AERIAL ACOUSTIC COMMUNICATIONS

Cristina Videira Lopes

Xerox PARC
Computer Science Laboratory
3333 Coyote Hill Rd, Palo Alto, CA 94304
lopes@parc.xerox.com

Pedro M. Q. Aguiar

Instituto Superior Técnico
Institute for Systems and Robotics
Av. Rovisco Pais, 1049-001 Lisboa, Portugal
aguiar@isr.ist.utl.pt

ABSTRACT

This paper describes experiments in using audible sound as a means for wireless device communications. The direct application of standard modulation techniques to sound, without further improvements, results in sounds that are immediately perceived as digital communications and that are fairly aggressive and intrusive. We observe that some parameters of the modulation that have an impact in the data rate, the error probability and the computational overhead at the receiver also have a tremendous impact in the quality of the sound as perceived by humans.

This paper focuses on how to vary those parameters in standard modulation techniques such as ASK, FSK and Spread-Spectrum to obtain communication systems in which the messages are musical and other familiar sounds, rather than modem sounds. A prototype called Digital Voices demonstrates the feasibility of this music-based communication technology. Our goal is to lay out the basis of sound design for aerial acoustic communications so that the presence of such communications, though noticeable, is not intrusive and can even be considered as part of musical compositions and sound tracks.

1. INTRODUCTION

Inter-machine communications have always been kept away from our own communication channel, audible sound in air. There are good reasons for this: the data rates are relatively low when compared to other media (e.g. electric wires, radio) and the sounds tend to be annoying. But as more and more devices support an audio channel for voice or music, that channel becomes a cheap option for transferring arbitrary information among devices that happen to be near each other. Sound is attractive for applications that do not require high bit rates and for which it is expensive to extend the hardware infrastructure with radio or infrared transmitters. It is also attractive for applications that require human awareness of the communication. Some examples of those applications are: toys; broadcasting information through the sound of TV and radio that can be picked up by devices at home or in the car; transferring names and phone numbers between cell phones; transferring business cards between PDAs; and broadcasting location-dependent information from rooms into PDAs and laptops. The design of such communication systems, however, must be carefully revised.

Most digital communication systems in use today are designed with goals such as the maximization of transmission data rate, the minimization of the probability of bit error, the minimization of the required bandwidth and the minimization of the required power [1]. Under those criteria, sound in air is a poor choice. Motivated by the specific characteristics of the aerial acoustic communication paradigm used by humans and other animals, we believe device-to-device aerial acoustic communications will be useful if the goals for such systems are refocused along the following criteria:

1. The messages of these communication systems should be pleasant to humans. They should either be imperceptible or, if perceivable, they should sound like music or familiar environment sounds such as birds, wind or water drops.
2. The systems are to be deployed in ordinary hardware. We should utilize the existing infrastructure for voice, avoiding extra costs.
3. The systems are to be used in ordinary environments. This means that the communication has to be reasonably robust in the presence of noise such as people talking.

In the Digital Voices project, we explore perceivable communications in audible sound. We started by analyzing common modulation techniques and the kinds of sounds they produce. We observed that some parameters of the modulation have a strong effect on the quality of the sound. Variations in those parameters allow us to obtain many different types of acoustic messages ranging from modem-noises to music.

The channel we target – air plus speakers and microphones included in palmtops/laptops/desktops/TVs – is far from ideal, not only because of ambient noise, but also because the hardware is faulty and the defects vary from device to device. Rather than seeing this as a set-back, we take it as a challenge to design sounds that are robust enough to survive the transmission through that imperfect channel.

This paper focuses on the application of standard modulation techniques to sound and the types of acoustic messages we can get. The issue of robustness is briefly addressed.

Paper organization. In section 2 we overview the use of sound in device communications and contrast it to our work. Section 3 revises standard modulation techniques and the receiver implementation. Section 4 describes the Digital Voices prototype and the emergence of music and other familiar sounds. In section 5 we state experimental observations. Section 6 concludes the paper.

2. SOUND IN DEVICE COMMUNICATIONS

The traditional uses of sound in device-to-device communications can be grouped in four categories: (1) sound as a way of utilizing the existing telephone networks for long-distance point-to-point communications (modems); (2) underwater communications; (3) ultrasonic remote controls [2]; and (4) speech recognition/synthesis and

other non-speech auditory displays that make the interaction of machines with humans more friendly [3].

In the last five years, there has been considerable work done in the new area of information hidding in audio [4]. The music industry has been trying to use audio watermarking as the key to preserve ownership of the music in electronic format. Also, in the last year, there have been some business ventures that took on that work and applied it to toys. There is a growing interest in this kind of communication, and we believe much is still to be done.

Our work fits in this new area of using audible sound to wirelessly transmit information between devices. But we take a different approach than that taken by watermarking. The goal of audio watermarking is to embed information in pre-existing sounds so that the data is imperceptible to the human ear. Digital Voices don't aim at hidding the data, they expose the data to the human ear. One consequence of this difference is that audio watermarking is constrained to techniques that preserve the important characteristics of the original sounds, and therefore can only transmit very low data rates. Public reports mention 32 bps or less – we don't know the numbers for proprietary technology used by recent start-ups. Most of our successful experiments transmit data at rates ranging from hundreds of bps to more than 1 Kbps.

Recently, Gerasimov and Bender [5] presented experiments in using the aerial acoustic channel for device-to-device communications. They have evaluated a number of variations of ASK and FSK according to the data rate, computational overhead, noise tolerance and disruption level. They report a maximum data rate of 3.4 Kbps using multiple-level B-FSK going into the low ultrasound band (18 KHz).

The novel idea in our work is to study how music and other pleasant sounds can emerge in the audible band by carefully choosing some parameters of the modulation. By doing so, we surpass the low data rates imposed by imperceptibility while preserving the property of using messages that are tolerable to humans.

3. DIGITAL AERIAL ACOUSTIC COMMUNICATIONS

3.1. Common Modulation Techniques

ASK. In amplitude-shift keying (ASK) modulation, the message is encoded in the signal amplitude. The number of levels of amplitude determines the number of bits encoded in each symbol. The aerial acoustic channel has particularly challenging characteristics such as the multiple reflections that corrupt the received signal with multiple echoes and the very fast decrease of the signal power. Due to these characteristics, we can't rely on using many levels of amplitude, at least without using expensive equalization techniques. For this reason, we have used binary amplitude-shift keying modulation (B-ASK).

We use multi-frequency B-ASK. We split the message $\{a_n\}$ in subsets of a pre-specified number of N bits, forming a N-vector-valued baseband signal. Each entry n of the vector signal modulates a sinusoidal carrier of frequency f_n in the time interval $t \in [0, T]$. The transmitted signal $s(t)$ is then given by

$$s(t) = \sum_{n=1}^{N} a_n \sin(2\pi f_n t), \qquad t \in [0, T]. \tag{1}$$

FSK. In frequency-shift keying (FSK) modulation, the message is encoded in the frequency of the signal. We use M-ary FSK (M-FSK), each frequency corresponding to one multi-bit symbol. The modulated signal $s(t)$ is given by

$$s(t) = a \sin(2\pi f_m t), \qquad t \in [0, T]. \tag{2}$$

We implemented a more general scheme that uses K tones per symbol, rather than a single one. In this case, the modulated signal $s(t)$ is written as

$$s(t) = a \sum_{k=1}^{K} \sin(2\pi f_{km} t), \qquad t \in [0, T]. \tag{3}$$

Spread-Spectrum. In spread-spectrum (SS) modulation, the the carrier frequencies are spread, over time, across a wide frequency spectrum, much wider than the minimum bandwidth required to transmit the information being sent [1]. The spreading is made according to a sequence, the hopping code, that is shared by the sender and the receiver.

3.2. Receiver Implementation

The task of the receiver is to recover the original bit sequence from the received acoustic signal. After synchronization, the problem reduces to detect the symbol transmitted over each time interval from the received signal $r(t), t \in [0, T]$.

In both ASK and FSK schemes described above, the transmitted signal was generated by summing sinusoids of known frequencies, see expressions (1) and (3). The main task of the detector is, then, to decide if each frequency component is present or not in the received signal. A number of approaches to this problem are available in the literature, see [1]. These include the coherent methods that require the knowledge of phase information, like the correlation receiver (matched filter) and the noncoherent ones, such as the use of bandpass filters (envelope detectors). We implemented a robust noncoherent detector by using the quadrature receiver.

The quadrature receiver sums the square of the integral of the quadrature components of each frequency f_n of the received signal $r(t)$. This is written in a compact way as[1]

$$R_n = \left| \int_0^T r(t) \exp(j2\pi f_n t)\ dt \right|. \tag{4}$$

The decision about the tone of frequency f_n being or not present in the signal is made by thresholding R_n. We use a calibration sequence to normalize the signal power for each frequency, so that the threshold value can be chosen independently of the frequency.

3.3. Symbol Duration, Frequency Spacing, and Data Rate

The choice and number of frequencies used and the symbol duration T both have influence on the transmission data rate. In what follows, we briefly discuss the choice of these parameters.

For band-limited channels, the symbol duration T is lower bounded by the Nyquist limit. In fact, to avoid intersymbol interference (ISI), the symbol duration T must be greater than $1/(2B)$, where B is the channel bandwidth, see [1].

Under the assumption of zero inter-symbol interference (ISI) and an ideal band-limited channel, the signal received by the detector to estimate each symbol is simply the time-windowed superposition of sinusoids expressed in (1) and (3). Due to the time-limited window of observation, the tones may interfere with each other.

[1] We use a continuous time notation for commodity. In practice, the signals are sampled and the integrals are replaced by appropriate sums.

This imposes a lower bound on the symbol duration T in terms of the spacing Δ_f of the frequencies. If we choose T in order to make zero the interference between the tones, we get $T = 1/\Delta_f$, as derived elsewhere.

The value of T determines the data rates achievable by our modulation schemes. For example, with B-ASK modulation using N frequencies, we transmit N bits per symbol and the data rate is $b = N/T$ bits per second. If we choose the N frequencies to be uniformly distributed over the channel band, the number of frequencies is $N = B/\Delta_f$, where B is the channel bandwidth, and the maximum data rate is given by

$$T = \frac{1}{\Delta_f} \qquad \Rightarrow \qquad b = B. \tag{5}$$

For example, for multiple-frequency B-ASK, $B = 10$ KHz, and to minimize tone interference, we can expect at most 10 Kbps.

4. DIGITAL VOICES

This section introduces some of the sound designs that we consider more promising to accomplish our goals.

Musically-Oriented Variations of ASK

We experimented with 8-frequency designs. For T=20 ms, the data rate is 400 bps. For T=100 ms, the data rate is 80 bps.

Case 1. The frequencies are related by a pentatonic scale[2] starting at 1000 Hz. For T=20 ms, the sound is similar to sounds of grasshoppers. For T=100 ms it sounds like a piece of music played by several instruments (soprano flutes, maybe). Fig. 1 illustrates this last case.

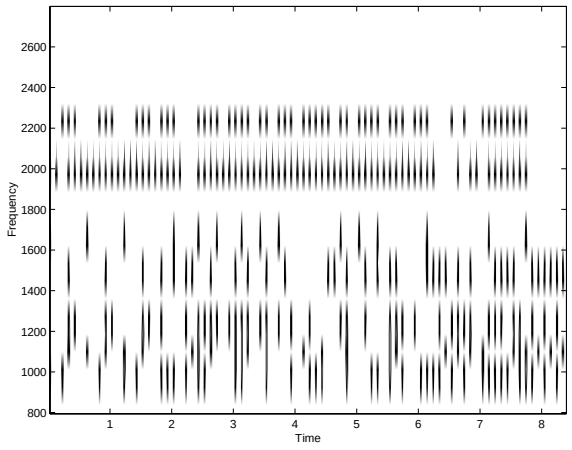

Figure 1: *Spectrogram of a multi-frequency B-ASK modulated message, with frequencies in pentatonic relation and $T = 100$ ms. This was a 7-bit ASCII message, the higher order bit is always 0.*

Case 2. The frequencies are in harmonic relation with the lowest frequency, 1000 Hz. For T=20 ms, the sound is, again, similar to sounds of grasshoppers. For T=100 ms, it sounds like a single instrument - a string instrument - playing the same note over and over again, although with the "string" being stroke in different ways.

Case 3. A third design uses 128 frequencies, all of them harmonics of 70 Hz, starting at 700 Hz. For T=100 ms, the data rate is 1280 bps. The sound is quite different from the previous designs: it sounds like an electronic-music drum beat.

Musically-Oriented Variations of FSK

Case 4. We used 256 frequencies are separated by intervals of 20 Hz, starting at 1000 Hz, to transmit an 8-bit value in each T =20 ms. The bit rate is 400 bps. The sound resembles one single grasshopper. When we increase T to 100 ms it sounds like a bird. Fig. 2 illustrates this case.

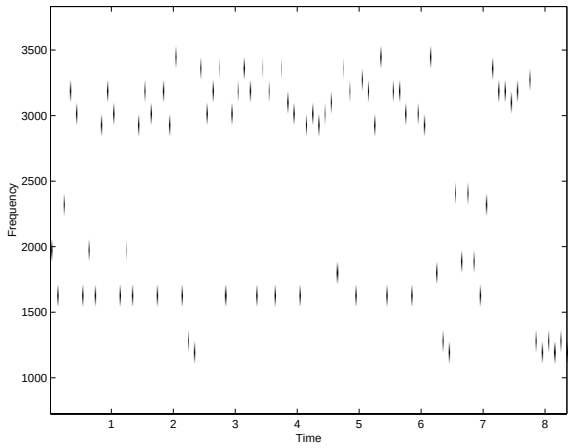

Figure 2: *Spectrogram of a 8-FSK modulated message, with Δf =20 Hz and $T = 100$ ms.*

Although the data rates for case 1 and case 4 are the same for the same values of T, the resulting sounds are quite different, as the spectrograms indicate. In case 1, there are usually several tones at each time, whereas in case 4, there is only one tone at each time. For small values of T, such as 20 ms, the human ear cannot distinguish the differences; they only become clear for higher values of T such as 100 ms.

Case 5. We use chords, rather than single frequencies, to transmit a 7-bit value in each T=200 ms, resulting in a bit rate of 35 bps (for 7-bit ASCII characters, this means 5 characters/s). We use the 7 tones of a major diatonic scale as the keys to form chords. The chords can be major or minor (presence of the 3rd major or the 3rd minor) and can include the 7th or not. Given a 7-bit value of the form $b_6b_5b_4b_3b_2b_1b_0$, we establish the following mapping that uses simultaneously FSK and ASK: b_6 determines whether the chord includes the 7th; b_5 determines the mode (major or minor); $b_4b_3b_2$ determine the key of the chord; and finally $b_1\,b_0$ determine which inversion to use (we assume 4 possible inversions, the 4th being the same as the 1st, but one octave above).[3] In order to make it more pleasant, we include a silence every 7 chords. The result is a sequence of familiar chords with a 4/4 rhythm that, although in arbitrary sequence, make the message sound like an ordinary musical composition.

Case 6. To increase the robustness of the above described scheme, we used a redundant code. The redundant information is sent on the 4th and 6th harmonics of the chord key, and it encodes the same information of the chord but in a rather different form. The key (bits $b_4b_3b_2$) is encoded in the frequency of the harmonics, since these are harmonics of the key frequency. As for the inversion, the mode

[2]The frequencies of the pentatonic scale are defined by the ratios $\{19/85/43/25/3\}$.

[3]For 7-bit ASCII characters, this mapping results in sounds that are humanly identifiable as corresponding to numbers vs. letters (presence or absence of the 7th) and capital letters vs. non-capital letters (minor/major).

and the presence or absence of the 7th, i.e. the remainder 4 bits, they are encoded in the time at which the harmonics are played. The result is the same musical composition overlaid with a xylophone-like melodic line that sounds slightly out of tempo.

Musically-Oriented Variations of SS

In sound, SS modulation takes a whole new meaning, as it is a key factor to produce melodic messages and, at the same time, can improve the robustness of the communication. We explain the mechanism with one of the variations we have implemented.

Case 7. The hopping code is a melodic line taken from a famous movie: B'-C#'-A'-A-E-E. Each tone lasts for about 1 sec, i.e the system hops every second. The data is B-FSK-modulated using the 4th and 8th harmonics of the hopping code frequency, representing 0 and 1 respectively. The data modulation period is about 7 ms, resulting in 143 bps. The hopping tones themselves are also included in the final signal. The result is a relatively slow melodic line with fast temporal variations in those two higher harmonics of each tone.

Similarly to SS for radio, hopping can make the communication more robust by using different parts of the spectrum at different times.

5. EXPERIMENTS

Digital Voices has been tested in Pentium PCs running Windows 2000, with Harman/Kardon speakers and off-the-shelf $5 microphones. We used a sampling rate of 22050 Hz and sound frequencies up to 10 KHz. Both Matlab and Java implementations of the coders/decoders were used. All case studies described here can be heard at [6].

We started by confirming in practice that variations of ASK, FSK, and SS that simply try to maximize the data rate result in annoying sounds. Then, we started experimenting with different values of frequencies and symbol durations T by implementing the schemes described in the previous section. In what follows, we summarize observations related to sound design, auditory perception and the communication channel.

In order to remove the sudden phase shifts from symbol to symbol, which result in annoying clicks, we have used two different strategies. One was to compute the phase so that there were no phase shifts from symbol to symbol. The other strategy was to multiply the signal by a smooth window such as the Blackman function. We observed, however, that the particular window function affects the quality of the sound. Therefore we also used those functions to fine-tune the timbre of the messages.

Sounds that have the same modulation schemes are perceived very differently depending on the symbol duration T. For the schemes we used, there is a sharp perceptual change between 80 ms and 30ms. We take advantage of that to obtain different voices. The more frequencies we use, the less sense of pitch we get. This is no surprise, considering the spectral analysis of existing sounds. What was less obvious was the threshold we observed. Up to 8 harmonically related frequencies the sound is pleasant; around 16 harmonically related frequencies the sound acquires an annoying characteristic; but after 25 or so frequencies, it becomes tolerable again, this time sounding like electronic drum beats.

The use of broad-spectrum messages such as case 4 is highly vulnerable to the non-uniform attenuation that the speakers and microphones introduce in the signal. The hardware has some hot spot frequencies that systematically get their power reduced to almost zero, especially when other frequencies are present. The problem is even more serious considering that different devices have different hot spots. This gives us some constraints and guidelines as to the kinds of sounds that we can use with his hardware and the data rates we can expect.

6. SUMMARY AND OPEN ISSUES

This paper describes experiments in using audible sound as a means for wireless device communications. It focuses on variations of B-ASK, M-FSK and SS modulations. The theoretical data rates of these modulations in the audible band are low, indicating, to no surprise, that the audible sound in air is a poor choice for transferring large amounts of data from device to device. Nevertheless, we envision several applications that can work within the range of 100 to 1000 bps, as long as the messages are not intrusive. Therefore we revised the criteria for such communication systems to include perceptual factors, wide availability and robustness. We proposed several acoustic message designs that resulted in relatively pleasant sounds.

We have started addressing the robustness and tolerance to noise issues, through the addition of redundant signals to the baseband signal. More work is necessary to understand how that affects the quality of the sounds and if it can be used to improve that quality from a perceptual point of view.

Another relevant issue concerns data compression. Just like human language's words usually take less time to transmit than the set of their individual characters, Digital Voices can also compress the individual symbols in higher-order symbols, if there exists additional knowledge about the data that's being transmitted. For example, if the data includes a fair amount of URLs and email addresses, it may make sense to compress the suffixes ".com", ".org" and ".edu" into higher-order symbols that take less than four characters time to transmit.

Acknowledgements. Horst Haussecker and Trevor Smith from PARC made important contributions to the implementation of some of the coders/decoders.

7. REFERENCES

[1] Sklar B., "Digital Communications," Prentice Hall, 1988.

[2] http://www.zenith.com/about_adler.html

[3] Kramer G. (ed.), "Auditory Display", SFI studies in the Sciences of Complexity, Proc. Vol XVIII, Addison-Wesley, 1994.

[4] Bender W., Gruhl D., Morimoto N., Lu A., "Techniques for Data Hidding," IBM Systems Journal, Vol. 35 Nos.3-4, 1996.

[5] Gerasimov V., and Bender W., "Things that talk: using sound for device-to-device and device-to-human communication," IBM Systems Journal, Vol. 39, Nos. 3-4, December 2000.

[6] http://www.parc.xerox.com/csl/members/lopes/digitalvoices