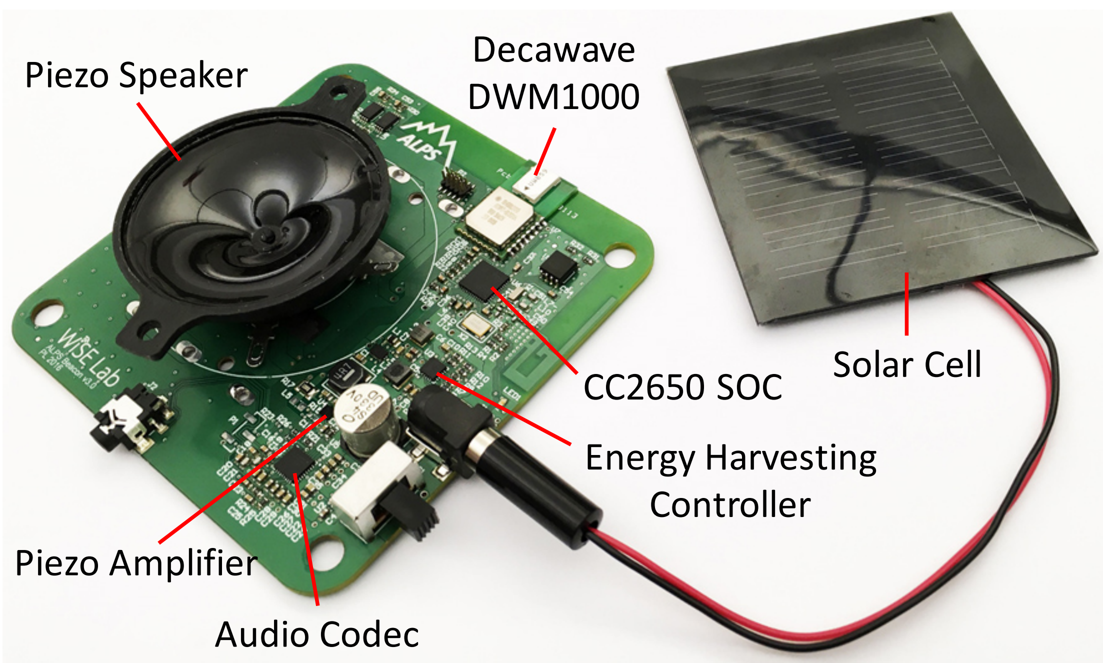
With the introduction of the CC2650 SOC an entirely new firmware had to be developed for this generation. The beacons ran TI-RTOS [31], but unlike the BLE daughter board of generation 2c (Section A.4.2), did not run a BLE stack. Instead, the advertisement packets necessary for the slot-level TDMA time synchronization were transmitted via direct commands to the radio. The beacon software application was segmented into multiple tasks (generally one per peripheral e.g. radio, audio, flash memory, etc.), which run at different priority levels and can be preempted. Since the CC2650 features a DMA controller, the CPU could run other tasks while audio is being played back. As opposed to older generations, the ultrasound waveforms were generated on the fly using ARM's CMSIS DSP library [3] instead of being externally generated and loaded into flash.

The application running on the phone remained largely identical to that of generation 2c (Section A.2.2).

A.5.3 Summary and Possible Improvements

This generation included major changes that drastically reduced a beacon's power consumption to be able to run on solar power. The lower power SOC and more efficient piezo speaker amplifier account for most of the power savings. The new speaker was not compatible with the omnidirectional horn of previous generations, but provided an adequate directionality to be used without one if the beacons were placed at strategic locations in the target environment (mostly corners). While the overall cost of the beacons is higher than previous generations, the ability to harvest solar power instead of needing to run wires drastically reduces installation cost, which is by far the highest expense in an ALPS deployment.

(a) Beacon



(b) Beacon PCB

Figure A.12: Generation 3a ALPS beacon hardware

A.6 Generation 3b

Generation 3b is the most recent generation of the ALPS platform and is described in Sections 3.2-3.5. Further pictures of this generation of beacons can be seen in Figures A.13-A.15 below.

Figure A.13: Generation 3b ALPS beacon with tile ceiling clip mount

Figure A.14: Generation 3b ALPS beacon with magnetic mount

Figure A.15: Generation 3b ALPS beacon with magnetic mount top view

Gen.	Description	Notable Features
1	Proof-of-concept with off-the-shelf audio components	• Tight beacon time synchronization via shared clock of 10 channel DAC • $<$ \$2 off-the-shelf piezo speakers
2a	First embedded beacon generation with wireless time synchronization between beacons	• Embedded beacons and network master with 8-bit AVR processor • Wireless beacon time synchronization via 802.15.4 • Ribbon tweeter with wide horizontal beam pattern
2b	Improved embedded beacon generation with omnidirectional speaker horn	• Embedded beacons and network master with 8-bit AVR processor • Wireless beacon time synchronization via 802.15.4 • RF amplifier for improved range • SoC with larger memory capacity for storing longer ultrasound signals • Audio codec with integrated amplifier • $<$ \$2 off-the-shelf piezo speakers • Speaker horn for omnidirectional ultrasound dispersion • Microphone for inter-beacon ranging
2c	Like generation 2b, but with BLE and battery powered	• Embedded beacons and network master with 8-bit AVR processor • Wireless beacon time synchronization via 802.15.4 • RF amplifier for improved range • BLE time synchronization with receivers to determine beacon IDs via daughter board • SoC with larger memory capacity for storing longer ultrasound signals • $<$ \$2 off-the-shelf piezo speakers • Audio codec with integrated amplifier • Speaker horn for omnidirectional ultrasound dispersion • Microphone for inter-beacon ranging • Battery powered
3a	All new embedded beacon generation with 32-bit ARM processor, BLE and energy harvesting	• Embedded beacons and network master with 32-bit ARM Cortex M3 processor • Multirole 802.15.4 and BLE radio on single SoC • Wireless beacon time synchronization via 802.15.4 • BLE time synchronization with receivers to determine beacon IDs • Drastically improved power consumption for battery operation with solar energy harvesting • $<$ \$1 off-the-shelf piezo speakers • Piezo speaker amplifier for improved range • UWB ranging capability on beacons and Network Master/Plug Forwarders • Network Master/Plug Forwarders with RF amplifier for improved range • Network Master/Plug Forwarders can be plugged directly into AC outlet
3b	Like generation 3a, but with four speaker array	• Embedded beacons and network master with 32-bit ARM Cortex M3 processor • Multirole 802.15.4 and BLE radio on single SoC • Wireless beacon time synchronization via 802.15.4 • BLE time synchronization with receivers to determine beacon IDs • Drastically improved power consumption for battery operation with solar energy harvesting • $<$ \$1 off-the-shelf piezo speakers • Piezo speaker amplifiers for improved range • Four speakers powered by two audio channels for improved coverage and AOA measurement • UWB ranging capability on beacons and Network Master/Plug Forwarders • Network Master/Plug Forwarders with RF amplifier for improved range • Network Master/Plug Forwarders can be plugged directly into AC outlet

Table A.1: ALPS platform generations and features

Bibliography

[1] Apple Maps: https://www.apple.com/ios/maps/ (viewed 7/13/2017). 2.3.2, 2.5.4

[2] Apple ARKit: https://developer.apple.com/arkit/ (viewed 8/2/2017). 7.1.5

[3] ARM CMSIS DSP Library: http://www.keil.com/pack/doc/CMSIS/DSP/html/index.html (viewed 7/19/2017). 3.3, A.5.2

[4] Clark, Nick. AUD1 Assistive Listening Software: http://www.aud1.com/ (viewed 2/13/2015). 5.3.1

[5] TI BLE Stack: http://www.ti.com/tool/ble-stack (viewed 7/19/2017). A.4.2

[6] Google Cardboard: https://vr.google.com/cardboard/ (viewed 8/2/2017). 7.1.5

[7] Microsoft Indoor Localization Competition 2017: https://www.microsoft.com/en-us/research/event/microsoft-indoor-localization-competition-ipsn-2017 (viewed 7/13/2017). 2.4.3

[8] Google Daydream: https://developers.google.com/vr/daydream/overview (viewed 8/2/2017). 7.1.5

[9] Decawave: http://decawave.com/ (viewed 7/13/2017). 1.2, 2.2.3, 2.4.4, 2.6(b), 7.1.1

[10] Drone RK Hardware Module: http://wise.ece.cmu.edu/redmine/projects/drone-rk/wiki/Hardware$_Module(viewed7/18/2017).A.2.1$

[11] 911 Wireless Services: https://www.fcc.gov/consumers/guides/911-wireless-services

(viewed 7/16/2017). 2.5.9

[12] Google Beacon Platform: https://developers.google.com/beacons/ (viewed 7/25/2017). 2.3.2, 7.1.1

[13] Estimote location beacons with UWB: http://estimote.com/ (viewed 7/14/2017). 2.4.4, 7.1.1

[14] Samsung Galaxy Gear VR: http://www.samsung.com/global/galaxy/gear-vr/ (viewed 8/2/2017). 7.1.5

[15] Gimbal: http://www.gimbal.com/ (viewed 2/13/2015). 1.2, 2.3.2, 2.5.4, 7.1.1

[16] Google Glass: https://www.x.company/glass/ (viewed 8/2/2017). . 7.1.5

[17] Google Maps: https://www.google.com/maps/about/ (viewed 7/13/2017). . 2.3.2, 2.5.4

[18] Microsoft Hololens: https://www.microsoft.com/en-us/hololens/ (viewed 8/2/2017). 7.1.5

[19] Cisco Hyperlocation: http://www.cisco.com/c/en/us/products/interfaces-modules/aironet-hyperlocation-module-advanced-security/index.html (viewed 7/13/2017). 2.3.1

[20] iBeacon: https://developer.apple.com/ibeacon/ (viewed 2/13/2015). 1.2, 2.3.2, 2.5.4, 7.1.1

[21] Eadie, Gavin. ios-ntp: https://github.com/jbenet/ios-ntp (viewed 2/13/2015). 5.3.4

[22] Microsoft Indoor Localization Competition Results 2014: https://www.microsoft.com/en-us/research/event/microsoft-indoor-localization-competition-ipsn-2014/ (viewed 7/15/2017). . 2.5.4, A.2.3

[23] Microsoft Indoor Localization Competition Results 2015: https://www.microsoft.com/en-us/research/event/microsoft-indoor-localization-competition-ipsn-2015/ (viewed 7/15/2017). . 2.5.4, A.4.3

[24] Microsoft Indoor Localization Competition Results 2016: https://www.microsoft.com/en-us/research/event/microsoft-indoor-localization-competition-ipsn-2016/ (viewed

7/15/2017). . 2.5.4

[25] Microsoft Indoor Localization Competition Results 2017: https://www.microsoft.com/en-us/research/event/microsoft-indoor-localization-competition-ipsn-2017/ (viewed 7/15/2017). . 2.5.4

[26] Nanotron: http://nanotron.com/EN/CO$_{t}echn - sdstwr.php/(viewed7/13/2017).$2.2.3

[27] Oculus Rift: https://www.oculus.com/ (viewed 8/2/2017). 7.1.5

[28] OpenChirp: http://openchirp.io/ (viewed 7/25/2017). 6.4, 7.1.4

[29] Quantitec Intranav: https://intranav.com/ (viewed 7/14/2017). 2.4.4

[30] Time Domain PulsON440: http://www.timedomain.com/ (viewed 7/14/2017). 1.2, 2.4.4, 2.6(a)

[31] TI RTOS: http://www.ti.com/tool/ti-rtos (viewed 7/19/2017). 3.3, A.4.2, A.5.2

[32] Ubisense Dimension 4: https://ubisense.net/en/products/Dimension4 (viewed 7/16/2017). 2.4.4

[33] HTC VIVE: https://www.vive.com/ (viewed 8/2/2017). 7.1.5

[34] WiFiSLAM: https://angel.co/wifislam (viewed 7/13/2017). 2.3.2, 2.5.4

[35] Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirement part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans). *IEEE Std 802.15.4a-2007 (Amendment to IEEE Std 802.15.4-2006)*, pages 1–203, 2007. doi: 10.1109/IEEESTD.2007.4299496. 2.4.4

[36] Ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE*

Std 802.11-2016 (Revision of IEEE Std 802.11-2012), pages 1–3534, Dec 2016. doi: 10.1109/IEEESTD.2016.7786995. 7.1.5

[37] Gaurav Bhatia Ethan Goldman Raj Rajkumar Lucio Soibelman James H. Garrett Jose M. F. Moura Anthony Rowe, Mario Berges. Sensor Andrew: Large-Scale Campus-Wide Sensing and Actuation. *IBM Journal of Research and Development: Special Issue on Smarter Cities and Sensed Infrastructures*, 2010. 6.4

[38] P. Bahl and V.N. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, volume 2, pages 775 –784, 2000. doi: 10.1109/INFCOM.2000.832252. 1.2, 2.3, 2.3.2, 2.5.4

[39] L.L. Beranek. *Acoustics*. McGraw-Hill electrical and electronic engineering series. McGraw-Hill, 1954. 3.7.2

[40] J. Biswas and M. Veloso. Wifi localization and navigation for autonomous indoor mobile robots. In *2010 IEEE International Conference on Robotics and Automation*, pages 4379–4384, May 2010. doi: 10.1109/ROBOT.2010.5509842. 2.5.4

[41] Pratik Biswas, Kim-Chuan Toh, and Yinyu Ye. A distributed sdp approach for large-scale noisy anchor-free graph realization with applications to molecular conformation. *SIAM Journal on Scientific Computing*, 30(3):1251–1277, 2008. 7.1.3

[42] Gaetano Borriello, Alan Liu, Tony Offer, Christopher Palistrant, and Richard Sharp. Walrus: wireless acoustic location with room-level resolution using ultrasound. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys '05)*, pages 191–203, New York, NY, USA, 2005. ACM. ISBN 1-931971-31-5. doi: 10.1145/1067170.1067191. URL `http://doi.acm.org/10.1145/1067170.1067191`. 2.2

[43] Bosch. Praesideo - digital public address and emergency sound system. URL http://resource.boschsecurity.com/documents/Catalog_enUS_1213647371.pdf. 4.3.2

[44] Health Protection Branch Health Canada. Environmental health directorate. guidelines for the safe use of ultrasound: Part ii - industrial and commercial applications - safety code 24. *Published by authority of the Minister of National Health and Welfare*, 1991. 4.3.2

[45] GEORGE CHAPMAN, Andrew Hadfield, and Neil Rhodes. *George Chapman: Homer's 'Odyssey'*, volume 21. Modern Humanities Research Association, 2016. ISBN 9781781881217. URL http://www.jstor.org/stable/j.ctt1g0b82z. 1.1

[46] Woo Cheol Chung and Dong Ha. An accurate ultra wideband (uwb) ranging for precision asset location. In *IEEE Conference on Ultra Wideband Systems and Technologies, 2003*, pages 389–393, Nov 2003. doi: 10.1109/UWBST.2003.1267870. 2.4.4

[47] F. Cristian. Probabilistic clock synchronization. *Distributed Computing*, 1989. 5.2

[48] Yichuan Ding, Nathan Krislock, Jiawei Qian, and Henry Wolkowicz. Sensor network localization, euclidean distance matrix completions, and graph realization. *Optimization and Engineering*, 11(1):45–66, Feb 2010. ISSN 1573-2924. doi: 10.1007/s11081-008-9072-0. URL https://doi.org/10.1007/s11081-008-9072-0. 7.1.3

[49] Gregory Dudek and Michael Jenkin. *Computational Principles of Mobile Robotics*. Cambridge University Press, New York, NY, USA, 2nd edition, 2010. ISBN 0521692121, 9780521692120. 6.3

[50] Jeremy Elson, Lewis Girod, and Deborah Estrin. Fine-grained network time synchronization using reference broadcasts. In *Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI '02)*, pages 147–163, New York, NY,

USA, 2002. ACM. ISBN 978-1-4503-0111-4. doi: 10.1145/1060289.1060304. URL `http://doi.acm.org/10.1145/1060289.1060304`. 5.2

[51] ETSI. Digital cellular telecommunications system (Phase 2+); Network Identity and Timezone (NITZ); Service description, Stage 1. Technical report, ETSI, 4 1999. 2.6.3

[52] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with Glossy. In *Proceedings of the 10th IEEE International Conference on Information Processing in Sensor Networks*, IPSN '11, pages 73–84, April 2011. 2.6.5, 5.2

[53] Brian D Ferris, Dieter Fox, and Neil Lawrence. Wifi-slam using gaussian process latent variable models. 2007. 2.5.4

[54] C. Di Franco, A. Prorok, N. Atanasov, B. Kempke, P. Dutta, V. Kumar, and G. J. Pappas. Calibration-free network localization using non-line-of-sight ultra-wideband measurements. In *2017 16th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 235–246, April 2017. 7.1.3

[55] 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 (SenSys '03)*, pages 138–149, New York, NY, USA, 2003. ACM. ISBN 1-58113-707-9. doi: 10.1145/958491.958508. URL `http://doi.acm.org/10.1145/958491.958508`. 5.2

[56] Zatti S. Gusell, R. The accuracy of clock synchonization achieved by tempo. *IEEE transactions on Software Engineering*, 1989. 5.2

[57] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. *Wirel. Netw.*, 8(2/3):187–197, March 2002. ISSN 1022-0038. doi: 10.1023/A:1013767926256. URL `http://dx.doi.org/10.1023/A:`

1013767926256. 2.5.7

[58] Mike Hazas and Andy Ward. A novel broadband ultrasonic location system. In *Proceedings of the 4th International Conference on Ubiquitous Computing (UbiComp '02)*, pages 264–280, London, UK, UK, 2002. Springer-Verlag. ISBN 3-540-44267-7. URL http://dl.acm.org/citation.cfm?id=647988.741478. 1.2, 2.5.3

[59] Mike Hazas and Andy Ward. A high performance privacy-oriented location system. In *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PERCOM '03)*, pages 216–223, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1893-1. URL http://dl.acm.org/citation.cfm?id=826025.826383. 1.2, 2.2, 2.5.3

[60] G. Kantor and S. Singh. Preliminary results in range-only localization and mapping. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, volume 2, pages 1818–1823 vol.2, 2002. doi: 10.1109/ROBOT.2002.1014805. 7.1.3

[61] Elliott Kaplan and Christopher Hegarty. *Understanding GPS: principles and applications*. Artech house, 2005. 5.4, 5.4

[62] Benjamin Kempke, Pat Pannuto, and Prabal Dutta. Harmonia: Wideband spreading for accurate indoor rf localization. *SIGMOBILE Mob. Comput. Commun. Rev.*, 18(3):19–25, January 2015. ISSN 1559-1662. doi: 10.1145/2721896.2721901. URL http://doi.acm.org/10.1145/2721896.2721901. 2.4.4

[63] Benjamin Kempke, Pat Pannuto, Bradford Campbell, and Prabal Dutta. Surepoint: Exploiting ultra wideband flooding and diversity to provide robust, scalable, high-fidelity indoor localization. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 137–149. ACM, 2016. 2.4.4

[64] Ochsenreiter W. Kopetz, H. Clock synchonization in distributed real-time systems. *IEEE Computer*, August 1987. 5.2

[65] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. Luxapose: Indoor positioning with mobile phones and visible light. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 447–458, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2783-1. doi: 10.1145/2639108.2639109. URL `http://doi.acm.org/10.1145/2639108.2639109`. 1.2

[66] Patrick Lazik and Anthony Rowe. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, pages 99–112, Toronto, Ontario, Canada, 2012. ACM. ISBN 978-1-4503-1169-4. doi: 10.1145/2426656.2426667. URL `http://doi.acm.org/10.1145/2426656.2426667`. 2.2.2, A.1.1

[67] Patrick Lazik, Niranjini Rajagopal, Oliver Shih, Bruno Sinopoli, and Anthony Rowe. Alps: A bluetooth and ultrasound platform for mapping and localization. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 73–84, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3631-4. doi: 10.1145/2809695.2809727. URL `http://doi.acm.org/10.1145/2809695.2809727`. 2.2.2, A.4.1

[68] Patrick Lazik, Niranjini Rajagopal, Bruno Sinopoli, and Anthony Rowe. Ultrasonic time synchronization and ranging on smartphones. In *Proceedings of the 21st IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2015)*, RTAS '15. IEEE, IEEE, 2015. 2.2.2, 2.6.2

[69] H. Lee, T. H. Kim, J. W. Choi, and S. Choi. Chirp signal-based aerial acoustic communication for smart devices. In *2015 IEEE Conference on Computer Communications*

(INFOCOM), pages 2407–2415, April 2015. doi: 10.1109/INFOCOM.2015.7218629. 3.7.1, 3.7.4

[70] Joon-Yong Lee and R. A. Scholtz. Ranging in a dense multipath environment using an uwb radio link. *IEEE Journal on Selected Areas in Communications*, 20(9):1677–1683, Dec 2002. ISSN 0733-8716. doi: 10.1109/JSAC.2002.805060. 2.4.4

[71] C. H. Lim, Y. Wan, B. P. Ng, and C. M. S. See. A real-time indoor wifi localization system utilizing smart antennas. *IEEE Transactions on Consumer Electronics*, 53(2):618–622, May 2007. ISSN 0098-3063. doi: 10.1109/TCE.2007.381737. 2.5.4

[72] Konrad Lorincz and Matt Welsh. Motetrack: a robust, decentralized approach to rf-based location tracking. In *Proceedings of the 1st International Conference on Location- and Context-Awareness (LoCA'05)*, pages 63–82, Berlin, Heidelberg, 2005. Springer-Verlag. ISBN 3-540-25896-5, 978-3-540-25896-4. doi: 10.1007/11426646_7. URL `http://dx.doi.org/10.1007/11426646_7`. 2.3, 2.5.5

[73] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. The flooding time synchronization protocol. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 39–49, New York, NY, USA, 2004. ACM. ISBN 1-58113-879-2. doi: 10.1145/1031495.1031501. URL `http://doi.acm.org/10.1145/1031495.1031501`. 2.6.4

[74] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. The flooding time synchronization protocol. In *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 39–49, New York, NY, USA, 2004. ACM. ISBN 1-58113-879-2. doi: 10.1145/1031495.1031501. URL `http://doi.acm.org/10.1145/1031495.1031501`. 2.6.4, 5.2

[75] Jim Martin, Jack Burbank, William Kasch, and Professor David L. Mills. Network Time

Protocol Version 4: Protocol and Algorithms Specification. RFC 5905, June 2010. URL https://rfc-editor.org/rfc/rfc5905.txt. 2.6.1

[76] U. Maurer, A. Rowe, A. Smailagic, and D.P. Siewiorek. ewatch: a wearable sensor and notification platform. In *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*, pages 4 pp.–145, April 2006. doi: 10.1109/BSN.2006.24. 2.3

[77] Michael McCarthy, Paul Duff, Henk L. Muller, and Cliff Randell. Accessible ultrasonic positioning. *IEEE Pervasive Computing*, 5(4):86–93, October 2006. ISSN 1536-1268. doi: 10.1109/MPRV.2006.65. URL http://dx.doi.org/10.1109/MPRV.2006.65. 1.2

[78] D. L. Mills. Internet time synchronization: the network time protocol. *IEEE Transactions on Communications*, 39(10):1482–1493, Oct 1991. ISSN 0090-6778. doi: 10.1109/26.103043. 2.6.1, 5.2

[79] P. Misra and P. Enge. *Global Positioning System: Signals, Measurements, and Performance*. Ganga-Jamuna Press, Lincoln MA, 2nd edition edition, 2006. 1.2, 2.6.2, 6.1, 6.2

[80] P. Newman and J. Leonard. Pure range-only sub-sea slam. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 2, pages 1921–1926 vol.2, Sept 2003. doi: 10.1109/ROBOT.2003.1241875. 7.1.3

[81] Nokia. Synchronization as a service. Technical report, Nokia, 10 2016. 2.6.3

[82] Scott Pace, United States. Office of Science, Technology Policy, and Rand Corporation. *The Global Positioning System: Assessing National Policies*. MR: Rand Corporation. RAND, 1995. ISBN 9780833023490. URL https://books.google.com/books?id=WBlSAQAAIAAJ. 1.2

[83] B.W. Parkinson and S.W. Gilbert. Navstar: Global positioning system - ten years later. *Proceedings of the IEEE*, 71(10):1177 – 1186, oct. 1983. 2.2, 2.5.1

[84] Chunyi Peng, Guobin Shen, Zheng Han, Yongguang Zhang, Yanlin Li, and Kun Tan. A beepbeep ranging system on mobile phones. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pages 397–398, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-763-6. doi: 10.1145/1322263.1322313. URL `http://doi.acm.org/10.1145/1322263.1322313`. 2.2.3, 2.5.8

[85] Plack,C.J. *The sense of hearing*. Lawrence Erlbaum Associates, Inc., 2005. 4.1

[86] R. Price and P.E. Green. A communication technique for multipath channels. *Proceedings of the IRE*, 46(3):555 –570, march 1958. ISSN 0096-8390. doi: 10.1109/JRPROC.1958.286870. 4.4.3

[87] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (Mobicom '00)*, pages 32–43, New York, NY, USA, 2000. ACM. ISBN 1-58113-197-6. doi: 10.1145/345910.345917. URL `http://doi.acm.org/10.1145/345910.345917`. 1.2, 2.2, 2.2.1, 2.5.2, 4.5.4

[88] Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rijurekha Sen. Zee: zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 293–304. ACM, 2012. 5.4.6

[89] N. Rajagopal, S. Chayapathy, B. Sinopoli, and A. Rowe. Beacon placement for range-based indoor localization. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, Oct 2016. doi: 10.1109/IPIN.2016.7743626. 7.1.4

[90] Niranjini Rajagopal, Patrick Lazik, and Anthony Rowe. Visual light landmarks for mo-

bile devices. In *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IPSN '14, pages 249–260, Piscataway, NJ, USA, 2014. IEEE Press. ISBN 978-1-4799-3146-0. URL `http://dl.acm.org/citation.cfm?id=2602339.2602367`. 1.2

[91] M.A. Richards. *Fundamentals of Radar Signal Processing*. Professional Engineering. Mcgraw-hill, 2005. ISBN 9780071444743. URL `https://books.google.com/books?id=VOvsJ7G1oDEC`. 2.4.1

[92] Fay RR. and Popper AN. Comparative hearing: Mammals. *Springer Handbook of Auditory Research Series*, 4, 1994. 4.3.3

[93] *LoRa Modulation Basics*. Semtech, 5 2015. Rev. 2. 5.7

[94] H. Shen, S. Machineni, C. Gupta, and A. Papandreou-Suppappola. Time-varying multichirp rate modulation for multiple access systems. *IEEE Signal Processing Letters*, 11(5): 497 – 500, may 2004. ISSN 1070-9908. doi: 10.1109/LSP.2004.826638. 4.4.1

[95] Gyula Simon, Miklós Maróti, Ákos Lédeczi, György Balogh, Branislav Kusy, András Nádas, Gábor Pap, János Sallai, and Ken Frampton. Sensor network-based countersniper system. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 1–12, New York, NY, USA, 2004. ACM. ISBN 1-58113-879-2. doi: 10.1145/1031495.1031497. URL `http://doi.acm.org/10.1145/1031495.1031497`. 2.2.2

[96] B. Sklar. *Digital communications: fundamentals and applications*. Prentice Hall Communications Engineering and Emerging Technologies Series. Prentice-Hall PTR, 2001. ISBN 9780130847881. URL `https://books.google.com/books?id=Bh4fAQAAIAAJ`. 2.3.2

[97] Aeroscout Enterprise Visibility Solutions. Aeroscout system: Bridging the gap between

wi-fi, active rfid and gps. *Printed Aug*, 27:3, 2004. 2.3.2

[98] P. Stavroulakis. *Third Generation Mobile Telecommunication Systems: UMTS and IMTter Stavroulakis (ed.).* Engineering online library. Springer Berlin Heidelberg, 2001. ISBN 9783540678502. URL `https://books.google.com/books?id=z39Bqp-0BgQC`. 2.6.3

[99] Stephen P. Tarzia, Peter A. Dinda, Robert P. Dick, and Gokhan Memik. Indoor localization without infrastructure using the acoustic background spectrum. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys '11)*, pages 155–168, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0643-0. doi: 10.1145/1999995.2000011. URL `http://doi.acm.org/10.1145/1999995.2000011`. 2.3

[100] Sebastian Thrun and Michael Montemerlo. The graph slam algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006. 7.1.3

[101] Sergio Verdu. *Multiuser Detection*. Cambridge University Press, New York, NY, USA, 1st edition, 1998. ISBN 0521593735. 4.4.3

[102] Peter Volgyesi, Gyorgy Balogh, Andras Nadas, Christopher B. Nash, and Akos Ledeczi. Shooter localization and weapon classification with soldier-wearable networked sensors. In *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services*, MobiSys '07, pages 113–126, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-614-1. doi: 10.1145/1247660.1247676. URL `http://doi.acm.org/10.1145/1247660.1247676`. 2.3.1

[103] Roy Want, Andy Hopper, Veronica Falcão, and Jonathan Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102, January 1992. ISSN 1046-8188.

doi: 10.1145/128756.128759. URL `http://doi.acm.org/10.1145/128756.128759`. 1.2, 2.3

[104] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42 –47, oct 1997. ISSN 1070-9916. doi: 10.1109/98.626982. 2.5.7

[105] P. Wille. *Sound Images of the Ocean: in Research and Monitoring*. Sound Images of the Ocean in Research and Monitoring. Springer Berlin Heidelberg, 2005. ISBN 9783540241225. URL `https://books.google.com/books?id=r5CUczPBlp0C`. 2.4.2