Chirp Signal-Based Aerial Acoustic Communication for Smart Devices

Hyewon Lee[†°], Tae Hyun Kim[°], Jun Won Choi[‡], and Sunghyun Choi[†]
[†]Seoul National University and INMC, [°]Soundlly, Inc, [‡]Hanyang University
Email: hwlee@mwnl.snu.ac.kr, tae@soundl.ly, junwchoi@hanyang.ac.kr, schoi@snu.ac.kr

***Abstract*—Smart devices such as smartphones and tablet/wearable PCs are equipped with voice user interface, i.e., speaker and microphone. Accordingly, various *aerial acoustic communication* techniques have been introduced to utilize the voice user interface as a communication interface. In this paper, we propose an aerial acoustic communication system using inaudible audio signal for low-rate communication in indoor environments. By adopting *chirp signal*, which is widely used for radar applications due to its capability of resolving multi-path propagation, the proposed acoustic modem supports long-range communication independent of device characteristics over severely frequency-selective acoustic channel. We also design a backend server architecture to compensate for the low data rate of chirp signal-based acoustic modem. Via extensive experiments, we evaluate various characteristics of the proposed modem including multi-path resolution and multiple chirp signal detection. We also verify that the proposed chirp signal can deliver data at 16 bps in typical indoor environments, where its maximum transmission range is drastically extended up to 25 m compared to the few meters of the previous research.**

***Index Terms*—Chirp signal, aerial acoustic communication, smart devices, software-based digital modem.**

I. Introduction

Today's various communication technologies can be categorized by the medium that is used to deliver information from transmitter to receiver. Separate from common wireless communication technologies such as WiFi and Bluetooth that deliver information through electromagnetic waves, *acoustic communication* refers to the communication technology that uses sound waves. Sound waves propagate through the mechanical vibration of a medium (e.g., water or air), and they can be transmitted and received by speakers and microphones.

Acoustic communication has been developed mainly for the underwater communication of sensor networks to monitor marine ecosystems and the status of underwater platforms like oil rigs [1, 2]. Digital modulation schemes that are used for electromagnetic wave communications, e.g., phase/frequency shift keying (PSK/FSK), can be applied to acoustic communications to modulate sound signals [2]. Underwater acoustic communication requires transceivers to operate in water, which typically means the use of special speakers and microphone modules, i.e., hydrophones.

In contrast, *aerial acoustic communication* delivers information through airborne sound waves [3, 4]. Depending on the operating frequency of aerial acoustic communication, the transceiver can be either an off-the-shelf device that plays/records audio (under 22 kHz in general), or an ultrasound transducer (over 22 kHz). It should be noted that the aerial acoustic communication is versatile because any audio interface-equipped device can be utilized as a communication device, i.e., hardware and operating system (OS) independent. Moreover, it provides an alternative communication interface for smart devices along with WiFi and Bluetooth, which often are turned off in order to save battery and/or prevent unintended connections. Given the widespread use of smart devices that play, record, and process sound signals through their voice user interface, aerial acoustic communication under 22 kHz has recently been studied in depth [5–14].

However, the communication range of the previous work is limited to few meters, as the acoustic channels, composed of speaker, air medium, and microphone, are heavily frequency selective. Moreover their signal modulation schemes do not work well over such frequency selective channels. In this paper, we focus on inaudible aerial acoustic communication for off-the-shelf audio interfaces in long-range indoor environments. Specifically, we assume that a speaker repeatedly broadcasts short-length ID and a smart device decodes the acoustic signal. The contributions of the paper are three folds.

- By adopting *chirp signal* that was originally used in radar applications [15], we drastically extend the communication range up to 25 m at maximum. The chirp signal also supports most smart devices equipped with microphones that have very selective frequency responses.
- We propose a software digital modem for smartphones that can efficiently demodulate the chirp signal by combining fast Fourier transform (FFT) and Hilbert transform. The proposed *envelope detector* can also compensate for Doppler effect.
- In addition, we introduce a TV content recognition service as an example application. We also implement a backend query server in order to overcome the low data rate (approximately 16 bps).

The rest of the paper is organized as follows. In Section II, we first review the previous work and briefly evaluate the performance of commonly used modulation schemes. Section III and Section IV present *chirp signal* in indoor acoustic channel and corresponding software modem for smart devices, respectively. We introduce a TV content recognition service based on the proposed modem and a backend server that compensates for the bit rate limitation in Section V. Through extensive experimental results, we evaluate the performance of the proposed system in Section VI. Finally we conclude the paper with discussion on the future work in Section VII.

978-1-4799-8381-0/15/$31.00 ©2015 IEEE

II. Related Work and Revisit of PHY Modulations

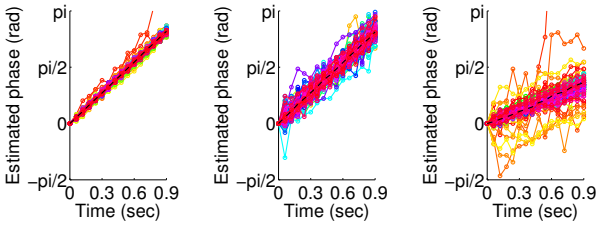
We investigate the previous research on aerial acoustic communication and discuss their drawbacks. Our experimental results show that the limited performance is due mainly to the signal modulation schemes.

A. Related Work

Aerial acoustic communication for data transmission among audio-enabled devices has been studied in depth [6–11]. In [6], multiple tones are used to transmit data at 5.6 kbps in an audible mode (735∼4,410 Hz), or a single tone for 1.4 kbps rate in an inaudible mode (18.4 kHz). Both modes work up to 2 m distance with line-of-sight (LOS) communication. M-ary FSK-based acoustic communication scheme, referred to as *Digital Voice*, is proposed in [7] to achieve data rates of tens to thousands of bps, where the frequency tones under 12 kHz audible band are chosen to resemble the sounds of clarinets and birds. [8] implements orthogonal frequency division multiplex (OFDM) with binary and quadrature PSK modulation schemes in the 6∼7 kHz band. The proposed system, Dhwani, aims to substitute near field communication (NFC) with a data rate of up to 800 bps in a very short range (less than 20 cm). NTT DoCoMo introduces Acoustic OFDM in [9] that hides approximately 240 bps of information in the 6.4∼8 kHz band of existing sound sources. [10] similarly embeds 600 bps data by modulating the phase of 6.4∼8 kHz band via modulated complex lapped transform. Due to the masking effect of the human hearing system, the embedded information in [9, 10] cannot be heard by people. Commercial off-the-shelf laptops are used to build a mesh network via aerial acoustic communications in [11], achieving a link distance and speed up to 20 m and 20 bps at maximum, respectively.

Commercial services using aerial acoustic communication technologies have been also introduced by a number of start-ups [12–14]. SonicNotify has developed small beacon devices that repeatedly transmit high frequency inaudible tones [12]. ShopKick is another start-up that is collaborating with retail stores such as Macy's to provide special offers via inaudible signals generated by ultrasound transmitters [13]. Naratte is a start-up that introduced Zoosh as a substitute to NFC. This is currently being used by Verifone as a mobile payment service for taxis, referred to as Way2Ride [14].

The aforementioned work has a few limitations. In [6–8], the sound signals of aerial acoustic communication is within the audible band. This has two main drawbacks: (1) the sound signals heard by human beings can be disruptive, and (2) the background noise in a typical environment distorts the acoustic signals. Acoustic signals designed in [9, 10] require an existing audible sound source to embed the signal, and hence, are not stand-alone techniques. In addition, the modulation schemes in many of these examples have a limited transmission range of up to a few meters even in ling-of-sight (LOS) environment. The only exception is [11], but the speaker and microphone of laptops used in the testbed have a very flat frequency response in the high-frequency band, which is not the case in most audio interfaces as presented in Section III-B.



(a) Optimus G2. (b) Sky Vega Iron. (c) Galaxy Note.

Fig. 1. Estimated phase shift of recorded sine tones in three different smartphone models. Phase basically drifts over time due to the sampling frequency offset, while its variance differs by model.

Extending the communication range is important to enable various types of services using aerial acoustic communication with off-the-shelf loud speakers. Note that most commercial services [12–14] employ specially designed transmitters to achieve a few meters of communication range.

B. Revisit of PHY modulation schemes

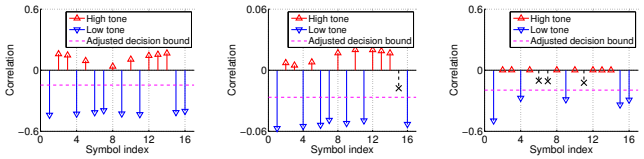
We revisit two widely used digital signal modulation schemes, namely, PSK and FSK, and discuss their limitations in aerial acoustic communications for smart devices.

1) Phase shift keying: It is well known that PSK requires exact carrier, sampling, and symbol timing recovery, and its performance degrades in time-varying fading channels [16]. We here present the time-varying phase shift of aerial acoustic channel through a simple experiment. In a typical office environment, we play a sine tone at 20 kHz center frequency using a reference speaker. Then we record the sine tone using smart devices at 12 m distance LOS channel.

Fig. 1 depicts the shift of the estimated phase (relative to the first sample) that are simultaneously recorded every 60 ms in three smart devices. Each line represents 50 iterations of recording. Basically the estimated phase linearly drifts over time due to the sampling frequency offset of the microphone [8] (Fig. 1(a)). For some models, however, the phase shift is not always linear; in the worst case, the phase shift is not predictable (Fig. 1(c)). This is due to the channel variation as well as non-ideal local oscillator of smart devices. Hence we claim that PSK is not applicable for long-range aerial acoustic communication as the receiver requires phase compensation algorithm such as phase locked loops, which complicates the receiver architecture.

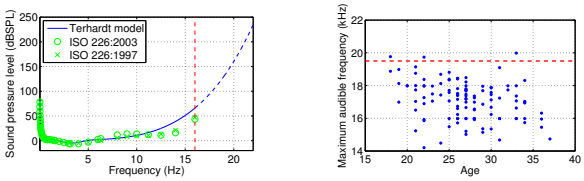
2) Frequency shift keying: FSK usually suffers from the frequency selectivity, since constellation points of received tones are asymmetric [17]. We set an experiment to validate the feasibility of FSK on aerial acoustic communication. Assuming binary FSK (BFSK), we sequentially play two types of sine tones using a reference speaker in inaudible band; low frequency (19 kHz for bit 0) and high frequency (21 kHz for bit 1) tones. Smart devices record each tone for 100 times at 12 m LOS channel, and we emulate non-coherent FSK receiver with adaptive decision bound [17] using MATLAB to present the received constellation point in Fig. 2.

Fig. 2 presents constellation points of 16 BFSK symbols received by three smart devices. The correlation of high and



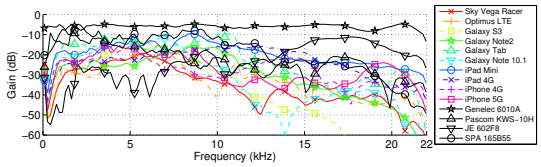
(a) Galaxy S3. (b) Sky Vega N6. (c) Galaxy Note2.

Fig. 2. Empirical constellation point of non-coherent BFSK receivers for three different smart devices. High frequency tones are hard to be received due to the frequency selectivity of microphones, so that adaptive decision bound might fail.

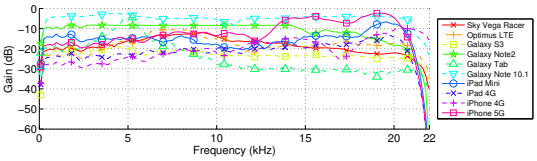


(a) Theoretic hearing threshold. (b) Empirical hearing threshold.

Fig. 3. Theoretic and empirical hearing threshold of human ears. Terhardt model in Fig. 3(a) is based on [19], and we obtain Fig. 3(b) via experiment with randomly selected 134 people.



(a) Frequency response of speakers.



(b) Frequency response of microphones.

Fig. 4. Frequency response of speaker and microphones of various smart devices and off-the-shelf speakers. Speakers are shown to be more frequency selective than microphones in general, and the selectivity of microphones is severe over 21 kHz.

low frequency tones are supposed to be the same in flat-frequency response. However, the constellation points are asymmetric due to the frequency selectivity of microphones, and hence, the receiver requires adaptive decision bound as shown in Fig. 2(a).[1] In very frequency selective channels as in Figs. 2(b) and 2(c), however, the adaptive decision bound fails to demodulate the high frequency tones due to the severe loss. Therefore, FSK is not a good option for highly frequency selective acoustic channel.

III. Acoustic Channel and Chirp Signal

In this section, we first define the frequency band for inaudible aerial acoustic communication. Then, we measure the frequency responses of speakers and microphones implemented in various smart devices, as well as the characteristics of typical indoor acoustic channel. In order to cope with the severe frequency selectivity and random phase distortion of the indoor acoustic channel, we adopt *chirp signal* for indoor aerial acoustic communication.

A. Hearing threshold of human beings

As we aim to design *inaudible* acoustic signal, we theoretically and empirically evaluate the maximum audible frequency of human ears. Fig. 3(a) presents the theoretic hearing threshold measured at 1 m distance [18], where Terhardt model over 16 kHz (vertical dashed line) shows estimated values. The sound pressure level that is required to stimulate cochlear varies over frequency. For instance, at 18 kHz, the sound pressure level should be at least 100 dBSPL to be heard; this is equivalent to the loudness of a heavy metal concert, and hence, signal over 18 kHz is claimed to be mostly inaudible in typical environments. To be more specific, we conduct an experiment with 134 people to determine the maximum frequency they can hear through earphones. From Fig. 3(b), we choose 19.5 kHz as the lower frequency bound of inaudible sound signal. Since the sampling frequency of most audio interfaces is 44.1 kHz, we set the upper frequency bound as 22 kHz.

B. Frequency response of various audio interfaces

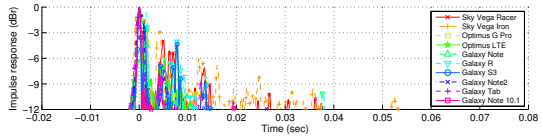
We measure the frequency responses of various audio interfaces in an anechoic chamber. As reference speaker and microphone with flat frequency responses, we use Genelec 6010A active speaker and Audix TM1 condenser microphone. Adopting linear frequency sweep (LFS) signal method proposed in [20], we play 50 sec sine signal linearly sweeping from 20 to 22,000 Hz via reference (or smart device/off-the-shelf) speaker, and record the sound using smart device (or reference microphone). Then we post-process the recorded audio using MATLAB to calculate the frequency response.

Comparing Figs. 4(a) and 4(b), we observe that speakers are more frequency selective than microphones in general. We also note that some smart devices can hardly record signals over 21 kHz. From Fig. 4, we conclude that inaudible acoustic communication in 19.5∼22 kHz band is mostly feasible, though it has to overcome the severe frequency selectivity of audio interfaces.

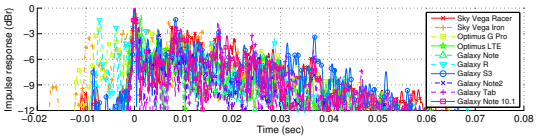
C. Delay spread of acoustic channel

Now we measure the impulse responses of aerial acoustic channels in a typical indoor office environment. We again adopt LFS signal method in [20]. We play LFS signal using Genelec 6010A and JE602F8 speakers for LOS and non-LOS (NLOS) acoustic channel, respectively, and record with various smart devices which are randomly located within 5 m distance from the speaker. From Fig. 5, we observe that the indoor aerial acoustic channel suffers from severe multi-path propagation. Especially in NLOS case, multi-path delay extends up to 40 ms for -6 dBr threshold. Therefore one needs guard interval between symbols or a channel equalizer in order to avoid inter-symbol interference (ISI).

[1]We assume that the first four symbols are known in off-line for the adaptation of decision bound in this experiment.

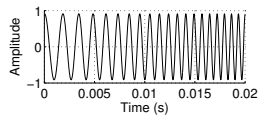


(a) Delay spread of LOS acoustic channel.

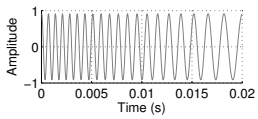


(b) Delay spread of NLOS acoustic channel.

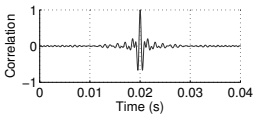
Fig. 5. Impulse responses of aerial acoustic channels in indoor office environment, measured using LFS signal [20]. Each measurement is repeated 10 times at random locations within 5 m.



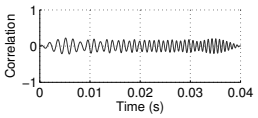
(a) Up chirp.



(b) Down chirp.



(c) Auto correlation.



(d) Cross correlation.

Fig. 6. An example of a pair linear chirps, sweeping from 500 to 1500 Hz with 20 ms symbol duration, and their auto and cross correlations.

D. Chirp binary orthogonal keying

The signal modulation scheme for aerial acoustic communication should be robust to the frequency selectivity, and symbol/sampling timing offset. As a solution, we adopt *chirp signal* [15]. Different from PSK and FSK signal which have time invariant frequency for a specific symbol, the frequency of a chirp symbol varies over time; up chirp and down chirp have increasing and decreasing frequencies, respectively, as shown in Fig. 6. Up and down chirps use the same frequency band, and hence, the frequency selectivity of acoustic channel distorts both up and down chirp symbols in a symmetric manner, which is not the case in FSK. We can also design a non-coherent demodulator which is robust to timing offset and Doppler shift [15], as to be discussed in Section IV.

As a chirp signal has good auto correlation characteristic (Figs. 6(c) and 6(d)), it has been widely used for radar applications as well as localizations [21, 22]. Based on the fact that up and down chirps are nearly orthogonal, the authors in [16, 23, 24] also propose chirp binary orthogonal keying (BOK) for wireless communications. [25, 26] uses acoustic chirp signals to measure the distance between speaker and microphone. To our best knowledge, however, this is the first literature that applies chirp BOK to aerial acoustic communication in order to cope with the frequency selective and time-varying acoustic channel.

A pair of chirp signals in frequency band $[f_1, f_2]$ Hz is defined as

$$s_1(t) = \cos\left(2\pi f_1 t + \mu t^2/2 + \phi_0\right), \quad (1)$$

$$s_2(t) = \cos\left(2\pi f_2 t - \mu t^2/2 + \phi_0\right), \quad 0 \le t \le T, \quad (2)$$

for Symbol 1 (up chirp) and Symbol 2 (down chirp), respectively, where T is symbol duration, and ϕ_0 is arbitrary initial phase which is assumed to be zero without loss of generality. μ is referred to as *chirp sweep rate* in bandwidth $B\ (= f_2 - f_1)$, i.e., $\mu = 2\pi B/T$. In the following section, we describe the acoustic modem architecture with matched filter for chirp BOK in detail, and discuss implementation issues in smart devices.

IV. Software Receiver for Chirp BOK Signal

We demonstrate the receiver architecture of chirp signals for smart devices. Due to the limitation of the computing power of smartphone, we keep the receiver architecture as simple and efficient as possible.

A. Matched filter and envelope detection

A common method to implement the *matched filter* (or equivalently, *correlator*) receiver is convolution. Received signal $r(t)$ is convolved with time-reversed versions of $s_1(t)$ and $s_2(t)$ to generate estimator c_1 and c_2, respectively, that is,

$$c_{1,2} = \int r(\tau)\, s_{1,2}(T-\tau)\, d\tau. \quad (3)$$

If $c_1 > c_2$, the receiver estimates that Symbol 1 is transmitted, and Symbol 2 otherwise.

The aerial acoustic communication is based on sound signals; the received signal $r(t)$ is obtained from recording process in OS kernel. Since the internal delay that takes to fetch recorded data from audio buffer cannot be tightly controlled, symbol-by-symbol recording might lose some of the acoustic signals in between recordings. Instead, we design the modem to process multiple symbols in a single recording with long duration. One problem of this bulk processing is the complexity of convolution ($O(n^2)$). We can simplify the matched filter using FFT with complexity $O(n \log n)$, i.e.,

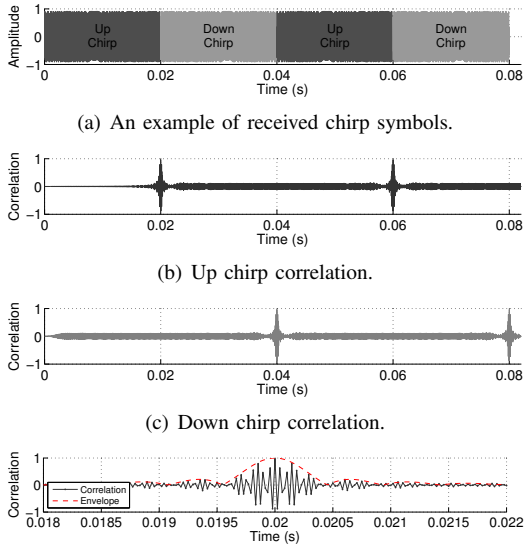
$$c_{1,2} = \mathcal{F}^{-1}\Big\{\mathcal{F}\{r(t)\}\mathcal{F}\{s_{1,2}(T-t)\}\Big\}, \quad (4)$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote FFT and inverse FFT (IFFT), respectively.

Fig. 7 depicts an example of the bulk audio signal processing, assuming that the received symbols are [up, down, up, down] chirps. Fig. 7(b) and Fig. 7(c) are obtained by convolving Fig. 7(a) with the time-reverse of up chirp and down chirp, respectively. Obviously the receiver can recover the transmitted symbols by comparing correlation peaks every 20 ms, i.e., symbol duration. We can efficiently find the correlation peaks of chirp signals as follows.

We can write the close-form auto correlation as

$$\psi(t) = T\Big(1 - \frac{|t|}{T}\Big)\mathrm{sinc}\left[\pi B\Big(1 - \frac{|t|}{T}\Big)\right]\cos(2\pi f_1 t),$$
$$\triangleq e(t)\cos(2\pi f_1 t), \quad (5)$$



(a) An example of received chirp symbols.

(b) Up chirp correlation.

(c) Down chirp correlation.

(d) Up chirp correlation magnified around 20 ms.

Fig. 7. An example of received chirp symbols and corresponding matched filter receiver. The receiver can demodulate chirp signals by comparing the correlation peaks every 20 ms, where the peaks can easily be read from envelopes.

which is a form of cosine-modulated time-decaying sinc function [15]. Due to the oscillating term $\cos(2\pi f_1 t)$ in Eq. (5), it is not easy to read clear peak values as shown in Fig. 7(d). We can smooth the auto correlation curve by finding its envelope. Eq. (5) is very similar to an amplitude modulated (AM) signal with carrier frequency of f_1. Therefore, its envelope $e(t)$ can be obtained via *Hilbert transform* [27] as follows.

$$\begin{aligned} |e(t)| &= |e(t)[\cos(2\pi f_c t) + \jmath \sin(2\pi f_c t)]|, \\ &= \left|\psi(t) + \jmath\hat{\psi}(t)\right|, \end{aligned} \tag{6}$$

where $\hat{}$ denotes *Hilbert transform* that can be calculated from FFT. More into detail, let $\Psi[w]$ and $\hat{\Psi}[w]$ be FFT representations of $\psi(t)$ and $\hat{\psi}(t)$, respectively. Note that

$$\hat{\Psi}[w] = -\jmath \operatorname{sgn}[w] \Psi[w], \tag{7}$$

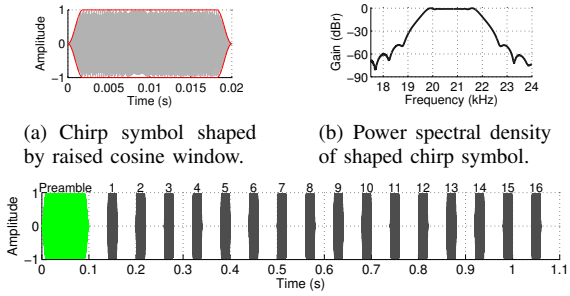
where $\operatorname{sgn}[\omega]$ is a signum function, i.e.,

$$\operatorname{sgn}[w] = \begin{cases} 1, & w > 0, \\ 0, & w = 0, \\ -1, & w < 0. \end{cases} \tag{8}$$

Hence, the right-hand-side of Eq. (6) is rewritten as

$$\begin{aligned} |e(t)| &= \left|\psi(t) + \jmath\hat{\psi}(t)\right|, \\ &= \left|\mathcal{F}^{-1}\{\Psi[w] + \jmath\hat{\Psi}[w]\}\right|, \\ &= \left|\mathcal{F}^{-1}\{2\Psi[w]\, u[w]\}\right|, \end{aligned} \tag{9}$$

where $u[w]$ is unit step function that $u[w] = 1$ for $w > 0$ and $u[w] = 0$ otherwise. It should be noted that we adopt FFT for convolution of received audio and up/down chirps. Therefore the envelopes in Eq. (9) can be easily detected after the convolution. Fig. 9(a) depicts the block diagram of envelope detector.



(a) Chirp symbol shaped by raised cosine window.

(b) Power spectral density of shaped chirp symbol.

(c) Frame structure.

Fig. 8. Envelope of raised cosine window for chirp symbol shaping. We set roll-off factor to 0.1 for -10 dB spectral masking at frequency boundaries. A frame begins with a preamble (long-chirp as a delimiter), and is followed by 16 data symbols. Each symbol is separated by 40 ms guard interval.

TABLE I
SYSTEM PARAMETERS

Parameter	Value	Number of samples	
		@44.1 kHz	@48 kHz
Frequency band	19.5∼22 kHz	-	-
Symbol duration	20 ms	882	960
Guard interval	40 ms	1764	1,920
Preamble	100 ms	4,410	4,800
Number of bits per frame	16 bits	-	-
Frame size	1.1 sec	48,510	52,800
FFT size	65,536 point	-	-
Raised cosine filter	roll-off 0.1	-	-

B. Preamble design and system parameters

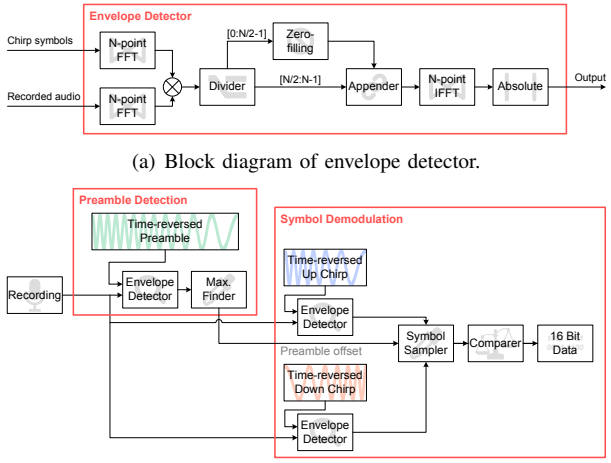
Considering the communication scenario, i.e., chirp signals for short ID broadcast, we assume that 16 bit ID is repeatedly played by speakers. In order for the receiver to find the beginning of a 16 bit ID, we deploy 0.1 sec up chirp signal as a *preamble* that can be also used as a frame delimiter. In addition, indoor acoustic channel typically suffers from multi-paths as discussed in Section III, and hence, we insert 40 ms guard interval between symbols (Fig. 8(c)). For 65,536 size FFT operation at 44.1 kHz or 48 kHz sampling rate, which are the most common sampling rates in audio interfaces, we set the symbol duration to 20 ms. Each data symbol is shaped using raised cosine window with roll-off factor 0.1 as shown in Fig. 8. The raised cosine window, which also works as an ingress filter at matched filter, is designed to reduce (1) out-of-band leakage to -10 dB at frequency boundaries and (2) ringing and rising effects at speakers and microphones [8]. Table I summarizes system parameters used in our design.

C. Receiver process and architecture

1) Recording: A receiver begins signal reception by recording. Considering 1.1 sec frame duration and 0.1 sec preamble, we set the recording size as 1.2 sec. This guarantees that the audio data includes at least one fully-recorded preamble.

2) Preamble detection: The receiver then searches the preamble in the recorded audio by *envelope detection* based on Eq. (9) as follows: (i) calculate FFT of recorded audio ($\mathcal{R}$) and preamble ($\mathcal{P}$); (ii) multiply $\mathcal{R}$ and $\mathcal{P}$, and substitute negative frequency terms to zero;[2] (iii) IFFT and take absolute

[2]The scalar factor 2 in Eq. (9) is omitted here for simplicity.



(a) Block diagram of envelope detector.

(b) Block diagram of receiver.

Fig. 9. Block diagrams of envelope detector and receiver. In Fig. 9(a), input chirp symbol can be either preamble, up chirp, or down chirp. FFT of recorded audio in three envelope detector modules of Fig. 9(b) can share the same memory.

values, and this yields envelope; (iv) find the maximum of envelope, which denotes the location of preamble, referred to as *preamble offset*. Fig. 10(a) depicts an example of preamble detection using its envelope. The receiver can find the peak of preamble envelope which denotes its location within the recorded audio.

3) Symbol demodulation: Now the receiver calculates the envelopes of up and down chirp symbols. The envelope detection in the previous step is repeated, where preamble is substituted by up and down chirps as shown in Fig. 9(b). That is, FFT of up and down chirps ($\mathcal{U}$ and $\mathcal{D}$, respectively) are multiplied by $\mathcal{R}$ instead of that of preamble ($\mathcal{P}$). Beginning from *preamble offset*, the receiver compares the envelopes of up and down chirps every symbol interval (upward and downward triangle marks in Fig. 10(b)), and recovers the received bit. If the receiver encounters end-of-buffer, then it wraps around to head-of-buffer and continues until 16 bits are decoded.

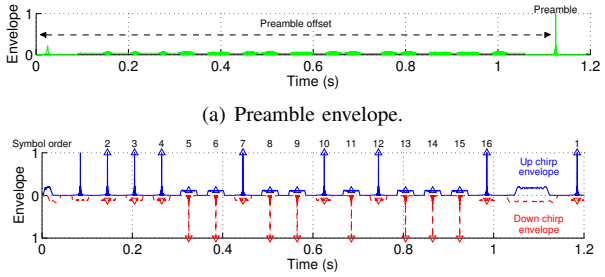
D. Demodulation example

Fig. 10 depicts an example of signal demodulation obtained via MATLAB. We plot up (down) chirp envelope output using solid (dashed) lines in upward (downward) direction for the ease of presentation. After finding the preamble at 1.125 sec, receiver decodes the first symbol (up chirp at 1.185 sec). As the receiver encounters end-of-buffer, it wraps around to the second up chirp at 0.145 sec.[3]

V. Backend Server and Service Scenario

The acoustic signal and modem design in Section IV supports approximately 16 bps data rate, which is sufficient for short ID transmission. Nonetheless, this low data rate is not enough to deliver detailed information to smart devices.

[3]This wrap-around peak sampling is feasible since the transmitter repeatedly broadcasts the same ID over time.



(a) Preamble envelope.

(b) Up and down chirp envelopes.

Fig. 10. Decoding example of the proposed modem. Fully recorded preamble is located at 1.125 sec. After decoding the first symbol (1.185 sec), the envelope hits end-of-buffer. Therefore the receiver wraps around to the second symbol at 0.145 sec.

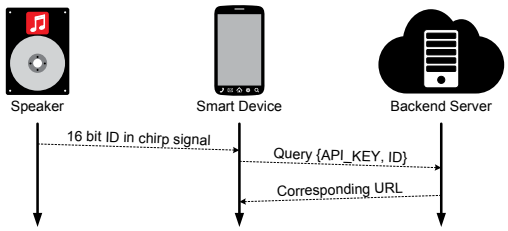


Fig. 11. Message flow between smart device and backend server.

In order to compensate the low bit rate of chirp BOK modem, we deploy a backend server to which a smart device can send a query for more information corresponding to the received 16 bit ID. We also present our on-going service scenario as a mobile commerce platform that reacts to TV programs.

A. System overview

We assume that the smart devices are connected to the Internet through WiFi and/or cellular interface, and the audio of multimedia content includes chirp signals as an inaudible watermark for *content recognition*.

Our backend server is implemented using Amazon web service.[4] The backend server basically manages a MySQL database that contains (i) API Key which is a unique ID of a specific app employing the chirp signal receiver, (ii) 16 bit ID received and demodulated by the receiver, and (iii) URL to be referred by the smart device. Fig. 11 presents the message flow between smart device and our backend server. On the reception of chirp signal and successful demodulation, the app in the smart device sends a query to the backend server with API Key and 16 bit ID. The backend server then responds with a corresponding URL. The smart device can contact to the URL to fetch virtually infinite amount of data.

B. Acoustic modem as a mobile commerce platform

We assume that the TV contents are embedded with the chirp signal that notifies their ID. We also assume that the proposed chirp signal receiver is employed by social commerce apps, so that they can decode the chirp signal in the TV contents. Social commerce app users who enabled the opt-in

[4]We run Apache 2.2.22 HTTP server with PHP 5.3.10 on Ubuntu 12.04.4.

chirp signal receiver can pervasively receive the information of product placement (PPL) during and/or after watching TV.

A challenge of this service is battery consumption issue of smart devices. Continuously recording and processing audio signal for 24 hours is not feasible since it would consume large amount of energy. Instead, we implement scheduling functionality for the chirp signal receiver, so that the apps can be scheduled to activate the receiver process based on the time table of TV programs. For instance, to check if a user actually watched an one-hour TV program, it would be enough to record and process audio signal every 3 minutes. Our experiment has shown that the receiver process consumes only 1% more battery energy when the processing interval is 3 minutes, where the detailed results are omitted in the paper due to the lack of space.

Another technical issue of chirp signal is encoding loss. Going through the TV encoder, the acoustic signal embedded in audio content could be partially lost due to the lossy encoding. Typical TV audio encoders such as advanced audio coding (AAC) and Dolby digital (AC3) have cut-off frequency around 21 kHz, where the detailed threshold varies depending on the bit-rate settings. Note that, however, partial frequency loss of the encoding process is similar to the effect of speaker and microphone's frequency response, and hence, we expect that the chirp signal can be delivered through the TV network. Leaving the field test through the actual TV network as our future work, we present in-lab test results below.

VI. Performance Evaluation

We present experimental results and validate the performance of the proposed chirp BOK modem for smart devices. We implement our chirp BOK modem on various Android devices with OS version from Froyo (Android 2.2) to Kitkat (Android 4.4). Most experiments are conducted in a static indoor office environment unless mentioned otherwise.

A. Transmission range in indoor environment

We first verify the maximum transmission range of the proposed modem. We deploy Genelec 6010A speaker and 15 different smartphones at the end side of 25 m corridor with 2.5 m width and 3 m height. MacBook Air laptop plays wave audio file through the Genelec 6010A speaker, where the volumes of laptop and speaker are set to 50%.

Fig. 12 presents the success probability of signal reception, averaged over 300 repetitions of 16 bit ID reception. It can be observed that most devices can successfully receive the chirp signal with 97% probability, which verifies that the proposed modem works in long transmission range. The low success rate of Galaxy Tab is due to its low volume of recording, as shown in Fig. 4(b). Considering the received audio level is approximately 10 dBSPL at 20 kHz, we expect that increasing the audio output level could further increase the transmission range.

B. Effect of lossy encoding

In order to study the effect of lossy encodings on the signal reception probability, we play chirp signals using a Samsung smart TV (UN46F8000). Three different types of audio files are used for the experiment; loss-less wave format (i.e., WAV), and AC3 and AAC for lossy codec.

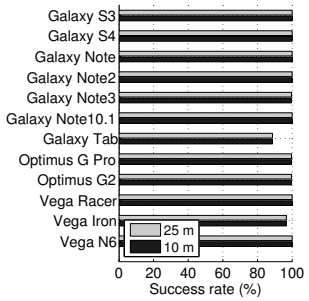


Fig. 12. Success probability of signal reception at 10 m and 25 m distances for various smart devices. Most devices can decode the transmitted signal with 97% probability at these distances.

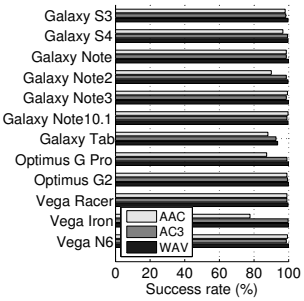


Fig. 13. Success rate of signal reception for AAC, AC3, and WAV audio formats in 5 m distance experiment. WAV and AC3 files perform the same, while the success rate of AAC is worse due to the signal loss in its frequency response.

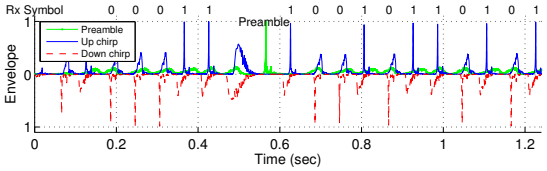
Fig. 13 presents the experimental results in a 5 m distance indoor environment. The direct path between TV and smart devices are partially blocked by partition screens. In case of WAV and AC3 formats, we observe that chirp signals can be successfully delivered despite of the partition screens due to the diffraction of sound waves. For AC3 format, however, the success rate degrades in Vega Iron, Optimus G Pro, and Galaxy Note2 models. This is due to the signal loss of AC3 encoding. As those models have small frequency gain in high frequency band, they sensitively reacts to the loss of high frequency signal compared to the others. Nonetheless, we claim that the encoded chirp signals can reach the smart devices through TV audio format with acceptable success probability in most cases.

C. Multi-path resolution

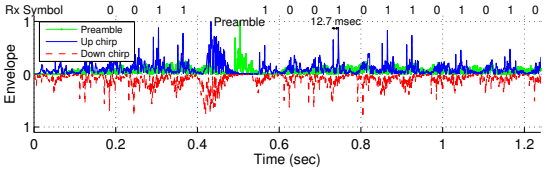
One of the reasons why chirp signal is robust in indoor environments is its capability of resolving multi-paths. Figs. 14(a) and 14(b) depict the envelopes at the chirp signal receiver in indoor single-path and multi-path environments, respectively. The envelope peaks in single-path environment clearly appear. The clustered peaks in Fig. 14(b) are separated by 12.7 ms (equivalently, 4.3 m path difference for 340 m/s propagation speed of sound). The receiver can differentiate the mingled signal from the envelope, and successfully recover the received symbols in multi-path environment.

More into detail, chirp signals sweeping B Hz can resolve two different chirp signals traversing with $1/B$ sec path difference [15]. Considering 2.5 kHz bandwidth of our signal design, the proposed chirp BOK modem can differentiate 0.4 ms difference of multi-path delays in an ideal case, which is equivalent to 0.14 m path difference. In practice, however, this could be worse due to the decreased chirp sweeping band caused by frequency selectivity, as shown in Fig. 15.

In order to assess the capability of multi-path resolution, we emulate the multi-path propagation of sound waves by using stereo speakers. We play chirp preamble through stereo

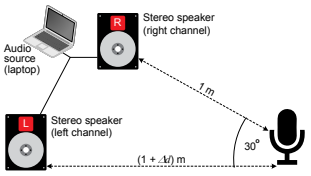


(a) Single-path environment.

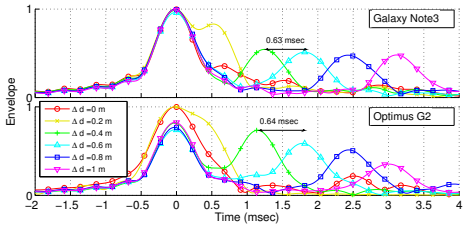


(b) Multi-path environment.

Fig. 14. Received chirp envelopes in single-path and multi-path environments. Received symbols are denoted above the envelope curves. Envelope peaks in multi-path propagation are separated by 12.7 ms.



(a) Experiment topology.



(b) Experimental results of Galaxy Note3 and Optimus G2.

Fig. 15. Experiment topology and results in stereo chirp transmission. Δd in Fig. 15(a) varies from 0 m to 1 m. Envelope peaks are approximately separated by 0.64 ms, which corresponds to 0.2 m path difference.

speakers connected to a common audio source in an open space, as shown in Fig. 15(a). We make the path difference of two signals by controlling the distance between microphone and left channel speaker. Fig. 15 presents the envelopes of Galaxy Note3 and Optimus G2, while those of the other devices are omitted as they have similar patterns. The minimum distance that we can observe distinct envelope peaks of stereo signals differs by model. In general, most devices are found to be able to differentiate the multi-path signals separated by 1.18 ms (equivalently, 0.4 m path difference). This is slightly worse than the theoretical capability 0.4 ms (equivalently, 0.14 m path difference) as chirp signals are partly lost due to the selective frequency response of speakers and microphones.

D. Local peak detection for Doppler shift compensation

Wireless communication usually suffers from Doppler shift due to mobility, and this is the same for acoustic communications. We study the effect of Doppler shift on chirp BOK modem in this section.

Doppler shift causes frequency shift to chirp signals, which results in shift of peaks in envelope output [15]. Fig. 16 depicts an example of envelope peaks shifted by Doppler shift. In a static environment, the envelope peak is supposed to appear at 20 ms, which is equal to the symbol sampling timing. However, in this example, Doppler shift makes the peaks move forward, so that the symbol sampling timing needs to

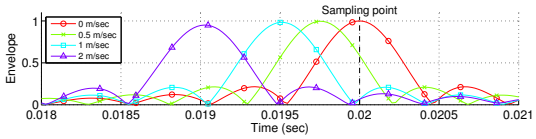


Fig. 16. Shift of peaks due to Doppler shift, assuming 20 ms symbol is transmitted. In a static environment, the envelope peak is supposed to appear at 20 ms symbol timing.

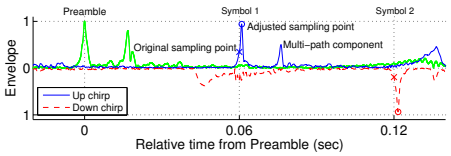


Fig. 17. Experimental results of Doppler shift and shifts of envelope output peaks. We slowly move a smartphone in front of a speaker, and this results in 1 ms peak shift of envelope.

be adjusted.

We adjust the symbol sampling timing by finding *local maxima* around the original sampling points. Considering that 2 m/sec walking speed causes approximately up to 1 ms peak shift depending on the moving direction [15], we choose 2 ms searching range to cover both directions moving forward and away.

Fig. 17 presents the experimental results of adjusted sampling timing. We slowly move a smartphone (Galaxy Note2) back and forth in front of a speaker, and then obtain the corresponding envelope output. According to the frame structure, the first symbol is supposed to appear at 60 ms after the preamble (x mark at 60 ms in Fig. 17). Due to the Doppler shift, however, the first peak appears approximately 1 ms later (o mark at 61 ms in Fig. 17). By searching for the local maxima, we can correctly sample the optimal peak.

It should be noted that extending the searching range is not always helpful. Especially in a severe multi-path environment, indistinct multiple peaks make the symbol timing decision even harder. We can design a *Rake receiver* in order to combine the multiple peaks of multi-path propagation. In this paper, we leave the design of Rake receiver as our future work.

E. Detection of multiple chirp signals

In practice, a smart device might receive multiple overlapping signals from more than one signal sources. The proposed receiver basically locks on the strongest chirp signal in such a case, as it detects the preamble from the maximum peak of the

TABLE II
DETECTION RATES OF TWO-SIGNAL SCENARIO (L/R)

		Time offset Δt (ms)				
		0	20	40	60	120
Amplitude difference (dB)	0	0.11/0.1	0.48/0.52	0.02/0.98	0.01/0.88	0.01/0.99
	1	0/0.99	0/1	0/1	0/0.99	0/1
	3	0/1	0/1	0/1	0/1	0/1

envelope. We here present the experimental results with two signal sources and examine the feasibility of multiple signal detection.

We set an experiment in Fig. 15(a), where $d = 0$ and play different signal sources in left and right channels. The stereo signal sources are skewed by Δt ms and the right channel signal is greater than the left channel one by $0 \sim 3$ dB. This emulates two chirp signals arriving at a smart device with time and amplitude difference in practice.

Table II summarizes the detection probability of left and right channel signals (L/R) out of 100 signal receptions. We verify that the proposed receiver detects the strongest signal in most cases. The exception is when two signals have exactly the same amplitude and precisely overlapping data symbols, which is very unlikely to happen in practice. The reception rates of the left and right channels in the 0 dB case with $\Delta t \geq 40$ ms time offset are uneven as we cannot perfectly match the signal strengths of two channels. Very small difference of signal strengths makes the receiver sensitively stick to the stronger signal (the right channel in this experiment). When the amplitude difference is 0 dB, some overlapping data symbols are lost in the 60 ms case as their preambles are partially overlapping. We can verify this from the result of 120 ms where the fully distinct preambles differentiate the overlapping data symbols. This trend repeats until $\Delta t = 1.1$ s, which is equivalent to $\Delta t = 0$ ms considering the frame duration, where we omit the results here due to the lack of space. Therefore, we claim that the proposed receiver works well in multiple-signal environments.

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose chirp signal-based aerial acoustic communication technique for smart devices. Considering severe multi-path propagation of indoor acoustic channel as well as device-dependent frequency selectivity, we adopt *chirp signal* for indoor acoustic communication. The proposed chirp BOK modem architecture can deliver information at approximately 16 bps up to 25 m distance with 97% success rate for most smart devices, and overcome Doppler shift by adjusting time points for symbol sampling. The chirp signals are also shown to be able to pass through lossy audio codecs such as AC3 and AAC. We additionally design backend server architecture to make up the low data rate of the proposed acoustic software modem.

In the future, we plan to design an equalizer and Rake receiver that can cope with multi-path propagation. It is expected to shorten and/or remove 40 ms guard intervals in the current frame structure such that the data rate can be enhanced. We are also developing *combined decoder* to improve the decoding capability, based on the fact that transmitter is assumed to broadcast the same bits repeatedly over time.

ACKNOWLEDGEMENT

The authors sincerely thank to Kyung-Kuk Lee, CEO of Orthotron in Korea, for his in-depth technical discussion on chirp signals.

REFERENCES

[1] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: propagation models and statistical characterization," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 84–89, 2009.
[2] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, no. 3, pp. 257–279, 2005.
[3] C. V. Lopes and P. M. Aguiar, "Acoustic modems for ubiquitous computing," *IEEE Pervasive Comput.*, vol. 2, no. 3, pp. 62–71, 2003.
[4] A. Madhavapeddy, R. Sharp, D. Scott, and A. Tse, "Audio networking: the forgotten wireless technology," *IEEE Pervasive Comput.*, vol. 4, no. 3, pp. 55–60, 2005.
[5] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, "BeepBeep: a high accuracy acoustic ranging system using cots mobile devices," in *Proc. ACM SenSys*, 2007, pp. 1–14.
[6] V. Gerasimov and W. Bender, "Things that talk: using sound for device-to-device and device-to-human communication," *IBM Systems Journal*, vol. 39, no. 3.4, pp. 530–546, 2000.
[7] C. V. Lopes and P. M. Aguiar, "Aerial acoustic communications," in *Proc. IEEE WASPAA*, 2001, pp. 219–222.
[8] R. Nandakumar, K. K. Chintalapudi, V. Padmanabhan, and R. Venkatesan, "Dhwani: secure peer-to-peer acoustic NFC," in *Proc. ACM SIGCOMM*, 2013, pp. 63–74.
[9] H. Matsuoka, Y. Nakashima, and T. Yoshimura, "Acoustic communication system using mobile terminal microphones," *NTT DoCoMo Tech. J*, vol. 8, no. 2, pp. 2–12, 2006.
[10] H. S. Yun, K. Cho, and N. S. Kim, "Acoustic data transmission based on modulated complex lapped transform," *IEEE Signal Processing Lett.*, vol. 17, no. 1, pp. 67–70, 2010.
[11] M. Hanspach and M. Goetz, "On covert acoustical mesh networks in air," *J. of Commun.*, vol. 8, no. 11, 2013.
[12] SonicNotify. [Online]. Available: sonicnotify.com/
[13] ShopKick. [Online]. Available: www.shopkick.com
[14] Zoosh. [Online]. Available: www.verifone.com/industries/taxi/way2ride/
[15] C. E. Cook, "Linear FM signal formats for beacon and communication systems," *IEEE Trans. Aerosp. and Electron. Syst. Mag.*, no. 4, pp. 471–478, 1974.
[16] A. J. Berni and W. Gregg, "On the utility of chirp modulation for digital signaling," *IEEE Trans. Commun.*, vol. 21, no. 6, pp. 748–751, June 1973.
[17] A. Viterbi, "Optimum detection and signal selection for partially coherent binary communication," *IEEE Trans. Inform. Theory*, vol. 11, no. 2, pp. 239–246, 1965.
[18] B. C. J. Moore, *An introduction to the psychology of hearing*, 5th ed. Academic Press San Diego, 2003.
[19] E. Terhardt, "Calculating virtual pitch," *Hearing research*, vol. 1, no. 2, pp. 155–182, 1979.
[20] A. Farina, "Simultaneous measurement of impulse response and distortion with a swept-sine technique," in *Proc. Audio Engineering Society Convention*, 2000.
[21] C. E. Cook and M. Bernfeld, *Radar signals: an introduction to theory and application*. Academic House, 1967.
[22] IEEE 802.15.4a, *Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), Chirp Spread Spectrum PHY Layer*, IEEE Std., 2007.
[23] M. R. Winkley, "Chirp signals for communications," in *WESCON Convention Record*, vol. 14, no. 2, 1962.
[24] S. E. El-Khamy, S. E. Shaaban, and E. Tabet, "Efficient multiple-access communications using multi-user chirp modulation signals," in *Proc. IEEE ISSSTA*, vol. 3, 1996, pp. 1209–1213.
[25] P. Lazik and A. Rowe, "Indoor pseudo-ranging of mobile devices using ultrasonic chirps," in *Proc. ACM SenSys*, 2012.
[26] Acoustic Ruler. [Online]. Available: https://iqtainment.wordpress.com/acoustic-ruler/
[27] R. N. Bracewell and R. Bracewell, *The Fourier transform and its applications*. McGraw-Hill New York, 1986.